Towards real MPC implementation in an office building using TACO

Filip Jorissen
KU Leuven, Leuven, Belgium, filip.jorissen@kuleuven.be

Damien Picard
KU Leuven, Leuven, Belgium; Boydens Engineering, Dilbeek, Belgium, damien.picard@kuleuven.be

Iago Cupeiro Figueroa
KU Leuven, Leuven, Belgium, iago.cupeirofigueroa@kuleuven.be

Wim Boydens
Boydens Engineering, Dilbeek, Belgium, wimb@boydens.be

Lieve Helsen
KU Leuven, Leuven, Belgium; EnergyVille, Waterschei, Belgium, lieve.helsen@kuleuven.be

Follow this and additional works at: https://docs.lib.purdue.edu/ihpbc
Towards real MPC implementation in an office building using TACO

Filip JORISSEN*, Damien PICARD, Iago CUPEIRO FIGUEROA, Wim BOYDENS, Lieve HELEN

* Corresponding Author
filip.jorissen@kuleuven.be

1 KU Leuven, Department of Mechanical Engineering, Leuven, Belgium
2 EnergyVille, Waterschei, Belgium
3 boydens engineering, Dilbeek, Belgium

ABSTRACT

Model predictive control (MPC) is a promising alternative to rule-based control since it is more suitable to control increasingly complex buildings and thereby realising energy savings and comfort improvement. Practical implementations are however hampered by the complexity of MPC and the expertise required for developing MPC. Therefore, a toolchain for automated control and optimization (TACO) has been developed that automatically translates an object-oriented Modelica model into an efficient MPC code. Since object-oriented models from the Modelica IDEAS library are used, the expertise requirement and development time are reduced significantly. TACO has, however, not yet been applied to a real building and its robustness in real operation still must be demonstrated. The purpose of this paper is to provide a comprehensive overview of the steps that are proposed for implementing an MPC using TACO. We therefore summarise our existing methodology and describe our future extension plans to implement an MPC in the Infrax office building in Brussels by September 2018.

1. INTRODUCTION

Building space heating and HVAC account for 15% of the world final energy use (International Energy Agency, 2015). Therefore, according to the European Union’s Directive 2010/31/EN (European Parliament, 2010), an increasing effort is spent at increasing the building efficiency by improving their insulation level, by installing HVAC systems with a high primary energy efficiency (e.g. a combination of heat pump and floor heating or concrete core activation), and by increasing the share of renewable energy sources in buildings. These measures however increase the complexity of building rule-based controllers (RBC). Firstly, the increase of the insulation level and often also the building thermal mass result in an increase of the building time constants, which is typically handled inadequately by RBC since such controllers usually do not anticipate the impact of future disturbances and control actions. Secondly, an efficient and economically viable HVAC system typically combines an efficient but expensive and slow base load system with a less ecological but cheaper and faster peak load system (Picard & Helsen, 2018). Such hybrid systems include many control variables such as valve openings, pump speeds. Efficiently exploiting the multitude of control options becomes too complex for an RBC. For example, a modern building can be equipped with a heat pump, a gas boiler, concrete core activation, one or more air handling units with cooling and heating capacity, and each zone might have an additional VAV box. In this case each zone can be conditioned by three different emission systems and the heat can be provided by either the heat pump or the gas boiler. In such buildings, it is a challenge to design an RBC that can provide comfort to each zone and which uses the HVAC system optimally (e.g. provide the required heating and cooling with the most appropriate system at each point in time). Moreover, thermal comfort problems often arise due to inadequate rule-based control.

This illustrates that state-of-the-art RBCs are not always capable to cope with the complexity of contemporary thermal systems in buildings. Model Predictive Control (MPC) is a fundamentally different type of controller, that has the potential to solve many, if not all, of the aforementioned problems. MPC is a control strategy that optimizes a system’s control inputs using a computer model of the system such that a cost
function (e.g. energy use) is minimized. Constraints can be enforced to bound the allowed solution space of the optimization problem.

In the literature, two types of MPC can be distinguished based on the type of HVAC and the type of building: MPC for light buildings with an air-based HVAC system and high cooling load, and MPC for heavy buildings with a water based HVAC system (e.g. geothermal thermally activated building systems - GEOTABS building). The former MPC focuses on saving energy by running the cooling machines at their optimal operating points by optimizing the supply air temperature and mass flow rates to the machines and to the zones. The MPC also saves cost by exploiting variable electricity prices and by shaving peak loads. The building controller model is typically very simplified and obtained by black or grey-box system identification and the MPC formulation is often non-linear. Due to the low mass content of the building and the fast reaction of the HVAC, optimal load shifting is not done at the building level but rather at a central storage tank level when available. Examples can be found in (Bengea et al., 2014; Ma et al., 2012; Risbeck et al., 2015; Gruber et al., 2014; Braun, 2003; Braun & Lee, 2006a; Armstrong et al., 2006; Zaheer-Uddin & Zheng, 2000). The latter MPC type focuses on saving energy and improving thermal comfort by optimally using the inertia of the building. The thermal comfort range and the building inertia are used to shift thermal loads, to maximize the use of inexpensive energy sources like solar gains or passive cooling, and to use slow reacting HVAC systems like TABS in an efficient way. The following paragraphs focus on this second type of MPC applied to (hybrid) GEOTABS buildings.

The energy use, energy cost and thermal discomfort saving potential of MPC for GEOTABS buildings has been investigated both in simulation environments and in real buildings. Sourbron et al. (2013b) considered two zones of a typical office building conditioned by a TABS and a ventilation system for which they developed an MPC controlling the TABS supply temperature while the ventilation was controlled by RBC. The MPC controller model was a second order resistive-capacitive (RC) model whose parameters had been obtained by grey-box system identification and the building model was developed in TRNSYS. Simulation results showed savings of 15% of the energy use compared to RBC. Oldewurtel et al. (2012) and Gyalistras & Gwerder (2009) investigated the MPC savings potential for office buildings by simulating different versions of a 12th order RC model (different orientations, construction types, building standards, window area fractions, internal gains levels, HVAC systems and climates were considered). The MPC optimized the operation of the blinds, the ventilation, the TABS and the supplementary emission system for one year. They found that for about 50% of the investigated building variants, MPC could save more than 40% of the non-renewable energy use. These high energy savings are an over-estimation of the real possible savings as the controller and the building models were identical (no model mismatch and perfect disturbance prediction) and they were relatively simplified. Sturzenegger et al. (2013) used a similar white-box controller model to control the HVAC of a real office building of 6000 m² floor area. Field tests showed that the implemented MPC could save 17% of the annual energy use. The MPC optimization variables were the heating and cooling powers delivered to the TABS, the solar transmission through the windows (blinds), the air flow through the recovery wheel or through its by-pass, and the flow through the ventilation heating and cooling coils. The resulting Optimal Control Problem (OCP) was bi-linear in both its inputs and its states. Vição et al. (2014) developed an MPC controlling the TABS of a 3000 m² real building. Experiments during the heating season showed energy savings of 17%. The controller model was an 8th order model representing three thermal zones (one per floor) and its parameters were obtained by means of system identification. As the controller model had been identified using only winter measurement data, the MPC was only used for the heating season. Privara et al. (2011) also proposed an MPC to control the TABS power of a large university building during the heating season. The controller model was obtained by subspace black-box model identification and savings between 17 and 24% were found. Finally, De Coninck & Helsen (2016) obtained more than 30% energy saving in a Belgian office building in Brussels during winter months.

Despite numerous demonstrations of MPC in buildings, commercially exploiting this potential is difficult due to the large amount of work incurred by setting up the MPC controller model (Sturzenegger et al., 2014; Cigler et al., 2013). Moreover, we lack a systematic approach that can be applied to any building, without requiring expertise of both building energy simulation and optimization. In order to address these difficulties, TACO, a Toolchain for Automated Control and Optimization (Jorissen, Boydens, & Helsen, 2018; Jorissen, 2018; Jorissen & Helsen, 2016), has recently been developed. TACO automatically generates a Non-Linear
Problem (NLP) MPC code from a Modelica building model such that the toolchain user is only exposed to a software with the complexity of a typical building energy simulation tool. The Modelica building models are developed using the IDEAS Modelica library, which has been verified using BESTEST and the TwinHouse experiment (Jorissen, Reynders, et al., 2018). TACO has been demonstrated for Solarwind, a GEOTABS office building (Jorissen, 2018), where electrical energy savings of 82 % are projected. However, the toolchain has only been demonstrated using simulations and therefore has to be complemented with additional software and hardware components for practical implementations. The purpose of this paper is therefore to provide a comprehensive overview of the steps that are proposed for implementing an MPC using TACO. We therefore summarise existing work and further outline future implementation plans of MPC in the Infrax office building in Brussels.

The structure of the paper is as follows: firstly, Section 2 describes the Infrax office building and building simulation model. Section 3 then presents TACO and the MPC controller model of Infrax. Finally, Section 4 proposes a software implementation that couples the MPC to an existing BACnet network.

2. BUILDING AND BUILDING MODEL DESCRIPTION

The proposed methodology is applied to an office building in Dilbeek, near Brussels (Belgium), which is illustrated in Figure 1. This section first presents the relevant aspects of the building and then explains a simulation model that has been developed to test the developed MPC and to benchmark it against the building RBC. This model is also modified in Section 3.2 to serve as controller model for the developed MPC.

2.1 Building

The Infrax building has 2232 m² of floor space spread over 4 floors. This office building contains open-plan offices, cellular offices and meeting rooms. The U-values for the outer walls and roof are between 0.18-0.25 and 0.14-0.15 W/(m².K) respectively. The windows have double glazing with a U-value of 1.0 W/(m².K) and g-values between 0.45-0.49. The air-tightness of the building is measured with a n-value of 1.3 ACH.

Solar gains can be controlled by means of movable horizontal fin shading on the 3rd and 2nd floors, and fixed horizontal fin shading combined with overhangs on the 1st and ground floors. On the building roof 61 PV-panels of 136 Wp (8.3 kWp) are installed and connected to the grid.

Figure 2 shows the hydraulic scheme of the building. Thermal energy is produced by means of 2 geothermal heat pumps of 70 kWth each. The source-side of the heat pumps is connected to a geothermal borefield that is composed of 38 vertical double-U boreholes of 94 m each. The sink-side of the heat pumps is connected to a 2500 l water storage tank and to a main collector, where heat is distributed to multiple distribution loops. In an initial stage, the building was designed to store and cool several server units. Consequently, a 150 kW cooling tower is installed on the source-side to relieve the cooling load of the borefield when needed. Due to the low demand of domestic hot water, a small electrical boiler is installed.

The building has a hybrid emission system that is composed of TABS as a hydronic slow-reacting base load.

---

1 The thermal balance of the borefield is not maintained for this implementation.
system and of an air handling unit (AHU) that can heat or cool as an air-based, fast-reacting secondary system. The AHU has a nominal supply volumetric flow rate of 10 000 m$^3$/h and a nominal extraction volumetric flow rate of 8850 m$^3$/h. The AHU is equipped with heating and cooling coils of 34 kW and 59 kW respectively, and a heat recovery wheel placed between these coils. The air is distributed over the zones of the building, of which some are equipped with Variable Air Volume (VAV) boxes. Air can be re-heated if needed by using heating coils inside the ducts.

For heating, the aforementioned collector distributes the thermal energy to 4 different heating loops: to the VAV heating coils in the air distribution system (45 kW), to the heating coil of the AHU (34 kW), to the TABS system (60 kW) and to the cooling tower (150 kW) in case of dissipation. For cooling, the source side is connected through counterflow plate heat exchangers to the TABS system (90 kW), and to the cooling coil of the AHU and to the server units (106 kW).

2.2 Building model

A white-box model of the building including the building envelope, the HVAC system, and the occupancy has been developed using the Modelica modelling language combined with the IDEAS (Jorissen, Reynders, et al., 2018) and Buildings (Wetter et al., 2014) libraries. Technical sheets were used to configure the component model parameters, and schematics determine the component interconnections.

The building envelope model is composed of 27 zones, of which 21 are conditioned. The 1$^{\text{st}}$, 2$^{\text{nd}}$ and 3$^{\text{rd}}$ floors are mainly open offices and separate zones exist for the north and south spaces, the individual meeting rooms and the bathrooms (not conditioned). The ground floor includes individual conference rooms and several facilities (first aid, cafeteria, storage and server rooms).

The HVAC model includes all important hydraulic components of the building, which are illustrated in Figure 2, and thus allows to control the model up to the component level. The model includes the pressure drops in the main circuits to correctly model the pressure driven flows and to compute the electrical power used by the circulation pump based on similarity laws (Wetter, 2013; Wetter et al., 2015). The valve models compute the pressure drop as a quadratic function of the mass flow rate and the flow-coefficient. This flow-coefficient depends on the control signal and on the valve characteristic. The heat pump model consists of a simplified vapor compression cycle with parameters that are calibrated from manufacturer data, as described by Cimmino & Wetter (2017). The borefield model allows simulating both the short and the long term thermal response (Picard & Helsen, 2014).

A rule-based controller was also developed based on the technical description of the building. The first
heat pump works with an hysteresis controller to keep the storage tank temperature at 32°C. If the desired temperature is not achieved within 10 minutes, the second heat pump is activated. The cooling tower control is adapted to dissipate heat from the storage tank or to split the load in the borefield in some scenarios. The TABS and the secondary system are operated independently. The TABS has a building climate mode that is computed each day, whereas the secondary system is turned on only during office hours. The supply temperature to the TABS is controlled using a three-way valve during heating mode or using the cooling heat exchanger during cooling mode. The supply mass flow rate of the TABS is controlled using four two-way valves, one for each floor. During the neutral mode, water circulates through the TABS in a closed loop. The AHU supply temperature set point is computed using a heating curve. When additional heating is required, the VAVs are opened to a zone-dependent fixed position and the heating coils supply heat up to a desired supply temperature set-point. When cooling is required, the heating coils are not used and the VAVs control the supply air mass flow rate. The building also includes control features for night ventilation and shading.

3. MPC METHODOLOGY

This section describes how the MPC is implemented, for which we rely on TACO, a Toolchain for Automated Control and Optimization (Jorissen, 2018; Jorissen, Boydens, & Helsen, 2018). The MPC methodology is first summarised in Section 3.1, after which its application to the Infrax model is presented in Section 3.2.

3.1 TACO

TACO is a toolchain that is derived from JModelica (Åkesson et al., 2010) and that converts a Modelica model into an MPC code and executable. The executable evaluates the objective and constraint values and derivatives using CasADi (Andersson et al., 2012), which can then be optimized using IPOPT (Wächter & Biegler, 2006). TACO uses a problem formulation that is tailored to the mathematical structure that building models typically have. This structure is illustrated in Figure 3a, which shows that three main equation groups exist.

The first group of equations are the boundary conditions, which depend on time only. I.e. they do not depend on state or on optimization variables. Since these variables and equations depend on time only, their values can be pre-computed before starting the MPC optimization. Therefore TACO automatically identifies what equations are a function of time only and generates an FMU from these equations. The FMU generation toolchain of JModelica supports algorithm sections and external C functions such as the Modelica data reader (CombiTimeTable) implementation such that (weather) data readers are supported. Moreover, since the boundary condition values are fixed during one optimization, these equations can be discontinuous
functions of time.

The second group of equations are the building dynamics that correspond to the heat transfer within the building envelope. We require that the building dynamics are linear such that linear algebra can be used to efficiently pre-compute the building dynamics. MPCs for buildings nearly always use linear (RC) envelope models such that this is common practice. The building envelope component models of the IDEAS Modelica library (Jorissen, Reynders, et al., 2018) have been parameterised such that all heat transfer equations can be linearised (Picard et al., 2015).

The third group of equations consists of the HVAC equations. These equations should be steady state and they should be twice continuously differentiable ($C^2$) with respect to the optimization and (building envelope) state variables. Integer decision variables are not yet supported.

The objective function and any number of inequality constraints can be defined as long as they are twice continuously differentiable functions of the state variables and the optimization variables. Jorissen (2018); Jorissen, Boydens, & Helsen (2018) describe in more detail how these equation groups are treated and integrated into a single efficient numerical code. Note that TACO automatically identifies what variables belong to what group.

3.2 Application to Infrax model

To be able to serve as the controller model, the Infrax simulation model is thus slightly modified by linearising the building dynamics and by removing HVAC dynamics. Furthermore, the algorithm section contained by the cooling tower model is currently not supported. Therefore it is excluded from the optimization by removing it from the optimization model implementation. The two on/off heat pumps are replaced by a single modulating heat pump with a maximum thermal power of 140 kW. Since integer decision variables are currently not supported, the thermal power can be controlled continuously up to 140 kW. Similarly, many on/off pumps are replaced by modulating pumps. Open/closed two-way valves that create a connection between the TABS circuit and the cooling or the heating circuit, are replaced by a single, continuously operated three-way valve that allows the MPC to switch between the two circuits. For these optimization variables, a post processing has to be defined that maps the continuous optimization variable into the boolean control signal. For pumps we therefore check whether the pump circuit is being used. E.g. when a heat exchanger in the pump circuit has a nonzero heat flow rate, or when a three-way valve in the pump circuit is not fully closed, the pump is enabled. For the heat pumps a hysteresis controller tracks the evaporator supply water temperature that is computed by the MPC. The cooling tower is not used, its pumps are disabled. For the two-way valves a 50% valve opening is chosen as a threshold for connecting that TABS circuit to the cooling or the heating circuit. Finally, pressure drops of many pipes are neglected such that no non-linear algebraic loops are formed. Pressure drops of the TABS and VAVs are computed. For an example of the type of simplifications that are applied, see Chapter 5 and Appendix A of (Jorissen, 2018).

Furthermore, constraints are defined that ensure a maximum temperature difference of 4 K across the heat pump evaporator and condenser, and that constrain the condenser heat flow rate to be positive and smaller than 140 kW. Zone temperatures are constrained between 21.5°C and 24.5°C.

The objective is to minimize the electrical power use of the building, which consists of all fan and pump powers and the heat pump compressor powers. For more details with respect to the mathematical implementation of these models we refer to Appendix A of (Jorissen, 2018).

The RBC and the MPC can be compared using the same simulation model. The MPC uses perfect state updates\(^2\). The MPC control signals are either directly applied to the model, or they are post-processed as outlined above. For the air handling unit cooling coil a PI controller is used to track the supply air temperature that is computed by the MPC. I.e. the MPC-computed cooling coil valve position is not applied directly, but implicitly by tracking the optimal supply air temperature that is computed by the MPC. The resulting software and model interdependencies are shown in Figure 3b.

\(^2\)All state variable values are read from the model and used to update the MPC internal state.
4. PRACTICAL IMPLEMENTATION

This section describes the original hardware configuration used by the RBC controller and the planned software architecture to allow the MPC to take over the control of the building.

The Infrax building is operational since 2011 and is controlled by an RBC and a Building Management System (BMS) implemented on HX controllers from company Priva (Priva, 2018). A local windows computer is furthermore connected to the controllers by an ethernet cable to provide a graphical user interface and to run the database on which measurement data is stored. The HX controllers are wire-connected to each controlled element, to each measurement equipment, and connected to each other through the network, as illustrated in Figure 4.

In contrast to Priva’s new products (e.g. Priva Blue ID S-line), HX controllers (Priva, 2018) do not have the option to share the control variables on BACnet/IP, which we intend to use as the communication layer. Therefore, a BACnet router (SX100L + TC8/510) was recently added allowing access to all control variables. The implemented RBC writes its control values on the lowest BACnet priority (priority 16). We control each valve, circulation pump, heat pump (on/off), register, compressor, etc., by simply writing a control signal on a higher BACnet priority than the RBC. We use priority 15 for this purpose. This approach has the advantage that the original RBC stays operational and it keeps writing its control actions on priority 16. MPC can therefore control a subset of the control variables while leaving the other untouched and the operational control can revert back to RBC by simply writing the value null on priority 15 to all MPC variables.

The MPC master algorithm is composed of three independent python processes.

**PROCESS 1** The first process is used to retrieve and store data from BACnet using a BACpypes application (Bender, 2018) and from a weather forecast website (Darksky team, 2018). The process also contains a BACpypes application which periodically broadcasts a watchdog signal on the BACnet network. The signal is read by an application running on the Windows computer. If the watchdog signal is not correctly received, the algorithm running on the Windows computer will reset the value of priority 15 of all controlled variables in order to return the control of the building to the original RBC.
PROCESS 2 is the core of the algorithm. **Every hour**³, at a predefined timestamp $ts$, the following steps are performed. Firstly, the weather forecast is read for the coming days and saved to a json file $ts_{\text{weatherforecast.json}}$ where $ts$ is the predefined timestamp. Secondly, a state update is performed using `Estimator.exe`, the measurement data from the database and the previous control actions $ts_{\text{umpc.json}}$. The estimation value of the states is written to $ts_{\text{initialStates.json}}$. Finally, the new control actions are computed by `MPC.exe` using the latest weather predictions and state estimation. The results are saved to $ts_{\text{umpc.json}}$, which contains the control actions of the coming 3 days at a sampling rate of 15 minutes. Each control action is saved in the json file including the timestamp at which it should be applied.

PROCESS 3 deals with the actual control of the valves and other HVAC components of the building. Every 15 minutes, at predefined timestamps, the folder containing all $ts_{\text{umpc.json}}$ is searched and the control values corresponding to the current timestamp are read from the latest $ts_{\text{umpc.json}}$. Since the MPC model does not directly compute set points for all control points (e.g., valves) of the system, the first step configures the binary valves such that the MPC control actions can be performed. Only when each binary valve is in its correct position, the MPC control actions are post-processed to actual control signals by calling `CTRL.fmu`, which implements the post-processing of the MPC controller, such as PI controllers. The FMU is evaluated with a sample time of one second until new MPC control actions are read. Since a large number of BACnet variables are written and read and as a failed BACnet request leads to a non-negligible timeout, the read and write requests are distributed over a pool of several threads.

Note that the developed software architecture is resistant to a failure of the blocks of PROCESS 2 because the different json files contain the required information for multiple hours. When a json file is not generated then the previous version can be used. The processes are called periodically and are closed again afterwards, which reduces the chance of memory leaks or other long term robustness problems of the developed software. Moreover, since all json input files will be stored, errors can be reproduced easily.

### 5. CONCLUSIONS

This paper presents a comprehensive overview of the steps that are proposed for real-life implementation of an MPC using TACO, a Toolchain for Automated Control and Optimization. We therefore summarise current work and further outline future implementation plans of MPC in the Infrax office building in Brussels. Firstly the building is described and a building simulation model is explained. Then main features of TACO are presented and we explain how the simulation model is modified to obtain an MPC controller model. TACO is used to generate an MPC from the controller model. The generated MPC has to be coupled to an existing BACnet communication infrastructure, for which an implementation is finally proposed. This implementation is designed such that it is simple, yet robust. Furthermore, the implementation is designed to be easy to debug since the master algorithm’s intermediate variables are stored as json input files. The presented methodology will serve as the basis for future implementations of MPC that are planned within the EU-Horizon 2020 hybridGEOTABS project.

### REFERENCES


³The interval is user-defined.


**ACKNOWLEDGMENT**

The authors gratefully acknowledge the funding of their research work by the EU within the H2020-EE-2016-RIA-IA programme for the project ‘Model Predictive Control and Innovative System Integration of GEOTABS;-) in Hybrid Low Grade Thermal Energy Systems - Hybrid MPC GEOTABS’ (grant number 723649 - MPC;- GT). The authors also thank Priva for providing they BACnet router for free and for they support.

5th International High Performance Buildings Conference at Purdue, July 9-12, 2018