Semi-automated Thermal Envelope Model Setup for Adaptive Model Predictive Control with Event-triggered System Identification

Lu Wan[1,2], Xiaobing Dai[3], Torsten Welfonder[1], Ekaterina Petrova[2], Pieter Pauwels[2]

[1]Robert Bosch GmbH, 71272 Renningen, Germany

[2]Group of Information Systems in the Built Environment,
Eindhoven University of Technology, 5600 MB Eindhoven, Netherlands

[3]Chair of Information-oriented Control, Technical University of Munich, 80333 Munich, Germany

Abstract

To reach carbon neutrality in the middle of this century, smart controls for building energy systems are urgently required. Model predictive control (MPC) demonstrates great potential in improving the performance of heating ventilation and air-conditioning (HVAC) systems, whereas its wide application in the building sector is impeded by the considerable manual efforts involved in setting up the control-oriented model. To facilitate the system identification (SI) of the building envelope as well as the configuration of the MPC algorithms with less human intervention, a semantic-assisted control framework is proposed in this paper. We first integrate different data sources required by the MPC algorithms such as the building topology, HVAC systems, sensor data stream and control settings in the form of a knowledge graph and then employ the data to set up the MPC algorithm automatically. Moreover, an event-triggered SI scheme is designed, to ensure the computational efficiency and accuracy of the MPC algorithm simultaneously. The proposed method is validated via simulations. The results demonstrate the practical relevance and effectiveness of the proposed semantics-assisted MPC framework with event-triggered learning of system dynamics.

Highlights

- Semantic web technologies
- Ontology-based data integration
- Adaptive model predictive control
- Event-triggered system identification

Introduction

The building sector takes up about 40% of the primary energy consumption and the greenhouse gas emissions worldwide (IEA, 2015), more than half of which occurs during the operational stage. heating, ventilation and air-conditioning (HVAC) systems account for a large part of the building's total energy use during the operation, inducing considerable CO_2 emission and monetary costs. Due to the increasing need for space cooling and heating globally, smart and economical control strategies are required for HVAC systems to decrease the carbon footprint.

Model Predictive Control (MPC) is a control method that uses building models and disturbance forecasts to solve constrained optimization problems in a dynamic manner (Drgoňa et al., 2020). It has gained increasing attention in building control these years due to its effectiveness in energy cost reduction and energy efficiency improvement (Eydner et al., 2022). The control-oriented prediction model for MPC includes the dynamic of the thermal envelope and HVAC systems. In this study, we mainly deal with thermal envelope modeling, where Resistance and Capacitance (RC) models are widely employed (Atam and Helsen, 2016). The lumped parameters of RC models, characterizing the heat transfer of buildings, are usually estimated through data-driven system identification (SI). In recent studies, adaptive SI methods are often employed, which update the RC model parameters regularly, performing moving horizon estimation (MHE) or model identification at a daily frequency (Arroyo et al., 2020; Blum et al., 2022). Because the performance of MPC deteriorates if the estimated RC parameters are inaccurate. We adopt an MPC algorithm with event-triggered model identification, to reduce the computational effort caused by frequent SI.

The wide application of MPC algorithms is however a challenge in the building sector. Because the control-oriented model configuration requires the interpretation of different data sources such as building geometry, HVAC systems, and sensor measurement (Blum et al., 2022), demanding different domain experts' knowledge. Furthermore, the MPC algorithm requires forecasts on disturbances caused by e.g.weather and occupancy, and appropriate hyper-parameter setting e.g. prediction horizon. As buildings have miscellaneous geometry, building energy systems, and geological locations, such a model-based approach is labor-intensive and difficult to transfer among buildings. Most of the previous studies have focused solely on the automatic control model setup for optimal structure and parameters, but with little regard to the practical implementation efforts required by both SI and MPC algorithms. For instance, De Coninck et al. (2016) has developed a python toolbox to identify the RC model for the building envelope, which is also deployed in (Arroyo et al., 2020) to automatically determine the optimal model structures for multi-zone buildings. However, the meta information required by the toolbox is manually configured.

Andriamamonjy et al. (2019) propose a tool-chain to generate RC model in Modelica, which automatically uses geometric data in the building information model (BIM) and monitoring data from the building management system (BMS). This study is insightful, but the connection between BIM model and BMS is established in a hard-coded manner and the implementation depends highly on the proprietary commercial software Revit. In Blum et al. (2022), a comprehensive open-source toolchain is developed that can generate automatically the MPC algorithm (including both model parameter estimation and optimal control formulation). Nevertheless, the data integration and interpretation process involved in the MPC algorithm generation is implemented specifically for the case study. Thus, the developed infrastructure is hard to transfer to another building. An integrated framework that collects and interprets the diverse data required by the MPC needs to be studied, in order to prompt the MPC applications.

There is no mature common platform yet to keep the heterogeneous data well-connected in the architecture, engineering, and construction (AEC) industry. Semantic technologies are considered promising to solve the data silo dilemma (Pauwels and McGlinn, 2022). Many previous attempts use Industry Foundation Classes (IFC) (Borrmann et al., 2018), which is an open BIM schema for data exchange in the AEC domain. However, IFC is not suitable for describing dynamic operational data such as sensor measurements, despite its strength in geometry modeling. Semantic web technologies (SWT) [1] enable the exchange and sharing of diverse data sources (semantic graphs) over the web, which is hard to achieve by using classical approaches. Semantic graphs also referred to as knowledge graphs or metadata schema, contain structured information describing the meaning of the underlying data (Fierro and Pauwels, 2022). Built by ontologies that mean a specification of a conceptualization (Gruber, 1993), semantic graphs are able to connect heterogeneous data. A few studies try to integrate the building data from the design and operation stages using SWT. Mavrokapnidis et al. (2021) propose an SWT-based methodology to link static building design data modeled in IFC with dynamic sensor data modeled by Brick Schema (Balaji et al., 2018). Building topology, product, and sensor data are connected using these two schemata for further exploitation. A similar approach is adopted in (Chamari et al., 2022) to integrate the IFC data and BMS sensor data, by first converting different data sources into knowledge graphs and then linking them together. They build a vendor-neutral web application to visualize the 3D BIM model and the spatial-related sensor measurements in a common platform. The ontology-based data integration can enable more efficient data-driven applications via properly linked data sources. However, to the best of the authors' knowledge, no existing study has investigated a semantics-aware framework for advanced building control, such as MPC.

In this paper, we propose a semantic-assisted control framework to support MPC applications in buildings. In the proposed framework, the data from diverse sources, including building geometry, building physics, and sensors, is first collected and managed in a machine-interpretable way and then employed to set up the algorithm automatically. Furthermore, an MPC algorithm with event-triggered SI is designed and implemented, in order to minimize operating costs with desirable indoor temperatures. The effectiveness of the proposed framework and algorithm is demonstrated through simulations.

The paper is structured as follows. In the section System modeling, we introduce the building and the HVAC system. In the section Semantic-assisted architecture for MPC, we explain the proposed control framework and elaborate on the designed MPC algorithm with event-triggered SI. The results section shows the effectiveness of the proposed approach via simulations. Finally, conclusions are drawn with remarks on future research.

System modeling

In this paper, we study a typical system of a European office building, consisting of a one-zone building envelope equipped with a Variable Air Volume (VAV) flow system and a radiator heating system. The physical model of the system is adapted from *"Buildings.Examples.ScalableBenchmarks.BuildingVAV.One_Floor_OneZone"* provided in the open-source Modelica Buildings library (Wetter et al., 2014). The existing control logic in the model is selected as the baseline, i.e., a rule-based controller (RBC) designed according to ASHRAE (2005), and compared with the proposed MPC algorithm in the result section.

Building model description

The BESTEST Case 600 (ASHRAE, 2007), a benchmark model for building energy simulation, is used in the study. The envelope consists of a single zone with a window on the south facade and a constant infiltration mass flow rate. A BIM model for the building envelope is manually created. The building represents a typical small-group office located in Stuttgart, Germany. Internal heat gains q_{int} and occupancy table are set according to the local standard (DIN, 2015). Moreover, bounds for desired zone temperature T_z are adjusted according to the occupancy. In particular, for the occupied time $t \in \mathbb{O}$ (8:00 to 18:00) during weekdays, the upper and lower bounds of the zone temperature are set as $T_{max}^{occ} = 27^\circ C$ and $T_{min}^{occ} = 21^\circ C$. For unoccupied time $t \notin \mathbb{O}$, the temperature bounds are relaxed to $T_{max}^{un} = 32^\circ C$ and $T_{min}^{un} = 17^\circ C$, respectively. The air in the thermal is assumed to be well-mixed.
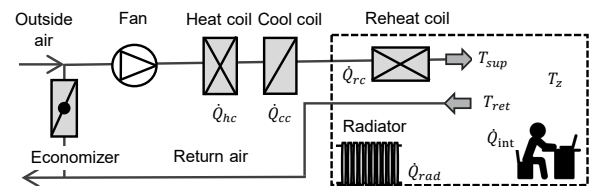

Figure 1: Schematics of the HVAC system

The building is controlled by the system as in Figure 1, in which the VAV system consists of a heating coil, a cooling coil, and a reheat coil with maximal power $\dot{Q}_{max}^{hc}$, $\dot{Q}_{max}^{cc}$ and $\dot{Q}_{max}^{rc}$, respectively. Moreover, an economizer is located between the main supply branch and the return branch. In the zone, the radiator heating system is deployed with a maximal power $\dot{Q}_{max}^{rad}$. The HVAC components are modeled as ideal devices with constant overall efficiency, which considers the energy loss in the hydraulic distribution system and the generator system by using a discount coefficient. The efficiencies are η_{hc}, η_{rc}, η_{rad} and COP_{cc}, respectively. The total thermal power delivered by the HVAC system $\dot{Q}^{hvac}$ to the zone is as in Eq. (1):

$$\dot{Q}^{hvac} = \boldsymbol{u}^T \boldsymbol{\Gamma} \dot{\boldsymbol{Q}}_{max} \quad (1)$$

where $\boldsymbol{u}^T = [u^{cc}, u^{hc}, u^{rc}, u^{rad}]$ refer to the normalized power matrix of the component (control variable), with values between 0 and 1. The efficiency matrix $\boldsymbol{\Gamma}$ is set as $\boldsymbol{\Gamma} = \text{diag}(COP_{cc}, \eta_{hc}, \eta_{rc}, \eta_{rad})$, while maximal power matrix set as $\dot{\boldsymbol{Q}}_{max}^T = [\dot{Q}_{max}^{cc}, \dot{Q}_{max}^{hc}, \dot{Q}_{max}^{rc}, \dot{Q}_{max}^{rad}]$.

Control problem setting

The control objective is to achieve low operating costs while satisfying the requirements for the desired zone temperature. For this purpose, the MPC controller is designed with a first-order RC model for prediction. The continuous time 1R1C model is expressed as in Eq. (2)

$$C_z \dot{T}_z = R_w^{-1}(T_{amb} - T_z) + \dot{Q}^{hvac} + q_{int}A + \alpha H_{glo} \quad (2)$$

where T_{amb} stands for the ambient temperature, H_{glo} for the global horizontal irradiation, and A for zone floor area. The parameter is set as $\boldsymbol{\theta} = [C_z, R_w, \alpha]^T$, a collection of the heat capacity C_z, thermal resistance of the walls R_w and solar irradiance coefficient α.

The lumped parameter $\boldsymbol{\theta}$ is identified using data-driven methods with historical measurements as the training data. SI aims to find the parameters that minimize the difference between the true state and prediction, defined in Eq. (3):

$$\hat{\boldsymbol{\theta}} = \arg\min_{\hat{\boldsymbol{\theta}}} J^s = \int \varepsilon^2(t, \hat{\boldsymbol{\theta}}) dt \text{ s.t. } \hat{\boldsymbol{\theta}} \in \boldsymbol{\Theta} = [\underline{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}] \quad (3)$$

where $\varepsilon(t, \hat{\boldsymbol{\theta}})$ represents the deviation of RC model prediction compared with measured data. Note that for the optimization, the proper setup of initial guess $\hat{\boldsymbol{\theta}}_0$ and boundaries $\boldsymbol{\Theta}$ are necessary for plausible results. As the real parameter $\boldsymbol{\theta}$ is time-varying, repeated SI is needed but induces more computational effort. To reduce the computational burden, we design the event-triggered SI strategy, updating the parameter via Eq. (3) only when the model error is large. The detail is discussed in section MPC service with event-triggered SI.

As SI described above and the MPC implementation afterward require both inputs from different sources, such as a proper setup of initial guess and continuous state measurements, this leads to the demand for an integrated data framework. We endeavor to integrate building design data (IFC file) and operational data (sensor measurements), in order to reduce the manual efforts required for the RC model set-up and the control algorithm. A semantic-assisted architecture is proposed, to realize the semi-automated setup of the proposed MPC algorithm with event-triggered SI.
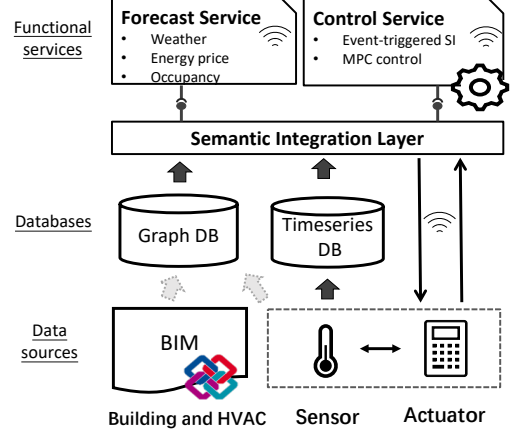
Semantic-assisted architecture for MPC

Figure 2: The architecture of the semantic-assisted framework for MPC with even-triggered SI

The proposed semantic-assisted framework adopts a layered service-oriented architecture as illustrated in Figure 2. Starting at the bottom, design data (BIM model) and operation data (data points of sensors and actuators) are first pre-processed and then delivered to the graph database and time-series databases accordingly. In the integration layer, semantic graphs for different chunks of entities e.g. buildings, HVAC systems, sensors and actuators are connected with each other and the overall information is integrated. The link between the virtual sensors in the graph DB and their measurements in the time-series DB is realized via the sensor ID. Via the semantic layer, the building design data and its operational data are seamlessly combined. Eventually, the functional service layer communicates with the semantic integration layer, exchanging corresponding data to execute the services. Currently, there are two services implemented, namely the forecast and control services.

As for the software implementation, all the components except the MPC service are implemented in Python due to its versatile packages available. The manipulation of the semantic graphs is realized using RDFLib [2]. We use MATLAB to develop the MPC service because of its powerful numeric solvers available.

The real-time communications between the semantic layer and the databases, and between the field measurements and time-series database are via RESTful-APIs, which rely on HTTP protocol and are secure. The communication between the semantic layer and the services, as well as between the semantic layer and the field actuators, on

---

[2]https://rdflib.readthedocs.io/en/stable/index.html

the other hand, is realized via User Datagram Protocol (UDP), which is based on the TCP protocol and has the advantage of fast communication speed. In this way, the field devices (i.e. sensors and actuators) and the MPC service form a closed-loop system with a local area network.

Individual components are specifically explained in the following subsections.

Data sources and generation of graphs

In this study, we take IFC file as the information source for data about envelope thermal properties, building geometry, and topology. The semantic graph for building-related information is automatically generated by the IFCtoLBD convertor (Oraskari et al., 2023). Since we use a simulation building with fictive HVAC systems, the graphs for the HVAC system and BMS data points are currently manually configured using Resource Description Framework (RDF) schema. Note that the RDF graph generation for these sources can theoretically also be automated via tools described in (Pauen et al., 2022) and (Chamari et al., 2022), which will be handled in the future study.

Databases

The generated semantic graphs for the building, HVAC system, and the metadata about sensors as well as actuators are stored in graph DB, because graph DB is fast in querying the relationships between entities. The schema used in the semantic graphs is detailed in the next section. On the contrary, the sensor measurements are stored in a time-series database, to enable more potential time-series analyses. We use GraphDB [3] and InfluxDB [4] for the specific implementation.

Semantic integration layer

In this subsection, we explain in detail how the semantic graph is modeled in terms of the adopted terminologies (T-box). This is fundamental for ontology-based data integration. Based on the controller design, we categorize the data and information that requires human inputs into the following five main aspects:

- **Building elements and topology.** The topological information and geometrical information about the thermal zones are required for RC model structure estimation and therefore modeled. In addition, the properties of building elements (walls, windows, doors etc.) such as area, thermal transmittance, solar transmittance and thermal capacitance are needed in calculating the initial guess and bounds of the parameter $\hat{\boldsymbol{\theta}}_0$. The information above is described using Building Topology Ontology (BOT)[5] (Rasmussen et al., 2021) and PROPS Ontology[5] (Bonduel, 2018).

- **Components and topology for HVAC systems.** The properties of the HVAC system components, such as the

nominal power $\dot{\boldsymbol{Q}}_{max}$, and the nominal efficiency $\boldsymbol{\Gamma}$ are modeled as `ssn:Properties`, using Semantic Sensor Network[5] (SSN) ontology (Compton et al., 2012). Furthermore, the interaction between the HVAC systems and zones is also modeled, so that the building envelope model is matched with the corresponding HVAC components efficiently. We use Flow System Ontology[5] (FSO) (Kukkonen et al., 2022) to describe the heat and fluid transfer among zones and HVAC components.

- **Sensor data collection.** The sensor data required by the MPC algorithm includes state measurement T_z. In addition, the set-points i.e., $T_{max}^{(\cdot)}$ and $T_{min}^{(\cdot)}$, are required to setup the constraints in MPC as in Eq. (4b). Such properties (e.g. temperature and power) observed by sensors are described by SOSA ontology[5] (Janowicz et al., 2019). The virtual data points attached to these properties are further modeled using Brick, with their identifier to the time-series measurements in the databases modeled via `brick:hasTimeseriesID`.

- **Forecast information.** Forecasts required by the MPC algorithm include weather, occupancy, and the energy price in the market. The link to the forecast models in the file system is described using SEAS forecasting ontology[5] and SEAS Procedure Execution ontology[5] (Lefrançois et al., 2017).

- **Controller setup.** The proposed adaptive control algorithm has 2 sub-modules as shown in Figure 2, namely MPC and the event-triggered SI. For MPC, the prediction horizon N_c is tuned according to the specific use case. For the event-triggered SI, the optimal setting for trigger horizon N_t and identification horizon N_s, as well as the trigger threshold ρ typically from domain experts' experience. The above hyperparameters for controller setup are represented by SEAS optimization ontology and Time ontology[5] (Hobbs and Pan 2004), and can be modified by domain experts easily.

With the proposed information modeling paradigm, the information needed in building MPC algorithm is organized in a uniform manner, reducing the redundant efforts in data preparation. How the aforementioned information benefits the MPC algorithm setup is detailed in section MPC service with event-triggered SI. A concrete instance model (A-box) is shown in the section Controller setup using the semantic graph.

The data from different sources are integrated and manipulated via the semantic integration layer, which serves as the coordinator between services and other components. The first task of the semantic integration layer is to calculate the initial guess $\hat{\boldsymbol{\theta}}_0$ and boundaries for RC parameters according to the material data, in which $\hat{R}_{w,0}$ is the sum of thermal resistance of all surfaces, and $\hat{C}_{z,0}$ for the total thermal capacitance of the surfaces and air within the zone, and $\hat{\alpha}_0$ for the solar irradiation through the window area. The lower and upper bounds of $\hat{\boldsymbol{\theta}}$ is set as $[0.1\hat{\boldsymbol{\theta}}_0, 10\hat{\boldsymbol{\theta}}_0]$. The second task of the semantic layer is to retrieve the relevant data from the databases and delivery it to the corresponding service, which is detailed in the specific service.

---

[3] https://www.ontotext.com/products/graphdb/

[4] https://www.influxdata.com/

[5] URIs for all used ontologies:
BOT: https://w3id.org/bot#; Brick: https://brickschema.org/schema/Brick#; PEP: https://w3id.org/pep/; FSO: https://w3id.org/fso#; PROPS: https://w3id.org/props#; SEAS forecasting ontology: https://w3id.org/seas/ForecastingOntology; SOSA: https://www.w3.org/ns/sosa/; SSN: https://www.w3.org/ns/ssn/; TIME: http://www.w3.org/2006/time/.

Functional service layer

Forecast service

The forecast service provides predictions on the weather environment, energy price and the occupancy of the building. We use the predictions made by existing models and store them in the file system. Test reference year weather data set[6] of Stuttgart is employed. We adopt the day-ahead German electricity price[7] for price forecast. The occupancy profile as defined in the norm (DIN, 2015) is employed for predicting occupancy and internal gains. The SPARQL [8] query against prediction-related information is demonstrated in Figure 3, running in the forecast service.

```
PREFIX seas: <https://w3id.org/seas/>
PREFIX weat: <https://bimerr.iot.linkeddata.es/def/weather#>
PREFIX pep:  <https://w3id.org/pep/>
SELECT DISTINCT ?forecast  ?filePath
WHERE
{
{  ?forecaster seas:forecasts :thermal_zone_observed. }
UNION
{  :thermal_zone_observed.  weat:locatedIn ?geolocation.
   ?forecaster seas:forecasts ?geolocation.  }
?forecaster a seas:Forecaster;
           pep:implements ?forecast.
?forecast pep:hasSimpleResult ?filePath.}
```
Figure 3: SPARQL query in the forecast service to retrieve geological local related and zone-specific forecasts

MPC service with event-triggered SI

The proposed algorithm in the control service consists of 2 sub-modules, namely MPC and event-triggered SI.

The MPC module is formulated as an optimization problem, aiming to minimize the operating costs of the HVAC system with desired zone temperature. The optimization problem with receding horizon N_c is written as Eq. (4):

$$\min_{\{u_i, \bar{s}_i, \underline{s}_i\}} = J_k^c = \sum_{i=k}^{k+N_c-1} (\hat{\lambda}_i \mathbf{\Gamma} \dot{\mathbf{Q}}_{max}^T \hat{\mathbf{u}}_i + \bar{\mu}\bar{s}_i + \underline{\mu}\underline{s}_i) \quad (4a)$$

$$\text{s.t. } T_{min}(t_i) - \underline{s}_i \leq \hat{T}_{z,i} \leq T_{max}(t_i) + \bar{s}_i \quad (4b)$$

$$\hat{u}_i^{cc} \in [0,1], \hat{u}_i^{hc} \in [0,1], \hat{u}_i^{rh} \in [0,1], \hat{u}_i^{rad} \in [0,1] \quad (4c)$$

$$\underline{s}_i \geq 0, \bar{s}_i \geq 0, \forall i = k, \cdots, k+N_c \quad (4d)$$

$$\hat{T}_{z,i+1} = f(\hat{T}_{z,i}, \hat{\mathbf{u}}_i, \hat{\mathbf{e}}_i, \hat{\boldsymbol{\theta}}_i), \hat{T}_{z,k} = T_{z,k} \quad (4e)$$

where $\hat{\lambda}_i$ is the predicted electricity price, $\dot{\mathbf{Q}}_{max}$ is the maximal thermal power, and $\hat{u}_i$ is control variables. The slack variables $\underline{s}_i$ and $\bar{s}_i$ stand for lower and upper bounds, and serve to guarantee the feasibility of the objective function by relaxing the hard constraints on T_z as equation Eq. (4b), with the penalty coefficient μ for violations. Eq. (4b) and Eq. (4c) are the constraints on the state $T_{z,i}$ and control variable $\mathbf{u_i}$, respectively. Eq. (4e) is the discrete form of Eq. (2) with the estimated parameter $\hat{\boldsymbol{\theta}}_k$.

The estimated parameter is updated in an event-triggered way, i.e., $\hat{\boldsymbol{\theta}}_k = \gamma_k \hat{\boldsymbol{\theta}}_k^* + (1-\gamma_k)\hat{\boldsymbol{\theta}}_{k-1}$, where the binary variable γ_k indicates whether the parameter is updated. The

newly estimated parameter $\hat{\boldsymbol{\theta}}_k^*$ is obtained from historical data with length N_s through the optimization as Eq. (5):

$$\hat{\boldsymbol{\theta}}_k^* = \arg\min_{\hat{\boldsymbol{\theta}}} J_k^s = \sum_{k-1}^{k-N_s+1} \xi_i \varepsilon_i^2(\hat{\boldsymbol{\theta}}) \quad (5a)$$

$$\text{s.t. } \hat{R}_w \in [\underline{R}, \bar{R}], \hat{C}_z \in [\underline{C}, \bar{C}], \hat{\alpha} \in [\underline{\alpha}, \bar{\alpha}] \quad (5b)$$

$$\varepsilon_i(\hat{\boldsymbol{\theta}}) = T_{z,i+1} - f(T_{z,i}, \mathbf{u}_i, \mathbf{e}_i, \hat{\boldsymbol{\theta}}), \forall i = k-1; \cdots, k-N_s+1 \quad (5c)$$

where $\underline{R}, \underline{C}, \underline{\alpha}$ and $\bar{R}, \bar{C}, \bar{\alpha}$ are the constant lower and upper bounds of R_w, C_z and α. The indicator γ_k is the result from the event trigger, which is designed by considering N_t previous data as Eq. (6):

$$\text{RMSE}_k = \sqrt{N_t^{-1} \sum_{i=k-1}^{k-N_t} \varepsilon_i^2(\hat{\boldsymbol{\theta}}_k)} > \rho \Leftrightarrow \gamma_k = 1 \quad (6)$$

where $\rho > 0$ is the trigger threshold. The choice of ρ is important but empirical, considering the trade-off between the trigger times and model-induced control accuracy.

According to the equations (4), (5) and (6), the setup of the MPC algorithm requires concrete data from (i) forecasts on electricity price, internal heat gains and ambient climate data, (ii) initial guess for the thermal envelope parameters and the boundaries, (iii) thermal zone and properties the HVAC components connected it, including nominal power and efficiency (iv) the specific control algorithm for the use case, e.g. horizons and threshold (v) the sensor measurements on states of the studied thermal zone, to which the semantic graph corresponds (described in section Semantic integration layer).

Regarding the sources, data (i) can be derived from the forecast service, data (ii) to (iv) come originally from BIM model and BMS and are stored in the graph DB, while data (v) is stored in the time-series DB. To fetch data (i)-(iv), SPARQL queries against the graph DB are sufficient. To retrieve data (v), the appropriate sensor ID needs to be firstly queried using SPARQL; afterward, the sensor ID as well as the time range need to be encoded into the FLUX query, to get the data out of InfluxDB. For illustration, the SPARQL query to retrieve all sensors related to the observed thermal zone is shown as in Figure 4.

```
PREFIX sosa:  <http://www.w3.org/ns/sosa/>
PREFIX ssn:   <http://www.w3.org/ns/ssn/>
PREFIX brick: <https://brickschema.org/schema/1.2/Brick>
SELECT DISTINCT ?property  ?quantity ?ID
WHERE {
:thermal_zone_observed ssn:hasProperty ?property.
?sensor sosa:observes | ssn:forProperty ?property;
        brick:measures ?quantity;
        brick:timeseries ?timeseriesRef.
?timeseriesRef brick:hasTimeseriesId ?ID.}
```
Figure 4: SPARQL query for all sensor points of the zone

With the assistance of the semantic integration layer and the underlying Semantic integration layer, the data required by the MPC service is easily redirected to the correct components and then merged to instantiate the algorithm depending on the specific use case. In this manner, the re-usability of the MPC among buildings is improved.
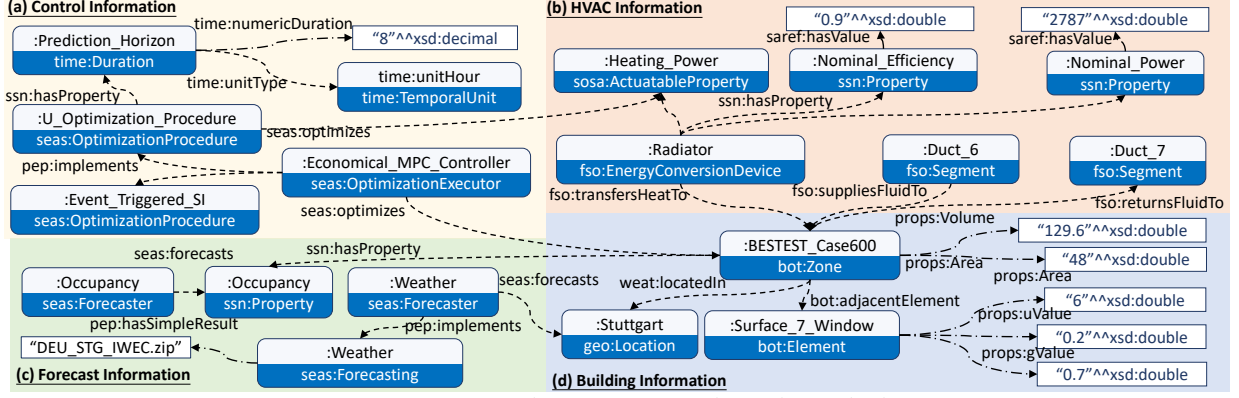
Figure 5: Excerpt of the semantic graph for the studied case

Results

Controller setup using the semantic graph

Here, we elaborate on how to set up the control algorithm with the semantic graph. An excerpt of the semantic graph deployed in the integration layer is shown in Figure 5.

The lower right corner of Figure 5 demonstrates the modeling of building-related information. The studied zone is modeled as a `bot:Zone` with a volume of $129.6\,m^3$ and an area of $48\,m^2$. It has 7 adjacent surfaces, including 4 walls, 1 floor, 1 ceiling, and 1 window. Taking the modeling of the window as an instance, the window (`bot:Element`) has an area of $6\,m^2$, a u-value of $0.2\,\frac{W}{m^2 \cdot K}$ and a g-Value of 0.7, whereas its thermal capacitance is neglected. The properties of other surfaces are modeled in a similar way with one more property on the area-specific capacitance. Using the data above, the initial guess and bounds of the RC parameters are calculated as $\hat{\boldsymbol{\theta}}_0 = [0.017\frac{K}{W}, 6.6\frac{MJ}{K}, 4.2m^2]$ and $\hat{\boldsymbol{\Theta}}_0 = [0.1\hat{\boldsymbol{\theta}}_0, 10\hat{\boldsymbol{\theta}}_0]$, which are used by the event-triggered SI module in the MPC service. Moreover, the information on sensor data points connected to the zone are queried using SPARQL described in Figure 4, and the results are listed in Table 1. In total, four sensors are related to the zone (`:BESTEST_case600`), measuring the T_{max}, T_{min}, T_z and occupancy head count. By using the sensor ID, the historic measurements made by the sensor are retrieved from time-series DB via Flux query. The historic measurements of T_z are taken as an example and shown in Table 2. The historical measurements on occupancy and T_z are sent to MPC service when SI procedure is triggered, while the real-time measurement of the state T_z is sent to the MPC service every 5 minutes. Note the time-series historical weather data related to the building geological location (`:Stuttgart`) is sent to MPC service in a similar way, and is not detailed here.

Table 1: Sensor data points connected to the studied zone

Property	Quantity	ID
:Set_Point_Cool	brick:Temperature	LR101.TR22
:Set_Point_Heat	brick:Temperature	LR101.TR23
:Temperature	brick:Temperature	LR101.TR21
:Occupancy	brick:Occupancy_Count	LR101.OC01

Table 2: Time-series data of zone temperature sensor (K)

Time	LR101.TR21
2018-07-31 23:45:00	293.17
2018-07-31 23:50:00	293.14
2018-07-31 23:55:00	293.12

The right upper corner part illustrates the HVAC systems. `Duct_6` and `Duct_7` exchange fluids directly with the thermal zone, whereas the radiator exchanges heat directly with the zone. The radiator has nominal properties, such as $\dot{Q}_{max}^{rad}$ of 2787 W and η_{rad} of 0.9, and dynamic property $\dot{Q}^{rad}$, which is monitored and also optimized by the MPC optimization procedure. The rest of HVAC system's topology and their properties are modeled in a similar way. The maximal powers for the heating coil, reheat coil, and cooling coil are $1477\,W$, $261\,W$, and $-1814\,W$, with respective efficiencies 0.8, 0.8, and 2.7, where negative power means cooling. The maximal power matrix $\dot{\boldsymbol{Q}}_{max}$ and efficiency matrix $\boldsymbol{\Gamma}$ are utilized to initialize the MPC at the beginning, and historical measurements $\hat{\boldsymbol{Q}}$ are retrieved by the SI module when a system update is activated, in the same way as described in the last paragraph.

The forecaster information is modeled as the left lower corner of Figure 5. Using the query in Figure 3, the forecast file paths are first retrieved via the forecast service (results in Table 3), and passed to the MPC service.

Table 3: Results of the SPARQL query in forecast service

Forecast	File
:Day_Ahead_Electricity_Forecast	Electricity_STG.mat
:Occupancy_Forecast	Occupancy_Case_600.mat
:Weather_Forecast	DEU_Stuttgart_IWEC.epw

The information about the hyper-parameter settings in the MPC algorithm is modeled in the left upper corner of Figure 5. The MPC service has two sub-modules: event-triggered SI and economic MPC. The former optimizes the RC model parameters to ensure the accuracy of the RC model, which is later used to predict the $\hat{T}_z$ in the MPC; the latter optimizes the relative power of all related components $\boldsymbol{u}$, to minimize the operating costs. For event-triggered SI, the trigger threshold ρ is set as $0.1\,°C$ with trigger horizon N_t set as 1 day to ensure accurate

daily prediction. The SI horizon N_s (training data length) is set to 7 days, as recommended in (Blum et al., 2019, 2022). For the economic MPC module, we set the prediction horizon N_c to 8 hours, because 1R1C model is not accurate for long-time prediction (Arroyo et al., 2020).

Performance of MPC with Event-triggered SI

The simulation model in Modelica is exported as Functional Mockup Units (FMU), defined by Functional Mockup Interface (FMI) standard , and simulated in Python with PyFMI library. The simulation results of July are demonstrated as follows.
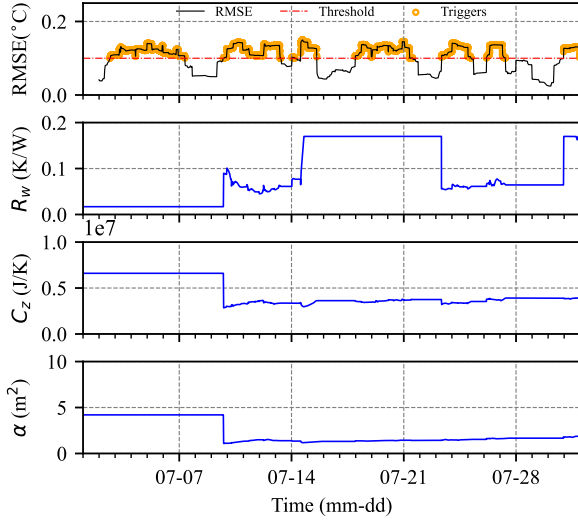

Figure 6: Event-trigger and estimated parameters in July

Figure 6 shows the values of RMSE in (6) with its threshold ρ and the estimated parameters $\hat{\boldsymbol{\theta}}_k$ along the time in July. The evaluation of RMSE starts after 1 day, while the SI after 7 days to collect enough data. The SI is triggered repeatedly in certain periods because the eventual convergence of $\hat{\boldsymbol{\theta}}$ demands a few new data. Moreover, more obvious changes are observed in R_w than C_z and α, which results from their different sensitivities to the environmental boundaries and different effects on the short-time prediction ($8h$) accuracy. Overall, a total of 3545 times of SI is activated among 6912 simulation steps by using the proposed event-trigger scheme, saving 49% computations. Results compared to the MPC with MHE prove that the proposed event-triggered SI achieves similar control performance while demanding less computational effort.

Figure 7 shows the performance of the MPC algorithm in comparison to RBC model in July, which is defined in the system modeling section. More specifically, diagram (a) shows the varying electricity price, diagram (b) for the measured zone temperatures, diagram (c) for the total heating power $\dot{Q}^h = \dot{Q}^{hc} + \dot{Q}^{rh} + \dot{Q}^{rad}$, and diagram (5) for the cooling power $\dot{Q}^c = \dot{Q}^{cc}$. According to the temperature profiles in Figure 7, the MPC algorithm controls the indoor climate better than the RBC, in terms of fewer violations of temperature constraints. Especially during the weekends (07-15,07-16 and 07-28), RBC model has forced the HVAC system to "Unoccupied-Off" mode, and
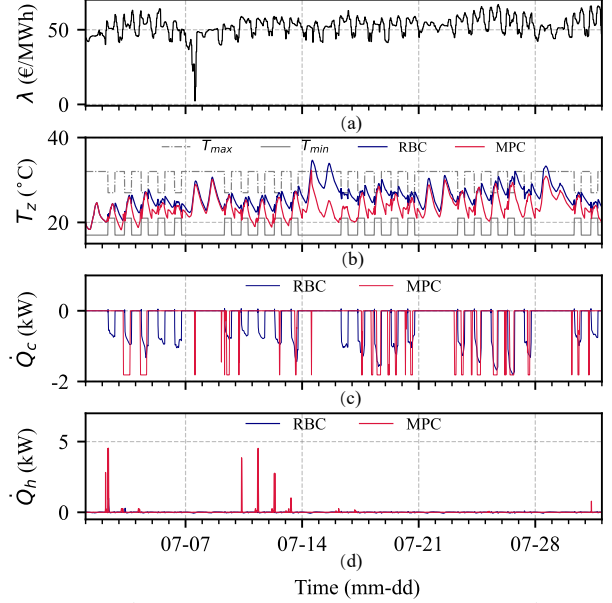

Figure 7: Electricity price, zone temperature, cooling and heating power from HVAC system in July

no cooling power is supplied, while MPC algorithm predicts the temperature peak is about to come and provide moderate cooling power. Overall, MPC achieves an operating cost reduction of 12% compared to RBC.

Combined with the successful integration of the building design data and sensor measurement explained, the results demonstrate the proposed semantic-assisted framework can be used practically to instantiate the algorithm.

Conclusion

In this paper, a semantic-assisted control framework for MPC algorithm with event-triggered SI is proposed. The framework facilitates MPC algorithm setup by integrating heterogeneous data sources via semantic modeling. To ensure computational efficiency and the accuracy of MPC model at the same time, an event-triggered SI scheme is designed, where an educated initial guess and reasonable boundaries of RC model parameters for the thermal envelope are automatically instantiated using the semantic graph. The effectiveness of the proposed MPC algorithm is validated via simulations, where lower operating costs and better indoor temperature control are achieved, compared to the legacy RBC sequence. In the future study, the proposed framework will be verified on a real building.

Acknowledgment

References

Andriamamonjy, A., R. Klein, and D. Saelens (2019). Automated grey box model implementation using bim and modelica. *Energy and Buildings 188*, 209–225.

Arroyo, J., F. Spiessens, and L. Helsen (2020). Identification of multi-zone grey-box building models for use in model predictive control. *J. Build. Perform. Simul. 13*(4), 472–486.

ASHRAE (2005). Sequences of operation for common HVAC systems.

ASHRAE (2007). ANSI/ASHRAE 140-2007 Standard method of test for the evaluation of building energy analysis computer programs.

Atam, E. and L. Helsen (2016). Control-oriented thermal modeling of multizone buildings: Methods and issues: Intelligent control of a building system. *IEEE Control systems magazine 36*(3), 86–111.

Balaji, B., A. Bhattacharya, G. Fierro, J. Gao, J. Gluck, D. Hong, A. Johansen, J. Koh, J. Ploennigs, Y. Agarwal, M. Bergés, D. Culler, R. K. Gupta, M. B. Kjærgaard, M. Srivastava, and K. Whitehouse (2018). Brick : Metadata schema for portable smart building applications. *Applied Energy 226*, 1273–1292.

Blum, D., K. Arendt, L. Rivalin, M. Piette, M. Wetter, and C. Veje (2019). Practical factors of envelope model setup and their effects on the performance of model predictive control for building heating, ventilating, and air conditioning systems. *Applied Energy 236*, 410–425.

Blum, D., Z. Wang, C. Weyandt, D. Kim, M. Wetter, T. Hong, and M. A. Piette (2022). Field demonstration and implementation analysis of model predictive control in an office HVAC system. *Applied Energy 318*.

Bonduel, M. (2018). Towards a PROPS ontology (2018).

Borrmann, A., J. Beetz, C. Koch, T. Liebich, and S. Muhic (2018, 09). Industry Foundation Classes: A standardized data model for the vendor-neutral exchange of digital building models.

Chamari, L., E. Petrova, and P. Pauwels (2022, May). A web-based approach to BMS, BIM and IoT integration: a case study. *CLIMA 2022 conference*.

Compton, M., P. Barnaghi, L. Bermudez, R. Garcia-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, et al. (2012). The SSN ontology of the W3C semantic sensor network incubator group. *Journal of Web Semantics 17*, 25–32.

De Coninck, R., F. Magnusson, J. Åkesson, and L. Helsen (2016). Toolbox for development and validation of grey-box building models for forecasting and control. *J. Build. Perform. Simul. 9*(3), 288–303.

DIN (2015). DIN EN 16798-1 Gesamtenergieeffizienz von Gebäuden Eingangsparameter für das Innenraumklima zur Auslegung und Bewertung der Energieeffizienz von Gebäuden bezüglich Raumluftqualität, Temperatur, Licht und Akustik .

Drgoňa, J., J. Arroyo, I. C. Figueroa, D. Blum, K. Arendt, D. Kim, E. P. Ollé, J. Oravec, M. Wetter, D. L. Vrabie, and L. Helsen (2020). All you need to know about model predictive control for buildings. *Annual Reviews in Control 50*, 190–232.

Eydner, M., L. Wan, T. Henzler, and K. Stergiaropoulos (2022). Real-time grid signal-based energy flexibility of heating generation: A methodology for optimal scheduling of stratified storage tanks. *Energies 15*(5), 1793.

Fierro, G. and P. Pauwels (2022). Survey of metadata schemas for data-driven smart buildings (Annex 81).

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition 5*(2), 199–220.

IEA (2015). Building energy performance metrics - Analysis and key findings.

Janowicz, K., A. Haller, S. J. Cox, D. Le Phuoc, and M. Lefrançois (2019). SOSA: A lightweight ontology for sensors, observations, samples, and actuators. *Journal of Web Semantics 56*, 1–10.

Kukkonen, V., A. Kücükavci, M. Seidenschnur, M. H. Rasmussen, K. M. Smith, and C. A. Hviid (2022). An ontology to support flow system descriptions from design to operation of buildings. *Automation in Construction 134*, 104067.

Lefrançois, M., J. Kalaoja, T. Ghariani, and A. Zimmermann (2017). *The SEAS knowledge model*. Ph. D. thesis, ITEA2 12004 Smart Energy Aware Systems.

Mavrokapnidis, D., K. Katsigarakis, P. Pauwels, E. Petrova, I. Korolija, and D. Rovas (2021). A linked-data paradigm for the integration of static and dynamic building data in digital twins. In *BuildSys 2021*, pp. 369–372.

Oraskari, J., M. Bonduel, K. McGlinn, P. Pauwels, F. Priyatna, A. Wagner, V. Kukkonen, S. Steyskaland, and J. Lehtonen (2023). IFCtoLBD v 2.40.1.

Pauen, N., J. Frisch, and C. van Treeck (2022, 9). IFC2TSO - Algorithmic processing, complexity reduction, and transfer of information regarding technical systems from IFC to TSO. In *33. Forum Bauinformatik*.

Pauwels, P. and K. McGlinn (2022). *Buildings and semantics: Data models and web technologies for the built environment*. CRC Press.

Rasmussen, M. H., M. Lefrançois, G. F. Schneider, and P. Pauwels (2021). BOT: The building topology ontology of the W3C linked building data group. *Semantic Web 12*(1), 143–161.

Wetter, M., W. Zuo, T. S. Nouidui, and X. Pang (2014). Modelica buildings library. *J. Build. Perform. Simul. 7*(4), 253–270.