EXCESS

FleXible user-CEntric Energy poSitive houseS

Deliverable 3.4:

EXCESS Model-Predictive Control Algorithms

LC-EEB-03-2019

New developments in plus energy houses (IA),

Proposal No. 870157

Title	Document Version
D3.4 EXCESS Model-Predictive Control Algorithms	0.7

Project Number	Project Acronym	Project Title
H2020-870157	EXCESS	FleXible user-CEntric Energy poSitive houseS

Contractual Delivery Date	Actual Delivery Date	Deliverable Type*-Security**
December 2021 (M28 2021)	February 2022 (M30 2022)	D-PU

*Type: P: Prototype; R: Report; D: Demonstrator; O: Other.

Security Class: PU: Public; PP: Restricted to other programme participants (including the Commission); RE: Restricted to a group defined by the consortium (including the Commission); CO: **Confidential, only for members of the consortium (including the Commission).

Responsible	Organisation	Contributing WP
Hassam ur Rehman	VTT	WP3

Abstract
This document acts as a blueprint for the model predictive control (MPC) base and the optimization algorithms that are planned to be used in different demos. It presents the methodology that will be used in demos to implement MPC and optimize the controls of the various components such as generation, storage and demand. In each demo, simulations using various available software are carried out; plus, the integration of prediction models to enable MPC and optimization of the models based on the inputs and defined objectives if needed. Afterwards, this is implemented at the demo site in real-case scenario depending on the local conditions. The main objective function is either cost reduction or imported energy reduction, depending on the local requirements. The current deliverable is a guide for the MPC integration, its optimization and implementation of the control component at the demos.

Keywords
Model predictive control, Control methods, Energy simulations, Energy models, Optimization algorithms

Revision	Date	Description	Author (Organisation)
V0.1	10.01.2022	*Template & draft TOC*	Mia Ala-Juusela (VTT)
V0.5	21.02.2022	*Ready for internal review*	Rakesh Ramesh, Hassam ur Rehman, Mia Ala-Juusela, Ala Hasan (VTT), Jarkko Järvinen (Gebwell), Victoria Lara Garcia (CENER), Koen Allaerts, Davy Geysen (VITO), Michael Gumhalter, Thomas Ramschak (AEE), Thomas Schwarzl (TSI), Manuel Strohmaier, Michael Kernitzkyi, (JR)
V0.6	25.02.2022	*Internal review completed*	Konstantinos Latanis (S5), Gabriel García Nevada (CENER)
V0.7	28.02.2022	*Final version submitted to EC*	Hassam ur Rehman, Rakesh Ramesh, Mia Ala-Juusela, Ala Hasan (VTT)

This project has received funding from the
European Union's Horizon 2020 Research and Innovation Programme
under Grant Agreement № 870157.

More information available at https://positive-energy-buildings.eu/

EXECUTIVE SUMMARY

This report describes the detailed model predictive control and optimization methods used in each demo pilot of the EXCESS project. In addition, the main components controlled after the prediction-based inputs, are presented as well. This deliverable is a direct outcome of the task, which aims to provide:

- The MPC framework for the demos and allow assessment of alternate strategies
- The energy models addressing each demo
- The optimization logic based on the objective function
- A base for the EXCESS control optimization decision support system
- Bases and guidelines for the demonstrator activity

Generally, in each demo, simulations are carried out using various available simulation software, integrate the prediction models to enable MPC and optimize the models based on the inputs and defined objectives if needed. It is followed by the implementation of the MPC and algorithms at the demo site in a real-case scenario, subject to the local conditions. The different modelling and simulation software, prediction-based inputs (for MPC) and optimization methods chosen for the demo cases are discussed. Different types of input data, prediction variables, constraints, design variables and optimization algorithms and objective functions are described based on the domestic requirements and conditions of the demos. The models and the methods described are based on the input from the reports and models generated in the earlier stages of the project (such as the reports Deliverable 1.1 "PEB as an enabler for the consumer-centred clean energy transition: shared definition and concept", Deliverable 2.6 "Report on advancing simulation-based energy performance assessment for optimal PEB design" and Deliverable 3.1 "EXCESS ICT Architecture Blueprint").

The simulation model developed in EXCESS is used in developing the Model Predictive Control methods, and the main inputs that are predicted are the weather forecast and the energy prices. This is used to optimize the control of the energy system to either reduce the energy cost or the energy purchased.

The main achievements of this task are the initial four developed MPC methods, optimization algorithm, energy model integration and its implementation in real pilot cases. This provides a powerful tool that aids in evaluating the performance of the controls for the PEB and the effectiveness of an MPC, which can then be compared with the rule-based method.

With the information provided in this deliverable, further research work could be undertaken to study and evaluate MPC strategies, optimization algorithms and to improve the controls of the components used in the building (to either reach the PEB level or reduce the energy cost).

At the time of writing this report, the Belgian and Austrian demos are seemingly at an advanced stage. For the Finnish demo, the MPC and optimization method development is ongoing, as the new building is under construction. Reaching the PEB level is a challenge according to the PEB definition defined in D1.1. Therefore, the objective function can be to minimize the energy cost. For the Spanish demo, the MPC and optimization methods are under development at an initial stage as the building construction is ongoing.

Finally, it is necessary to know that the MPC for each demo building must be designed separately due to different local requirements such as weather profile, cost profile and technology used. This is important to achieve PEB standards based on local requirements, as well as the evaluation of optimization of control strategies. Another lesson learned is that the user behaviour, energy demand for appliances/lighting and set points can impact the performance of the energy system and the operational cost. Therefore, it is necessary to engage the end-user to achieve the PEB requirements and cost reductions.

This report will provide input to other tasks of WP3, as it will define the controls strategies that would be implemented in the demos and WP4 as it will provide input for testing activities.

TABLE OF CONTENTS

EXECUTIVE SUMMARY 3

1 INTRODUCTION 8

1.1 Purpose of the document 8

1.2 Scope of the document 8

1.3 Structure of the document 8

2 BACKGROUND 10

3 BASIC COMPONENTS IN MPC 11

3.1 Simulation programs suitable for MPC 11

3.2 Optimization methods and programs 11

3.3 Data sources 11

4 DEMO SITE MPCS 13

4.1 MPC elements in Finnish demo 13

4.1.1 Simulation and optimization layer 13

4.1.1.1 Optimization tool 14

4.1.1.2 Algorithm for optimization 14

4.1.1.3 Objective function 15

4.1.1.4 Inputs forecasting 15

4.1.1.5 Automation process 16

4.1.2 Organization layer 18

4.2 MPC element in Belgian demo 18

4.2.1 Approach 18

4.2.2 State-space representation building model example 19

4.2.3 Combined linear models 20

4.2.4 Model Predictive Control (MPC) 21

4.2.5 Optimization objectives 23

4.2.6 Rolling horizon control 24

4.2.7 Software 24

4.3 MPC elements in Austrian demo 25

4.3.1 Forecast layer 25

4.3.2 Supervisory building MPC layer 28

4.3.3 Smart device MPC layer 29

4.3.4 Base control layer 30

4.3.5 User layer 30
4.4 MPC elements in Spanish demo 30
4.4.1 Simulation layer 31
4.4.1.1 Virtual building model 31
4.4.2 Optimization layer 32
4.4.2.1 Optimization tool 32
4.4.2.2 Algorithm for optimization 32
4.4.2.3 Objective function 32
4.4.3 Organization layer 32
4.4.3.1 Automation process 32
4.4.3.2 Inputs -forecasting 33

5 CONCLUSIONS 34

6 REFERENCES 36

LIST OF FIGURES

Figure 1: Current framework and components of the Finnish demo 13
Figure 2: The flow diagram of the optimization process and its integration with the simulation 15
Figure 3: Schematic of the current model of the Finnish demo 16
Figure 4: First order RC building model 20
Figure 5: Combined building and HP model 20
Figure 6: Schematic overview of an MPC system for building control 22
Figure 7: Schematic overview of the MPC models configuration for the Belgian demonstrator 23
Figure 8: Rolling time horizon principle 24
Figure 9: Layered implementation structure for Austrian demonstrator control scheme 26
Figure 10: Simplified flowchart of the forecast infrastructure 27
Figure 11: MPC controller testing strategy with co-simulation 28
Figure 12: MPC Spanish Demo Scheme 31

LIST OF TABLES

Table 1: The data sources options for the input parameters and the prediction variables 12
Table 2: Weather forecasting variables, energy cost, building occupancy data and possible providers 16
Table 3: The MPC components in the energy system and possible functionalities options 17
Table 4: Used software components in the forecast layer 26
Table 5: Subcomponents of supervisory building MPC 28
Table 6: Weather forecasting variables and possible providers. 33
Table 7: Different approaches to system modelling and incorporating MPCs by the partners 35

1 Introduction

1.1 Purpose of the document

This work configures a base-ground MPC framework and optimization logic, consisting of an appropriate building Model Predictive Control (MPC) component to allow for the assessment of alternative control scenarios and strategies for PEBs, based on data collected through the building systems and simulations by the EXCESS Data Management Platform. The objective of the report is to present the methods used in the demos to implement MPC and optimize the controls of various components based on the prediction of the variables. Thus, this component will act as the EXCESS Control Optimization Decision Support System that provides the main functionalities upon any control optimization and automation application for end-users. In contrast to Task 2.6 [1], which focuses on further optimizing energy performance simulation results for improving the design, this task focuses on controls, appropriately extending existing MPC algorithms with context-aware models (cost, occupancy, comfort) and optimizing the operation of the building or the energy system or both.

The models and methods described in this deliverable are based on the input from the D1.1 (PEB as enabler for consumer centred clean energy transition: shared definition and concept), D2.6 (Report on advancing simulation-based energy performance assessment for optimal PEB design) and D3.1 (EXCESS ICT Architecture Blueprint) and integrated in this deliverable. Software integration in the demo buildings and districts will be performed during WP4, where demo-relevant partners will undertake the responsibility to deliver functional instances of the MPC component, properly integrated into the demo buildings assets and systems, to ensure stable operation under real-life conditions.

The partners contributing to this Task are AEE, TSI, JR, VITO, GebWell and CENER. VTT is the task leader of this deliverable and has contributed to the energy modelling, MPC development and optimization method development for the Finnish pilot. On the other hand, AEE, TSI and JR are the partners responsible for MPC and optimization method development for the Austrian Pilot, VITO for the Belgian Pilot and CENER for the Spanish Pilot.

1.2 Scope of the document

The deliverable provides a base-ground MPC framework and optimization logic, consisting of an appropriate building model predictive control (MPC) component logic to allow for the assessment of alternative control scenarios and strategies for PEBs. In month 42, an updated version, based on findings from the demonstration activities of the project, will be available.

1.3 Structure of the document

The contents of the document are the following:

- the explanation of the background: what is MPC, what benefits are achieved by using it (by VTT)
- the description of the potential /typical elements: what simulation programs are available, where to get the data for e.g., the weather forecasts or occupant behaviour patterns, typical format for the data and transfer, optimization protocols and maths used, etc. (by VTT)

- current situation on the demo sites in the development: which of the above elements will be used for each demo and planned timeline (by demo MPC developers)

2 Background

Model Predictive Control (MPC) is a feedback control algorithm that makes predictions about future outputs of a process through a model. It has the ability to handle systems having multiple inputs and outputs. Being a multi-variable controller, it controls the outputs simultaneously by considering all interactions (static and dynamic) between the system variables. In addition, it can handle multiple constraints as well. Similar to a feed-forward controller, MPCs have a preview capability. To improve controller performance, they incorporate future reference information into the control problem [2], [3].

At a specific timestep, an MPC applies the first optimal control and then disregards the rest of the sequence. As the current prediction horizon moves forward to the next timestep, it then computes the new optimization problem from that timestep. This is commonly referred to as the *receding horizon philosophy* [4]. The strategy of an MPC to compute the input is to predict the future. An optimization problem is solved by an optimizer to reduce the error between the reference setpoint and the predicted output. It is executed in such a way as to avoid the sharp ramping of controls.

If the timestep chosen is too high, the MPC would not be able to react swiftly to disturbances, whereas, if the chosen timestep is too low, the computational power required is immense. Thus, this demands a trade-off between faster reaction time and computing complexity. If the prediction horizon is too short, the MPC will not be able to relay control methods and thus, the system will not have time to react. On the other hand, if the prediction horizon is too long, an unforeseen disturbance occurring might render the computations useless and the simulation must be discarded. [2].

Buildings and energy systems, owing to the variances and uncertainties in weather forecasting, activities of the inhabitants, etc., pose several challenges for an optimized, energy efficient performance while maintaining indoor thermal comfort. Ever since its conception, there have been several attempts to utilise MPCs to combat this problem [5]. The results of several works on the simulation and implementation of MPCs in buildings have displayed an increased energy savings potential and reduced energy consumption whilst maintaining the desired thermal comfort indoors [6]– [11]. The objective (maximizing on-site renewables or reducing energy costs or reducing imported energy), in addition to a cost-benefit analysis, helps in determining if one should implement an MPC in a building-energy system or adhere to a Rule-Based Control (RBC) [12]. The development of an MPC model might also have additional associated costs including the cost of hardware installation, procuring weather forecast, adjusting for uncertainty etc. which should be considered in the cost-benefit analysis [13].

3 Basic components in MPC

3.1 Simulation programs suitable for MPC

To improve the performance of buildings, several active (HVACs, PVs) and passive (orientation, natural ventilation, luminance) strategies have been developed. In addition, indoor thermal comfort, sustainability, cost are other parameters considered when designing a building with improved energy savings potential [14]. Identification of the most efficient design while striking a balance amongst the conflicting objectives is a strenuous task. Building Performance Simulation tools reproduce the different facets of the performance of the building using a computer-based, mathematical model. Some examples of such software include TRNSYS, IDA-ICE, EnergyPlus, VIP Energy, BV^2 etc. [15]. Each of them has its pros and cons, and it is worthwhile to determine the objective of our work before choosing the simulation tool. Some of the above-mentioned tools are used by the partnering organizations for the demo building in their respective countries.

3.2 Optimization methods and programs

As stated earlier, for the optimal performance of a building, the decision variables have to be optimized such that the objective functions are solved subject to constraints. In a building, the HVAC system and the local generation systems are the variables that are optimized. The objective function could be minimizing the energy cost, maximizing self-sufficiency, maximizing renewable energy utilization, or a combination of these while satisfying indoor thermal comfort. Thus, these single or multi-objective functions are optimized by some optimizing tools available. A few tools, such as Opt-E-Plus, GENE_ARCH, BEopt, GenOpt are quite common and have been utilized in this work. These tools make use of genetic or evolutionary or hybrid optimization algorithms like GPSPSOCCHJ, NSGA-II, MOPSO, SPEA-2, etc. [14]. More details about the algorithms used in this work are present in the specific sub-sections.

3.3 Data sources

For implementing MPCs in the building and the energy system, procurement of relevant data is necessary. For instance, for a comfort profiling component that measures the indoor thermal comfort of the inhabitants - the data from the sensors (room temperature, room humidity, room luminance, room CO_2 concentration, occupancy status, etc.) are necessary. The above-mentioned factors are also dependent on external weather conditions, such as the ambient temperature, humidity, and solar irradiation levels. This data must be obtained from the local weather stations, metrological institutes, with the shortest possible lead times, ensuring an improved forecast on energy produced. For the energy system, the data for the current status of the PV/T, heat pumps, buffer tanks are obtained through sensors. The cost of electricity from the grid also plays a chief role in deciding when to import electricity and use the heat pump. This data could be acquired from power markets such as Nord Pool, which offers day-ahead and intraday markets. These input parameters, along with the possible sources are tabulated in Table 1.

Table 1: The data sources options for the input parameters and the prediction variables

Prediction Variable	Possible Source
Outdoor Temperature, Relative Humidity	Local Weather Station, Online Data
Solar Irradiation	Local Weather Station, Online Data
Demand of the building, space heating, cooling and HVAC	Sensors -> EXCESS Data Management Platform
Room temperature, Room humidity, Room luminance, Room CO_2 concentration, Occupancy status, Lighting devices	Sensors -> EXCESS Data Management Platform
Price of electricity	Power Trading Markets (e.g., Nord Pool)
PV/T data and set points	Sensors -> EXCESS Data Management Platform
Heat pump status and set points	Sensors -> EXCESS Data Management Platform
Storage tanks set points and data	Sensors -> EXCESS Data Management Platform

4 Demo site MPCs

Section 4 describes the current situation of the demo sites in development. Additional details about the optimization method, the algorithm, the objective function, and the forecasting data are communicated in this section.

4.1 MPC elements in Finnish demo

In this sub-section, we will look at the present state of MPC development for the Finnish Demo. The Finnish demo MPC scheme can be divided into two main layers:

- Simulation and optimization Layer: this layer is based on the simulation software, its integration with the optimization software; thereby enabling the benefits of MPC. This is explained in section 4.1.1.
- Organization Layer: this layer is responsible for the development and integration of the MPC onsite, at the demo. It includes managing different inputs and outputs, data collection, the connection between the MPC and Building Management System (BMS). This layer is still under development by the integrator company. The plan is explained briefly in section **Error! Reference source not found.**.

4.1.1 Simulation and optimization layer

For the positive energy building in Kalasatama, Finland, some modifications were made to the building-energy system model described in Task 2.6. The current framework and components of the building and the energy system are shown in *Figure 1*.

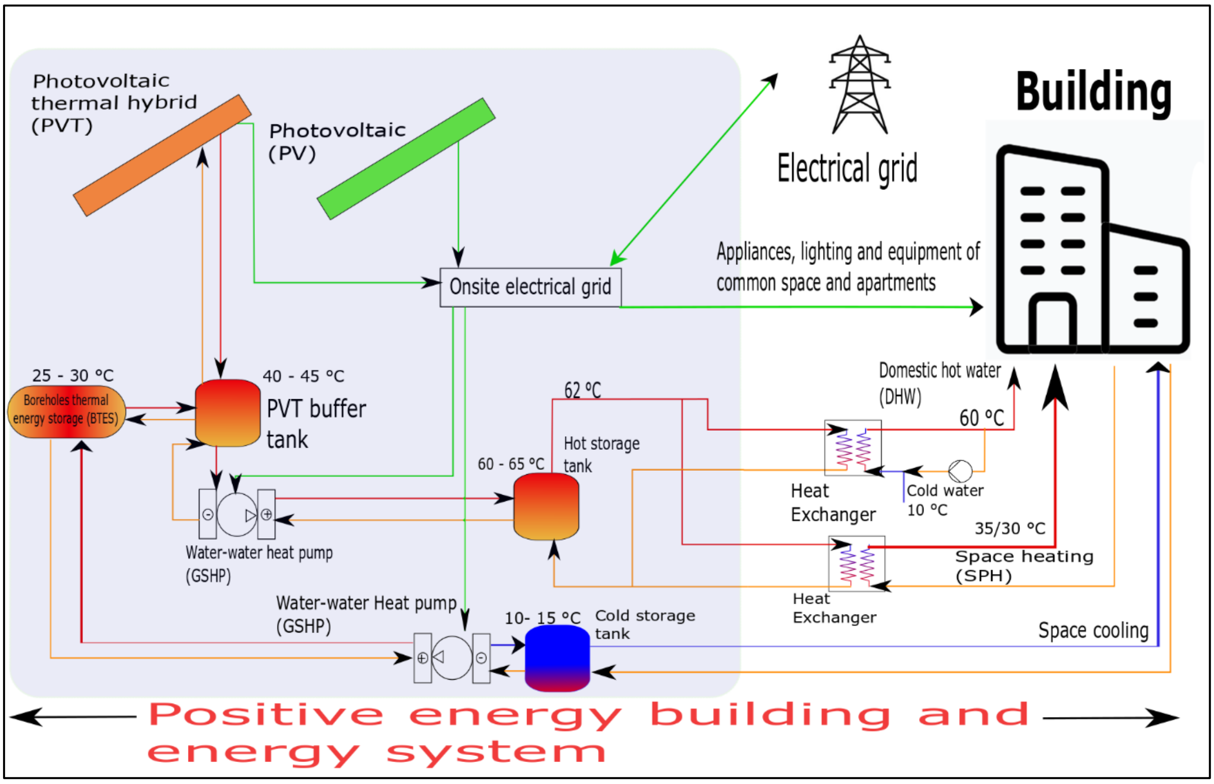

Figure 1: Current framework and components of the Finnish demo

The building model is developed in IDA-ICE simulation software, and the energy system is modelled and simulated in TRNSYS software. The description of the modelling packages is described in detail in Task 2.6 [1]. The IDA-ICE software generates the building's demand profiles, which is then integrated with the TRNSYS simulation model as an external text file.

The Finnish demo building is designed to satisfy and surpass the prequisites warranted by the national building regulations by reducing the energy demand and making it a positive energy building. Since district heating contributes mainly to the emissions in Finland, renewables are instead used to satisfy the demand. The building's load consists of electricity used for heating and cooling demands and appliance and lighting loads. Based on further studies and construction constraints, there could be some changes to the model.

4.1.1.1 Optimization tool

The selected optimization tool is MOBO. MOBO is a building optimization software capable of handling both single and multi-objective optimization problems with the added ability to handle multiple constraints automatically [16]. Being a generic optimization tool, MOBO can be coupled with several building simulation programs (in our case, TRNSYS). The optimizer is devised to attain the optimum objectives, by varying the design parameters.

The optimization tool for a single objective function is used to provide better control values such as set points values for the energy system component, e. g. the heat pumps, hot water and tanks setpoints based on weather and cost data [17]. Furthermore, this is particularly important for a progressive decision-making approach where the input values may change within the decision-making process. The results can be post-processed to identify the sensitivities of the decision variables if needed. It defines how various independent design variables impact a particular outcome under a given set of assumptions and inputs. Lastly, the optimization tool is used to decide which parameters need more in-depth analysis and those for which standard values could be used.

4.1.1.2 Algorithm for optimization

There is more than one algorithm that can be utilised. Genetic algorithm (GA) is one method, while the deterministic with hybrid is another method such as Hooke-Jeeves. At the time of writing this report, it is planned to proceed with the non-domination based genetic algorithm (NSGA) algorithm. In the present approach, the TRNSYS system models and multi-objective building optimizer (MOBO) are combined to perform the optimization. MOBO [18] is freeware optimization software that can handle discrete and continuous variables and allows evolutionary and classical optimization algorithms. For this study, the NSGA-II algorithm is selected [19]. The NSGA-II algorithm is selected because it solves a multi-objective problem while handling the constraints, discrete and continuous variables. Furthermore, parallel computing is possible with this algorithm. It is not only computationally expensive to explore all designs, but also computationally infeasible. Hence, a multi-objective non-dominated sorting genetic algorithm (NSGA-II) is used to perform the exploration [20]

An automated simulation-based optimization method is performed using the NSGA-II algorithm combined with TRNSYS. It keeps all the iterations in an archive, and uses them in a non-dominated sorting process. *Figure 2* shows the flow diagram of the optimization process. All the design variables random values are generated by the algorithm to be evaluated in the TRNSYS/Python/IDA-ICE/MATLAB simulation software and later the results are sorted by MOBO based on the objective functions. The integration, logic, and flow of the MOBO optimizer and simulation software is shown in the *Figure 2*. The values of the proposed design variables are created by the MOBO, these values are written by MOBO in the simulation model files. The simulation runs the model files and provides the results. These results are then evaluated by MOBO in order to meet the objective functions.

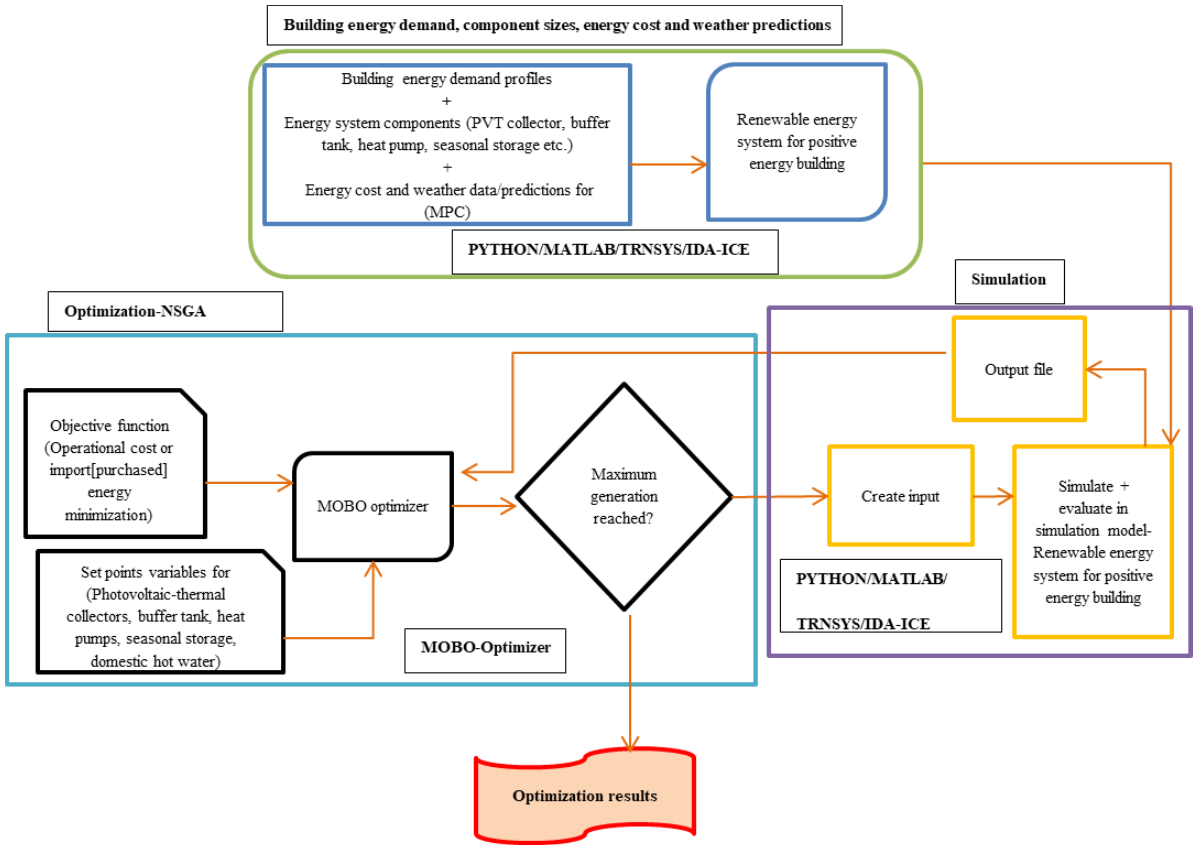

Figure 2: The flow diagram of the optimization process and its integration with the simulation

4.1.1.3 Objective function

At the time of writing of the report, the objective function is yet to be finalised between operational cost or imported (purchased) energy. In either case, this is to be minimized while ensuring the sustenance of indoor thermal comfort. A tolerance band will be chosen, and any thermal discomfort of the inhabitants is to be penalised accordingly.

The optimization problem can be formulated as follows:

$$\text{Min } \{C_{OC}\,(x) = \text{Operation cost (OC) or } E_{PUR}\,(x) = \text{Purchased electricity}\}, \text{ for all } \boldsymbol{x} = [x_1, x_2, \dots, x_n],,$$

where C_{OC} is the operational cost of the system and EPUR is the purchased electricity for the system together with the building demand, and 'x' is the vector of the design variables (set points). To provide an overall performance of the building, it includes both the energy system and building appliances demand. The purchased electricity includes both the factors. The C_{OC} includes the operational costs, the import and export energy cost.

4.1.1.4 Inputs forecasting

Several possible inputs, such as weather conditions, cost, occupancy profiles and the associated variable predictions are tabulated in Table 2. These are the possible input variables that can be predicted for input during the implementation phase of the controls in demo building (as discussed in section 4.1.2) and in the models as needed.

Table 2: Weather forecasting variables, energy cost, building occupancy data and possible providers

	Prediction Variables	Units	Source
1	Outdoor Temperature	º C	[21], [22]
2	Relative Humidity (or dew temperature)	%	[21], [22]
3	Direct Normal Beam Radiation	W/m²	[23]
4	Total Radiation on horizontal surface	W/m²	[23]
5	Demand of the building, space heating, cooling and HVAC (target temperature)		*EXCESS Data Management Platform*
6	Room temperature, room humidity, room luminance, room CO2 concentration, occupancy status (yes/no), lighting devices		*EXCESS Data Management Platform*
7	Price of electricity		[24]

4.1.1.5 Automation process

The energy system in the Finnish demo is comprised of photovoltaic-thermal (PV/T) panels, buffer tanks, boreholes thermal energy storage (BTES), heat pumps (HP) and the electrical grid. It is designed in such a way to satisfy and cater to the heating, cooling, domestic hot water and electrical energy demands of the building. A flowchart of the current model is presented in *Figure 3*. The idea behind such an onsite energy system is to provide maximum energy to the building. Any excess electricity is exported to the grid. In addition, any shortfall is met by importing electricity from the grid.

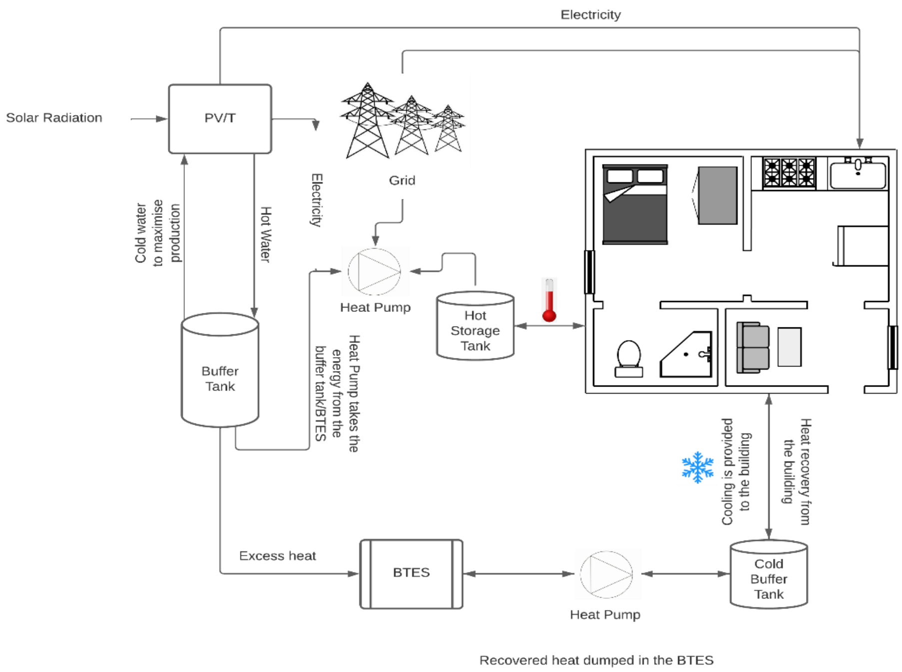

Figure 3: Schematic of the current model of the Finnish demo

The control framework of the energy system is designed so that, cold water from the buffer tank cools down the PV/T and thereby, maximizes its electrical and thermal production. When the buffer tank is charged beyond a certain level, the excess heat energy is unloaded in the BTES. Space heating and domestic hot water to the building is delivered by the heat pump, which in turn takes the energy from the buffer tank or from the BTES. On the other hand, the cooling is provided to the building by using a combination of a heat pump, cold tank, and a ventilation unit. In this work, the heat from the building is recovered and deposited in the cold tank, and the heat pump in turn takes the heat from the cold tank and dumps the heat energy in the BTES.

In order to control the energy system and to optimize the performance, attempt is made to incorporate MPCs in set points of the heat pumps or PV/T or buffer tanks or hot tank. Table 3 prescribes some of the current operational controls (based on rule-based controls) and MPC based functionalities options that can be implemented in the energy system based on the inputs (as discussed in Table 2). These controls and data can be provided through the cloud service to the MPC controller for optimization of the component operation in the organization layer (4.1.2).

Table 3: The MPC components in the energy system and possible functionalities options

Component	Type of control	Current operation/functionality	MPC based operation/functionality
PVT	Buffer Tank temperature and set points	• The PVT is used to charge the buffer tank mainly if the buffer tank temperature is lower than 55 ºC, where it is heated to 60 °C. • The PVT pump is used when the solar radiation is above 700 kJ/hr.m2 and the PVT flow is recirculated so that the buffer tank inlet temperature is higher than the tank. • Any excess energy present in the buffer tank is transferred to the BTES when the buffer tank temperature is higher than 35 °C until the buffer tank temperature drops to 30 °C	The set point temperature of the tank and BTES can vary between the range of 2-5°C from the current setting, based on weather and energy cost prediction.
Hot Tank	Tank and domestic hot water set points	• If the hot tank temperature is lower than 60 °C, it is heated to 65 °C by the heat pump. • Domestic hot water is provided at 60 ºC.	The set point temperature of the hot tank and domestic hot water can vary between the range of 2-5°C from the current setting, based on weather and energy cost prediction.

Heat pump	Source for Heat Pump	• The heat pump takes energy from the buffer tank or BTES at maximum 25 °C to charge the hot tank.	The set point of heat pump can vary between the range of 2-5°C from the current setting, based on weather and energy cost prediction.

4.1.2 Organization layer

The automation process for file processing of inputs and outputs, communication with weather forecasting providers, connections between MPCs' outputs and BMS, in addition to system integration will be developed by Tom Allen Senera. Several inputs have been considered, such as weather conditions, energy cost and occupancy profiles, as mentioned in Table 2. The horizon of these predictions will be 24-48 hours. The most likely choice as a weather forecasting provider would be a non-commercial website.

The MPCs are likely to be used to maintain the room temperature, with the aid of temperature sensors. A black box modelling technique is proposed for temperature forecasting, along with data from the Norwegian Meteorological Institute. Based on the data from the sensors in the building, the prediction values for the next hours and estimations the controls of the heat pump can change to increase the user comfort and reduce the energy cost.

Gebwell Oy along with Tom Allen Senera Oy has been developing a machine learning algorithm (recurrent neural network) that takes weather forecasts and predicts the indoor temperature into the future. They have been trialling the algorithm with their Aries size heat pumps, used in single-detached houses. In addition, they have trialled several different prediction periods, and a conclusion was derived that predicting 8 hours into the future seemed to give the best results. Furthermore, it is necessary to measure the indoor temperature to ensure that the indoor living conditions do not suffer significantly (or stay within agreed limits). If future indoor temperatures are predicted with sufficient accuracy, this algorithm could be used for demand response purposes as well.

4.2 MPC element in Belgian demo

4.2.1 Approach

The development of energy management strategies for the demos in EXCESS involves developing optimization algorithms that consider all the system components. For the Belgian demonstrator, the following components are included: both innovative prototype and commercial heat pumps, various thermal buffers, PVT panels, wind turbine, space heating satellite systems and domestic hot water boilers. In the near future, a stationary battery system and an EV charging infrastructure will also be installed.

To develop an energy management strategy for such a complex system, one needs to account for the interactions between the components, as well as the coordinated optimal operation of each of these components, with respect to the global objective. This can be achieved by using a Model Predictive Control approach.

In the MPC framework applied at the Belgian demo site, all models are represented as linear models by using the standardized state-space matrix representation. It states that the relationship between inputs and outputs of any linear model can be described by a state-space representation in the following form:

Equation 1: $State - space\ representation$

$$\dot{x}(t) = A.x(t) + B.u(t)$$
$$y(t) = C.x(t) + D.u(t)$$

With **x:** vector of all n states [n x 1] **A:** state matrix [n x n]

u: vector of all p inputs [p x 1] **B:** input matrix [n x p]

y: vector of al q outputs [q x 1] **C:** output matrix [q x n]

D: feedthrough matrix [q x p] or zero matrix in case no direct feedthrough

$\dot{\boldsymbol{x}}(\boldsymbol{t}) := \frac{d}{dt}x(t)$

In this representation the elements of the matrices can be time dependent, and the time variable t can be continuous. To use it in a simulation environment or in a real pilot however, we assume the matrices are time-invariant and we discretize to small fixed timesteps, e.g. 5 minutes, 15 minutes or 1 hour. In this case *Equation 1* can be rewritten as:

Equation 2: Discrete time-invariant state-space representation

$$x(k+1) = A.x(k) + B.u(k)$$
$$y(k) = C.x(k) + D.u(k)$$

The new state x at step (k+1) is dependent on the previous state x(k) and the inputs applied to the model at the current timestep u(k). The feedthrough matrix D can be used if it is necessary to output extra information of the model which cannot be found in the state variables. VITO already had some experience in developing a generic framework able to work with this type of models from the work done in the IndustRE project [25]and Res4Build [26].

Describing all linear models in this form makes it possible to combine and interconnect different linear models together into another hierarchical linear model.

4.2.2 State-space representation building model example

Taking the example of a first order RC building model, shown in *Figure 4*, of which the ordinary differential equation (ODE) is given in EXCESS deliverable 2.6 [1]:

- State (x)
  - T_i: indoor temperature [°C]
- Input (u)
  - Q_h: heat input into the building [W]
  - T_e: outdoor temperature [°C]
- Output (y)
  - T_i: indoor temperature [°C]
- Parameters

- ○ C_i: thermal capacitance of the indoor air [J/K]
- ○ R_{vent}: thermal resistance (wall) between indoor air and outdoor air [K/W]

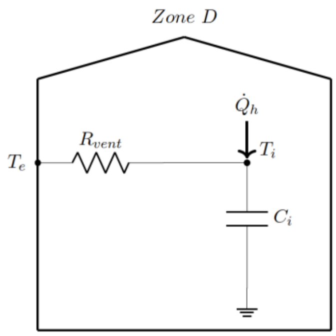

Figure 4: First order RC building model

This leads to the following state-space representation:

$$x(k+1) = \left[\frac{1}{C_i.R_{vent}}\right].T_i(k) + \left[\frac{1}{C_i.R_{vent}} \quad \frac{1}{C_i}\right].\begin{bmatrix} T_e(k) \\ Q_h(k) \end{bmatrix}$$
$$y(k) = [1].T_i(k) + [0 \quad 0].\begin{bmatrix} T_e(k) \\ Q_h(k) \end{bmatrix}$$

In this example the feedthrough matrix D is the zero matrix as the output only consists of the state T_i and does not include any hidden states. Using this representation, any linear model can be added or connected to another linear model in our framework, an example is given in the next section.

4.2.3 Combined linear models

In a building environment, often similar types of models will have to be combined, for example buildings which are heated by a Heat Pump (HP), buildings having a DHW buffer, etc. Using these combined models makes it easy to change the parameters of both the building model and the HP/buffer model individually, creating a new combined model for a different setup in an instant. *Figure 5* shows a simple model combining a fixed Coefficient of Performance (COP) HP model with a building model.

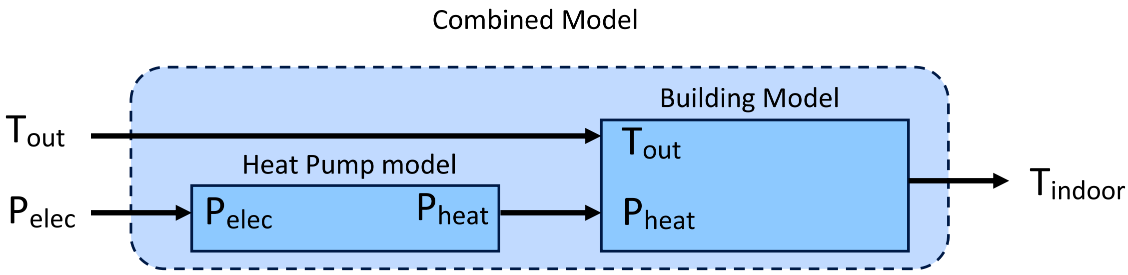

Figure 5: Combined building and HP model

If both sub models are available, the inputs/outputs of the existing models need to be connected in order to instantiate the hierarchical combined model. For the example in *Figure 5*, the following relations are defined:

- Global input to sub model input

- T_{out} is the outdoor temperature of the building model
- P_{elec} is the electric power of the HP model

- Sub model output to sub model input
  - P_{heat} is the heat power output of the HP model going into the building model
- Sub model output to global output
  - Indoor temperature of the building model is the global output T_{indoor}

The outcome is a combined linear model which again can be used in combination with other individual or combined models, enabling the creation of nested linear models. Looking at the state-space representation of a combined linear model, the different individual ABCD matrices will be combined into one large matrix. Taking the example of the combined model in *Figure 5*, the matrices will look like:

$$A_{combined} = \begin{bmatrix} A_{building} & 0 \\ 0 & A_{HP} \end{bmatrix} = A_{building} \quad (A_{HP} = [\,])$$
$$B_{combined} = \begin{bmatrix} B_{building} & 0 \\ 0 & B_{HP} \end{bmatrix} = B_{building} \quad (B_{HP} = [\,])$$
$$C_{combined} = \begin{bmatrix} C_{building} & 0 \\ 0 & C_{HP} \end{bmatrix} = C_{building} \quad (C_{HP} = [])$$
$$D_{combined} = D_{building}$$

The combined state matrix is equal to the state matrix of the building because the HP does not hold any state information in this case. Because of this, the input matrix B of the HP is also empty as the input to the HP model does not influence the (non-existent) HP state. The combined feed through matrix D is also equal to the matrix D of the building despite the fact that matrix D of the HP is not empty, this is due the fact that only the output of the building model (T_{indoor}) is an output of the combined model while output P_{heat} from the HP is not. The combined matrices, as shown above, are just a combination of the different individual matrices.

By combining different types of models, we can configure various setups to be used in our Model Predictive Control (MPC) approach, this is described in the next section.

4.2.4 Model Predictive Control (MPC)

In order to calculate an optimal control plan for the different models, based on a certain business objective or use case, these models will be used in an MPC approach.

MPC is a method for controlling a process satisfying a set of input and state constraints within an optimal control setting. An important feature of this approach is that it allows to optimize the current timeslot while keeping constraints of future timeslots into account. It is based on an iterative, finite-horizon optimization and relies on:

- The internal dynamic models
- Constraints on the inputs and states of the models
  - $\mathrm{u_{min}} < u(\mathrm{k}) < \mathrm{u_{max}}$
  - $\mathrm{x_{min}} < x(\mathrm{k}) < \mathrm{x_{max}}$
- A cost function to optimize over the optimization horizon (N) (for example cost optimization)

- $min \sum_{k=0}^{N-1} u(k) * cost(k)$

- Forecasts of the non-controllable inputs [$\hat{d}$]
  - Weather data: solar irradiation, outdoor temperature, etc.
  - Uncontrollable load
    - Electricity consumption
    - Hot water consumption
- State estimator
  - If the building has a hidden state, e.g., building mass, the state estimator will estimate this hidden state based on data of the visible state

Figure 6 shows a schematic overview on how the above MPC components can be used for optimal building control [27]. The building model gets two types of inputs: the current (estimated) state of the building and the forecasted non-controllable inputs such as weather and consumption forecasts. On top of this an objective function is applied, e.g., cost optimization, which is extended with the input and state constraints to form the complete optimization problem. The outcome of this optimization problem is an optimal input profile for the complete optimization horizon N. Once this is available the input of the current timeslot k will be applied on the building. At the same time the real non-controllable inputs d(k) will have their effect on the building. Both the impact of u(k) and d(k) will lead to a new building state x(k+1). This new state will then be used by the state estimator to estimate the new state of the building and the complete MPC cycle will start again.

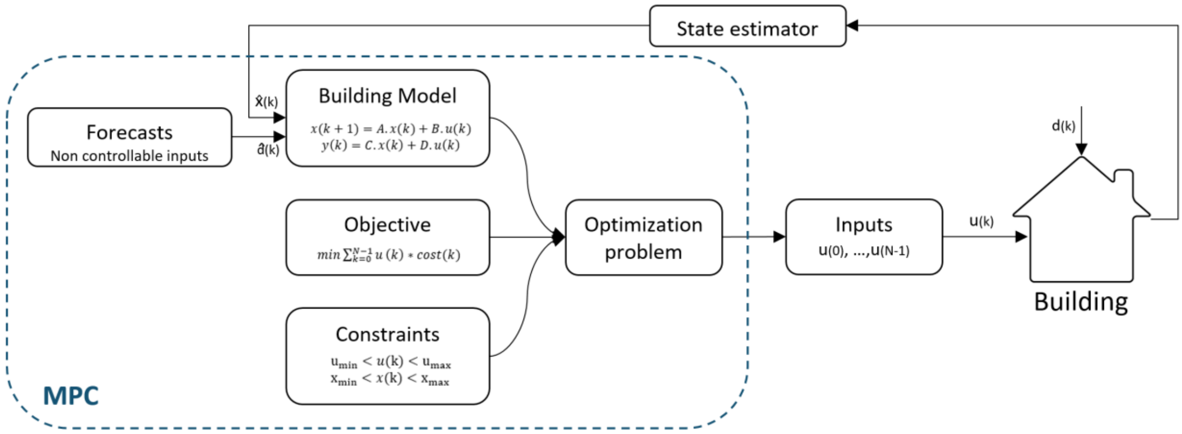

Figure 6: Schematic overview of an MPC system for building control

In the Belgian EXCESS pilot, the above sequence will be used for controlling the different assets in a configuration as described in *Figure 7*.

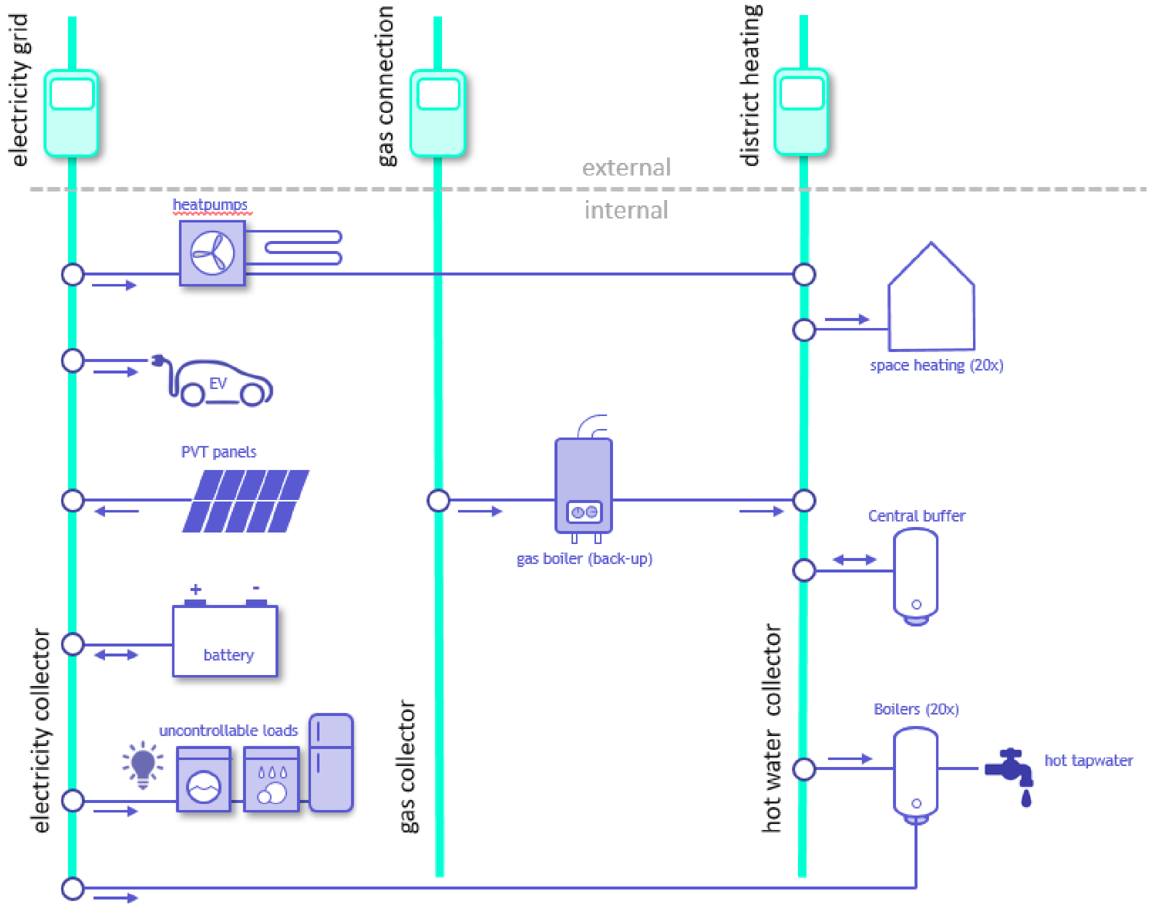

Figure 7: Schematic overview of the MPC models configuration for the Belgian demonstrator

As it is optimized for multiple timesteps in each iteration, the total combined matrix of the system would be extended to take into account all timesteps in the optimization horizon.

These matrices can become very large in more complex systems. Once these matrices are constructed, the input, state and output constraints are defined for each timeslot. This is done using CVXPY [28], [29], which is a Python-embedded modeling language for convex optimization problems. These constraints are then transformed by CVXPY into the restrictive standard form that is required by the underlying solvers it uses.

An advantage of using CVXPY together with its underlying solvers is that it can handle very large sparse matrices. As an example, an optimization problem consisting of an hourly 1-year time horizon can be solved in minutes using a commercial solver like Gurobi [30].

Now that the model dynamics and constraints are added to the optimization problem, the next section will discuss the optimization objectives that can be applied on the problem.

4.2.5 Optimization objectives

The main use case implemented in the optimization framework is a generic cost optimization objective. This implies that multiple price signals can be applied on one or more inputs and or outputs of the system. It allows for defining different price signals for both offtake (positive values, cost) and injection (negative prices, return) in the system, the latter if the system includes a local electricity producer such as solar panels. When electricity is taken from the grid, the offtake price will be taken into account, if electricity is fed into the grid, the injection price is applied. The following formula shows the cost optimization objective in case of both offtake and injection of electricity:

$$min \sum_{k=0}^{N-1} P_{injection}\,(k) \times price_{injection}(k) + \ P_{offtake}(k) \ \times price_{offtake}(k)$$

$$with\ P_{injection} \ \geq 0\ and\ P_{offtake} \geq 0$$

Besides cost optimization, we can also calculate the business case of maximizing local self-consumption. This can be obtained by using the cost minimization objective and defining a high positive price on injected power. Because we are minimizing cost, this implies that local production will be consumed on site as much as possible.

Based on the optimization objective, an optimal plan is calculated as an output of the optimization problem. The next section explains how this plan will be used in a rolling horizon approach for controlling a real building setup.

4.2.6 Rolling horizon control

The main outcome of the optimization problem is an input sequence that needs to be followed by the local BMS or controllers. When controlling a real system, this input sequence is sent to the system components (building, boilers, storage tanks, heat pumps etc.) by using an API. Next, the optimal input sequence is translated into the correct underlying control signals required by the different devices in the building by a local PLC system or gateway. Optimal planning normally happens in day ahead mode with time horizons of 24 or 36 hours (in steps of 15 minutes). Once an input sequence for the complete horizon is available, it is sent to the local PLC which will convert the profile and send the control signals to the building according to the time schedule of the profile. The entire process described above is then repeated in a rolling horizon fashion. That is, after a predefined interval, for example 1 hour, the internal states of the building are queried again, and an updated planning is calculated. The resulting input sequence is then activated for the next period of 1 hour until the next planning is done. This is visualized in *Figure 8*.

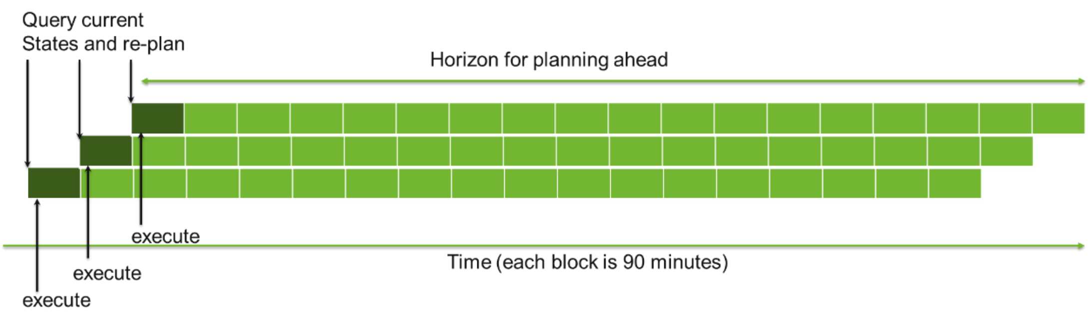

Figure 8: Rolling time horizon principle

4.2.7 Software

The optimization framework is implemented using Python 3.6 [31]. An important part of the solution is data structuring and cleaning, this is mainly done using the pandas library version 1.1.2 [32]. In order to write out the complete optimization problem, version 1.1 of the CVXPY modelling language for convex optimization is used [28], [29]. This package can call different underlying solvers, both free and commercial. For the optimization objectives defined for the Belgian Excess demonstrator, we mostly use the Gurobi solver [30] to calculate the optimal solution.

4.3 MPC elements in Austrian demo

The current status of the MPC development for the Austrian demonstrator has passed the conception stage and most sub-features are already in development, in some parts already in a testing phase. This section shall provide an overview of the MPC framework's structure and describe the used software components and technologies.

The aims of the MPC technology used within the Austrian demonstration case can be divided into two main categories. Firstly, seen from an **energy centric perspective,** we want to achieve the maximum local solar energy share for the thermal conditioning of the building. This implies the active usage of building mass storage which is readily available due to the innovative façade integrated heating system. The optimal utilization of this thermal storage capacity by harvesting flexibility out of user's comfort bandwidth is therefore a major objective. Secondly, when taking a **user centric perspective**, other needs like the minimization of energy costs and maintaining comfortable living conditions are most important. Also, active behaviour feedback and a community effect will be pushed by the usage of predictive energy systems.

As the energy focused objectives need more external and supervisory information (e.g., from energy Markets, DSOs, PV and other local RE generators, average building temperature) and the user focused part demands for more detailed thermal zone information (e.g., temperature, shading status, occupancy) and also personal information (e.g., personal thermal comfort boundaries, willingness to accept flexibility, occupation habits), it was decided to split the MPC task into different system layers. This reduces mathematical model sizes and allows for more scalability of the solution as the energy focused optimization can be seen as a supervisory MPC that delivers a demand response signal to all single Zone smart device MPC units. In *Figure 9*, the proposed system structure is illustrated including the two MPC Layers, Forecast-, Base Control-, and User Layer. In the following sub-chapters these individual Layers are described accordingly. Communication between the layers is web based and relies on REST-APIs.

This current state differs slightly from the one previously described in deliverable D3.1. Other buildings on the area which were planned to be integrated into black- or grey-box model based MPC by JR are now only treated via the prediction of load profiles without control functionality. This focus on prediction is now the core of the forecast layer which is described in section 4.3.1. The above-described splitting of the building internal MPC into two cascaded layers is a decision based on experience during development and states a difference to the report in deliverable D3.1.

4.3.1 Forecast layer

Since the basis for all MPC levels is the forecast layer, it is necessary to create these forecasts as precisely as possible. The Forecast Layer consists of Weather, Load, PV, and Radiation. The weather module is responsible for forecasting air temperature, cloudiness, and radiation. In order to ensure a high quality of the weather forecasts, various established weather models (ICON, GFS, ECMWF, MOSMIX) are compared and statistically processed. Energy profiles for commercial and domestic buildings are estimated for the Load Forecast. The PV Forecast will use AI and statistical methods to predict the consumption and production of PV energy. Since the solar radiation on the building at a certain position of the sun is a relevant factor for the heating of the facade and the interior, the shading or solar radiation on the building is calculated on a daily and seasonal basis using a digital surface model with a resolution of 1m². This radiation calculation is also based on the weather forecasts and weather variables such as the global horizontal irradiance (GHI). The result is a 2-day

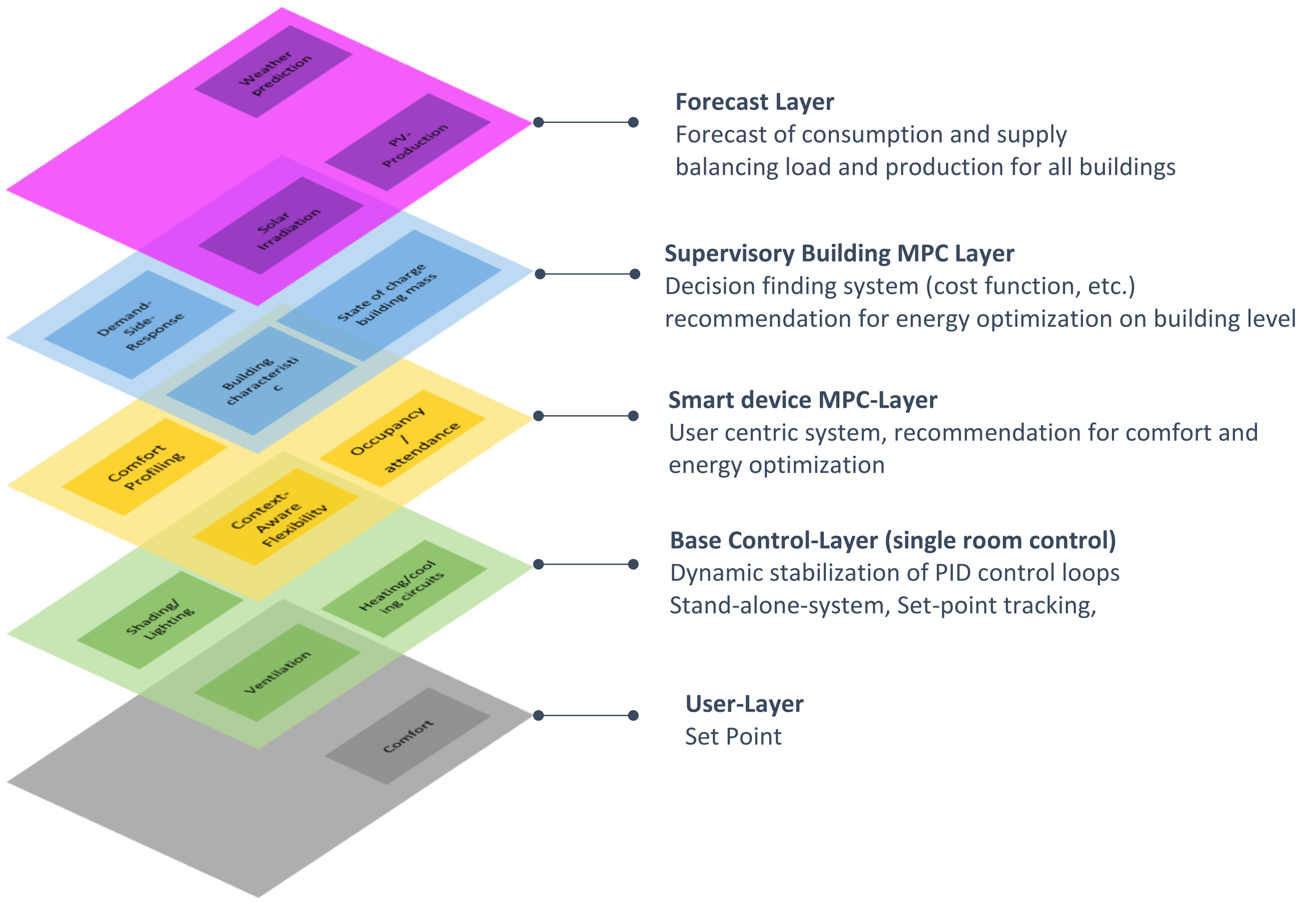

Figure 9: Layered implementation structure for Austrian demonstrator control scheme. Source: AEE INTEC.

forecast of solar radiation in hourly resolution, with the first hours being calculated in 15-minute resolution. In *Figure 10* a flowchart can be found that visualizes this process and Table 4 lists the used software components.

Table 4: Used software components in the forecast layer

	Software tools
Environment	• Docker container with Linux Ubuntu20 • Programming languages: Python 3.8.5, R 4.1.2
Database	• postgreSQL • Database Client: pgModeler
API	• Gunicorn as HTTP server and flask as middleware, nginx as proxy
Weather forecasts	• Main Python libraries: pvlib, dwdGribExtractor, wetterdienst
Radiation forecast	• 3D Models: EnergyPlus

	• 3D Model Visualization and GUI: SketchUp 2017 with Euclide extension • Geographic information system to handle georeferenced data like digital surface models: QGIS 3.22.1 • Main Python libraries: pvlib, pyrano, geomeppy, eppy, mpl_toolkits, osgeo (Geographical files handler) • External command line programs: radiance-online
PV forecast	• Main R libraries: solaR, mgcv, suncalc, lubridate, matrixStats • Main Python libraries: pvlib, rpy2 (R <-> Python interface), sklearn (ml models), dill (object loader)

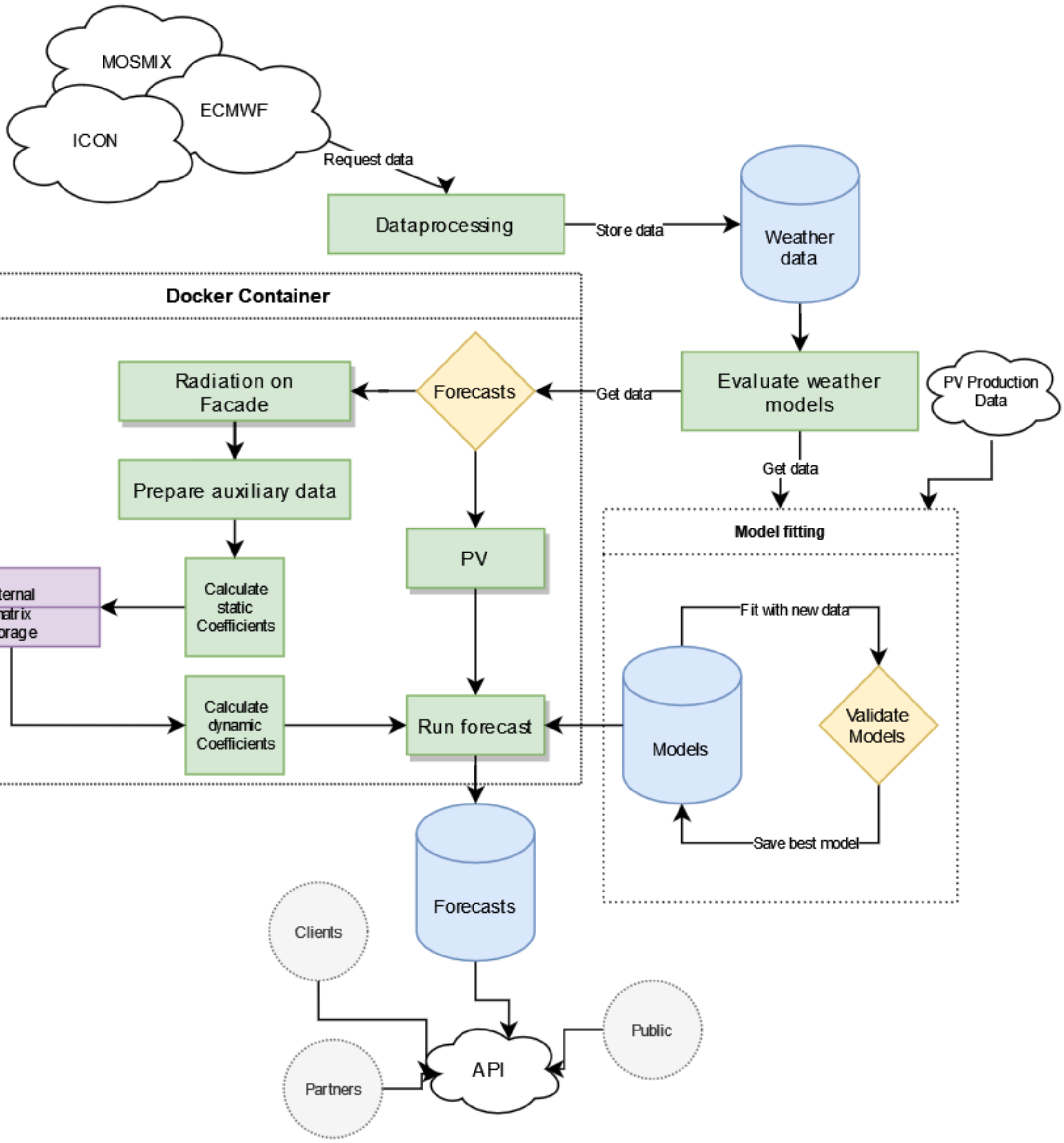

Figure 10: Simplified flowchart of the forecast infrastructure. Source: JR-LIFE.

4.3.2 Supervisory building MPC layer

The supervisory building MPC is a control system which is delivering the optimal operation strategy for heating and cooling systems from the energy centric perspective and delivers a demand response control signal. It features a semi physical grey-box model which represents the entire Building as a big thermal storage with the heating system and the thermal comfort as boundary conditions. The optimization goal is to utilize as much locally generated renewables as possible, which is reached by using the thermal capacity to store heat in times of high production and letting it dissipate into the zones during times of low production.

The core functionalities of the Supervisory Building MPC Layer are already implemented and currently in a testing phase with the help of an IDA ICE building co-simulation. *Figure 11* schematically shows the structure of the control loop. The MPC algorithm and all connected data exchange is based on a 15-minute interval whilst the co-simulation is using shorter and dynamic timesteps to increase the details on transient events in the building thermals and HVAC system simulations. The duration of the MPC internal prediction horizon is 48 hours with increasing timesteps. The whole application is Python-based and uses GEKKO Optimization Suite [33] as the main modelling and optimization tool. The core parts are listed and briefly described in Table 5.

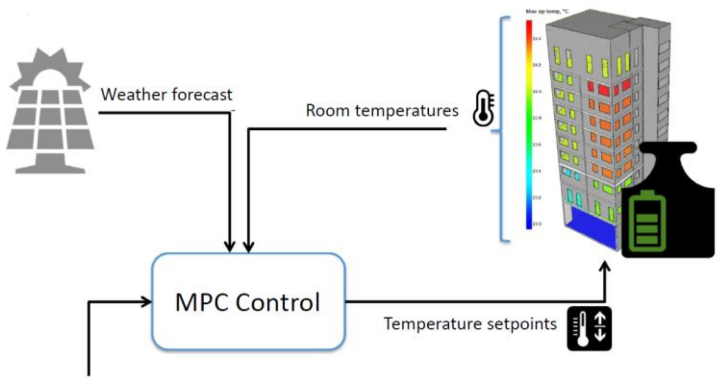

Figure 11: MPC controller testing strategy with co-simulation. Source: AEE INTEC.

Table 5: Subcomponents of supervisory building MPC

	Functionality	Software used	Status
Excitation tool	Uses PRBS or continuous random set-point temperatures to excite either co-simulated or real buildings to obtain time series data which can be used in system identification. This is a temporary solution to test the workflow and models as long as no real measurement data is available from the building.	IDA ICE 5.0b21 Python 3.8.8	testing
Building model structure class	Defines the structure of the control oriented model which is used by system identification, MPC and MHE applications. This central management allows fast and easy adoptions of the semiphysical model structure for the whole Layer.	Python 3.8.8 Gekko V1.0.2	testing

System identification tool	Used on historic measurement data or recorded co-sim data this tool provides parameter values which can be used to initialize the MPC core routine or for model development.	Python 3.8.8 Gekko V1.0.2	testing
MPC Core	Contains the main MPC routine which can be used for co-simulation or raltime use.	Python 3.8.8 Gekko V1.0.2	testing
MHE Core	Provides parameter updates for every control cycle of the MPC to reduce model missmatch and adopt to seasonal effects.	Python 3.8.8 Gekko V1.0.2	testing
Forecast data import	Data import function called by the main MPC routine to gather external forecasts form various sources. For the co-simulation case it provides artificially disturbed pre-known data.	Python 3.8.8 API	development
Control data export	Data export function triggered by the main MPC routine to transfer control signals to either the real building control or the co-simulation.	Python 3.8.8 API	development
Model forecast data storage	Used to store all import, export and state variables including their predictions at every control iteration. This information is used to analyse the model fidelity and prediction performance.	Python 3.8.8 PostgreSQL	testing
Data visualization and performance analysis	Web based tool to access historic controller data from the database and mend it into charts which allow performance evaluation.	Python 3.8.8 PostgreSQL DASH	development

4.3.3 Smart device MPC layer

The Smart Device MPC Layer is the interconnection between Supervisory Building MPC, optimizing the energy management on building level, the Base Control Layer, responsible for maintaining the requested room comfort, and the User Layer, providing the user's comfort requirements and behaviour forecasts. Basically, it combines the energy optimization strategy at building level with the comfort requirements of the users, taking into account the control options of each room.

The development of the Smart Device MPC Layer is currently in the conceptual design phase aiming to separate the energy optimization strategy of the building and the dwelling. We expect two beneficial effects: First, the complexity of each MPC Layer is reduced, since the Smart Device MPC Layer will handle each dwelling separately. Second, by emphasizing the user centric comfort management, users might readily accept a high level of automation, thus, allowing the Supervisory Building MPC to take full advantage of user flexibilities.

In the first step the Smart Device MPC Layer will be developed as a cloud-based service, allowing an easy integration with the OBS App and provide a high level of flexibility concerning MPC frameworks. A future step might be the integration of this layer into selected embedded devices if performance results indicate that the computing capacity is sufficient.

The Smart Device MPC Layer will process each dwelling separately by using sensory data from the Base Control Layer, temperature, and shading limits from the OBS App and multiple forecasts: occupancy, Heat influx by electric loads, both from the OBS APP, outdoor temperature, heat by solar gains, and charging and discharging instructions for thermal storages provided by the Supervisory Building MPC. As output, the Smart Device MPC Layer will report to the Base Control Layer targets for room temperature, shading level, and possibly air exchange rate. It will also provide the Supervisory Building MPC with the expected energy consumption profile.

4.3.4 Base control layer

The base control layer consists of sensors, actuators, and control units. The logic that connects these components is combined in the room comfort control system. At room level, temperature, brightness, humidity, and VOC are measured directly via an interactive room control panel and forwarded to the room comfort control system. Indirectly, forecasts about user behaviour are available via the OBS App, see Task 3.6/D3.5.

The room comfort control manages the distribution of the heating/cooling loads, regulates the shading, and communicates with the ventilation control. In this way, the room comfort control automatically maintains the room temperature, brightness and air quality at desired values and enables automatic optimization as far as the users allow. Sensor data is widely available down to a granularity of seconds.

To enable this optimized control, the room comfort control uses predictive control logic based on a physical model. A specific room temperature is achieved by balancing heat fluxes estimated from sensor data, internal and external forecasts via a room-dependent factor. This factor is re-evaluated at regular intervals and is similar to the heat-transfer coefficient.

The room comfort control in combination with the room control panel is setting hard boundaries for target values, such as temperature and shading level. It can accept external control suggestions from a connected MPC, as far as the user granted automation ranges for explicit values like temperature and shading.

4.3.5 User layer

The user layer focusses on two interfaces, the room control panel and the OBS App. The room control panel allows a manual override of control strategies. The OBS app provides modules to enable the user to configure the automation limits for heating and the degree of automation for shading. Both directly affect the room comfort control. Additionally, users can schedule their electricity consumption, which in turn provides feedback on heat generation in the dwelling. Based on the activity of the user, the provided data is used as forecast for MPC. A detailed description can be found in Deliverable D3.5 covering the OBS App.

4.4 MPC elements in Spanish demo

MPC are often used to help controlling complex system in real-time. In buildings the thermal inertia of the building, the HVAC systems and the energy fluxes makes this MPC philosophy not feasible. An interesting and more productive approach is to pre-simulate a set of control options, applied to the building operating at a predicted meteorological condition. This way, the energy consumption and other outputs can be optimized taking advantage of the Building Model, overcoming the restrictions of real-time simulations for a building.

The functional structure will be:

- A climate prediction Module.
- A building thermal behaviour simulation Module.
- A results, outputs and inputs files management Module.
- An interface Module with the physical control system of building.

Synthetically, the Spanish Demo MPC Scheme can be organized in three basic layers as depicted in *Figure 12*:

- Simulation Layer: this layer is based on TRNSYS software.
- Optimization Layer: this layer is based on GenOpt software.
- Organization Layer: this layer is responsible to manage different inputs and outputs (text files) and the connection between the MPC and Building Management System (BMS). This layer is still under development, and it could be based on Python Scripts or *.bat files.

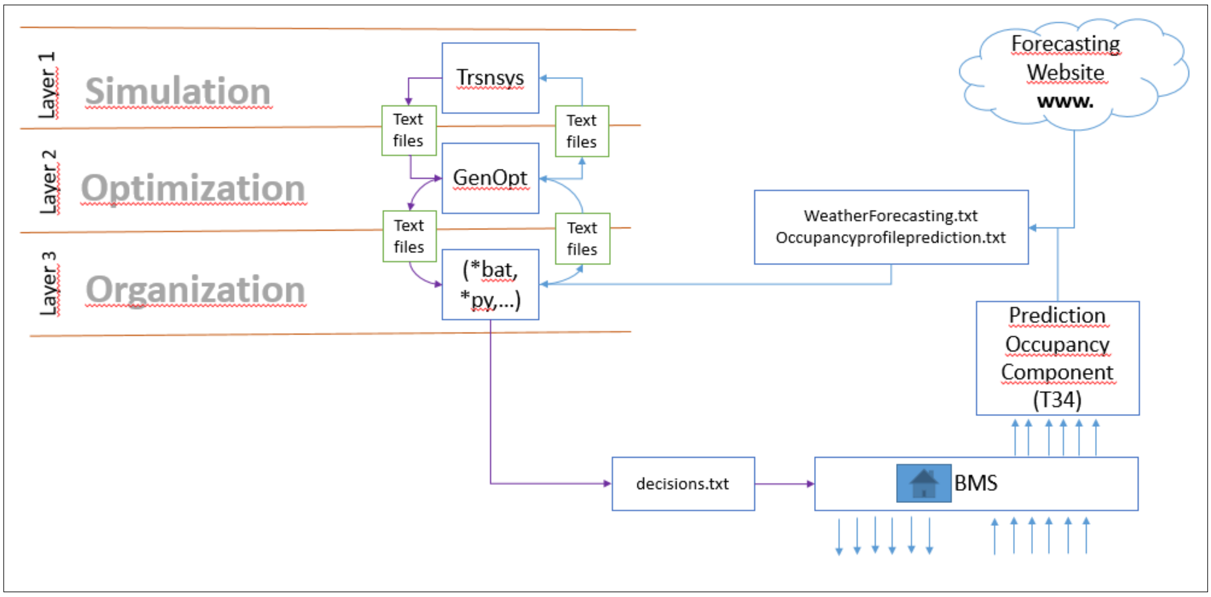

Figure 12: MPC Spanish Demo Scheme. Source: CENER

Below, the current situation of MPC development for Spanish Demo will be explained.

4.4.1 Simulation layer

4.4.1.1 Virtual building model

Since the development of a TRNSYS model for the building was carried out in Task 2.6, the virtual building model of the MPC will make use of it. .

Several modifications have been introduced in the last few months, compared to the model developed in Task 2.6., to enable that the model can read weather forecasting and occupancy profile prediction files.

In the version developed by Task 2.6, a statistical meteorological year and a typical occupancy profile based on Spanish Building Code were considered. These components (types) have been replaced by readers of text files. So, the model is enabled to read these predictions in text format files (*.txt files).

On the other hand, some changes in the model might have to be made after the final development/design of building and their facilities to reflect the behaviour of building as well as their HVAC facilities and control system with high accuracy. The final building design could suffer changes after the writing of this report.

Also, after the building construction have been finalized and the inhabitants are living in, the Virtual Building Model would need to be validated with real data (in real operation).

4.4.2 Optimization layer

4.4.2.1 Optimization tool

The selected Optimization tool is GenOpt. This software allows multidimensional optimization of an objective function.

The optimization will be done by systematic variation of specified design parameters in order to minimize the objective function.

The GenOpt´s interface is easy to use, and the program can be coupled to any simulation program (such as TRNSYS, EnergyPlus, Modelica etc.) that reads its input from a text file and writes its output to a text file

4.4.2.2 Algorithm for optimization

At this early stage, the algorithm selected for the optimization process is GPSPSOCCHJ algorithm.

The algorithm GPSPSOCCHJ is a hybrid multidimensional optimization algorithm which uses generalized pattern search (GPS) for the first stage search and particle swarm optimization (PSO) Hooke Jeeves algorithm as a fine search for the defined discrete and continuous variables function solution.

4.4.2.3 Objective function

The objective function is still to be decided because this function is highly linked to the actual possibilities of building control system.

The most probably option is minimizing the energy consumption while preserving comfortable internal conditions. So, the cost function will be defined in such as a way as to penalize the degree of thermal discomfort of the occupants. The proposed approach is a multi-objective optimization, gathering in a single value or merit figure, low consumption and comfort, which are opposite. The objective function will be a comprising decision balancing between these two criteria.

In case the final cost function will be those described above, the building control will act about the operation setpoint temperature of heat pump(s). At this early stage, the range of variable values is to be decided.

However, another objective function could be considered as minimizing the energy cost (€/kWh) taking into consideration a daily energy prices profile.

4.4.3 Organization layer

4.4.3.1 Automation process

The automation process for file processing of inputs and outputs, communication with weather forecasting providers, connection with EXCESS Data Analytics Framework (Task 3.3 & 3.4), connections between MPC's outputs and BMS will be developed by the IT CENER Department.

4.4.3.2 Inputs -forecasting

Several inputs have been considered such as weather conditions and occupancy profiles. The horizon of these predictions will be 24-48 hours. The most likely choice as weather forecasting providers would be a non-commercial website. By the time being, several websites (national and international) are being checked. The needed variables predictions for Model about weather conditions are presented in Table 6.

Table 6: Weather forecasting variables and possible providers. Source: CENER.

Prediction Variables	Name	Units	Sources	Comments
1	Outdoor Temperature	º C	www.meteogalicia.gal www.ecmwf.int	To be determined
2	Relative Humidity (or dew temperature)	%	www.meteogalicia.gal www.ecmwf.int	To be determined
3	Direct Normal Beam Radiation	W/m^2	www.ecmwf.int ¿others?	To be determined
4	Total Radiation on horizontal surface	W/m^2	www.ecmwf.int ¿others?	To be determined

Because of the current status of Spanish Demo, it could be the most available option that the occupancy profile would be provided directly by users. In this sense, the users of every dwelling could provide a file in which the most probably occupancy for the following day in a format of text file.

However, if real data for inhabitants is available, we could be provided with profile patterns calculated by the EXCESS Data Analytics Framework.

5 Conclusions

This deliverable discusses the activities of Task 3.5 towards the EXCESS Model-Predictive Control algorithms. The outcome of this report is the description of the tools used for the development of the MPC for each demo case. Afterwards, this is implemented at the demo site in a real scenario, depending on the local conditions. This deliverable comprises a direct outcome of the D2.6 (Report on advancing simulation-based energy performance assessment for optimal PEB design), D1.1 (PEB as an enabler for the consumer-centred clean energy transition: shared definition and concept) and D3.1 (EXCESS ICT Architecture Blueprint).

The Austrian and Finnish demo cases use IDA-ICE for the building simulation, and the Spanish demo case uses TRNSYS software. The Finnish and Spanish demo cases use TRNSYS, and the Austrian demo uses IDA-ICE to model and simulate the thermal and electrical system. In the Belgian demo case, the RC model is used to model the energy system and building models. Depending on the local requirements and challenges different MPC and control optimization is implemented in each demo case. The weather and the energy cost data will be predicted based on open-source data available. Depending on the local use case, this data will then be input into the simulation software. Based on this, the optimization study is carried out for a better control strategy if needed. The selected Optimization tool for the Spanish case is GenOpt. This software allows multidimensional optimization of an objective function. At this early stage, the GPSPSOCCHJ algorithm is selected as the algorithm for the optimization process. The objective function is still yet to be finalized. The most likely option is minimizing energy consumption while preserving comfortable internal conditions. In the Finnish demo, the selected optimization tool is MOBO. This software can be integrated with TRNSYS/IDA-ICE/PYTHON/MATLAB etc. The selected algorithm is the NSGA or Hooke-Jeeves. The objective function is either to minimize the operational energy cost or import electricity while meeting the energy demand of the building. In Finland and Spain, the MPC is likely to be implemented in two phases. In the first phase, simulation and optimization are carried out to identify the benefits of MPC. In the second phase, the MPC module is designed and implemented in the demo building. The benefit of this is that the outcomes of the first phase (simulation-based learnings) can support the second phase during the implementation of MPC in a real case environment. In the Austrian demo, it is planned to use GEKKO Optimization in PYTHON. In the Belgian demo, it is planned to use CVXPY modelling language for linear programming optimization. Some of the research work can continue further to evaluate the benefits of MPC and optimization algorithm if needed or if rule-based control is enough based on the requirements, ease of implementation and monetary value (such as in Spanish and Finnish demos). Afterwards these improved methods are implemented in the demo buildings.

The specific details about the MPC component and logic are discussed in section 4. The Spanish and Finnish demo building control methods are being developed as the building construction progresses. On the other hand, the Austrian and Belgian demo building control methods are at an advanced phase.

One lesson learned is that the MPC for each demo building has to be designed separately due to different local requirements such as weather profile, cost profile and technology used. It is important to achieve the PEB standard based on local requirements, as well as the evaluation of optimization of control strategies.

Another insight gained was that the user behaviour, energy demand for appliances/lighting, and set points can impact the performance of the energy system and the operational cost. Therefore, it is of paramount importance to engage the end-user to achieve the PEB requirements and cost reductions.

Table 7 provides a summary of the different approaches for achieving the Positive Energy Building by the partnering countries.

Table 7: Different approaches to system modelling and incorporating MPCs by the partners

	Spain	Finland	Austria	Belgium
Input data generation (prediction)	Weather data, occupancy profile	Weather data, cost data	Weather data	Weather data, Variable load
Optimization method	GPSPSOCCHJ	NSGA, Hooke-Jeeves	Mixed-integer and differential algebraic equations	Linear Programming
Optimization program	GenOpt	MOBO/ Python/MATLAB	Gekko	CVXPY, Gurobi
Modelling program	TRNSYS/ Python	TRNSYS/IDA-ICE/Python/MATLAB	IDA-ICE/ Python	RC Building Model / Python
Control of the component	To be decided	Set points of heat pumps, storage, domestic hot water	Room temperature, brightness and air quality	Building, boilers, storage tanks, heat pumps
Objective function	Energy Consumption / Energy Cost	Energy Cost / Imported (purchased) Electricity	Local solar energy share / Energy Cost	Energy Cost (Self - sufficiency)

Necessary refinements and further details on the functionalities of the different ICT components, controls and their implementation will be documented in the rest of the deliverables of the WP3. This report will provide input to other tasks of WP3, as it will define the controls strategies that would be implemented in the demos and WP4 as it will provide input for testing activities. A second version of D3,4 will be delivered in M42 with updates and any additional functionalities based on users feedback during testing in demo sites.

6 References

[1] V. Lara, H. U. Rehman, A. Laitinen, F. Hengel, K. Allaerts, and J. Diriken, "Deliverable 2.6: Report on advancing simulation-based energy performance assessment for optimal PEB design ," Nov. 2020.

[2] MathWorks, "Understanding Model Predictive Control - MATLAB & Simulink," *https://se.mathworks.com/videos/series/understanding-model-predictive-control.html*.

[3] M. L. Darby, M. Harmse, and M. Nikolaou, "MPC: Current Practice and Challenges," *IFAC Proceedings Volumes*, vol. 42, no. 11, pp. 86–98, 2009, doi: 10.3182/20090712-4-TR-2008.00014.

[4] R. Enríquez, M. J. Jiménez, and M. D. R. Heras, "Solar Forecasting Requirements for Buildings MPC," *Energy Procedia*, vol. 91, pp. 1024–1032, Jun. 2016, doi: 10.1016/J.EGYPRO.2016.06.271.

[5] Y. Yao and D. K. Shekhar, "State of the art review on model predictive control (MPC) in Heating Ventilation and Air-conditioning (HVAC) field," *Building and Environment*, vol. 200, p. 107952, Aug. 2021, doi: 10.1016/J.BUILDENV.2021.107952.

[6] T. Hilliard, L. Swan, and Z. Qin, "Experimental implementation of whole building MPC with zone based thermal comfort adjustments," *Building and Environment*, vol. 125, pp. 326–338, Nov. 2017, doi: 10.1016/J.BUILDENV.2017.09.003.

[7] J. Ma, J. Qin, T. Salsbury, and P. Xu, "Demand reduction in building energy systems based on economic model predictive control," *Chemical Engineering Science*, vol. 67, no. 1, pp. 92–100, Jan. 2012, doi: 10.1016/J.CES.2011.07.052.

[8] W. J. Cole, K. M. Powell, E. T. Hale, and T. F. Edgar, "Reduced-order residential home modeling for model predictive control," *Energy and Buildings*, vol. 74, pp. 69–77, May 2014, doi: 10.1016/J.ENBUILD.2014.01.033.

[9] J. Široký, F. Oldewurtel, J. Cigler, and S. Prívara, "Experimental analysis of model predictive control for an energy efficient building heating system," *Applied Energy*, vol. 88, no. 9, pp. 3079–3087, Sep. 2011, doi: 10.1016/J.APENERGY.2011.03.009.

[10] S. R. West, J. K. Ward, and J. Wall, "Trial results from a model predictive control and optimisation system for commercial building HVAC," *Energy and Buildings*, vol. 72, pp. 271–279, Apr. 2014, doi: 10.1016/J.ENBUILD.2013.12.037.

[11] D. O. Woo and L. Junghans, "Framework for model predictive control (MPC)-based surface condensation prevention for thermo-active building systems (TABS)," *Energy and Buildings*, vol. 215, p. 109898, May 2020, doi: 10.1016/J.ENBUILD.2020.109898.

[12] J. Clauß, C. Finck, P. Vogler-Finck, and P. Beagon, "Control strategies for building energy systems to unlock demand side flexibility – A review," *IBPSA Building Simulation 2017, San Francisco, 7-9 August 2017*, Aug. 2017, Accessed: Feb. 10, 2022. [Online]. Available: https://researchrepository.ucd.ie/handle/10197/9016

[13] D. Sturzenegger, R. Smith, and M. Morari, "Implementing MPC on a Building: Experimental results and a toolbox for fast modelling," Zurich, 2021. Accessed: Feb. 21, 2022. [Online]. Available: https://www.youtube.com/watch?v=IpZ-PHBXLj0

[14] M. Manni and A. Nicolini, "Multi-Objective Optimization Models to Design a Responsive Built Environment: A Synthetic Review," *Energies 2022, Vol. 15, Page 486*, vol. 15, no. 2, p. 486, Jan. 2022, doi: 10.3390/EN15020486.

[15] A. Vadiee, A. Dodoo, and L. Gustavsson, "A Comparison Between Four Dynamic Energy Modeling Tools for Simulation of Space Heating Demand of Buildings," pp. 701–711, 2019, doi: 10.1007/978-3-030-00662-4_59.

[16] M. Palonen, M. Hamdy, and A. Hasan, "MOBO A NEW SOFTWARE FOR MULTI-OBJECTIVE BUILDING PERFORMANCE OPTIMIZATION," Aug. 2013.

[17] H. ur Rehman, J. Hirvonen, and K. Sirén, "A long-term performance analysis of three different configurations for community-sized solar heating systems in high latitudes," *Renewable Energy*, vol. 113, pp. 479–493, Dec. 2017, doi: 10.1016/J.RENENE.2017.06.017.

[18] M. Palonen and A. Hasan, "MOBO Beta 0.3b Manual Version 1.4," Espoo, Jul. 2014. Accessed: Feb. 14, 2022. [Online]. Available: https://www.researchgate.net/publication/273307166_MOBO_Beta_03b_Manual_Version_14

[19] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002, doi: 10.1109/4235.996017.

[20] M. Hamdy, A. Hasan, and K. Siren, "A multi-stage optimization method for cost-optimal and nearly-zero-energy building solutions in line with the EPBD-recast 2010," *Energy and Buildings*, vol. 56, pp. 189–203, Jan. 2013, doi: 10.1016/J.ENBUILD.2012.08.023.

[21] "Hourly weather forecast in Helsinki (Finland) - 14 days." https://en.tutiempo.net/helsinki.html?data=hourly (accessed Feb. 14, 2022).

[22] "Sää Helsinki - Ilmatieteen laitos." https://www.ilmatieteenlaitos.fi/paikallissaa (accessed Feb. 14, 2022).

[23] "Solar radiation in Helsinki (Finland)." https://en.tutiempo.net/solar-radiation/helsinki.html (accessed Feb. 14, 2022).

[24] "Market data | Nord Pool." https://www.nordpoolgroup.com/Market-data1/Power-system-data/Production1/Procuction-Prognosis/FI/Hourly/?view=table (accessed Feb. 14, 2022).

[25] IndustRE, "Simplified assessment methodology for optimal valorization of Flexible Industrial Electricity Demand," 2016. Accessed: Feb. 18, 2022. [Online]. Available: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwil2MOFnon2AhV2SPEDHVfDBpQQFnoECAcQAQ&url=http%3A%2F%2Fwww.industre.eu%2Fdownloads%2Fdownload%2Fsimplified-methodology-for-optimal-valorization-of&usg=AOvVaw0H_8ohGTxBjGLIBfX5AkQK

[26] D. Geysen, J. Lago, and G. Suryanarayana, "D3.3: Optimization Strategies," 2021.

[27] J. Drgoňa *et al.*, “All you need to know about model predictive control for buildings,” *Annual Reviews in Control*, vol. 50, pp. 190–232, Jan. 2020, doi: 10.1016/J.ARCONTROL.2020.09.001.

[28] A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd, “A rewriting system for convex optimization problems,” *https://doi.org/10.1080/23307706.2017.1397554*, vol. 5, no. 1, pp. 42–60, Jan. 2018, doi: 10.1080/23307706.2017.1397554.

[29] S. Diamond and S. Boyd, “CVXPY: A Python-Embedded Modeling Language for Convex Optimization,” *Journal of Machine Learning Research*, vol. 17, pp. 1–5, 2016, Accessed: Feb. 18, 2022. [Online]. Available: http://www.cvxpy.org/

[30] Gurobi Optimization LLC, “GUROBI OPTIMIZER REFERENCE MANUAL,” 2020, Accessed: Feb. 18, 2022. [Online]. Available: https://www.gurobi.com/wp-content/plugins/hd_documentations/documentation/9.0/refman.pdf

[31] G. van Rossum, “Python tutorial.” Jan. 01, 1995.

[32] W. Mckinney, “Data Structures for Statistical Computing in Python,” 2010.

[33] L. D. R. Beal, D. C. Hill, R. Abraham Martin, and J. D. Hedengren, “GEKKO Optimization Suite,” *Processes 2018, Vol. 6, Page 106*, vol. 6, no. 8, p. 106, Jul. 2018, doi: 10.3390/PR6080106.