Modelica based Design and Optimisation of Control Systems for Solar Heat Systems and Low Energy Buildings

Stephan Seidel[1]   Christoph Clauss[1]   Jürgen Haufe[1]   Kristin Majetta[1]   Torsten Blochwitz[2]
Edgar Liebold[3]   Ullrich Hintzen[4]   Volker Klostermann[5]

[1] Fraunhofer IIS EAS, Zeunerstraße 38, D-01069 Dresden, GERMANY
[2] ITI GmbH, Schweriner Straße 1, D-01067 Dresden, GERMANY
[3] NSC GmbH, Äußere Zwickauer Straße 8, D-08064 Zwickau, GERMANY
[4] FASA AG, Marianne-Brandt-Straße 4, D-09112 Chemnitz, GERMANY
[5] Provedo GmbH, Schweriner Strasse 1, Mottelerstraße 8, D-04155 Leipzig, GERMANY
{stephan.seidel, christoph.clauss, juergen.haufe, kristin.majetta}@eas.iis.fraunhofer.de

Abstract

The goal of the research project enerMAT is the reduction of energy consumption and CO2 emissions of buildings. Especially solar heating systems are installed in more and more buildings. This paper introduces a novel approach for simulation and optimisation that aims to improve the performance of building controllers and especially solar heating controllers by simulation and model-in-the-loop tests. A new generation of energy-aware optimised building energy management systems (BEMS) will be discussed and its advantages over the older controllers highlighted. The energy-aware optimisation will be shown on a model-based approach with an overall building system model enabling the assessment of the energy performance for different design and operation alternatives of the building automation system in interaction with the building. This system model will allow a simulation-based, energy-aware, global, dynamic, multi-criterial optimisation of BEMS. In this paper, the idea, the approach, and the actual state of the project research is presented with a focus on solar heating controllers.

Keywords: Building, Energy Management, Solar Heat, Controller

1   Introduction

As mentioned in the public media and many scientific studies the energy demand of buildings is responsible for about 40% of the primary energy consumption (European Commission, 2008). This demand is caused by many energy consuming devices and systems such as lighting, water heating, and consumer electronics. The main consumer however is the heating system. There is a two-step approach to reduce the demand of fossil energy in buildings. The first step is to equip buildings with insulation layers and reduce the amount of energy that is emitted into the environment. This passive measure was and still is the main procedure for energy reduction but yields only a small positive effect on the overall energy balance in case the reduction is compared to the expenditure in manufacturing. Insulation however is a prerequisite for low energy houses as it reduces greatly the required heating power. But as explained in the following insulation should not be the only measure for saving energy. Instead of further improving insulation levels renewable energy sources should be employed to reduce the impact on the environment. This is the second step in which fossil fuel heating systems are replaced with renewable energy sources after insulation has been upgraded. Solar energy plays a major role as it is readily available and easy to harvest. Photovoltaic and solar thermal energy systems are installed in low energy houses in order to assist or replace fossil energy consuming systems. Other renewable energy harvesting systems such as heat pumps are an alternative or a support for solar energy. Heat pumps can harvest thermal energy (ground or air) although depending on the heat source their installation can be expensive. Micro wind turbines can be installed as well but are more difficult than solar systems. Their approval by authorities might require costly surveys, moving parts cause higher servicing costs and the noise might cause resentment with the neighbours. Therefore they are not very common in urban areas. The installation of solar energy systems is, depending on size and system, relatively cheap and requires only the installation area on a roof or other exposed area.

Currently almost every newly built detached house in Germany is equipped with a solar thermal heating system in order to meet requirements introduced with the Energieeinsparverordnung (Energy Savings Regulation) (BmWI, 2014). While most houses feature small solar thermal systems which support the warm water supply a growing percentage of the houses built are energy-plus or solar houses that generate most of the energy required for heating from renewable energy sources (solar and geothermal). These houses are in general very well insulated and rely heavily on an optimised heating system that does provide sufficient heat not only in summer but also in winter when the supply of renewable energy is generally low. Therefore it is crucial for these systems that their hardware and software parameters are adapted to the individual building where they are installed. There is a multitude of parameters (w.r.t. to solar thermal heating systems) including size and direction of collector array, type of collectors, threshold temperatures for switching, buffer size and charging procedure as well as feed line temperatures, occupation profiles and heating times and many more. Many if not all of these parameters are established during the planning and construction phase of the building and they are based mainly on experience, especially parameters in the controller. Once installed these parameters cannot be changed at all or only by experts. Therefore it is vital to establish optimal values during the development procedure by using simulation and optimisation.

A solar thermal heating system typically consists of a collector array, a huge water-filled buffer storing the thermal energy, a circulation pump on the source side and floor heating, hot water heating and circulation pumps on the consumer side. All of these devices are controlled by a dedicated stand-alone controller. Among others the universal controller 1611 (Universalregelung UVR 1611) from the Austrian manufacturer Technische Alternative is a well-

Figure 1: *Universal controller UVR1611*

known and widely used device. This device is programmed by using a PC and parameterising and connecting function blocks in the programming tool from a preconfigured library.

This paper will present an approach to use simulation and model-based optimisation to validate the planned solar-heating system and find ideal parameters for installation and control unit.

Furthermore this paper will highlight the use of simulation and optimisation for low energy buildings in general and for controllers for solar thermal heating systems in special. Section 2 will explain the controller in some detail and will also discuss the targets for simulation and the expected benefits. The following section 3 will shed some light on the approach for simulating the solar thermal heating system in conjunction with control systems and the surrounding house. This will be further elaborated in section 4 where two use cases and demonstration projects are presented. Section 5 will give an overview of the current status of the project and an outlook. The paper will conclude with section 6.

2 Targets

The UVR 1611 control unit introduced in section 1 is a typical controller for HVAC systems and was developed by the Austrian company Technische Alternative since 2000. It is tailor-made for HVAC and solar heating systems and features a wide set of functions for this purpose. The controller has 16 sensor inputs (typically temperature sensors), 4 speed outputs (e.g. for circulation pumps), 7 relay outputs (for opener/closer switches or valves), a CAN bus connection and several extension modules to add more relay outputs or LAN connectivity for viewing and operating the controller via internet.

Although described and marketed as freely programmable this should be understood in a different way. Programming languages such as IEC 61131-3's instruction list or ladder logic are not available. The closest equivalent in 61131-3 would be the function block diagram (FBD) although the function blocks in the UVR cannot be connected with each other. A UVR configuration (or sometimes called program) consists of several function blocks that process inputs and flag values and write outputs and flag values. Typical function blocks are provided by a library but it is not possible to write own function blocks. This is in most cases not necessary because typically the UVR is not programmed by an engineer but rather configured by a technician. Functions blocks are ranging from simple logic functions (AND, OR, FlipFlop), compare function, timers,

counters and clocks to control functions such PID control. Many function blocks are designed for typical HVAC control tasks such as heating circuit controller, mixer control, load pump control, solar control, boiler cascade control and many more. This concept is part of the UVR's success (tens of thousands sold units) because typical tasks can be solved by configuring a few function blocks instead of programming everything from scratch. The configuration, which is done on a PC and downloaded onto the device, is fairly easy to do. Function blocks are pulled from the library and the parameters are edited. Many function blocks provide switches for behaviour changing (e.g. P-, PI- or PID controller) and cover a wide range of required functionality. Internet forums are full of example configurations and also the manufacturer provides a range of configuration examples for download.

The UVR's success is due to its versatility and its scalability. Typical use cases are heating systems in detached houses that go beyond the standard natural gas powered heating system which are controlled by the integrated controller of the condensing boiler such as solar thermal heating systems, geothermal heating systems or older systems that require an extension or a retrofit. Within the enerMAT project UVR controllers were installed by the FASA AG as control unit for the heating system of so called solar houses and offices with the trade name ENERGETIKhaus100. These houses are not standard low energy houses, in fact they have a higher demand in heating energy than typical low energy houses, but the heat is provided entirely by renewable energy sources. Typically the UVR would control the solar heating system, its circulation pump and the corresponding valves, the loading of the buffer with solar heat and the circulation pump of the floor heating. The solar heating system in such houses by FASA has the aim of providing up to 92% of the heat required by the house and inhabitants by means of solar energy. The remainder is covered by a wood-burning stove. Typically one UVR is sufficient to cover all the pending control tasks in a detached house.

Other use cases, especially more sophisticated or larger installations can be resolved by installing several UVR units which communicate by CAN bus with each other. Hence UVR 1611 controllers can be installed wherever an off-the-shelf solution for heating systems is not available or not desired because of compatibility issues between components of different manufactures or missing features. Since the UVR 1611 is an older controller type it is currently phased out and replaced by a newer type, the UVR16X2

which is similar but not identical in form and function to the 1611 model.

3 Approach

As mentioned already in Sections 1 and 2 the lack of a methodology to obtain optimised parameters before the actual implementation on the HVAC controller leads to non-optimal behaviour of the heating system which could result in poor heating performance and energy wasting. Parameters are often estimated or based on empirical values which are not necessarily ideal for the particular heating system. Therefore a simulation-based commissioning and optimisation is required to obtain ideal functionality and parameters for the heating system controller such as the UVR 1611. Simulation or virtual commissioning is almost non-existent in the field of such control units. This is vastly different for industrial programmable logic controllers (PLC) such as PLCs for intralogistic systems (Seidel, 2012) which are tested and optimised on a virtual model of the plant before the real commissioning takes place. This has various advantages such as thoroughly tested and mature software with fewer bugs, shorter commissioning and project time and reduced costs. Unfortunately virtual commissioning doesn't play any significant role in building control systems. One of the enerMat project's aim was to provide engineers with the tools required for a continuous workflow for design, test and optimisation of building controllers and systems of which one subsystem the UVR 1611 is.

Several options for simulation, emulation and virtual commissioning have been analysed. The lack of a software simulator for the UVR 1611 controller prevents any form of software-in-the-loop (SiL) approach. Software available for the UVR is limited to the programming system TAPPS which features no simulation or emulation mode. If such an emulation tool would exist a co-simulation solution would be feasible. Hardware-in-the-loop (HiL) approaches would be feasible by connecting the hardware controller by means of an I/O interface and FMI to the simulation. The big issue however is the missing time synchronisation between controller and simulation tool. The hardware controller would always be restricted to real-time, and simulation runs over a long time span such as a year would be difficult because of the required amount of time. Therefore a model-in-the-loop (MiL) was favoured and found to be the most promising approach which avoids the error-prone coupling of tools or systems. By applying MiL the engineer can concentrate on the simulation tool and the designed controller and building

model. Other advantages are: symbolic preprocessing of the model and thus a more reliable and faster simulation.

To create a model of the UVR controller (and any other controller) a detailed functional description is required. This can be either the program code or a system specification. Since the source code was not available the function block specification (TA, 2014) was used as a basis for the design of the controller model. Such a model consists of two submodels. One is the functional model; the other is the behavioural model (Seidel, 2009). The latter describes how the controller works internally, for instance what amount of time is consumed for calculating new outputs from a change of inputs (cycle time) or in which sequence the user program is executed. Knowledge of this behaviour is important for fast processes such as manufacturing machines (Seidel, 2011) or car engines. In the case of building controls this can be neglected and the designed behaviour model has no cycle time because the controller's cycle time which is in the range of milliseconds is very small compared with time constants of the building and heating system (minutes to hours).

The functional model consists of the user program which is the function block structure with corresponding inputs, outputs and parameters. Therefore to create a model of the user program a library of the available function blocks is required. Components of this library are models of each function block having the same inputs, outputs and parameters as their counterpart in the real-world controller. The specification of each function block was analysed, inputs and outputs connectors were created and the function was modeled by means of algorithms and equations or in case of more complex blocks as a group of interconnected elements from the Modelica Standard Library (MSL). As already mentioned the complexity of the function blocks was very diverse. Simple blocks, such as the comparison block, required just a few lines of code as algorithm section as shown below.

```modelica
when valueA > valueB + diffOn then
    valAGreater := true;
elsewhen valueA < valueB + diffOff then
    valAGreater := false;
end when;
valALower:=not valAGreater;
if enable then
  valAGreaterOut := valAGreater;
  valALowerOut := valALower;
else
  valAGreaterOut:=false;
  valALowerOut:=false;
end if;
```

Modelica Code fragment of comparison block

It is worth noting that many of the function blocks implement margin parameters for signal inputs to filter noise or minor oscillations and to prevent a frequent switching of outputs.

On the other hand, the PID control block was considerably more complex because the function block can be set to three different behaviours (absolute value controller, differential value controller and event triggered controller). The final version of the PID block consists of 11 MSL blocks and several lines of code. A third example is the synchronisation block which generates user defined outputs signals according to different times of day or week. Although this function is not very complex the final version contains 230 lines of code but no MSL blocks.

Testing was very important as testing was an iterative process during the development of the function blocks. In a first step for each block an individual
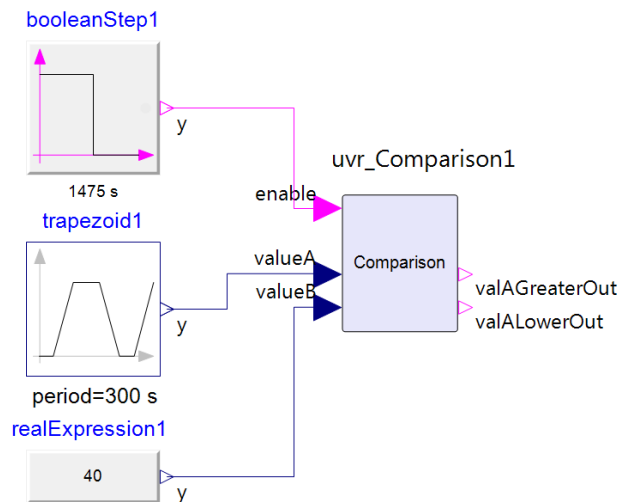
Figure 2: Comparison Function block test model

test model was designed which consisted mainly of signal functions from the MSL to generate input signals and used to test and verify the function blocks (test model for Comparison block is shown in Figure 2). In a second step we generated test signals by using the UVR 1611 simulation board which is an additional hardware module for the UVR and features 16 potentiometers with which temperature inputs can be set manually. After loading user programs consisting of just the function block under test into the UVR the inputs were altered and the corresponding outputs signals were recorded. These test signals were then fed to the function block under test and the recorded outputs were compared to the simulated ones.

As a third step a model of a detached house was created consisting of several parts:
- model of the UVR control program

- model of the solar heating system
- model of the buffer for storing thermal energy
- model of the building including a single room with floor heating.

This model was developed using components from the GreenBuilding Library (Schwan, 2012) which is part of SimulationX. This library provides model components for buildings and renewable energy systems. This model (Figure 3) was kept simple in order to detect errors in the UVR controller model which consists of several different control blocks (solar control, comparison, timer and others). The focus was on the interaction of the UVR control blocks with the rest of the model components and on the ratio of simulation time vs. computing time. The required amount of computing time was relatively small and in a few cases where an increase of computing time was registered, this was due to programming errors or faulty parameters. Both problems caused the generation of too many events mainly because thresholds were too close or states in the function blocks not stable.

Not all available function blocks for the UVR controller were modeled because there were too many of them and some are rarely used. Therefore we created a subset of the 12 most used function blocks which were modeled in Modelica and implemented into a library:

- Solar control
- Start function
- Cooling function
- Comparison
- Load pump
- PID control (speed control)
- Analog function
- Timer

- Time Switch
- Synchronisation
- Heat quantity counter
- Counter

The library of UVR function blocks is thus useable and can be extended to its full extend. However development was halted after around 60% of function blocks had been modelled because the new controller UVR 16X2 which replaces the older 1611 will feature a slightly different set of functions and will not be compatible with the 1611. In addition the main function blocks required in the project for UVR 1611 were modelled.

4 Use Cases and Demonstrators

After the required function blocks were modelled and tested we were able to implement the larger model of a refurbished office building in Chemnitz which is also a demonstrator in the enerMat project. The main building components that were included in the model are:

- Solar heating system with 270m² solar thermal collector area providing up to 80% of the annual heat energy (90% was planned)
- 110m³ heat storage buffer
- Building with 2 floors and 1150m² heated area
- Wood burning stove as backup heating system for cold and darker winter months (providing the remaining 20% heat energy)
- Heat pump to rearrange heat distribution in the buffer which also extends the range of the buffer. It has no external heat source and is used solely with water from the buffer.
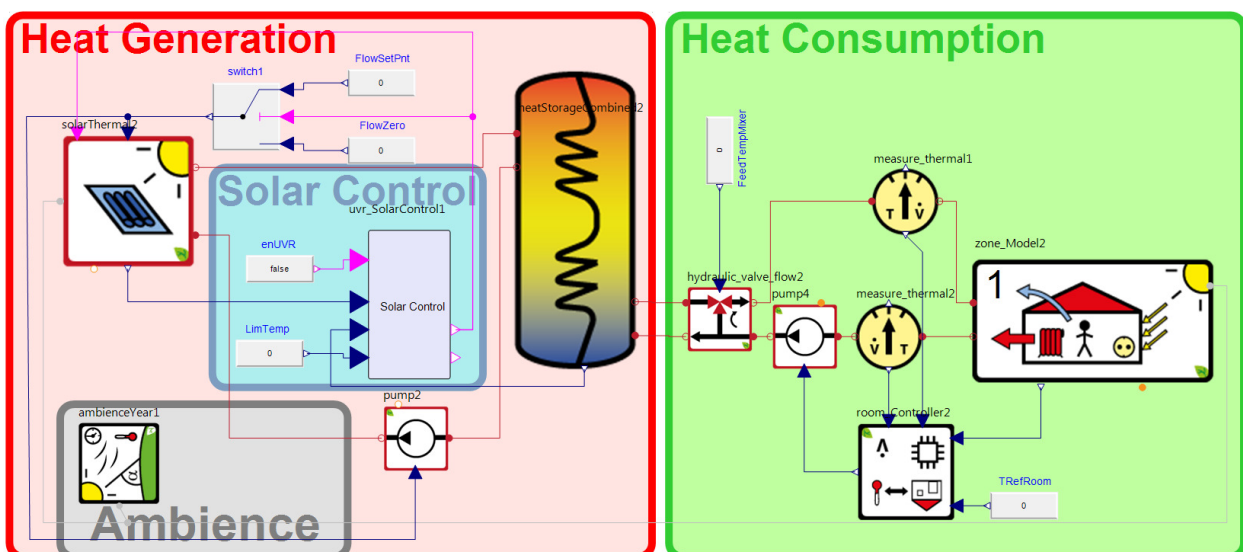

Figure 3: Model of a house with solar heating system and a UVR function block Solar Control

The refurbishment of the office building was planned and designed not as a low energy building but as a building that is sustainable and requires almost no fossil fuel for heating apart from the electrical energy of the heat pump which is active for around 10 days per year during the winter months. Therefore the building lacks typical features of low energy buildings such as air condition, automatic windows, etc.

The Modelica model for this building is split into the energy source and energy demand model which were developed in parallel.

4.1   Energy source model

The energy source model contains the solar thermal collector array, the UVR controller, the storage buffer, heat pump and wood burning stove along with several pumps and valves.

Combined with a dummy heat sink model consuming heat from the buffer this model was used to validate the UVR controller model by comparing simulation data with sensor readings obtained from the real controller. Typical signals were temperatures in the collector and buffer, pump and valve signals, power and heat levels. Simulation and measured values were relatively similar with some deviations that can be attributed to the temporal resolution of the meteorological data which was one hour (mean values for one hour: temperature, solar radiation, wind, etc…) and the one-point character of the collector model. The UVR controller model was created by using real UVR controller in the office building as template and was modeled one to one w.r.t. to function blocks, inputs and parameters.
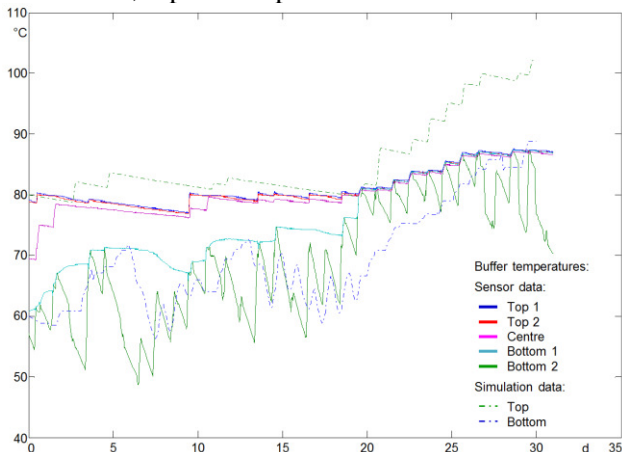
Figure 4: Temperature in 110m³ buffer (30 days)

Figure 4 shows the simulated buffer temperature distribution from the energy source model compared to measurements over a simulation period of 30 days. Input for the model was meteorological data recorded during the same period as the measurements were taken. (Note: Local weather data was not available,

weather data was recorded at a weather station around 60km from building). The general trend between simulation model and measurement data is identical and deviations are due to a very simplistic model for heat consumption which extracts heat differently from the buffer than the real-world heating system. Real world data from the consumption side which would make the model more exact is not available.

4.2   Heat consumption model

Components from the GreenBuilding library were used to model the office building complete with building zones, heating system, inner loads and shading. Several parts of the model had to be created from scratch where the GreenBuilding library did not provide sufficient components. These were models for:

- Heat pump and control
- Wood burning stove and control
- Distribution valve
- Heat meter
- Heating circuit manifold
- Time switches
- Two point temperature controller

and several other purposes. The solar heated office building has 53 rooms on 2 floors. Each room is equipped with a heating circuit which is supplied with warm water from a central heat storage tank. This tank is heated with solar power produced by a solar heating system which is part of the energy source model.

The model of the overall building covers 25 building zones each equipped with a heating system (heat consumer). Several neighbouring office rooms had to be combined into one building zone to speed up simulation and because the maximum amount of building zones in one model was limited. Modelling itself was quite time consuming because every boundary had to be parameterised manually with the corresponding parameters (thickness, thermal transmittance, area, material, …) taken from Excel sheets. After the completion of the demand part of the model several simulation runs were analysed in order to assess the models conformity to the real-world and numerous test cases were defined such as:

- Test of the heating behaviour in room:
  - maximum heating
  - No heating (neighbouring rooms are heated)
- Room occupation with different number of persons

- Windows opened and closed for certain amount of time
- Lighting (halogen bulbs) in room activated

Sensors at different positions recorded the temperature during the tests and the readings were later compared to the simulation results. Typical deviations initially observed between measured and simulation data were:

- Slope of temperature too steep due to simplified heating model not containing insulation layer between room and underfloor heating
- Simulated temperature 1K too low due to not modelled internal gains and/or misplaced temperature sensors (main logging sensor was attached to ceiling)
- Solar gains because of sunlight shining into the room too low resulting in lower temperatures in simulation
- single point room model too coarse (just single temperature instead of temperature distribution)

It was the aim of these tests and simulations to determine the accuracy of the simulation model. All tests were conducted in one office room because the building was in constant use and tests with the heating system would have caused discomfort for the inhabitants. The aforementioned deviations led to some improvements in the simulation model such as an added first order delay for the heating power. An upgraded heating model would have produced better results but the heating model from the library was protected and could not be extended. Results from the adjusted model are compared to measured data in Figure 5.
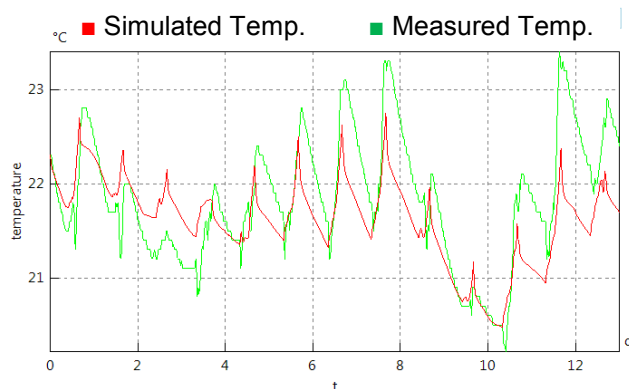

Figure 5: Simulated temperature compared to measured temperature in office room (13 days)

The complete building model with heating system was also simulated for one year and meteorological data for a test reference year (TRY) was used to obtain the energy demand of the building. Results (annual heat consumption in kWh/m²) were then compared with values calculated by the architect for the buildings energy performance certificate and were remarkably close (simulation 57kWh/m²; certificate 52kWh/m²). The 57kWh/m² are provided by solar heat and in case the solar heat buffer is depleted by a wood burning stove or the heat pump.

Simulation was then employed to find answers to the following question. Is it useful to program the heating controller to lower the set point temperature at night and/or at weekends?

Simulation results did not provide a simple answer as shown in Figure 6. The underfloor heating system is characterised by a couple of attributes that make it very slow: two-point controller with an on/off valve; long time constants of up to 12h, low feed temperature (around 32°C but down to 29°C in winter). All these factors contribute to a very slow heating system which makes energy saving measures like lowering the set point temperature difficult. The main issue with a lowered set point (red line) is that reheating the room in the morning or after weekends takes too long and a comfortable room temperature (blue and green lines) is not reached in time as shown in Figure 6.
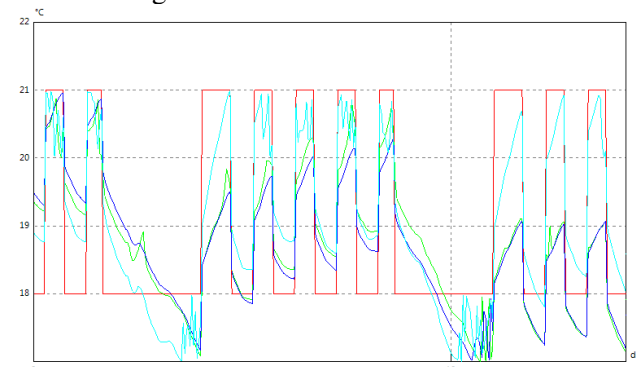
Figure 6: Temperature setpoint lowered to 18°C on weekends and after working hours

This result was an important finding and also a proof that energy saving by lowering the temperature during night and weekends comes at a price of reduced comfort at least in this particular building. Subsequently this also demonstrates the need for an intelligent heating controller algorithm that calculates the required preheating time w.r.t. current room temperature, outside temperature and available heating power (feed temperatures) and next occupation. Such a controller is currently under development and cannot be implemented using the function blocks of the UVR controller as they are not flexible enough. Any form of intelligent algorithm must therefore be implemented on a top level controlling system such as a BEMS or BACS.

4.3   Combined energy and building model

The energy source model and demand model were then combined into one single model. This model was rather large and separate development was crucial for handling and verification of each part. The one year simulation produced results that were already observed by the buildings occupants. Although the solar heat harvested during summer and stored in the 110m³ water buffer will last until December or January depending on weather conditions, it is not sufficient to provide enough heat until spring. It was therefore necessary to heat the buffer by means of a powerful but manually operated wood burning stove. In addition the concept of installing a heat pump to rearrange the temperature distribution in the buffer was simulated and evaluated and subsequently implemented. While this is a reasonable approach and helps to provide the required feed temperatures for the heating system the amount of electrical energy needed by the heat pump cannot be neglected. Therefore several tasks have to be targeted with regard to the control system:

1. The UVR controller has to be as effective as possible, thus harvesting as much solar energy as possible.
   The validation of the UVR parameters with the combined model was done along with a manual variation of certain UVR parameters to find better values. Due to the limited accuracy of the collector model this was found to be not promising. Especially time constants could not be validated because of the one-point collector model. Therefore the reaction of the collector model was too fast and for a model-based optimisation of the UVR parameters the model of the collector was inadequate. Development of a more detailed collector model was not part of the project.
2. The heating system should save energy by lowering set points in non-occupied rooms or during nights and weekends. As discussed in section 4.2 a possible solution would be an intelligent controller or BEMS algorithm. An approach to find a solution to this task was developed in this project as well and is discussed in (Majetta, 2015).
3. The heating support from the wood burning stove has to be optimised while the runtime of the heat pump to save electrical energy must be minimised. As a solution for this problem a BEMS algorithm has been developed which constantly monitors the feed temperatures of the heating system from the buffer and informs the occupant when additional heat from the wood burning stove is required. Such a system is currently not installed and the occupant must decide for himself whether it is time to start the stove or not. The BEMS module informs the inhabitants about the right time to start the stove and will in addition start the heat pump if the stove's heat is not sufficient or if no occupant is around to start the stove (during night or on weekends). The heat pump however is only used for redistribution of the temperature layers in the buffer and takes heat from the colder bottom of the buffer to heat up water at warmer top of the buffer where the feed connector to the heating circuits is installed.

Especially for tasks 2 and 3 simulation-based optimisation was employed but not really achievable due to very long simulation times of the model. Initially it was planned to simulate a whole year with the combined model and use this model also for optimisation. But a year simulation required around 5 hours computing time which is far too long for optimisation purposes. Therefore the model was analysed in order to find the part that requires the most computing time. It was found that the different building zones and their heat exchange through boundaries were responsible for roughly 80% of the required time. Hence a less detailed model was developed with just 2 building zones instead of 26. The 2 zones are representing the ground floor and the first floor. The energy demand of this model was higher due to the size of each floor and the set point temperature of 22°C whereas in the detailed model several rooms had a set point temperature of only 18°C such as storage rooms and libraries. The simplified model was much faster and required around 20 min for a year simulation and was thus fast enough for simulation-based optimisation.

4.4   BEMS development

Development of the BEMS was done in two steps. At first a statechart was created which represents the different states of the buffer and triggers the start of the stove and heat pump depending on certain conditions and parameters. The correct function of the statechart was then tested together with the simplified model. Then the fixed parameters were replaced with a function that evaluates buffer condition and weather forecast. This function takes into account whether warm and sunny weather is expected or cold and overcast weather. Depending on the result the heating command will be given earlier or later. In order to enable optimisation the function was equipped with certain weighting factors and a target function to be minimised.

Optimisation was then employed to find ideal parameters for these factors. In order to make the mod-

el ready for the optimisation framework it had to be exported as an FMU (functional mock-up unit) and was then executed within the MOEA framework (multiobjective evolutionary algorithms).

A PSO (particle swarm optimisation) algorithm (Kennedy, 1995) from the MOEA framework was employed to find optimised parameters for the BEMS function.

	Fixed Param.	Non-Optimised Param. Funct.	Optimised Param.Fnct.
Minimised Model			
En. Heat Pump [kWh]	4437	3976	3877
En. Stove [kWh]	14440	16320	17437
Temp.Violation [days]	66	64	62
Full-scale Model			
En. Heat Pump [kWh]	3638	3632	3656
En. Stove [kWh]	13881	15114	16732
Temp.Violation [days]	75	64	52

Table 1: Simulation and Optimisation Results for BEMS controlled stove and heat pump

A typical optimisation run with a 6 months simulation period and the following PSO parameters (6 particles, 50 iterations) results in 300 simulation runs and takes around 23 hours to complete. The FMU of the simplified model was used for the optimisation. Optimisation targets were to minimise the operation time of the heat pump and thus reduce the amount of electrical energy required and to reduce the temperature violation which is triggered if the feed temperature of the heating system is below a certain threshold (30°C). An important restriction for the overall system is that the stove can only be operated between 7am and 5pm on weekdays because it requires manual loading and starting.

The optimisation results in Table 1 show a small reduction in the energy consumption of the heat pump compared to the non-optimised or fixed parameters. In addition number of days where low feed temperatures were registered is also lower which can be seen as a higher level of comfort for the occupants. This results in a higher heat production of the stove which is equivalent to a longer operation time (stove is started on more days in winter). The required firewood was not taken into account as an optimisation target as because it is available for free. An important question was however if the simulation and optimisation results (parameters) of the reduced model can be transferred directly to the full scale model and will lead to improvements as well. With 5 hours computing time the optimisation for this model would have taken around 60 days. The optimised parameters from the reduced model were having the similar effect in the full scale model (see Table 1) however the heat pump energy was not significantly reduced but the temperature violation was lower. It

was found that the limited operation time slot of the stove is the main factor which prevents a further reduction of the heat pump energy.

5 Actual State and Outlook

The UVR library for Modelica has been tested and used in this project but it is not finished as of yet due to the discontinuation of the controller from the manufacturer. The function blocks however are useable not only in UVR related projects but also for other purposes because they implement typical functions that are required in building control systems. It has been found that the design of a model from scratch for a single family house will be too time consuming and the subsequent energy savings will not cover the cost of simulation and optimisation. Therefore it is planned to create models for standard UVR applications and standard houses of one manufacturer. The simulation of these models does not require modelling skills and can be used to test UVR parameters for certain different configurations such as different collector area or buffer size, different location, usage patterns and house parameters. This will enable the HVAC planner or engineer to install the controller with optimised parameters and will subsequently cause a higher performance of the solar heating system combined with an increased convenience for occupants without the need to develop a new model for each project.

BEMS development is currently under way and the simulation results are very promising. It has been shown in this paper that the implementation of a BEMS can solve certain tasks that cannot be easily conquered with a stand-alone controller such as the UVR. While the UVR and similar controllers are great for standard applications more difficult tasks require control systems that go beyond the limits of stand-alone control units. Our BEMS approach can be such a system. Simulation and optimisation will play a significant role in the future. This paper tried to highlight the use of both strategies in the development and validation for building controls.

6 Conclusion

In this paper several ideas w.r.t building control systems, simulation and optimisation were discussed. As of yet control systems in buildings are stand-alone units and energy efficiency cannot be fully achieved with these units because the standard commissioning procedures rely heavily on experience rather than optimised and tested parameters. There-

fore a library of controller function blocks has been developed that enables the engineer to test and optimise a stand-alone building controller in a simulation environment. This library is compatible to the program on the real-world controller so that the parameters established and optimised in simulation can then be transferred to the controller. The limits of this approach were also discussed and a different approach for the implementation of additional functionality into a Building Energy Management System (BEMS) was demonstrated. An optimisation algorithm was implemented to find BEMS parameters that are highly energy efficient and ensure a better comfort for occupants.

Acknowledgement

References

Bundesministerium für Wirtschaft und Energie *Energieeinsparverordnung* 2014

European Commission: *EU Energy and transport in figures, statistical pocket book* 2007/2008

Kennedy J., E. R.: *Particle swarm optimization*. IEEE International Conference on Neural Networks, (S. 1942 - 1948). Bur. of Labor Stat., Washington, DC, USA, 1995.

Majetta, K., Clauss, C., Haufe, J., Seidel, S., Blochwitz, T., Liebold, E., Hintzen, U., Klostermann, V.: *Design and Optimization of an Energy Manager for an Office Building*, ASIM/GI-Section Workshop – Simulation of Technical Systems & Methods in Modelling and Simulation, Stralsund, June 2015

Schwan, T.; Unger, R.; Bäker, B.; Mikoleit, B.; Kehrer, C.; Rodemann, T.: *„Green Building" – Modelling renewable building energy systems ans electric mobility concepts using Modelica*. 9th International Modelica Conference, Munich, September 2012.

Seidel, S., Donath, U., Haufe, J.: *Approach to a Simulation-based Verification Environment for Material Handling Systems*, 17th IEEE Conference on Emerging Technologies and Factory Automation, Krakow, September 2012

Seidel, S., Klotz, T., Donath, U., Haufe, J.: *Modelling the Real-Time Behaviour of Machine Controls using UML Statecharts*, 15th IEEE Conference on Emerging Technologies and Factory Automation, Bilbao, September 2010

Seidel, S. Donath, U.: *Error-free Control Programs by means of Graphical Program Design, Simulation-based Verification and Automatic Code Generation*, 8th International Modelica Conference, Dresden, March 2011

Technische Alternative: *UVR 1611 Frei programmierbare Universalregelung Bedienung, Programmierung, Montageanleitung* Ver.4 2014