Automatic Optimization of Energy Supply Systems in Buildings and City Quarters based on Modelica Models

Dipl.-Ing. Torsten Schwan[1] M. Eng. David Feige[2] Dipl.-Ing. Leonhard Wenzel[1] Dipl.-Ing. Charlotte Voelckner[1] M.Sc. Martin Leuschke[1]

[1]EA Systems Dresden GmbH, Germany, {torsten.schwan, leonhard.wenzel,charlotte.voelckner,martin.leuschke}@ea-energie.de

[2]HKL Ingenieurgesellschaft mbH, Germany, d.feige@hkl-ingenieure.de

Abstract

The evaluation and analysis of complex energy supply systems with Modelica models is more and more an integral part of the building design processes. Dynamic system modeling became there especially important regarding analyses of the use of storage and the integration of volatile renewable resources as well as intelligent control.

However, this still requires extensive engineering work and time-consuming modeling efforts, although the basic work steps are largely comparable and based on the same fundamentals. Therefore, the open interfaces to and from Modelica offer extensive possibilities for automation and generalization of these processes.

This paper describes such a new integrative and automated optimization framework for energy systems of buildings and districts, which uses Modelica models and FMUs iteratively for the identification of optimal system configurations.

Keywords: System Optimization, HVAC System Models, Python Automation

1 Introduction

Energy supply systems in buildings become more and more complex. Therefore, huge knowledge and a high number of professionals in different trades must be coordinated. This adds further challenges for architects, engineers and designers especially regarding the increased requirements on energy efficiency and availability.

These extensive engineering tasks can only be solved with adequate calculation tools. These tools must be able to deal with an increasing variety of solution options and degrees of freedom as well as influencing factors.

For example, the required heat of new buildings is nowadays provided by renewable energies and no longer by individual boiler systems. Renewable heat sources often require heat pumps to provide the necessary temperature level. Both the volatile heat sources (such as waste heat, solar heat, geothermal heat) and the heat pumps must be considered in detail.

An engineer can now no longer focus on a singular balancing of the necessary natural gas consumption of the boilers. Design decision now need an influence analysis of weather and site conditions as well as the necessary power requirements for the heat pumps. These design analyses must often include an additional coverage by local renewable power production (e.g. by photovoltaics or wind power) which is another important boundary condition and influencing factor. In case of an additional seasonal storage (e.g. ice storage) in a system, engineers need to solve a multi-valent and cross-domain design problem already only regarding the singular task of heating system design.

Keywords like multi-valent and cross-domain quickly bring simulation-savvy engineers to the versatile, multi-physics modeling language Modelica. Therefore, Modelica already provides a large number of well-proven library solutions, e.g. Buildings, BuildingSystems, IBPSA, AixLib, IDEAS, Green City, etc. (c.f. Wetter 2009, Müller et.al. 2016). Furthermore, the various Modelica simulation environments (like SimulationX, Modelon Impact, Dymola, Open Modelica, etc.) offer a variety of open interfaces for the simulation automation. They can help to automate the necessary extensive variant analysis. This saves a significant amount of engineering time. Additional export options, especially like the Functional Mockup Interface Standard (FMI), enable IP protection and tool-independent development of the necessary design tools.

Engineers can design the energy supply for each building individually. However, thinking on a district level often provides more efficient solutions. It allows to use synergy effects between different buildings and surrounding energy sources. This is the core topic of the Smood (Smart Neighborhood) network (Roselt, and Büttner, 2019).

The Smood joint project consists of a group of German engineering companies and local scientists. It tries to develop a holistic value chain of tools and processes for the decarbonization of the energy supply of residential neighborhoods. Another important research and development aspect is the market launch of the developed toolsets.

The whole project considers four core development goals. SmoodQIM represents a holistic neighborhood information model for data management (i.e. comparable to BIM for buildings). Another core piece, i.e. SmoodManage, includes automated process steps for the building retrofit planning. Besides these software and process components, the SmoodHardware part considers the development of novel storage systems in both the thermal and the electrical domain. For the Smood collaborators, all these components are necessary for a future sustainable energy supply to residential neighborhoods.

Another key component of the developed tool chains is a new methodological simulation approach (i.e. SmoodSimulator) that evaluates buildings and HVAC systems together in an iterative optimization process. This automated process results in a holistic retrofit strategy including a therefore optimized HVAC system configuration. It includes three main components.

A building-focused analysis tool (i.e. Caala) provides basic retrofit options for the particular building envelope with respect to CO_2 savings and gray energy demand. Then, the automated tool chain derives a model matrix with different system configurations based on the building requirements. This matrix is linked to a huge set of pre-configured HVAC system models developed in Modelica. These models run automatically after an automated setup of the necessary components parameters. All automation steps are based on the versatile scripting language Python. This Python framework also includes an optimization algorithm which iteratively adjusts the HVAC system parameters depending on the chosen optimization goals.

Such iterative approaches which connects Python-based optimization algorithms and Modelica models are not new. Leimeister, 2019 describes in her paper a combined optimization framework that uses both components. However, this work primarily focuses on the optimization of a singular system component, a wind turbine in its operating environment. A more general link between Python-based machine learning libraries and the simulation tool EnergyPlus was introduced by Christiaanse et.al. 2021. Eckstädt et.al. 2020 investigated extensively the use of simulation-based methods and optimization approaches with respect to different application scenarios in the context of building design process.

The use of multi-criteria optimization approaches with focus on architectural design was introduced by Dan Hou et.al. 2019 among others. The BeDOT tool (c.f. Bergel et.al. 2019) uses some the scientific approach of holistic software tools for energy system optimization. An alternative view on building controls with a holistic framework is also part of Arroyo et.al. 2021.

Comparable approaches to Smood regarding a holistic view of all these topics on a neighborhood scale is also part of the research at the University of Innsbruck, Austria (c.f. Dermentzis et.al. 2019).

2 General Concept

The entire Smood nucleus includes a variety of companies and research groups as well as many advanced technologies and tools. One key component of the master plan is a novel, holistic simulation environment for the automated generation of retrofit strategies and energy concepts for larger residential districts, i.e. SmoodSimulator.

This tool needs to take into account the great variety of requirements of the building structure and the energy supply as far as possible. The identified solutions have to look for the local and global optima with regard to the joint consideration of life cycle costs (LCC) and life cycle assessment (LCA). Therefore, it can consider retrofit measures for the building envelope as well as the use of novel (renewable) supply technologies. Very often, potentials solutions can also be mixed forms of both approaches. The main goal of the SmoodSimulator is to automatically find exactly these solutions on the basis of the existing data.

Designated users of the tool can also be city and district planners as well as architects. High engineering and simulation know-how can therefore not be assumed. Requirements resulting from the use of Modelica models have to be handled as automated as possible by the SmoodSimulator itself or corresponding simplifications.

Figure 1 shows the general structure of the SmoodSimulator workflow and its abstract software components. This also includes application-specific tools from Smood partners (i.e. Caala) and third-party providers (i.e. Rhino 3D).

The Rhino 3D tool enables architects to plan the structure of new buildings from the design point of view or to optimize existing buildings with the help of a graphical interface. It provides only information about the building envelope. The tool Caala (via Grasshopper plugin) evaluates exactly this structural data of the 2D and 3D

DOI
10.3384/ecp204789

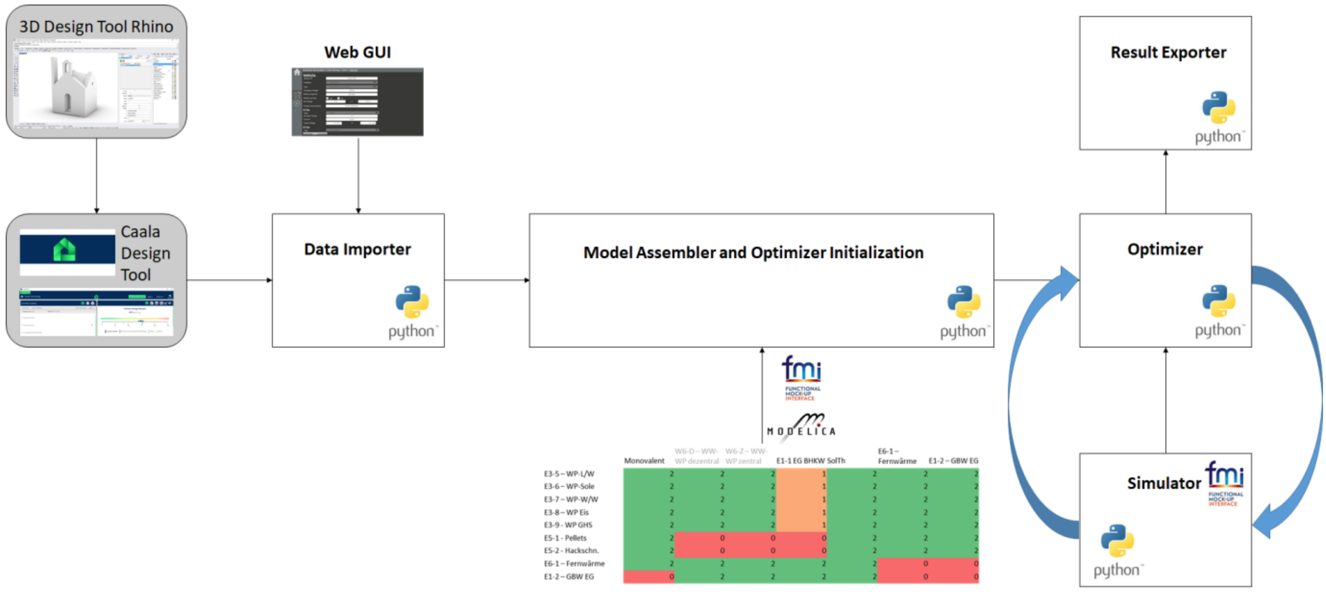

Figure 1: Overview of smoodSimulator workflow and abstract software components incl. the integration of Modelica models and FMUs (sources of figures: Constantino and Pepe 2021)

building representations in order to run a first analysis of LCC and LCA based on simple key figures.

Its strength is primarily in the underlying databases on the necessary building material costs and their LCA statistics. In this manner, it enables a balanced quantification of the so-called "gray energy". This is a measure of necessary CO_2 emissions regarding the required building materials and amounts (i.e. incl. insulation). However, the energy supply system in Caala only uses simple and constant efficiency factors. This is an essential limit of the HVAC and power supply system evaluation in Caala. Especially, volatile production of renewables and local storages with nonlinear efficiency characteristics and availability are thus hard to assess. A more specific modeling was necessary for appropriate engineering evaluation.

The SmoodSimulator was designed to fill this gap using the available interfaces of Caala as well as its results and outputs. The main idea is to add a seamless automated toolset which takes the available information and adds the necessary complexity in the HVAC and power supply system part of the evaluation strategy. Therefore, Modelica models seemed to be most promising in accuracy but lacked regarding handling and usability for non-professional simulation engineers, like architects and designers. Therefore, adequate automation, e.g. via Python, was necessary.

This process starts with the building-specific data, such as thermal and electrical energy requirements, as well as the initial results of the LCA/LCC analyzes in Caala. The SmoodSimulator reads them via a specific interface API using the Data Importer. This process defines the caalaConfig. Furthermore, it reads additional information regarding optimization process parameters and the stored simulation models from other configuration files (i.e. userConfig, optiConfig). In the future, a graphical user interface will be available for this specific import task.

The core piece of the Python framework is the Model Assembler and the Optimization Initialization. It has a variety of tasks, especially in the area of automatic selection, preparation and instantiation of the required models.

The Modelica models have to represent as accurately as possible the interdependencies of any dynamic energy system with its variety on possible details. The building(s) are considered as static load profiles. Their optimization with respect to possible retrofit steps for the building envelope was part of the prior optimization loop in the Caala tool. However, these Modelica models of HVAC systems require consumption curves with high temporal resolution that go beyond the available outputs of the Caala tool (i.e., monthly balances only). Therefore, the Model Assembler has to generate automatically suitable input data sets for the models, i.e. heating and power loads, weather data.

The Model Assembler has access to an extensive model library of predefined energy supply system configurations. The variants choice for the optimization process is based on a dynamic requirements matrix. This is matched with the requirements of the building(s) under consideration (e.g. temperature conditions).

Furthermore, the Model Assembler prepares the required set of model configurations for the optimization process. This is a working step which has been optimized regarding the total optimization speed for several times. One of these steps included the execution of exported FMUs (Functional Mockup Units) via Python instead of running Modelica models via remote control. This provides a very performant solution which enable a parallel execution of models with different configurations on different platforms (i.e. using PyFMI or FMPy interface in Python platform). In order to perform appropriate parameter optimization (i.e., equipment technology performance data) for each configuration, each FMU provides appropriate variable parameters.

Another important initialization step is the initial parameterization of all system variants which is based on the requirements of the generated building loads. The Model Assembler checks all generated load profiles regarding potential critical values, such as peak power (e.g. dynamic heating load). Then, it derives the necessary preset parameters of each FMU with respect to the requirements of the variant matrix and some specific heuristics.

The optimization process runs iteratively and parallelized. Each model configuration and parameter variant runs an annual simulation as an executable FMU in Python. Because of the necessary high number of system configurations and suitable parameter numbers, the requirements on the model performance are very high. Section 3 therefore shows the most relevant steps of model performance optimization.

Each system configuration is first simulated as a baseline with its initial parameter set. LCC and LCA are then calculated based on the energetic results of the executed FMUs in post-processing. With the Python hyperparameter optimization framework Optuna, the variable parameters (i.e., power classes) of all system variants are then optimized in parallel using an iterative loop and a high number of year simulations. Special attention is given to bivalent equipment configurations such as district heating grid and heat pump etc. during optimization.

If the optimizer reaches the previously defined termination conditions, the optimization process ends. The results of the analysis, i.e. a selection of the different optimized system configurations and the presentation of the best result, are displayed. This includes an export of suitable graphics and necessary data for post-processing.

3 Model Concept

The developed framework focuses on Modelica-based simulation models of technical equipment for heat and power supply of (residential) buildings. The building itself is only a simple look-up table based load profile because of highly accurate tool sets (i.e. Caala, Rhino 3D) before the Modelica models in the tool chain. However, these simple models also require suitable interfaces to the still necessary dynamic model components.

The Modelica-based Green City library in SimulationX therefore provides some relevant interfaces and a suitable data integration (c.f. Schwan et.al. 2017).

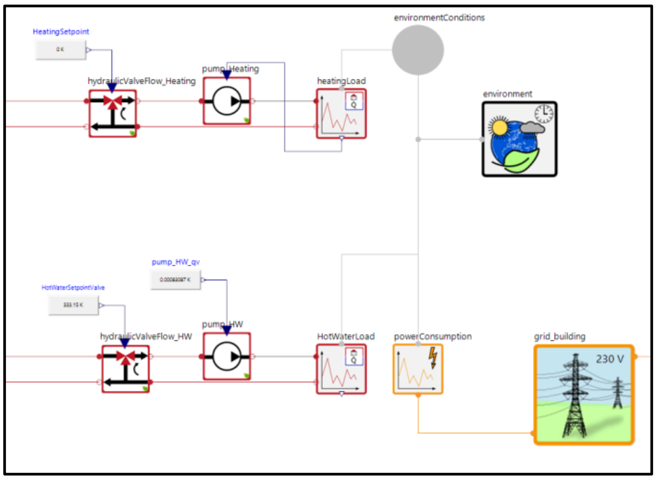

Figure 2: Simple modelling approach of building load curves with Green City (i.e. Modelica-based library) components

Regarding the co-simulation of HVAC systems with independent building models, Nicolai and Paepcke 2017 have already shown a first adequate solution with the Green City library. The presented framework uses the same premises as model base.

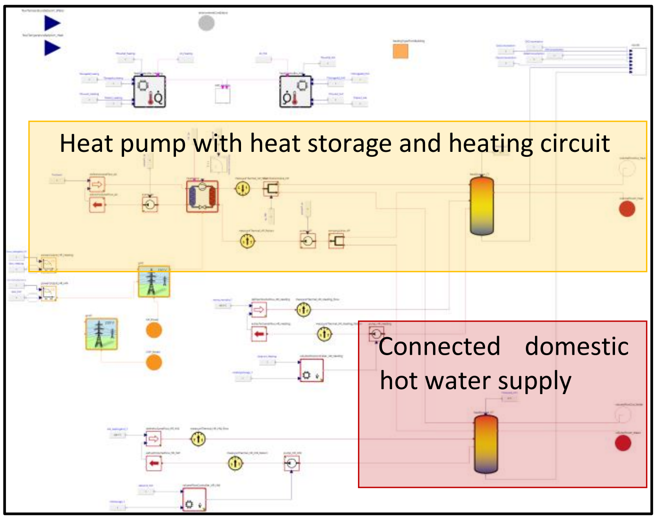

Figure 3: Detailed HVAC system model component

The chosen modeling concept is based on the main task to enable simulations of each possible system configuration of the model matrix (cf. section 2). A first approach used detailed single Modelica models. There were saved in specific library as a model template for the Model Assembler. In these templates all model-internal parameters (e.g., pump parameters, storage tank sizes, etc.) depend on a reduced set of relevant parameters, like performance categories. If these performance values are changed during the optimization process, the model internally adapts automatically. This is important for both numerical stability and a realistic representation of real-world behavior.

Figure 3 shows an example model of a simple heat pump system with geothermal collectors as renewable heat source based on Green City components. The model only has a unified interface to the heating circuits and the local power grid which are compatible to the general load profile model (c.f. Figure 2). However, there are strong mathematical dependencies between the different temperature levels of both heating circuits (i.e. heating and domestic hot water) due to the availability of only one heat supply component (i.e. one heat pump). This model is comparatively accurate as individual controls define the heat pump output depending on individual storage temperatures and the respective dynamic loads. However, this significantly lowers the simulation performance. Simulation of one entire year thus requires about 10 to 15 min for each variant and system configuration.

In an optimization process with 10++ different system configurations and various options of discrete system parameterization (especially in bivalent system configuration), 1,000s to 10,000s simulation runs may be necessary. However, the optimization period should not exceed a frame of hours to a few days.

The first important performance optimization step considered the decoupling of dependencies between the two heat supply tasks (i.e. heating and hot water production). This approach required to focus on each template model of SmoodSimulator’s system matrix. Obviously, it results in a higher deviation between the simulated system behavior and exact simulation results or real-world measurements. However, this loss of accuracy is acceptable because the intended field of toolset application are preliminary design phases of existing neighborhood districts. The level of detail of all assumptions is there quantified with +/-40% and higher (c.f. Kochendörfer et.al. 2010). Deviations from different efficiencies of the simulated systems (e.g. temperature-related COP of heat pumps) or limited availability due to simultaneity will be significantly lower.

The presented approach already reduced the average simulation times of an entire year to 2 to 3 min. However, this was still not fast enough. Further optimization steps were necessary. The model concept update still considered a coupled forward-backward modelling approach. Heat and power load characteristics defined a backward model of the considered building(s). The energy system template models still represented a forward model which operated depending on internal control on temperature levels of the decoupling storage tanks (i.e. internal system capacities).

The next optimization step consisted of a complete conversion of the Modelica model approach to a backward model. This required a redesign of the individual system components of the HVAC system models in the templates. Now, these models only represent a nonlinear, dynamic efficiency characteristic and are thus only conversion models (e.g. heat pump - heat/electric energy).

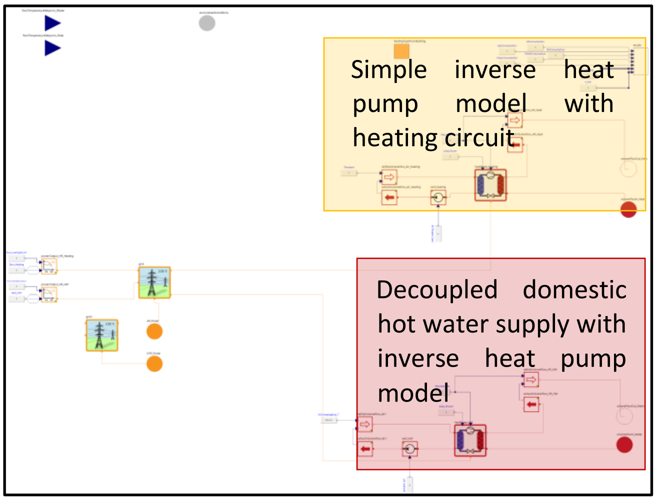

Figure 4: Simplified HVAC system model component

Model dependencies on external control functions were removed as far as possible. However, since the interfaces of the model templates stay the same, reuse of the more detailed model options for detailed analyses within the framework will be possible at any time.

In this way, the average simulation time for a year simulation per system model template could be reduced to less than 30 s (i.e. using CVODE solver with common settings).

Now an acceptable performance range was reached. Further simplifications in the models themselves were almost no longer possible. However, further performance potential could still be identified during model execution.

On the one hand, the direct execution of Modelica models requires the use of a suitable simulation environment (in this case SimulationX). Communication with and automatic execution of models in this environment also requires a certain time period. The faster the models, the higher is the share of these communication time periods on execution total time. On the other hand, the execution of simulation models in the simulation environment takes place exclusively on one computational core of the respective computer/server. Parallelization is difficult to realize and also requires extended licensing costs, especially in the area of optimization for the simultaneous execution of a large number of models.

Therefore, the use of FMUs in Python with the help of the PyFMI framework provides an alternative solution. This allows the execution of parameterizable FMUs directly in Python without an additional running simulation environment. By exporting the models with solvers (FMI 4 co-simulation), similar or even better performance can be expected without the communication overhead.

However, the biggest potential advantage is the parallelizability of model execution. By using multi-threading approaches, the Python framework supports parallel execution of models within an optimization loops.

This last optimization step again provided another significant reduction of the necessary simulation time periods per model (FMU). With about 10 to 20 s per annual simulation to be performed, the framework now has a sufficient computing speed even for large problems. Common optimization tasks with about 5,000 model iterations can usually be performed in 2 to 3 hours.

The developed approach now represents a powerful simulation and optimization framework. However, due to the necessary simplifications, the model accuracy is now somewhat reduced. However, it is still in an acceptable range regarding the available degree of accuracy of necessary assumptions in an early building design phase (i.e. +/-40%).

Furthermore, the developed model concept represents another huge advantage regarding its upward compatibility. Because of the consistent interface definition, the models are always available in different accuracy levels (c.f. Figure 3).

If a more detailed consideration of some system variants with the help of the optimization framework is required in a later design phase (e.g. detailed planning), a more accurate simulation model with the same interfaces can simply be generated and analyzed with the help of the identified system configurations. These adapted models can then also be edited manually, refined and developed during the entire design period.

4 Examples of Optimization and Validation Results

The developed SmoodSimulator approach is predestinated for all scalable tasks of an automated design regarding retrofit and energy concepts of buildings up to urban quarters. However, the current version focuses primarily on residential quarters and thus on the use of buildings as living space.

The toolset is currently used in a comparatively limited field. The required input data sets for the simulations of buildings are thus often similar.

As a starting point of any optimization run, the Caala tool performs a simplified energy analysis of a given building structure and type. The results are monthly energy demand characteristics of the considered building. However, this is only the basis of input data set generation for the HVAC and power supply system models.

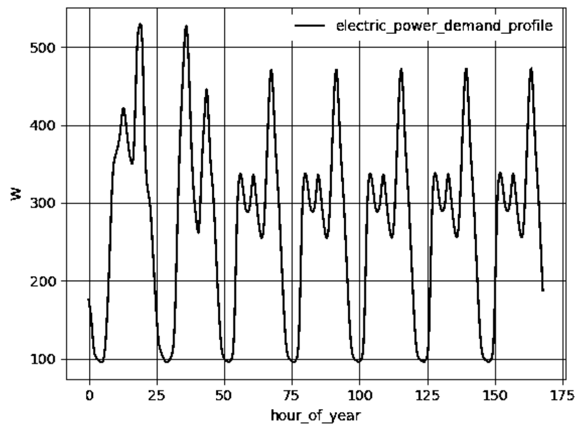

Figure 5: Example power consumption characteristic (1-week detail) of H0 standard load profile

Because of the similarity of energy consumption patterns in all potential households, generalized time series (i.e. standard load profiles – c.f. Figure 5) are the base to create load curves with higher temporal resolution. The Caala results only provide scaling factors, such as the cumulative energy demand values. This approach is used in the smoodSimulator for both electricity consumption and hot water demand.

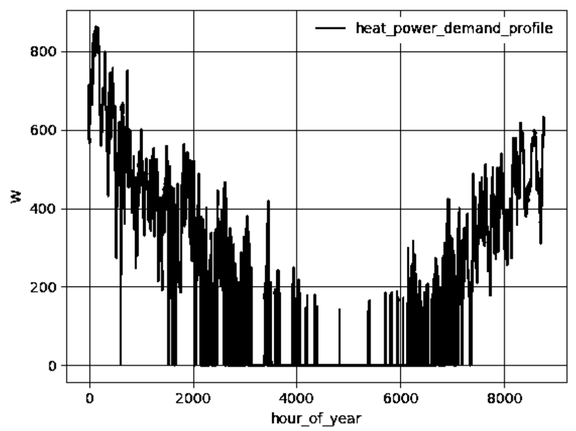

Figure 6: Example heat consumption profile based on TRY weather data

In addition to primarily occupancy-related energy consumption, buildings also consume climate-based heat. In order to be able to provide these with a higher temporal resolution, weather data (e.g. test reference year - TRY) with an hourly resolution are used as scaling base. The cumulative heat demands, again a result of the Caala tool, are weighted and distributed to a corresponding year profile with respect to the outdoor temperature and corresponding generic heat demands.

 DOI
10.3384/ecp204789

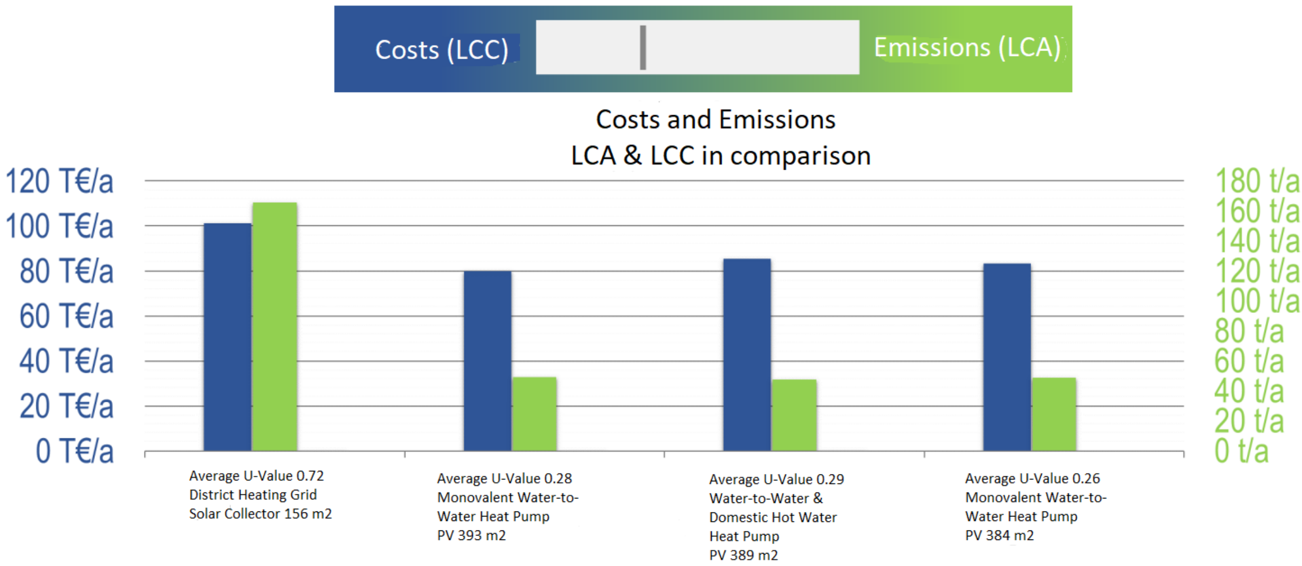

Figure 7: Example results of a holistic LCC/LCA analysis of an urban living quarter

However, the optimization framework does not only consider energy aspects. The goal of optimization is always a holistic system analysis, also in the direction of life cycle costs (LCC) and life cycle assessment (LCA). In this regard, adequate estimated values for the necessary investments are required in addition to suitable assessment factors.

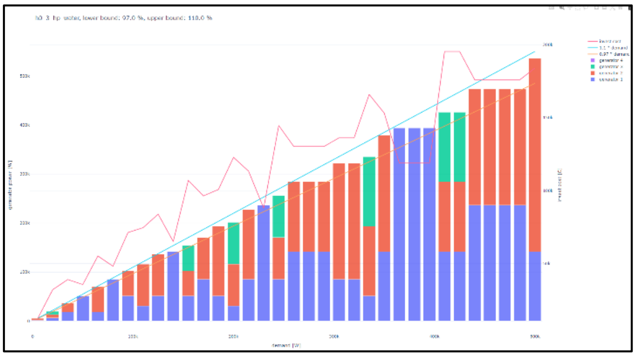

Figure 8: Cascading of heat supply systems depending on available system sizes and total power demands

The investment costs always depend primarily on the component size. However, not every system/unit size is available on the market as standard. Typically, the availability of the individual size follows a series development by power classes (e.g. 10 - 20 - 50 kW). Therefore, certain capacities can be achieved either with one unit or by cascading several units. Oversized units will cause additional costs. A cascade of several units is more expensive than one unit with the same total power. From this static optimization problem, an optimal component configuration can always be found for each (discrete) power class prior to any simulation run.

Figure 8 therefore shows an example configuration matrix of air-to-water heat pumps with different total (discrete) system sizes and individual optimal cascade solutions (i.e. blue / red / green – individual maximum power output of heat pumps).

The Model Assembler also organizes the optimal cascading of each analyzed system configuration automatically during the initial simulation period and the later iterative optimization process.

This also results in a necessary requirement for the parameterization of the models. No matter which models base is used, i.e. Modelica model or FMU, before the start of each simulation run all components parameters must be editable regarding new power categories and characteristics of the individual system configurations. However, this represents a huge challenge for the instantiation of the models, especially in the optimized FMU mode.

FMUs are not structurally changeable after export. Only constant parameters (unchangeable during simulation) can be adjusted. The modeling approach used in Green City is thus advantageous. It maps the cascadability of the individual component models by internal integer parameters.

Figure 8 shows exemplary results of one of the first test analyses. It considered a small urban living quarter in mid-Germany.

On the one hand, the analysis took into account the possible retrofit of the building envelope with different building materials (i.e. use of different U-values). On the other hand, different system configurations were analyzed

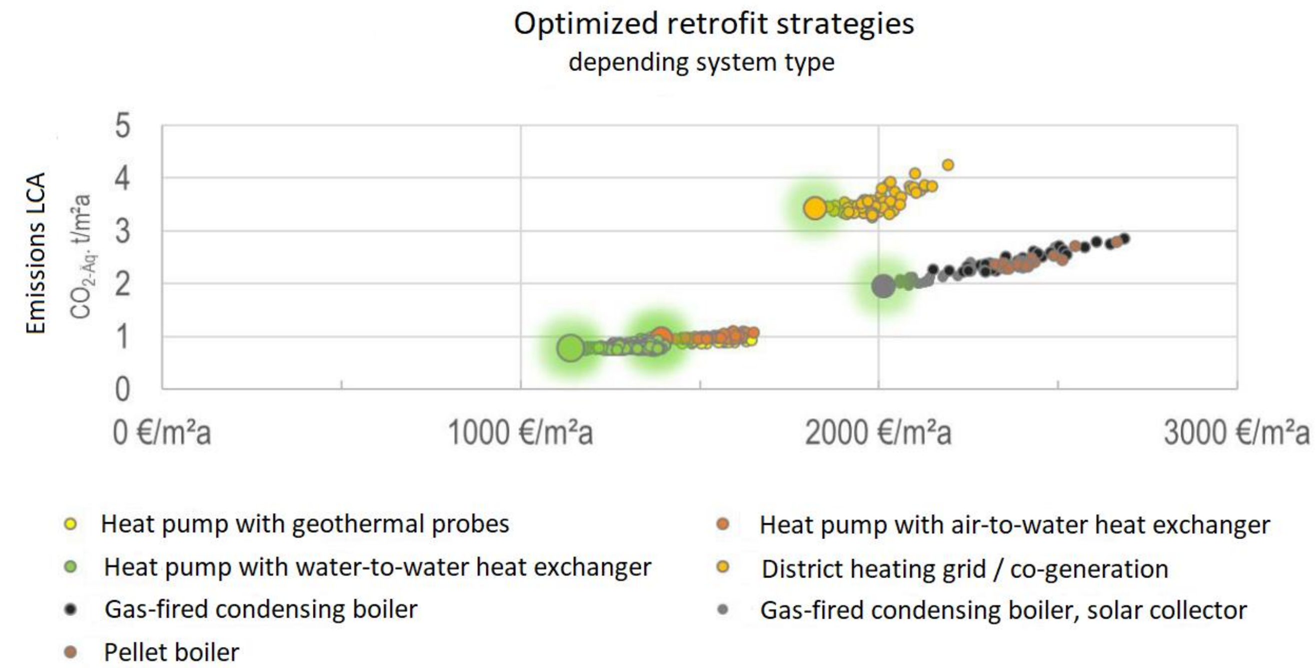

Figure 9: Full results of LCC/LCA optimization process of an urban living quarter

and their performance parameters were optimized with the help of the framework:

- District heating grid plus solar collectors
- Monovalent water-to-water heat pump and photovoltaics
- Bivalent heat pump system (air-to-water) for heating and domestic water supply

The existing system is district heating, which currently supplies heat monovalently only to buildings with simple insulation. In a simple optimization strategy, this is supplemented exclusively by thermal solar collectors. The retrofit carried out in this way generates the lowest investment costs. Over the lifetime, however, both the life cycle assessment (LCA) and the life cycle costs (LCC) are disproportionate to comparable heat pump systems.

Through iterative parameter optimization in the optimizer, the respective optimal system configuration with regard to LCAs and LCCs is found for all alternative supply concepts. The respective retrofit conditions of the building (i.e. the selected insulation standard) are also included in this optimization.

The example in Figure 8 shows the three best options, all describing different heat pump configurations. Depending on the efficiency and necessary costs of the selected technical systems, the energy standard of the building envelope can or must be adapted. Therefore, all variants describe slightly different insulation standards (i.e. U-values). The same applies to the use of local renewable generators (e.g. photovoltaics). Depending on the efficiency of the used heat pumps, slightly smaller or larger generator sets must be installed.

Figure 8 only shows the final results of the optimization process. It compares the baseline, i.e. the existing energetic standard and HVAC system, with the three best retrofit options of the optimization process.

Figure 9 adds to this plot a dot plot showing all parameter configurations simulated by the framework with the results of the LCC/LCA analysis for all evaluated system configurations.

It also shows different clusters of efficiencies and costs for different system configurations. The influence of parameter optimization is presented very precisely for each configuration from the deviations of the points of the same color. It is noticeable that the change of the HVAC system configuration has a significantly higher influence on efficiency and costs in each case than an optimal identification of the respective component parameters (e.g. power categories).

5 Conclusions

The presented approach of a holistic, automated optimization framework for energy systems of buildings and districts shows again the versatility of Modelica models.

On the one hand, multiphysical, noncausal modeling with Modelica is also widely used in the fields of energy supply and building services engineering. On the other hand, both the models themselves and their derivatives (especially

DOI
10.3384/ecp204789

FMUs) can be integrated into automated software applications independently of tools.

Multiphysics simulation of power systems using Modelica allows very detailed evaluation of system behavior, especially in the presence of volatile generation and storage. However, these models are still very complex for use in optimization loops. Simplification to a minimum of necessary complexity is still required to ensure a reasonable time frame for optimization.

Further developments in the field of solver technologies are still necessary. This relates primarily to the computational speed of the models. In addition, knowledge-based potentials for increasing speed must continue to be tested and implemented. This mainly refers to the simplification of model equations by application-specific information. In this way, solvers can be relieved and high speed increases can be achieved. These approaches can be supported by the use of artificial intelligence and automated expert systems.

Another important next step is to extend the approach to non-residential buildings. These entail a significantly higher complexity and new dependencies for the optimization procedure and especially for the integrated simulation models. These arise primarily in the area of ventilation and air conditioning systems as well as refrigeration technology.

Since the approach of decoupling dependencies has already led to a significant acceleration of the existing optimization framework, this also seems to be a promising solution in this case. However, even such a solution can quickly reach its limits, especially due to increased complexity and deeper dependencies.

6 Acknowledgements

Supported by:

Federal Ministry for Economic Affairs and Energy

on the basis of a decision by the German Bundestag

The presented research work and results were partly encouraged by subsidies of the German Federal Ministry of Economics and Technology.

References

Wetter, Michael (2009): Modelica-based Modeling and Simulation to Support Research and Development in Building Energy and Control Systems. In: *Journal of Building Performance Simulation 2(2), p. 143-161. DOI:10.1080/19401490902818259*.

Dirk Müller, Michael Lauster, Ana Constantin, Martin Fuchs, Peter Remmen (2016): “AixLib – an open-source modelica library within the iea-ebc annex 60 framework”. In: *BauSim Conference 2016: 6th Conference of IBPSA-Germany and Austria. September 14-16, Dresden, Germany.*

Roselt, Kersten and Büttner, Christiane (2019). “Wachstumsmarkt energetischer Quartiersumbau”. In: *DIB – Deutsches Ingenieursblatt 10-2019. October 2019, Germany.*

Leimeister, Mareike (2019). “Python-Modelica Framework for Automated Simulation and Optimization”. In: *13th International Modelica Conference. March 04-06, 2019, Regensburg, Germany.*

Eckstädt, Elisabeth; Paepcke, Anne; Hentschel, Alexander; Schneider, André; Nicolai, Andreas; Schumann, Falk (2020). “Simulationsszenarien für Gebäudeenergiesimulation in frühen Planungsphasen“. In: *BauSIM 2020, 8th Conference of IBPSA Germany and Austria, September 23-25, Graz, Austria.*

Dan Hou, Wei Yan, Gang Liu, Zhen Han (2019). “Primary-energy Based Optimization of a New Building District through Simulations on Flat, Building, Block and District Level“. In: *16th International Building Simulation Conference, September 02-04, Rom, Italy.*

Ramón Bergel, Giovana Fantin do Amaral Silva, Max Tillberg, Angela Sasic Kalagasidis (2019). “Energy Performance Modelling: Introducing the Building Early-stage Design Optimization Tool (BeDOT)”. In: *16th International Building Simulation Conference, September 02-04, Rom, Italy.*

Georgios Dermentzis, Fabian Ochs, Alexander Thuer (2019). “Primary-energy Based Optimization of a New Building District through Simulations on Flat, Building, Block and District Level”: In: *16th International Building Simulation Conference, September 02-04, Rom, Italy.*

Theodor Victor Christiaanse, Paul Westermann, Will Beckett, Gaelle Faure, Ralph Evins (2021). “BESOS: a Python library that links EnergyPlus with optimization and machine learning tools”. In: *17th International Building Simulation Conference, September 01-03, Brugges, Belgium.*

Javier Arroyo, Carlo Manna, Fred Spiessens, Lieve Helsen (2021). “An OpenAI-Gym environment for the Building Optimization Testing (BOPTEST) framework”. In: *17th International Building Simulation Conference, September 01-03, Brugges, Belgium.*

Domenico Costantino and Massimiliano Pepe (2021): “Scan-to-HBIM for conservation and preservation of Cultural Heritage building: the case study of San Nicola in Montedoro church (Italy). In: *Journal of Applied Geomatics. DOI 10.1007/s12518-021-00359-2.*

Torsten Schwan and René Unger (2017): “Complex Large-Scale Heating, Cooling and Power Supply Models – Virtual Renewable Power Plant Simulation with Modelica“. In: *15th International Building Simulation Conference, September 07-09, San Francisco, USA.*

Andreas Nicolai and Anne Paepcke (2017): “Co-Simulation between detailed building energy performance simulation and

Modelica HVAC component models". In: 12th International Modelica Conference. May 15-17, Prague, Czech Republic.

Bernd Kochendörfer, Jens H. Liebchen, Markus G. Viering (2010): "Bau-Projekt-Management: Grundlagen und Vorgehens-weisen. (Leitfaden des Baubetriebs und der Bauwirtschaft)". 4. Auflage. Vieweg & Teubner Verlag, ISBN 978-3-8348-0496-9.

 DOI 10.3384/ecp204789