Ontology-based Optimization of Building Automation Systems

Stefan Gaida, Wolfgang Kastner
Institute of Computer Engineering
TU Wien
Vienna, Austria
{sgaida, k}@auto.tuwien.ac.at

Filip Petrushevski, Milos Sipetic
Center for Energy
Austrian Institute of Technology
Vienna, Austria
{filip.petrushevski, milos.sipetic}@ait.ac.at

Abstract—**Modern buildings are equipped with various complex automation systems that produce vast amounts of data. The operation of these systems has significant influence on energy demands and user comfort in the premises. Therefore, an optimal configuration of the systems to increase energy efficiency is desired by the operators of the buildings. An automated decision-making process that evaluates and analyzes the bulk of available data can help to achieve these objectives. Hence, this work introduces a framework for the automatic generation of recommended configuration changes to increase energy efficiency and user comfort. The approach makes extensive use of semantic building information modeling as basis for the selection of potentially useful measures as well as for their assessment. Expected consequences of configuration changes are forecasted via simulation. Additionally, current results of an application of the proposed solution in an office building are discussed.**

Index Terms—**Buildings, building automation, data processing, data analysis, energy efficiency, information management**

I. Introduction

Residential and commercial buildings account for about 40% of total energy consumption [1]. Between 20% and 60% thereof are used for operation of heating, ventilation and air conditioning (HVAC) systems. Hence, energy-efficient indoor climate control can make significant contributions towards global energy savings. Monitoring tools enable building managers to access huge amounts of data regarding energy consumption and indoor climate. Based on their experience, they can take measures to increase energy efficiency and reduce occupants' dissatisfaction. However, as the amount of data as well as the rate at which the data are collected increase, it becomes impossible to analyze the data manually and make appropriate decisions. For this reason, an automated system that supports the decision-making process through data analytics algorithms, simulation, and optimization can help building operators to improve building performance and efficiency more dynamically.

The idea of applying data analytics on building-related data for the detection of inefficiencies and faults is already becoming an established practice. Yu et al. [2] use the association rule mining algorithm FP-growth to explore associations and correlations between climatic data, building operational data, and physical parameters of buildings. The results support building operators who have the required domain expertise to gain useful knowledge for improving energy efficiency. Peña et al. [3] utilize data mining algorithms to extract knowledge from historical data of buildings. Based on these data and additional expert knowledge, rules for the detection of power consumption anomalies are defined. The rules are then used as part of a decision support system which has access to monitoring data. Also Chen et al. [4] perform fault detection in the building sector with data-driven methods including weather based pattern matching and feature based principal component analysis. Their main focus is on whole building systems rather than on sub-systems or components. To handle a large number of data points, feature selection is performed using regression and a genetic algorithm. In [5], anomalous behavior is detected using clustering algorithms and fuzzy logic rule extraction. An ontology for modeling control logic implemented in building automation systems (BASs) is proposed in [6]. They use the ontology as a knowledge base for rule-based verification of designed BAS control logic. Another ontology, called BASont, addresses use cases over the whole life cycle of a BAS [7]. Based on this work, a systematic engineering approach for cooperating, adaptive BASs covering the design, commissioning, and operation phases is presented in [8]. In [9], a cloud-based approach for fault detection and diagnosis in building management systems that takes event and error messages from other buildings into account is described. Zeng et al. [10] use data mining algorithms to create predictive models of HVAC energy consumption and comfort of multiple building zones. The models are required for a heuristic search algorithm which allows to derive optimal settings of setpoints under defined operational constraints.

In our previous work, the architecture of a suggested advanced data analytics framework for energy efficiency is presented [11]. The overall goal of the framework is to automatically generate recommendations improving energy efficiency and comfort in buildings. Figure 1 shows that the sub-systems are grouped into three sections which represent the iterative data analytics steps, namely descriptive analytics, predictive analytics, and prescriptive analytics.

The main component of the descriptive analytics part is the data preparation and data description module. It collects data from BASs and weather forecast providers (WFPs) as well as additional semantic information about the building

978-1-5090-6684-1/18/$31.00 ©2018 IEEE

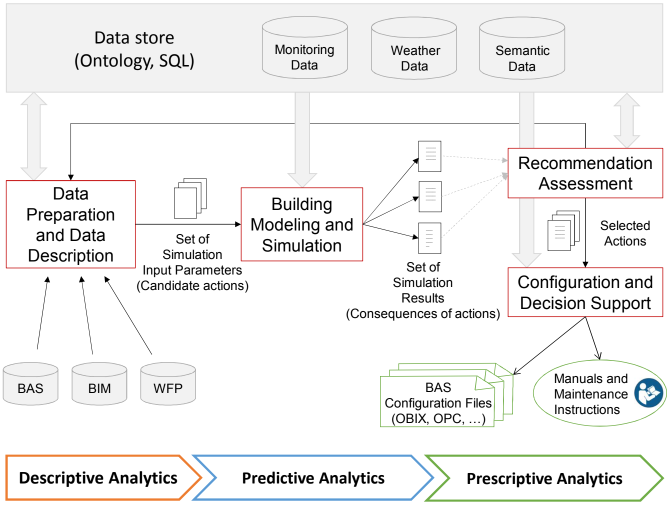

Fig. 1. Advanced data analytics framework architecture (adapted from [11])

including building structure and description of the deployed automation systems. After data preparation, which tries to improve the quality of collected data by methods such as time series alignment, boundary checks, and interpolation, all these information are stored in an internal data store which consists of an Web Ontology Language (OWL) ontology for semantic information and SQL databases for time series data such as sensor readings and weather forecasts. This data store serves as source for all parts of the developed system. Double-sided arrows in Figure 1 indicate that some modules are also writing information to the data store. The second task of the data preparation and data description module is to detect inefficiencies and optimization potential by applying data analytics algorithms on the available data. Based on the results, sets of potential measures to reconfigure the BAS, called candidate actions, are generated. For predictive analytics, simulation is chosen as the preferred method because building characteristics can be integrated more easily than in other forecasting mechanisms. Therefore, a building model is created based on semantic data from external sources such as building information models (BIMs), plant schematics, and device manufacturer information. This simulation model is used to predict the effects of candidate actions on the building's behavior. Additionally, a process for transformation of candidate action sets into sets of simulation inputs is required. Subsequently, the reconfiguration measures are rated in the recommendation assessment module as part of prescriptive analytics. It takes sets of simulation results or consequences of actions as input and selects the potentially best candidate action set. Results of the recommendation assessment can also be used as feedback for the data preparation and data description module to improve the generation of candidate actions. Finally, configuration and decision support converts the selected actions into human readable instructions and configuration files.

In this work, methods for the introduced modules are described in detail with focus on candidate action generation and recommendation assessment. Section II explains the generation of candidate actions including the detection of inefficiencies and optimization potential. Section III gives an overview on the simulation environment, while Section IV is dedicated to the assessment process. Afterwards, results are discussed in Section V, and conclusions drawn in Section VI.

II. Candidate Action Generation

The first step of the proposed approach is to detect optimization potential in the configuration of the BAS and to find options in terms of configuration changes to be used. These options are called candidate actions. In the following, the generation of these actions is described.

In [12], we present an approach for the detection of building performance inefficiencies based on semantic modeling of building systems and expert knowledge. The core component of this preliminary work is an ontology based on OWL 2 which reuses and extends the smart control ontology presented in [13]. The main aspects that are modeled are (1) building structures, (2) devices and appliances, (3) data services, (4) control services, (5) expert rules, and (6) detected shortcomings. The building structure is described by zones (e.g. floors, rooms) which are arranged to each other in a relative manner. Within these zones, devices and appliances like sensors and actuators can be placed. They are modeled as sub-classes of the concept *BuildingResource* and act as service providers which host data and control services. A data service represents provided data such as time series of temperature values including their semantic descriptions like the parameter type (e.g. room temperature) or an engineering unit. A control service, on the other hand, is intended to influence parameters like the room temperature in some zone. In favor, one or more individuals of *ParameterVariation* describe how and under which conditions a parameter can be changed. The class *Rule* is used to model expert rules which are essential for the detection of building performance inefficiencies. A rule has a condition (guard) and at least one conclusion. The guard that evaluates either to *true* or *false* is provided as a formula which can contain variables to address data (via data services) and constants. Conclusions are used to model faults that can occur within the BAS, e.g. room temperature is too high. This enables the definition of rules which express that a particular problem exists when some condition is met. Finally, the detected shortcomings are represented in the ontology as *Incidents*. Important properties of an incident are time, reference to the service or building resource that caused the problem, and a so-called *AnomalyType* that describes the type of the problem (e.g. *High*, *Low*, *BreakdownLikely*).

The described semantic model was successfully used to detect various building performance inefficiencies like comfort violations (e.g. "CO_2 concentration is too high"), system dimensioning issues (e.g. "too many occupants present for proper operation mode of the ventilation system"), and energy inefficiencies (e.g. "ventilation system running although no occupants are present"). For this purpose, an application that interprets the rules and applies them on time series data was developed. However, expert rules for the detection of these issues are defined for each zone in the building separately which is not practicable for large buildings. To overcome this shortcoming and to enable the generation of candidate actions, the *Rule* concept as well as the related application are extended in this work. Firstly, specializations are specified in the ontology (cf. Figure 2). The *ZoneBasedRule* concept allows the definition of rules that cover multiple zones at once. For the selection of involved zones, a so-called *zoneSelector* is used, which is basically a SPARQL query that returns only zones. For instance, the following query returns all offices in the third floor:

```
SELECT ?zone WHERE {
        ?zone a example:Office .
        ?zone example:isPartOf+ example:3rd_Floor }
```

As previously mentioned, data services can be included into rule conditions via variables, more specifically data variables. However, this mechanism is not suitable to address data that is specific for each zone covered by the selector, e.g. each room has an own data service for the representation of its room temperature. To fix this problem, another variable type called *ParameterTypeVariable* is created. Such a variable has a property *hasParameterType* and is used as reference to a data service of the provided parameter type (e.g. RoomTemperature) in the currently processed zone. Additional properties like *hasOrientation* and *hasDeviceType* can be used to address data services more accurately. The former is used to specify the spacial relation of the parameter to a zone. This enables, for example, the distinction between supply air temperature modeled as variable with parameter type *Temperature* and orientation *In*, and exhaust air temperature having type *Temperature* with orientation *Out*. The latter specifies that the addressed data service must be provided by a device of a certain type. When a *ZoneBasedRule* is loaded into the application that performs the analysis of monitoring data, it is split up into separate standard rules for each zone. This process also includes the transformation of parameter type variables into particular data variables. Similar mechanisms are designed for the definition of rules for multiple devices (e.g. all heat pumps).

Conditions like that comfort is not important in non-working hours are considered by active periods. They define season (e.g. cooling season), weekdays, and time when a rule is active. Additionally, the three following sub-classes of incidents that can be detected by an expert rule are defined:

1) *Optimization potentials* describe inadequacy in the BAS configuration that shall be addressed by the generation of candidate actions, e.g. "CO_2 concentration in a zone is too high". The anomaly type of optimization potentials is currently restricted to the values *Low* or *High*.
2) *Faults* are a problem that are not necessarily solved by candidate action generation, but reported to the facility manager.
3) *Critical faults* are similar to faults, but shall be reported to the facility manager more recently and prevent the developed system from creating candidate actions until the situation is resolved.

Analogous to the three incident types, sub-classes of *Conclusion* are specified in the ontology. *OptimizationConclusion*, *FaultConclusion*, and *CriticalFaultConclusion* state that a corresponding incident shall be created when the condition evaluates to *true*. Finally, an occurrence threshold can be specified for each of these conclusions. Only if the occurrence of an incident over some definable time exceeds the threshold it is reported to the facility manager or taken into account for candidate action generation.

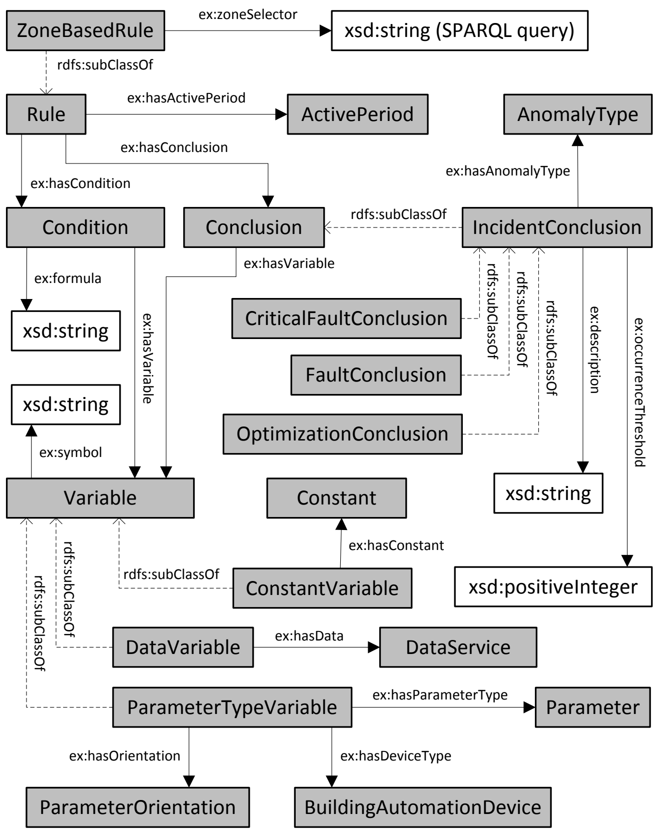

Fig. 2. Ontology representation of zone-based rules

An example for the application of zone-based rules making use of parameter-based variables is to check if the actual room temperature exceeds the desired room temperature by some predefined value (tolerance). Thereby, the rule shall be applied to all offices in the building. The rule condition is related to two instances of *ParameterTypeVariable*. The first one has parameter type *RoomTemperature* and orientation *Inside*. Optionally, the device type can be defined as *TemperatureSensor*. A second variable has parameter type *RoomTemperatureSetpoint* and also orientation *Inside*. The tolerance is provided as a constant or another variable that depends on the zone. In addition, a conclusion with anomaly type *High* that references the variable with parameter type *RoomTemperature* is modeled.

In the following, the process for detection of optimization potential is called performance evaluation. Before this process is started, a fault detection is performed to ensure that no critical faults are present. The actual candidate action generation starts directly after the performance evaluation was executed. For this purpose, all detected optimization potentials that occur more often than the related occurrence threshold are fetched from the ontology. As mentioned before, each optimization potential states that some parameter is too high or low in some zone (or device). The basic idea is to find all possibilities to increase or decrease this parameter. In other words, all control services that are potentially capable to perform this change are fetched from the ontology. If the anomaly type is *High*, all control service that provide a *ParameterVariation* to decrease the concerned parameter in the zone are queried. In case that the type is *Low*, a service has to be able to increase the parameter. Each of these control services is transformed into a candidate action by setting a new value. For this purpose, the current value of the service is fetched from the monitoring database. Range of valid values, the direction in which the value needs to be changed to influence the target parameter in the desired direction, as well as the extent of the change are available in the ontology. An example of a control service together with one parameter variation to lower room temperature is shown in Figure 3. If performance evaluation detects overheating in the zone ”Office1”, this service provides a potential countermeasure. ”DownTrend” and ”LowerValue” state that the value of the setpoint must be lowered to potentially decrease the room temperature. The extent of the value change is set to 0.5.

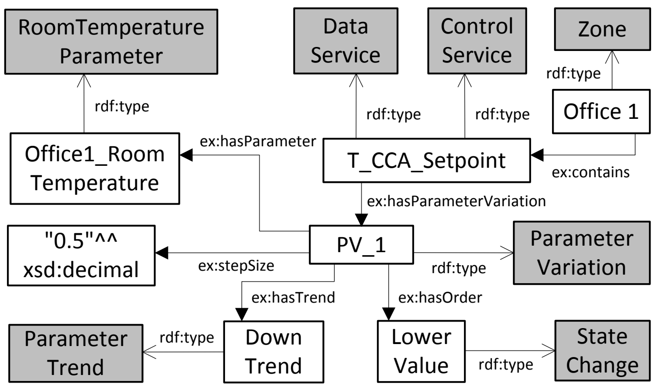

Fig. 3. Excerpt of a control service for candidate action generation

Subsequently, sets of candidate actions are built by combining the created actions. Such a set includes at most one action for each detected optimization potential. Finally, the sets of candidate actions are written to the ontology for further processing.

III. Simulation

The expected impacts of changes in the system configuration on energy consumption and user comfort are determined individually for each candidate action set by means of building simulation. Therefore, an accurate model of the building and the installed HVAC systems is required. As mentioned before, several available external sources such as BIMs, plant schematics, and device manufacturer information are used for the modeling process. In case that accurate BIM models are available (e.g. in Industry Foundation Classes (IFC) format), it is possible to automate parts of the process by using model transformation. Existing approaches show how IFC models can be reused to create building models including geometry and thermal behavior for simulation environments like EnergyPlus and TRNSYS [14] [15]. In this work, TRNSYS was chosen as the preferred simulation software. The parametrization as well as the calibration of the model are supported by the ontology. Before simulation can be used to determine effects of configuration changes it must be ensured that the simulation results correspond to the real world [16]. For this purpose, system configurations, monitoring data (temperature, air quality, and energy consumption), and weather data from the past are used within a calibration process. All data must be available in the ontology at this stage. For each simulation run, the model has to be parametrized with the system configuration to be simulated and a weather forecast including ambient temperature, wind (speed and direction), irradiation, humidity, and air pressure. Hence, an interface for the transformation of configuration data and candidate actions into simulation parameters is needed. This interface consists of two parts. First, a CSV file containing BAS configurations for all sets of candidate actions is generated. Such a configuration includes, for example, setpoints, offsets, or curves (e.g. heating curve). For the creation of this file, the current configuration is fetched from the ontology. This information is needed for two reasons. On one hand, it is used to simulate the behavior of the building without any configuration changes, the so-called baseline. Hence, it is written to the CSV file. The result of this simulation run is later used to decide if one of the proposed changes led to a better building performance than the current configuration. On the other hand, it is adopted by the sets of candidate actions which are again queried from the ontology. Each adopted configuration is then also written to the CSV file. In a second step, the file is processed with a Python tool using the *PyBPS* package to create simulation jobs for TRNSYS. The tool is responsible to start the parametrized simulation runs and to store the results in the file system. Also parallel simulation of different configurations is supported. Finally, the outputs of each simulation run are post-processed for use in following sub-systems. This includes tasks like aggregation of energy consumption for different sources as well as transformation of result data to an expected format. Post-processed simulation results contain at least data for room temperature and CO_2 concentration in all building zones, and energy use for all available energy sources.

IV. Assessment process

In the next step, the simulation results form the basis for an assessment process that enables the selection of the potentially best candidate action set. The assessment is based on the calculation of key performance indicators (KPIs). In the following, the process is introduced followed by the definition of actually applied KPIs. Generally, one comprehensive KPI that can consists of several sub-KPIs is computed for each set of candidate actions. The set with the best result is chosen for further processing. KPIs are defined in the ontology as individuals of the *KeyPerformanceIndicator* concept, which was presented in [12] and is extended in this work as shown in Figure 4. The formula for the computation of a particular KPI

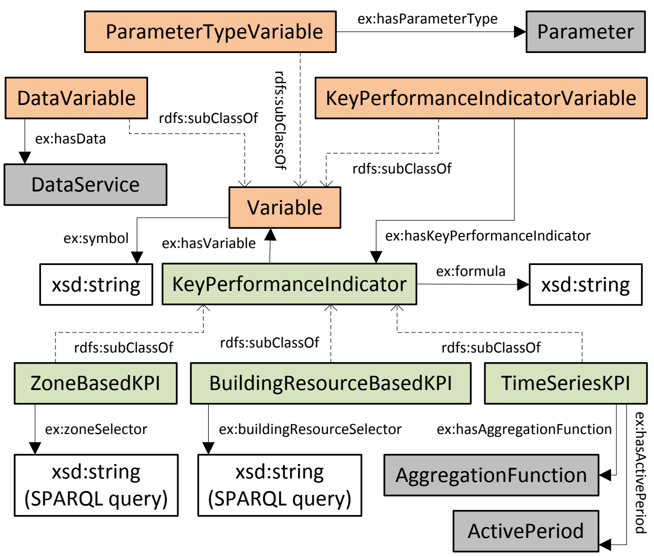

Fig. 4. Ontology concepts for key performance indicators

is represented as a data property and variables like previously described for rule conditions. However, variables are handled in an adapted fashion. If data for some data service is available in the simulation results, it is taken from there, otherwise the last value from the monitoring database is fetched. The new *KeyPerformanceIndicatorVariable* concept is intended to address results of other KPIs. The definition as well as the calculation is independent of the building structure and the utilized energy sources. In other words, they can be used for every building where the proposed solution is deployed and common comfort-related data like room temperature measurements are available. For this purpose, two specializations of *KeyPerformanceIndicator* are introduced. *ZoneBasedKPIs* are calculated for a selection of building zones. The principle is similar to zone-based rules. A selector is used to address the desired zones and parameter-based variables to address data of a specified parameter type. The typical application is the computation of an expected comfort value (e.g. as degree of dissatisfaction) for all zones. *BuildingResourceBasedKPIs*, on the other hand, are intended to compute KPIs for a set of building resources. As energy sources are modeled as building resources, they can be used to calculate energy related KPIs for all available sources. Examples are expected primary energy demands or CO_2 emissions. Most of the data that are used in the assessment process are time series. Hence, also a typical sub-KPI must be computed for time series and not only one point in time. To take this into account, the *TimeSeriesKPI* is specified as a third subclass. Results are calculated for each step in the time series and are then aggregated via an aggregation function (e.g. sum, average) to get a single value. An individual of the class *ActivePeriod* can be used to specify time windows where the KPI is active. Thereby, conditions like that comfort is unimportant when the building is not occupied are taken into account. While *ZoneBasedKPI* and *BuildingResourceBasedKPI* are disjoint classes, which means that a particular individual cannot be of both types, they can be combined with *TimeSeriesKPI*. The actual calculation is performed by an application written in Java.

This approach is used to rate the candidate action sets as follows. The rating for each set is computed as a weighted sum of total primary energy demand, total CO_2 emissions, and comfort-related measures. A lower results leads to a better rating. All influences are expressed as relative changes to the baseline to take different orders of magnitude into account. The baseline is defined as the empty candidate action set and therefore the results for the current BAS configuration. Weights or in other words the importance can be defined by the facility management. For each energy source that is available in the building, factors for the conversion of energy consumption (e.g. in kilowatt hours) into primary energy demand as well as CO_2 emissions are stored in the ontology. This enables the calculation of the first two influences by the utilization of two building resource-based KPIs, where all building resources representing energy sources are addressed. The formula is simply a multiplication of the amount of consumed energy from one source with the corresponding factor. As energy consumption is provided from simulation as a time series, also the *TimeSeriesKPI* concept is used, whereby the aggregation function is set to build the sum over time. Subsequently, two sums over all energy sources are used to get the total values for the building. The assessment of comfort is focused on room temperature and air quality. Humidity can be included in the future. KPIs for both of them must be calculated per zone, and required data is again provided as time series. Hence, KPIs of types *ZoneBasedKPI* and *TimeSeriesKPI* are used. The measure for comfort regarding room temperature, is based on the degree of individual dissatisfaction (DID) [17]:

$$DID(vote) = \frac{1+tanh(2|vote|-3)}{2}$$

where vote is defined as follows:

$$vote(T) = \begin{cases} +3 & , T > T_0 + 2\Delta T \\ -3 & , T < T_0 - 2\Delta T \\ 1.5\frac{T-T_0}{\Delta T} & \text{otherwise} \end{cases}$$

The vote is a function of the room temperature T where T_0 stands for the desired individual temperature, and ΔT for the individual temperature tolerance. Time series for T in each zone are part of simulation results, while T_0 and ΔT are stored in the ontology. Results of the vote function can be interpreted as a scale between cold (-3) and hot (+3). The DID transforms these values to a measure of dissatisfaction which lies between 0 (satisfied) and 1 (dissatisfied). This measure is used as KPI for room temperature related comfort. For air quality, the dimensions of CO_2 concentration violations over working hours are calculated in all zones. First, the extent in ppm to which the CO_2 limit (CO_2_lim) is exceeded is determined for each sample t during working hours in the time series.

$$exc(t) = \begin{cases} CO_2(t) - CO_2\_lim & , CO_2(t) > CO_2\_lim \\ 0 & \text{, otherwise} \end{cases}$$

The result is used to calculate the overall violation in the following way, where $sample\_period$ is the time between two samples in the time series and $working\_hours$ the total number of working hours in the time series:

$$CO_2\_violation = \frac{\sum_t exc(t) * sample\_period}{working\_hours}$$

To take differences between zones into account, a weighted sum of average, maximum, and standard deviation is computed for the degree of dissatisfaction and CO_2 violations. Finally, the relative changes to the baseline are calculated for all four influences before the overall sum is formed. After the calculation is finished, the set of candidate actions with the best result is marked as selected in the ontology. The results of all KPIs and sub-KPIs are also stored in the ontology to support future improvements of candidate action generation.

V. Results

To show the feasibility of the approach, recommendations for BAS configuration adaptions in an office building located in Vienna are generated. The ontology is populated with information regarding building services and structure in a semi-automatic manner. Some of the required data including devices and their locations, data points, zone hierarchies, and addresses can be extracted automatically from the BAS. However, other important data such as parameter types or affected parameters for control services have to be added manually using additional sources like schematics. The building model for simulation is created utilizing an available BIM model and plant schematics. Note that the degree of automation for both modeling tasks heavily depends on the used BAS software as well as on the availability and accuracy of other data sources. To gain monitoring data and weather forecasts, two external interfaces are developed. Historical data are fetched directly from the SQL database of the BAS. Afterwards, data sanitation including time series alignment, and interpolation for missing values is performed. Weather forecasts, on the other hand, are obtained from a weather data provider via Web Services.

For the detection of energy inefficiencies and comfort violations, four zone-based rules that cover the following undesired situations are defined:

- *Room temperatures during working hours are too low in heating season* which leads to dissatisfaction among the occupants.
- *Zones are overheated during heating season*. This causes preventable energy consumption. If the situation occurs during working hours, there are also negative effects on occupants' comfort.
- *Air quality in office rooms is bad during working hours* due to high CO_2 concentration which also leads to occupants' dissatisfaction.
- *CO_2 concentration in rooms is unnecessarily low* which is an indicator that there is potential for more energy efficient operation of ventilation systems.

By applying the rules to monitoring data covering approximately two month in the heating season, optimization potential in several zones is found. Most of the time, room temperatures are significantly higher than the desired values. As mentioned before, this can have negative effects on energy efficiency and comfort. Hence, candidate actions to lower the temperature in affected rooms shall be generated. In addition, it turns out that CO_2 concentrations are far below the limits for high air quality standards in office rooms, both during working and non-working hours. Therefore, there could be potential to save energy by decreasing the air volume flow into zones.

The room temperature of a typical zone in the selected building can be influenced via concrete core activation (CCA) and the ventilation system. For CCA, there exists a valve per zone that controls the volume flow based on a room temperature set point. The water temperature of the CCA system is set for the whole building dependent on the outside temperature. Therefore, there are two possibilities to influence the operation of the CCA in heating season, namely changing the set point per zone (cf. Figure 3) and adapting the heating curve that describes the relationship between outside temperature and water temperature. Both options are modeled in the ontology as control services. Additionally, there are six control services to influence volume flow and supply air temperature of the ventilation system. The developed algorithm for candidate action generation finds all of these eight services in the ontology and creates actions by setting new values for them. The number of values that should be tried out for each control service by simulation is set to three. Currently, actions are only generated for one zone at once. Overall, 39 candidate actions to lower the room temperature in the selected zone are generated. The relatively high number of actions comes from changes of multidimensional control services like heating curves. Such a curve is piecewise linear and is defined as a series of points with Cartesian coordinates which can be changed on both axis. For the second optimization potential, control services to decrease the air volume flow into the zone are fetched from the ontology. The only option is to change a curve controlling the volume flow controller. By choosing zero or one action for each optimization potential and taking conflicts between them into account, a total of 139 sets of candidate actions are generated, including the empty set which represents the baseline. Afterwards, the effects on the building are predicted for each set by simulation in TRNSYS for three days in advance, which took about 20 minutes on a laptop with four CPU cores.

Regarding the assessment process, good weight factors for occupants' dissatisfaction, air quality, primary energy demand, and CO_2 emissions are essential. In a first run, all impacts are weighted equally. Also the factors for average, minimum, and standard deviation of dissatisfaction and air quality KPIs over all zones are set to equal values. With these settings, the most appropriate option is to adapt the CCA heating curve by changing the coordinates of selected points by a relatively large extent. All points, which are close to the outside temperatures which were present while optimization potential for candidate action generation was detected, are moved to the left on the x-axis by 6°K. This means that the water temperature of the CCA will be much lower for the affected outside temperature range. The results are significantly lower primary energy demands and therefore lower CO_2 emissions, but also worse comfort because room temperatures drop to 19°C. This is of course no satisfying solution. Therefore, the weights of comfort-related KPIs are increased. With factors of 0.5 for occupants' dissatisfaction, 0.3 for air quality, and 0.1 for primary energy demand as well as for CO_2 emissions the results are much better. The best option is still to adapt the heating curve of the CCA, but by a smaller extent. Instead of 6°K, the points are shifted to the left by 2°K. This leads to about 6% less temperature-related dissatisfaction and decreases primary energy demand as well as CO_2 emissions from HVAC operation by about 30%. The value of the assessment KPI is lower and therefore better than for the baseline by 9%. Note that the calibration of the simulation model is an ongoing investigation and still in progress. Hence, the exact values must be taken with caution.

Finally, a human-readable report for the facility manager is generated automatically. This document contains a description of detected problems and optimization potentials, the recommended configuration changes, and expected effects on energy consumption as well as user comfort.

VI. Conclusion

Growing amounts of data from various sources such as BASs, BIMs, or WFPs available in modern buildings make it difficult to analyze them manually and to make appropriate decisions. Thus, this work presents an approach for automatic generation of recommendations to improve energy efficiency and user comfort. Rule-based data analysis to detect inefficiencies and optimization potential in buildings serves as starting point for the generation of sets of potential measures to reconfigure the BAS. The effects of these measures are predicted using simulation of the building behavior. For interpretation of the simulation results and to select the best option, an assessment method is introduced. Finally, the approach is applied to parts of a real-world office building to generate human-readable recommendations. The whole process relies on the use of semantic information about the building and its systems that are modeled in an ontology.

In future work, more advanced options for the generation of candidate action sets using feedback from previous runs will be examined. The current approach leads to an infeasible number of sets when problems are detected in several rooms. Also possibilities to apply selected measures directly to the BAS after they are checked and confirmed by a facility manager are investigated. OPC UA or BACnet/WS are potential integration technologies for this purpose. Additionally, comprehensive evaluation of the results by applying the recommended actions in a real-world building is required and planned.

Acknowledgment

This work was funded by FFG (Austrian Research Promotion Agency) under the *ADA-EE* project (849968).

References

[1] D. Zhai and Y. C. Soh, "Balancing indoor thermal comfort and energy consumption of air-conditioning and mechanical ventilation systems via sparse firefly algorithm optimization," in *International Joint Conference on Neural Networks (IJCNN)*, May 2017, pp. 1488–1494.

[2] Z. J. Yu, F. Haghighat, B. C. Fung, and L. Zhou, "A novel methodology for knowledge discovery through mining associations between building operational data," *Energy and Buildings*, vol. 47, pp. 430–440, 2012.

[3] M. Pea, F. Biscarri, J. I. Guerrero, I. Monedero, and C. Len, "Rule-based system to detect energy efficiency anomalies in smart buildings, a data mining approach," *Expert Systems with Applications*, vol. 56, pp. 242–255, 2016.

[4] Y. Chen and J. Wen, "A whole building fault detection using weather based pattern matching and feature based PCA method," in *IEEE International Conference on Big Data*, Dec 2017, pp. 4050–4057.

[5] D. Wijayasekara, O. Linda, M. Manic, and C. Rieger, "Mining building energy management system data using fuzzy anomaly detection and linguistic descriptions," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 3, pp. 1829–1840, Aug 2014.

[6] G. F. Schneider, P. Pauwels, and S. Steiger, "Ontology-based modeling of control logic in building automation systems," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 3350–3360, Dec 2017.

[7] J. Ploennigs, B. Hensel, H. Dibowski, and K. Kabitzsch, "BASont - a modular, adaptive building automation system ontology," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, Oct 2012, pp. 4827–4833.

[8] A. M. Gibney, S. Rea, M. Lehmann, S. Thior, S. Lesecq, M. Hendriks, C. Gardeux, L. T. Mai, F. Pacull, J. Ploennigs, T. Basten, and D. Pesch, "A systematic engineering tool chain approach for self-organizing building automation systems," in *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, Nov 2013, pp. 7696–7701.

[9] S. Lazarova-Molnar and N. Mohamed, "A framework for collaborative cloud-based fault detection and diagnosis in smart buildings," in *7th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO)*, April 2017, pp. 1–6.

[10] Y. Zeng, Z. Zhang, and A. Kusiak, "Predictive modeling and optimization of a multi-zone HVAC system with data mining and firefly algorithms," *Energy*, vol. 86, pp. 393–402, 2015.

[11] D. Schachinger, S. Gaida, W. Kastner, F. Petrushevski, C. Reinthaler, M. Sipetic, and G. Zucker, "An advanced data analytics framework for energy efficiency in buildings," in *Emerging Technologies and Factory Automation (ETFA), IEEE 21st Int. Conf. on*, 2016, pp. 1–4.

[12] F. Petrushevski, S. Gaida, B. Beigelboeck, M. Sipetic, G. Zucker, C. Schiefer, D. Schachinger, and W. Kastner, "Semantic building systems modeling for advanced data analytics for energy efficiency," in *Building Simulation*, Aug 2017, pp. 622–627.

[13] D. Schachinger and W. Kastner, "Semantics for smart control of building automation," in *IEEE 25th International Symposium on Industrial Electronics (ISIE)*, June 2016, pp. 1073–1078.

[14] G. I. Giannakis, G. N. Lilis, M. A. Garcia, G. D. Kontes, C. Valmaseda, and D. V. Rovas, "A methodology to automatically generate geometry inputs for energy performance simulation from IFC BIM models," in *Building Simulation*, Dec 2015, pp. 504–511.

[15] K.-U. Ahn, Y.-J. Kim, C.-S. Park, I. Kim, and K. Lee, "BIM interface for full vs. semi-automated building energy simulation," *Energy and Buildings*, vol. 68, pp. 671–678, 2014.

[16] S. Li, K. Deng, and M. Zhou, "Sensitivity analysis for building energy simulation model calibration via algorithmic differentiation," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 905–914, April 2017.

[17] S. Ari, I. A. Cosden, H. E. Khalifa, J. F. Dannenhoffer, P. Wilcoxen, and C. Isik, "Constrained fuzzy logic approximation for indoor comfort and energy optimization," in *Annual Meeting of the North American Fuzzy Information Processing Society*, June 2005, pp. 500–504.