

Adaptable Room Automation Using Multi-agent System

Ganesh Ramanathan
ganesh.ramanathan@student.unisg.ch
University of St.Gallen
St. Gallen, Switzerland

ABSTRACT

Building automation (BA) systems ensure comfort and safety of the occupants while optimizing energy consumption. However, the requirements of space comfort and the capabilities of the equipment which serve it often change during the lifetime of the building. This is seen especially for systems like heating, ventilation, and air conditioning (HVAC), lighting, etc. in rooms. The current approach of manually re-engineering the room automation system to realign with the changes is time-consuming (and hence, higher costs), and misaligned automation can result in energy loss. To address this challenge, we show how goal-driven multi-agent systems which use procedural reasoning in their planning process enable automation systems to recognize and dynamically readjust to changing requirements and environment capabilities. The novelty of our approach lies in decomposing the available high-level feature requirements of the rooms to goals for the agents while providing it semantic description of the room environment. We validated our approach by implementing an automation system for a real-life office room and subjecting it to changes in requirements and equipment capabilities.

KEYWORDS

building automation, fault detection, semantic data

ACM Reference Format:

Ganesh Ramanathan. 2021. Adaptable Room Automation Using Multi-agent System. In *Proceedings of The 1st St. Gallen Pervasive Computing Forum (SgPervConf '21)*. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

1 INTRODUCTION

We first describe in brief a broad outline of engineering procedures in BA, and then highlight the challenge that the requirements, the environment, and the automation system are difficult to remain aligned in the face of changes in each of them.

Engineering of BA systems are based on design descriptions of mechanical, electrical, and plumbing (MEP) systems like HVAC, lighting, fire-safety etc. along with high-level description of equipment monitoring, controls, and coordination requirements (including references to norms and regulations).

The BA engineering process is carried out in parallel for each of the disciplines like HVAC and fire-safety by one or more vendors,

wherein the respective BA engineers create the bill of materials and functional specification of the automation system. This functional specification is provided in the following documents:

- (1) **Data point list** showing inputs, outputs, and virtual data points, their role in control logic, and their interface to respective automation controllers.
- (2) **Sequence of operation** which describes high-level goals and control strategy to achieve it (see [17] for further elaboration)
- (3) **Process schematic** which shows the equipment and its energy and mass flow relationships. This is augmented with the placement of sensors and actuators.

During the usage of the building spaces, the facility managers often encounter new requirements or errors in the designed behavior. In a survey at a BA project organization of Siemens AG where the author is affiliated, it was found that medium-sized buildings like offices and hotels require re-engineering of control behavior in about a third of the automation controllers in the first two years of operation, whereas large-sized installation like airports require constant re-engineering throughout its life (an informal term *continuous commissioning* is used in the BA industry to denote this situation). Most of the re-engineering of the control logic is required because of change in usage of room spaces, change in its installed MEP equipment, or need for changes in control strategy in the room. **In almost half-the cases, the need for re-engineering is realized late because the facility managers assumed based on the sequence of operation that the changes will not need re-engineering, whereas the exact control strategy was found to be inconsistent with the changes.** Such inconsistencies are often hard to discover and lead to sub-optimal performance and energy wastage.

Hence, our research goal was to find out how room automation system can be achieved based on goals derived from the functional specification.

In the following sections, we first describe the existing approaches to enable adaptable automation systems, and then describe our approach based on multi-agent systems (MAS) followed by its evaluation in a real-life deployment.

2 STATE OF THE ART

We surveyed existing approaches in the industry and research to identify potential reuse of the following building blocks needed in our work:

- (1) Tools and information model to capture BA functional specifications
- (2) Ontology to describe the room environment
- (3) Programming models and architecture that drive automation behavior based on goals derived from functional specification

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SgPervConf '21, December 23, 2021, St.Gallen, Switzerland

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.

<https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

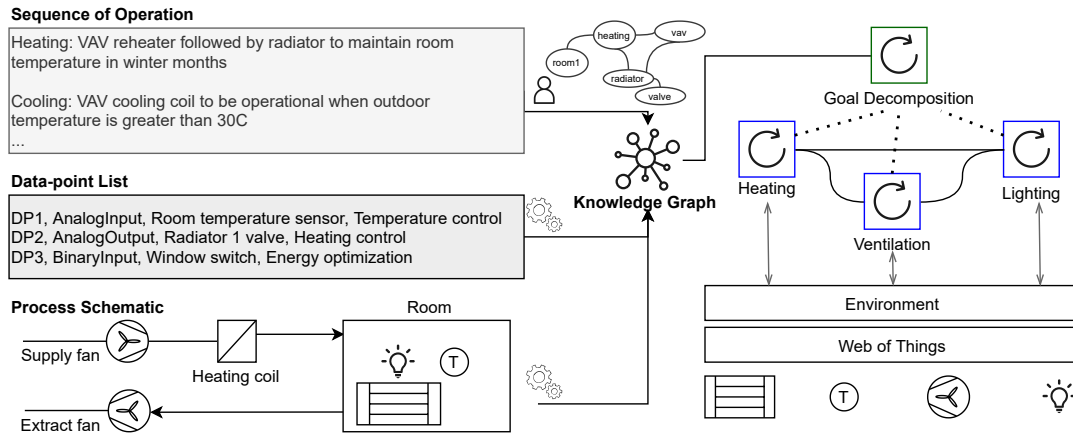


Figure 1: Our implementation of a MAS-based room automation whose goals are obtained from functional specifications.

In one of the earliest work on the topic, [17] proposes an information model to describe the *sequence of operation* by defining a vocabulary of well-known control functions (like temperature control) and linking them using predicates describing the coordination relationship (e.g. energy management *specifies set point* for temperature control). The control functions are also linked to the data points required for its operation. However, it does not specify rules or formalization of the semantics. [7] proposes generation of abstract *design templates* for controls based on rules defined by domain experts to map functional requirements to function blocks. This approach however requires presence of extensive rules to cover all possible combination of physical process in the room. Similarly, [4] considers semantic models of standard room automation strategies so that they can be assigned to rooms based on feature requirements. The feasibility to link abstract room functions to the space description has been described in [2], while [8] (using IEC 61499 model), [19], and [18] advocate the need to describe the exact functioning of the control logic.

In the last decade, ontologies to describe BA systems has seen significant adoption. Project Haystack¹ provides an extensive syntactic tagging vocabulary to describe the equipment and data points, but lacks in richness to express relationships. The BRICK [1] was the first openly available ontology to describe the building topology, the equipment, and physical process relationship. It however lacks the ability to describe automation functions, and its vocabulary is restricted mostly to terms in HVAC. A more extensive survey of ontologies in the building domain can be found in [12].

Adapting behaviour of automation systems to changing needs and environment has been explored by several researches. [15] proposes the use of mediators which communicate in a hierarchical structure using messages with pre-defined semantics to find appropriate functions to meet the goals. The architecture however relies on the specialized mediators to interact in a tightly coupled manner as it advocates centralized orchestration of function (called *home agents*) which is not practical in large installations. Autonomous and goal-driven behaviour by software agents which are social and proactive has been explored for automation systems too [14, 16, 20].

While they draw on the benefits of MAS to adapt to dynamic environments, the agent goals, behaviour, interaction, and organization are manually engineered (i.e. in code). Though some research address the use of MAS to achieve adaptive controls in BA [16], they have not looked at reusing the vast repertoire of control strategies and simulation models for different equipment and physical processes known in the BA domain to facilitate the agent planning process.

Inspiration for adaptive automation systems can also be drawn from the smart manufacturing and robotics [6] domain. [13] shows how top-down planning can be achieved where actions are composed using semantically described web services. [10] describes how service compositions can be created based on specification of the product to be manufactured (the equivalence to application in BA can be viewed as choice of process control that achieves a state of the environment in its goal).

3 APPROACH

Room automation involves multiple functional disciplines like HVAC, lighting, fire safety etc. which are engineered independent of one another, but need to operate in co-ordination based on local and global goals. Hence, as compared to centralized orchestration of functions or service composition, the paradigm of multi-agent systems where agents operate autonomously, but coordinate (and in some cases, compete) to achieve their local goals (process control) as well as global (system-wide constraints and coordination) goals provides better scalability of solutions in BA. (An interesting discussion on the applicability of MAS can be found here [9], and [11] makes a similar argument for the case of home automation).

For our implementation, we chose JaCaMo² as the multi-agent-oriented programming (MAOP) framework as it offers in addition to multi-agent programming, the possibility to model both the environment and the agent organization as first-class entities (see [3] for details).

An office room with a typical combination of HVAC and lighting equipment was considered as a candidate for implementation and evaluation of our approach. We obtained the functional specification for the room automation of this office in the form of *data*

¹<https://project-haystack.org/>

²<http://jacamo.sourceforge.net/>

point list (CSV file), *sequence of operation* (human-readable text), and *process schematic* (CAD file). A software program was created to translate the *data point list* and the *process schematic* to RDF. The translator referenced the BRICK ontology to depict data points, equipment, and process relationships in the room. Based on the terms proposed by [17], the sequence of operation was manually translated to RDF. As opposed to automated translation of process diagram and datapoint list, manual translation of sequence of operation was required because there is neither a standard structure or formal language to model this specification. We facilitated the translation by providing a GUI tool which suggested terms and relationships from BRICK ontology augmented with proprietary terms for describing functions.

A heterogeneous device ecosystem like in BA, where multiple communication protocols and information models are employed, creates a challenge for their integration to the agent's environment [5]. To address this, data points representing the sensors and actuators were described using Web of Thing (WoT) Thing Description (TD)³ and integrated into the environment program of JaCaMo. In this way, the agents could then use semantic query (encoded in SPARQL) like "room temperature sensor" or "valve(s) of the radiator(s) in the room" to find artifacts in the environment and interact with them (e.g. read sensor value, or send actuation command).

A single agent was assigned the responsibility to instantiate (based on need) agents specialized for heating, cooling, ventilation, and lighting controls in the room. This *management agent* used the semantic description of the *sequence of operation* to determine which specialized agents are required and assigned them local goals (e.g. to maintain thermal comfort, or constant light level) as well as global goals (e.g. energy efficient heating and cooling).

The *application-specific agents* had access to different plans which reflect possible control strategies. For the immediate purpose of our research we hard-coded these plans in the agent's program. However, in our future work, we see the possibility of providing access to such plans in form of a common library.

The agents were deployed on distributed computational nodes on the network. We used automation devices from Siemens AG which provided the capability to host the JaCaMo program (which needs a Java run-time environment).

4 EVALUATION

We evaluated the ability of our MAS-based room automation to adapt to changes in goals and capabilities of the environment. In this process, apart from validating the correct functioning of the MAS-based system, we also compared the time required to re-engineer the system using conventional BA programming workflow versus the time required to update the functional specification (and consequently the knowledge graph) which enables the agents to re-plan. We considered the following test cases which reflect the scenarios of change in usage, regulations, system-wide goals, and available means of acting upon and perceiving the environment:

- (1) Room usage: Change the usage of the room from *office* to *electronics lab* which requires higher ventilation, negative room pressure, humidity control, and constant light output.

Changes	Conventional	MAS-based
Room usage	90 min	30 min
Regulation	5 min	10 min
Global goals	30 min	15 min
Means of perception	30 min	5 min
Means of action	30 min	5 min

Table 1: Comparison of time required for re-engineering the automation system

- (2) Regulation: Update ventilation standard for number of air-changes per hour. This influences the working of the ventilation and energy management system.
- (3) Global goal: Specify energy consumption limit for the room, which should cause the individual systems in the room to constrain their operation to meet the goal.
- (4) Means of perception: Add or remove a second temperature sensor, and also a humidity sensor in the room, which should result in the systems choosing a more appropriate control strategy. For example, the additional temperature sensor should cause the averaged value to be used, whereas the humidity sensor should cause the heating and cooling systems to achieve humidity control goal if required.
- (5) Means of action: Add or remove a radiator and a humidifier to the room. Depending on the load conditions, the second radiator should either be operated sequentially or in parallel to the first. The presence of the humidifier should cause an additional control system to be instantiated (based on regulations) to maintain minimum room humidity.

We first confirmed the functioning of the MAS-based room automation for each control scenario on a reduced-scale physical model of the room, and then deployed it in the actual office room installation. We then compared the time taken to re-configure or re-engineer the automation system using traditional workflow (conducted by an experienced BA engineer) versus the time taken for the agents to obtain update goals and realign their planning (see table 1)

Changes in both room usage and global goals affect multiple disciplines (heating, ventilation, lighting) in their local operation as well as their coordination with others. To deal with this, traditional BA engineering involves extensive manual refactoring of the control programs. In some specific cases, like the update of regulation for ventilation, BA control strategies may already encapsulate this feature and it requires only an update of a parameter. In contrast, re-engineering of our MAS-based system is driven by update of description of sequence of operation which is then captured in the knowledge graph. The following snippet of triples shows an example:

```
:room1 :hasUsageKind :lab
:lab :hasProcessNeed :ventilationControl.
:lab :hasProcessNeed :humidityControl.
:lab :hasProcessNeed :pressureControl.
:ventilationControl :hasMinLevel 500.
:pressureControl :hasSetpoint -7
:lab :hasMaxEnergyLimit 10
```

³<https://www.w3.org/TR/wot-thing-description/>

This results in new agents being instantiated for pressure and humidity control, whereas the existing agent responsible for ventilation control receives an updated goal. The pressurisation and ventilation agents exchange beliefs to synchronize their operation (e.g. ventilation agent would delay increasing the airflow if the pressurization agent is in the process of ramping up the extract fans). Such co-ordinations do not need to be specified explicitly because they are known in the control strategies.

In the traditional BA engineering process, adding of a second temperature sensor with the intent of aggregating measurements (min, max, or average) required binding the control program to the new data source and assigning an additional function block for the math function. Whereas adding a humidity sensor (with the implicit intention of controlling the room humidity) required changing the heating and cooling control logic. Similarly, change in heating capability by adding or removing an additional radiator in the room required binding the control program output to the device (i.e. radiator valve) plus changing the control logic to sequence the radiators. In the case of the MAS-based systems, availability of new artifacts in the environment caused the goals (e.g. control humidity if sensor is available) and plans (e.g. use averaging of temperature) to become valid in the agents. Hence, the effort required to re-engineer the system in this case was limited to specifying the presence of the new sensors and devices in the environment. In case of change in regulation, we realized that operational parameters of BA functional blocks in traditional BA engineering tools were linked to a library of norms (or regulations). Hence, change in norms required only update of its parameters, whereas in our case we had to extend the knowledge graph with description of the norm and assign it to the agent organization (thus needing additional design-time effort).

Our study was conducted in a functional scenario limited to a single room. This naturally demands less computational resources on behalf of the management agent, and similarly less overhead for translating the source specification and updating the knowledge graph, both of which are expected to be significantly more complex in a large-scale deployment. We have manually converted control strategies for room automation to code, and made it available to the agents. Also, agent-to-agent communication for various coordination scenarios were hard-coded in the agent program. This results in higher memory footprint on the device in comparison to the traditional deployment where only the required control logic is compiled and downloaded. Finally, the agent roles, global goals, and social coordination was managed by the management agent. This function should have ideally belonged to an organisation management abstraction in the MAOP (topic for further work).

5 CONCLUSION

The challenge of achieving room automation system which is adaptable to changing usage and environment capability was examined in the light of availability of functional specifications that can be decomposed to goals for autonomous agents. We have shown that goals derived from the specifications along with description of the environment can enable agents to reason about procedures to achieve the goals. Our proof-of-concept implementation demonstrated that this method clearly helps reduce the time required for

re-engineering. We have also seen that the agents' planning process can be made responsive to changes in the environment's capabilities, thereby making it adopt the most optimal control strategy (which in turn helps avoid energy loss). However we see further work required in the area of generating specifications of agent organization based on the BA functional specification. Also, the ability of the agent to access function blocks as resources to compose its control strategy will enable not only reuse of existing know-how in BA, but allow agents to run on embedded hardware devices.

REFERENCES

- [1] Bharathan Balaji, Arka Bhattacharya, Gabriel Fierro, Jingkun Gao, Joshua Gluck, Dezhong Hong, Aslak Johansen, Jason Koh, Joern Ploennigs, Yuvraj Agarwal, et al. 2016. Brick: Towards a unified metadata schema for buildings. In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*. 41–50.
- [2] G Benndorf, N Réhault, M Clairembault, and T Rist. 2017. Describing HVAC controls in IFC—Method and application. *Energy Procedia* 122 (2017), 319–324.
- [3] Olivier Boissier, Rafael H Bordini, Jomi F Hübner, Alessandro Ricci, and Andrea Santi. 2013. Multi-agent oriented programming with JaCaMo. *Science of Computer Programming* 78, 6 (2013), 747–761.
- [4] Björn Butzin, Frank Glatowski, Christoph Niedermeier, Norbert Vicari, and Egon Wuchner. 2014. A model based development approach for building automation systems. *19th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2014* (2014). <https://doi.org/10.1109/ETFA.2014.7005365>
- [5] Andrei Ciortea, Olivier Boissier, and Alessandro Ricci. 2018. Engineering worldwide multi-agent systems with hypermedia. In *International Workshop on Engineering Multi-Agent Systems*. Springer, 285–301.
- [6] Matthew Crosby, Michael Rovatsos, and Ronald PA Petrick. 2013. Automated agent decomposition for classical planning. In *Twenty-Third International Conference on Automated Planning and Scheduling*.
- [7] Henrik Dibowski, Joern Ploennigs, and Klaus Kabitzsch. 2009. Automated design of building automation systems. *IEEE Transactions on Industrial Electronics* 57, 11 (2009), 3606–3613.
- [8] Victor Dubinin and Valeriy Vyatkin. 2006. Towards a formal semantic model of IEC 61499 function blocks. In *2006 4th IEEE International Conference on Industrial Informatics*. IEEE, 6–11.
- [9] Viviana Mascardi, Danny Weyns, Alessandro Ricci, Clara Benac Earle, Arthur Casals, Moharram Challenger, Amit Chopra, Andrei Ciortea, Louise A Dennis, Álvaro Fernández Díaz, et al. 2019. Engineering multi-agent systems: State of affairs and the road ahead. *ACM SIGSOFT Software Engineering Notes* 44, 1 (2019), 18–28.
- [10] Simon Mayer, Dominic Plangger, Florian Michahelles, and Simon Rothfuss. 2016. UberManufacturing: A goal-driven collaborative industrial manufacturing marketplace. In *Proceedings of the 6th International Conference on the Internet of Things*. 111–119.
- [11] Gianluca Morganti, Anna Maria Perdon, Giuseppe Conte, and David Scaradozzi. 2009. Multi-agent system theory for modelling a home automation system. In *International Work-Conference on Artificial Neural Networks*. Springer, 585–593.
- [12] Marco Pritoni, Drew Paine, Gabriel Fierro, Cory Mosiman, Michael Poplawski, Avijit Saha, Joel Bender, and Jessica Granderson. 2021. Metadata Schemas and Ontologies for Building Energy Applications: A Critical Review and Use Case Analysis. *Energies* 14, 7 (2021), 2024.
- [13] Juha Puttonen, Andrei Lobov, Maria A. Cavia Soto, and José L. Martínez Lastra. 2015. Planning-based semantic web service composition in factory automation. *Advanced Engineering Informatics* 29 (2015), 1041–1054. Issue 4. <https://doi.org/10.1016/j.aei.2015.08.002>
- [14] Bing Qiao, Kecheng Liu, and Chris Guy. 2006. A multi-agent system for building control. In *2006 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*. IEEE, 653–659.
- [15] Michele Ruta, Floriano Scioscia, Giuseppe Loseto, and Eugenio Di Sciascio. 2013. Semantic-based resource discovery and orchestration in home and building automation: A multi-agent approach. *IEEE Transactions on Industrial Informatics* 10, 1 (2013), 730–741.
- [16] Ueli Rutishauser and A Schafer. 2002. Adaptive Building Automation. A multi-agent approach. *Research project* (2002).
- [17] Jeffrey Schein. 2007. An information model for building automation systems. *Automation in construction* 16, 2 (2007), 125–139.
- [18] Georg Ferdinand Schneider, Pieter Pauwels, and Simone Steiger. 2017. Ontology-Based Modeling of Control Logic in Building Automation Systems. *IEEE Transactions on Industrial Informatics* 13 (2017), 3350–3360. Issue 6. <https://doi.org/10.1109/TII.2017.2743221>

- [19] Michael Wetter, Milica Grahovac, and Jianjun Hu. 2019. Control description language. In *Proceedings of The American Modelica Conference 2018, October 9-10, Somberg Conference Center, Cambridge MA, USA*. Linköping University Electronic Press, 17–26.
- [20] Tehseen Zia, Roland Lang, Harold Boley, Dietmar Bruckner, and Gerhard Zucker. 2009. An autonomous adaptive multiagent model for building automation. *IFAC Proceedings Volumes* 42, 3 (2009), 250–254.