A Semi-Decentralized Self-Adaptive IoT Architecture for Energy Efficiency in Smart Households

Samuele Giussani[1]

[1]Linnaeus University, Växjö, Sweden

Abstract
The advent of IoT opened up countless possibilities for the realization of smart devices, architectures and systems. In several circumstances, however, these components are designed to provide a good experience in terms of performance as well as usability, often choosing them over other features, such as energy efficiency. In this paper, a model-based architecture derived from the Collect-Organize Pattern for Self-Adaptation is presented through a use case involving a smart household, and a brief discussion on its applicability in a realistic scenario is provided.

Keywords
MAPE-K, Runtime models, Energy efficiency, Smart homes

1. Introduction

Nowadays, many domains are transitioning towards or have already incorporated Internet of Things (IoT) devices in their operations, such as agriculture [1], manufacturing [2], and home automation [3]. One of the main strengths of IoT is the capability to collect, transfer and process large amounts of data with a relatively simple structure composed of sensors and, often, actuators orchestrated by devices located in the fog, in the cloud, or both layers [4]. Unlike the layers, which possess a definite classification and standardization, at the moment of writing a widely adopted and recognized reference architecture for IoT systems is missing, with the literature overflowing with several frameworks and suggestions that may sometimes be inconsistent or contradictory to each other [5]. Moreover, many of these proposed architectures concentrate their novelty on either performance improvements or energy consumption optimization, but rarely on the balance of both aspects and they often do not take into consideration the variability of the environment the architecture is implemented in and its necessity of being resilient to change.

In this short paper, a use case involving IoT devices in a smart household is presented to discuss the feasibility of a decentralized self-adaptive architecture that is able to comply with changing optimization requirements for energy consumption through the use of architectural and adaptation goals models. The self-adaptation mechanism is based on multiple instances of MAPE controllers [6] that have access to both a local knowledge (K) component and one with a wider scope as well as a global one, each comprising its runtime models.

MeSS@STAF 2023
samuele.giussani@lnu.se (S. Giussani)

2. Architecture and Use Case

In Figure 1, the proposed architecture is shown applied to the considered use case. The analyzed scenario involves a group of flats, each one composed of several apartments equipped with various IoT devices. In particular, each housing unit can count on multiple sensors and actuators that monitor and regulate the heating system, the electrical system, and the water supply, along with smart devices that can acquire data based on the users' preferences and offer basic services such as media content streaming. Plus, a control panel offers a visualization of the gathered information as well as shows the targets of the self-adaptation process, and can influence it with manual settings that change the adaptation goals.

Each of the described components is linked to a local controller responsible for performing self-adaptation operations locally, with its own MAPE-K feedback loop mechanism [7]. In other words, each sensor and actuator pertaining to the electrical system is connected to a dedicated managing component, and the same applies to the heating, the water supply, and the other devices. The control panel, instead, is conceptualized as a unified view for the many views offered by all the subsystems and, similarly, is a centralized control unit of the apartment for all the manual adjustments that may apply to each subdomain. In this way, it allows editing threshold values on the adaptation models contained in the knowledge base (K) of a single controller without influencing the other ones.

Since the main purpose of this architecture is to keep a desirable performance standard while guaranteeing a good level of energy efficiency, the selected architectural pattern itself has to be as energy efficient as possible and, at the same time, able to effectively support the adaptation operations. Considering the literature existing on proposed self-adaptive patterns for Cyber-Physical

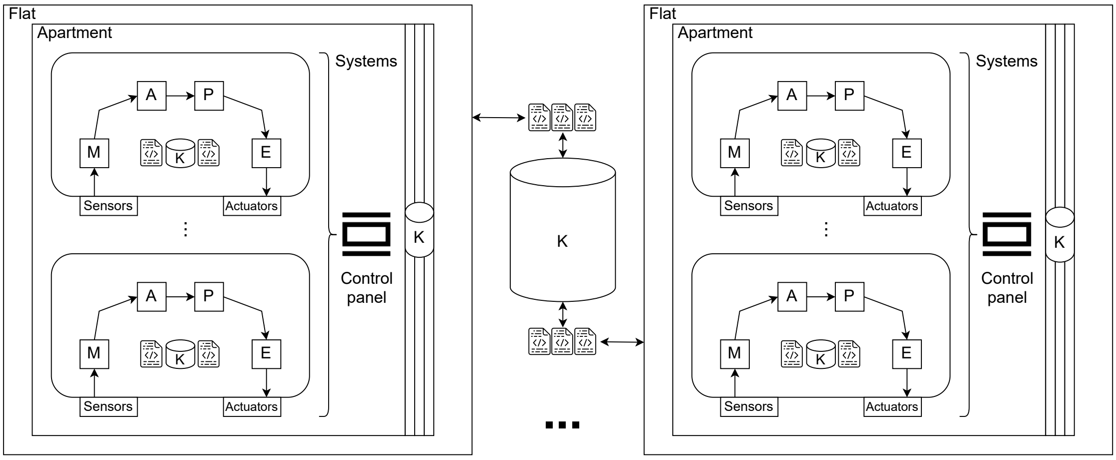

Figure 1: The semi-decentralized architecture applied to the considered use case

Systems [8, 9] and for IoT technologies [10, 11, 12], an information sharing structure complying to the Collect-Organize pattern is chosen, which has generally a lower energetic impact on the IoT sensors compared to other patterns [13]. As such, the knowledge base of the entire block of flats is fragmented into multiple layers, while the controllers are only present in the subsystems of each apartment and can benefit from the usage of models, may they be architectural or related to adaptation goals, coming from any of the knowledge bases. In particular, the knowledge layers in the proposed use case may be local to the controllers or contain, respectively, apartment-wise, flat-wise, or global information.

3. Runtime operations

Once deployed, and in a standard runtime situation, the architecture must aim at regulating the multiple energy flows (primarily electrical and thermal) in a way that minimizes their consumption and nonetheless assures a reasonably good quality of living for the apartment's occupants. To do so, each knowledge base in the architecture is provided with architectural models, from a holistic one for the global base to a local one for each component's base, adaptation goals models, and requirements models, all able to be read and written at runtime [14]. The adaptation goal models contain, at the various layers, rules and strategies for adaptation. In contrast, the requirements models define thresholds for parameters that are to be mandatorily respected, with the local layer's ones possibly set through each apartment's control panel.

In a standard scenario, each system of a single apartment is driven by its own controller, that tries to maintain the energy consumption optimal for the given requirements. For instance, the MAPE-K loop that manages the heating system might, in the absence of other constraints, choose a strategy for the smallest possible energy need from the heaters to keep a living temperature that is over the minimum threshold for that apartment's requirement model and, at the same time, reasonable in relation to the other decision factors, like the hour of the day, the external temperature or the presence or momentary absence of occupants. These factors may be collected by sensors pertaining to other systems, but they are available to all the controllers, at need, as the data collected by them are shared through the apartment-wise knowledge base.

The layered knowledge bases allow the self-adaptation process to take place, at need, at multiple levels. This is possible because the controllers get knowledge, constraints, requirements, and goals up to the global knowledge base and are consequently capable of reacting to substantial changes in them by including this information in their analysis and planning phases. Considering, for instance, the sudden increment in energy demand from one of the flats composing the neighborhood and knowing that, in the global knowledge base, it exists an indication of the desired maximum consumption per hour for the system as a whole, a temporary lower threshold of energy usage might be set in each other flat-wise constraint model. This, in turn, would lead every eligible controller, at the moment of fetching the new information into its loop, to consider the limitation and, possibly, adapt to the change in consideration of the local constraints and goals.

At the same time, the decomposition in layers of the available knowledge in contrast with an architecture possessing uniquely centralized knowledge allows the controllers to operate continuously even when some knowledge bases are not reachable due to, for instance, connectivity issues or temporary unavailability. In this scenario, the control loops cannot benefit from updated general information about the whole system but still have access to the latest data coming from the local sensors they are attached to, hence continuing the adaptation operations without interruption focusing momentarily on the local objectives and constraints only. On top of that, the layered structure combined with the runtime usage of architectural models is resilient to the failure, permanent removal, or new addition of one or more controllers, as the others are promptly informed of the architectural change in the system and can balance their contribution in keeping the expected goals.

However, it is still unknown, at the current stage, how severe might be the interferences on the system caused by defective sensors that return realistic, but imprecise, values. In fact, if not individuated and repaired swiftly, they might trigger a chain of adaptation cycles that unknowingly push the system far away from the expected goals, leading to instability. To face this eventuality, the optimal energy efficiency of the pattern might be sacrificed to introduce a further, centralized, control mechanism that is able to detect such anomalies and isolate them before they can influence the adaptation process.

Considering that the given use case describes a scenario that also involves the quality of life of the occupants of the given apartments, it might be unfeasible to leave to the adaptation mechanism an almost absolute control on the operations taking place in the block of flats. Until now, the application of the architecture gave the opportunity for the residents to act on the adaptation process only through their control panel, which could define tighter or more permissive constraints, but they were still subject to the global goals in a perhaps too strict manner. In the case of too-conservative settings, indeed, the performance along with the quality of service might drop significantly. For this reason, an alternative solution would be to display on each control panel the recommended settings to adopt in order to reach the adaptation goals as a simple suggestion, leaving the users full control of the effective settings. This would logically reduce the global constraints and goals to mere indicative values but might be the key to raising the occupants' awareness in terms of their energy consumption.

4. Conclusion

In this short paper, a semi-decentralized self-adaptive architecture for energy efficiency in IoT systems was presented through a use case in the domain of smart households. Following the Collect-Organize Pattern [8, 11], the architecture aims at optimizing the energy consumption of a block of flats while preserving optimal performance and quality of service by employing localized adaptive controllers with multiple layers of knowledge bases. A short discussion highlighted the possible strengths and weaknesses of the proposed approach and leaves space for future work in addressing the presented concerns and building a base for a future evaluation of the methodology.

References

[1] C. Brewster, I. Roussaki, N. Kalatzis, K. Doolin, K. Ellis, Iot in agriculture: Designing a europe-wide large-scale pilot, IEEE Communications Magazine 55 (2017) 26–33. doi:10.1109/MCOM.2017.1600528.

[2] C. Yang, W. Shen, X. Wang, Applications of internet of things in manufacturing, in: 2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2016, pp. 670–675. doi:10.1109/CSCWD.2016.7566069.

[3] J. Li, Y. Lin, Iot home automation – smart homes and internet of things, in: 2021 3rd International Academic Exchange Conference on Science and Technology Innovation (IAECST), 2021, pp. 294–298. doi:10.1109/IAECST54258.2021.9695788.

[4] A. A. Laghari, K. Wu, R. A. Laghari, M. Ali, A. A. Khan, A review and state of art of internet of things (iot), Archives of Computational Methods in Engineering 29 (2022) 1395–1413. doi:10.1007/s11831-021-09622-6.

[5] S. R. Bader, M. Maleshkova, S. Lohmann, Structuring reference architectures for the industrial internet of things, Future Internet 11 (2019). doi:10.3390/fi11070151.

[6] IBM, An architectural blueprint for autonomic computing. white paper, 2005. URL: https://www-03.ibm.com/autonomic/pdfs/AC%20Blueprint%20White%20Paper%20V7.pdf.

[7] Y. Brun, G. Di Marzo Serugendo, C. Gacek, H. Giese, H. Kienle, M. Litoiu, H. Müller, M. Pezzè, M. Shaw, Engineering Self-Adaptive Systems through Feedback Loops, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 48–70. doi:10.1007/978-3-642-02161-9_3.

[8] A. Musil, J. Musil, D. Weyns, T. Bures, H. Muccini, M. Sharaf, Patterns for Self-Adaptation in Cyber-Physical Systems, Springer International Publishing, Cham, 2017, pp. 331–368. doi:10.1007/978-3-319-56345-9_13.

[9] H. Muccini, M. Sharaf, D. Weyns, Self-adaptation for cyber-physical systems: A systematic literature review, in: Proceedings of the 11th International Symposium on Software Engineering for Adaptive and Self-Managing Systems, SEAMS '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 75–81. doi:10.1145/2897053.2897069.

[10] H. Muccini, R. Spalazzese, M. T. Moghaddam, M. Sharaf, Self-adaptive iot architectures: An emergency handling case study, in: Proceedings of the 12th European Conference on Software Architecture: Companion Proceedings, ECSA '18, Association for Computing Machinery, New York, NY, USA, 2018. doi:10.1145/3241403.3241424.

[11] J. Cámara, H. Muccini, K. Vaidhyanathan, Quantitative verification-aided machine learning: A tandem approach for architecting self-adaptive iot systems, in: 2020 IEEE International Conference on Software Architecture (ICSA), 2020, pp. 11–22. doi:10.1109/ICSA47634.2020.00010.

[12] I. Alfonso, K. Garcés, H. Castro, J. Cabot, Self-adaptive architectures in iot systems: a systematic literature review, Journal of Internet Services and Applications 12 (2021) 14. doi:10.1186/s13174-021-00145-8.

[13] H. Muccini, K. Vaidhyanathan, Leveraging machine learning techniques for architecting self-adaptive iot systems, in: 2020 IEEE International Conference on Smart Computing (SMARTCOMP), 2020, pp. 65–72. doi:10.1109/SMARTCOMP50058.2020.00029.

[14] T. Vogel, A. Seibel, H. Giese, The role of models and megamodels at runtime, in: J. Dingel, A. Solberg (Eds.), Models in Software Engineering, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 224–238. doi:10.1007/978-3-642-21210-9_22.