Towards a Predictable and Cognitive Edge-Cloud Architecture for Industrial Systems

Mohammad Ashjaei*, Saad Mubeen*, Masoud Daneshtalab *, Victor Casamayor†, Geoffrey Nelissen‡
*Mälardalen University, Sweden
†Technical University of Vienna, Austria
‡Eindhoven University of Technology, the Netherlands
*firstname.lastname@mdu.se, †v.casamayor@dsg.tuwien.ac.at, ‡g.r.r.j.p.nelissen@tue.nl

***Abstract*—In this paper, we present a conceptual proposal for a novel predictable and cognitive edge-cloud computing architecture for industrial cyber-physical systems. Timing predictability in this multi-layer architecture is envisioned to be supported by cognitive adaptation mechanisms in various computing layers of the edge-cloud computing continuum, including the communication among the layers. We also discuss our preliminary plan to realize the proposed architecture. Furthermore, we conceptualize the proposed architecture on a use case from the automation industry to show its applicability.**

I. Introduction

The edge and cloud computing technologies have already become an integral part of many industrial Cyber Physical Systems (CPS), from infrastructure monitoring, smart automation and construction equipment to telecommunication infrastructures [1], [2]. In such industrial systems, the environment is considered to be highly dynamic. A typical case would be that of construction quarries which are subject to frequent changes in their environment (due to weather, layout modification, etc.) and that accommodate battery-operated construction vehicles that regularly join and leave the site to, for example, fulfill variable charging requests. Beside their requirements for adaptation to varying operating conditions, most of these systems require timing predictable services in various computing layers, e.g., edge and cloud computing layers, as often industrial systems possess strict timing requirements, i.e., a hard service deadline should be met.

The computing continuum, from edge to cloud, that is available today lacks a holistic support for *cognitivity*, *adaptivity* and *timing predictability* in these industrial CPS. In this context, cognitivity refers to processes that monitor the environment, intelligently perceive the situation, and autonomously adapt the overall utilisation of resources, including computation and communication resources. A system is considered timing predictable if it is possible to prove or demonstrate that it meets all the specified timing requirements [3]–[5]. The primary objective of the overall adaptation is to obtain optimal resource utilisation and reduction of overall cost and energy.

To meet these requirements, this paper proposes a novel predictable and cognitive edge-cloud computing architecture for industrial CPS. Timing predictability is supported by cognitive adaptation mechanisms in the edge-cloud architectures. Therefore, we use the term predictable cognitive edge-cloud computing continuum throughout the paper to spotlight on the novel feature of the proposed architecture. That is, timing predictability of services, also known as timeliness of services, will be supported through the use of cognitive and adaptation mechanisms in various layers of the computing continuum. We will leverage AI-enabled technologies that allow the development of computing continuum, from edge to cloud [6]. Achieving such an architecture is of paramount importance to many Original Equipment Manufacturers (OEMs) in their path towards providing not only timing predictable services but also energy- and cost-effective systems that are cooperative and support zero downtime.

II. Related Work

The concurrent execution of an application through all computing continuum layers increases the complexity of the entire system. Thus, holistic approaches have been considered in the literature which tackle the design of the architecture from different aspects, e.g., for mobile models [7], ad-hoc computing establishment [8], and orchestration mechanisms for the edge-cloud computing systems [1]. The architecture proposed in this paper will also focus on utilising AI technologies to allow a seamless integration of adaptive mechanisms into the architecture, while maintaining timeliness of services in all computing layers. Therefore, the proposed architecture considers timeliness aspects, both in computation and communication, unlike the previously proposed architectures, that makes it suitable for time-critical industrial systems.

Considering solely the application orchestration, a comprehensive survey presents various proposals [1]. There are orchestration solutions for cloud computing, which are also evaluated on edge computing, encompassing the edge-cloud computing continuum [9]. One of the key elements of orchestration is the scheduling of services, where few proposals have shown some development in this direction [10]. Predictability in cloud computing has been a focus of several works [11], [12], mainly targeting timeliness for applications in the cloud. A few recent works further extend the predictability for the computing continuum [13].

Focusing on the network technologies in the edge and fog computing, there are very few works that address both timeliness and adaptivity of the communication services. For instance, a self-configuring time-sensitive network (TSN) is proposed for fog and edge computing in the automation domain [14] and for enabling fog computing to use TSN technologies [15]. In this paper, we will focus on developing communication infrastructure for the edge-cloud computing architecture with the goal of enabling them for utilising predictable communication technologies, such as TSN and wireless TSN, and at the same time providing dynamicity in the configuration of the network.

There are several large EU projects that have initiated the development of edge-cloud computing continuum. For example, the AI@Edge[1] project aims at developing reusable, secure and trustworthy AI solutions for the network edge. The SERRANO[2] project aims at developing an abstraction layer for automated and cognitive orchestration from edge to cloud computing. Development of a set of tools are the main goal in the DITAS[3] project, while Fog-protect[4] targets data protection through the computing continuum. The SESAME[5] project aims at combining solutions of network virtualization with edge computing to develop multi-service 5G small cells to obtain low-latency communication.

To the best of our knowledge, none of the existing architectures particularly present how timeliness and predictability for services can be achieved in various computing layers. Timeliness is a paramount requirement that is imposed by industrial systems as they have many timing requirements to fulfill. Timeliness of services is often left as a secondary objective, while providing computing resources is commonly the primary goal. Therefore, we aim at building an edge-cloud computing continuum that essentially supports predictability in all computing layers including communication among the computing layers. We exploit the concept of cognitivity, including monitoring and adaptation mechanisms, to enable support for predictability of services.

III. Envisioned Architecture

The predictable and cognitive edge-cloud architecture envisioned in this paper is depicted in Fig. 1. In this architecture, the edge-cloud computing continuum consists of several layers of interconnected computing resources. In general, the edge nodes provide services to end-systems or devices, while a cluster of fog nodes aggregate several edge nodes providing enhanced computing capabilities and connectivity. The cloud computing layer, which can consist of several layers itself including enterprise (private) and public clouds, has a vast quantity of computing resources and can steer large computations with massive storage capabilities. The edge node provides services with strict timing requirements, while less time-critical services are deployed to fog or cloud layers. With such an architecture, enterprises can take advantage of multi-layer computing systems for their applications with different requirements on timing.

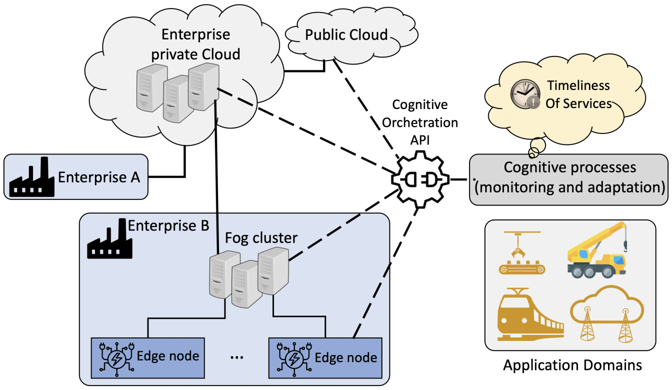

Fig. 1. Envisioned predictable & cognitive edge-cloud computing architecture.

We aim at leveraging cognitive methodologies in order to support predictable services within the presented multi-layer architecture. This computing architecture can provide services to the end nodes from different computing layers with different requirements. In this work we focus on timing requirements, however, we envision an architecture able to deal with other type of requirements, such as reliability or security requirements. For instance, a service for high time- and safety-critical requirements may only be provided by the fog cluster, while less-critical services can be deployed on the private or even public cloud. The novelty of this architecture is to provide timeliness in all layers, including the private and public cloud, with different time-criticality levels.

In order to provide such timeliness, during run-time of the system without any service disruption, an entity is envisioned to perform cognitive processes. This entity can reside in different layers adhere to either a centralized, a distributed or a hybrid model. The cognitive orchestration Application Programming Interface (API) provides an interface between the cognitive processes. Moreover, AI and Machine Learning (ML) techniques will be utilized in the cognitive processes to ensure intelligent, autonomous and time-predictable changes during the run-time of the system. The cognitive processes are categorized into three phases: monitoring the environment, intelligent perception of the monitored environment, and on-the-fly adaptation of the computing continuum.

We aim at achieving the following objectives to realize the envisioned architecture:

- to develop a multi-layer edge-cloud computing architecture with an intelligent cognitive capacity to adapt the system autonomously during run-time, while maintaining the timing requirements of the services imposed by industrial systems;
- to adopt suitable AI- and ML-based techniques to perform run-time monitoring, intelligent perception, and adaptation of the multi-layer edge-cloud computing architecture;
- to develop techniques to verify the timing predictability of the system, in terms of computation and communication, during offline and run-time of the system; and
- to demonstrate the proposed architecture together with cognitive processes on realistic or industrial use cases.

[1] https://cordis.europa.eu/project/id/101015922
[2] https://ict-serrano.eu/
[3] https://www.ditas-project.eu/
[4] https://fogprotect.eu/
[5] https://cordis.europa.eu/project/id/671596/results

IV. Preliminary Plan to Realize the Architecture

The envisioned architecture will be realized as a hierarchical model consisting of several computing layers from edge nodes to a cluster of fog nodes, until the cloud (including enterprise private and public clouds).

A. Predictable run-time environment and communication

The novel essence of this architecture is its support for timing predictability in all layers. Hence, predictable run-time environments, such as real-time operating systems and real-time orchestration systems, will be utilized. In addition, timing predictability will be considered in communication among several computing layers that should also handle high-bandwidth and low-latency communication over wired and wireless networks. For instance, TSN will be considered for wired communication within and between computing layers [16]. Utilizing 5G, as a wireless technology with low-latency and high-bandwidth support, will be considered where wired connection cannot be established. In addition, converged TSN and 5G communication will also be considered [17].

B. Cognitive Orchestration API

We will investigate various solutions to develop an entity to provide capacity for cognition, for instance, a centralized, a distributed or a hybrid model. A centralized model has the advantage of requiring less synchronization, while a distributed model scales better for large systems. In general, edge-cloud computing systems are complex and large, hence a hybrid model might seem a priori a better solution. In any case, these trade-offs will be investigated to select a suitable model to deploy the cognitive entity.

In brief, the cognitive entity will follow the execution and communication of the distributed application auditing its timeliness and proposing adaptive measures. Then, the communication interface between the edge-cloud system and the cognitive entity will be realized by the cognitive orchestration API. The intention is not to implement an entirely new orchestration API, instead to adopt and enhance the existing solutions to provide such an interface, e.g., Kubernetes (like KubeEdge[6]).

Once the cognitive orchestration API is designed, it will be furnished with a set of ML techniques to provide its cognitive behavior.

C. Cognitive Capacity: Monitoring and Dynamic Adaptation

Cognition requires three phases: monitoring the edge-cloud system and its environment; intelligently perceiving the situation; and autonomously adapting the configuration accordingly.

A set of techniques will be used to efficiently monitor the computation and communication utilization. Different parameters will be considered when such monitoring techniques are designed, e.g., periodicity of monitoring, metrics to monitor, and update rates. Open-source tools, such as Prometheus[7], can be adopted to monitor lower-level metrics of the system.

Half-way between monitoring and intelligent perception can be found techniques of predicting monitoring, which by means of ML/AI techniques are able to predict future outcomes of the monitoring module to enhance the system perception. Then, intelligently perceiving the situation is achieved by inferring the current and future edge-cloud system states with respect to its requirements, anticipating possible Service Level Objectives (SLOs) violations. Finally, the cognitive entity selects the most appropriate adaptive mechanism. To do so, a set of techniques based on ML/AI techniques will be adopted to autonomously adapt the edge-cloud system, in terms of computing or communication capacities. For example, a monitoring technique can regularly inspect the network utilisation within fog nodes and an adaptation technique can adjust the routing of traffic based on the gathered (or historical) data. It can also predict potential congestion and recover proactively. Similarly, high-level metrics for software in a computing platform can be monitored, such as the efficiency of the resources used, which can lead to adaptation techniques based on AI that can automatically optimise their deployment.

D. Support for Timing Predictability

We assume that many control applications that will be deployed on edge, fog or cloud require to meet timing requirements such as deadlines, age and reaction constraints [18]. Similarly, delivering information from a remote computing node to an end station should follow different timing constraints. The timing requirements of systems are usually verified during the design phase of the systems and any changes during the system will negate the verification of the timing. In order to continuously support timing predictability of the services in the proposed architecture, online timing verification mechanisms are required. Any changes that the adaptation mechanism proposes as per the cognitive processes should be verified with respect to timing requirements before their deployment. Hence, we will develop analytical and formal techniques to verify timing predictability of services before deployment of proposed changes by the adaptation mechanisms. Such an online verification will be done on both communication and computation to ensure guaranteed timing for the whole system. For instance, pseudo-polynomial response-time analysis techniques for distributed embedded systems and end-to-end resource analysis techniques [19], [20] can be used to verify the timeliness properties of distributed systems with low time-complexity.

V. Conceptualization on an Industrial Use Case

The envisioned architecture can be applied to any application domain with strict and non-strict timing requirements that employs edge-cloud computing for adaptive and predictable systems, including construction vehicles, railway, telecommunication, and many more. In this section, we conceptualize

[6] https://kubeedge.io/en/

[7] https://prometheus.io/docs/introduction/overview/

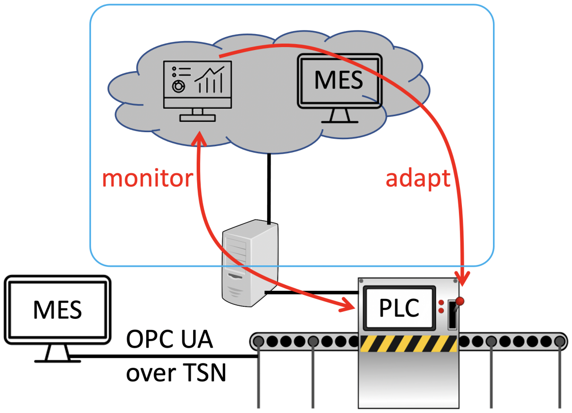

Fig. 2. Conceptualization of the envisioned framework on an industrial automation use case.

the architecture on a use case from the automation industry to demonstrate its applicability and usability.

The use case comprises an automation assembly line in which a set of machinery collaboratively produces cell phones. The machinery, including punching machine, cutting machine, assembly machine, etc, are connected via a wired network based on TSN switched Ethernet running Open Platform Communication - Unified Architecture (OPC UA)[8] in the application layer. The Manufacturing Execution System (MES) on the edge is utilized to deploy the orders and make changes in the assembly lines. However, the MES works statically and any changes in the order, communication between the machines, and processes, should be done offline. This can potentially cause some delays and disruption in the production. Therefore, the main idea of this use case is to further develop the automation system to offload some of the intelligent controls to the fog or enterprise cloud, while the changes in the processes and communication among the machinery become automatic and online. This will potentially help the production system to be dynamic in the sense that when different products are ordered the changes in the production system becomes quick and automatic.

We aim to showcase the envisioned architecture and accompanying techniques on this use case, including the edge-cloud computing architecture, cognitive processes to make the system dynamic and automatic, and respect the timing predictability requirements that are imposed by the processes in the assembly line. Within this use case, we plan to use a private in-house cloud to build such an edge-cloud continuum and deploy the MES on both edge and cloud to work collaboratively. Fig. 2 shows the conceptualization of the proposed framework on an existing industrial automation use case.

VI. Conclusion and Future Work

In this paper, we proposed a novel predictable and cognitive edge-cloud computing architecture for industrial CPS. Furthermore, we presented our plan to realize this architecture and develop accompanying techniques. We also provided a conceptualization of the proposed architecture on an industrial automation use case to show its applicability in practice. We believe, the proposed architecture would be beneficial for OEMs in their path towards providing timing predictable and energy- and cost-effective systems that are cooperative and support zero downtime. In the future, we plan to execute the presented plan to realize the proposed architecture.

[8] https://opcfoundation.org/

References

[1] B. Costa, J. Bachiega, L. R. de Carvalho, and A. P. F. Araujo, "Orchestration in fog computing: A comprehensive survey," *ACM Comput. Surv.*, vol. 55, no. 2, 2022.

[2] R. Chaâri, F. Ellouze, A. Koubâa, B. Qureshi, N. Pereira, H. Youssef, and E. Tovar, "Cyber-physical systems clouds: A survey," *Computer Networks*, vol. 108, pp. 260–278, 2016.

[3] J. A. Stankovic and K. Ramamritham, "What is predictability for real-time systems?" *Real-Time Sys.*, vol. 2, no. 4, pp. 247–254, Nov 1990.

[4] D. Grund, J. Reineke, and R. Wilhelm, "A Template for Predictability Definitions with Supporting Evidence," in *Bringing Theory to Practice: Predictability and Performance in Embedded Systems*, ser. OpenAccess Series in Informatics, vol. 18, Dagstuhl, Germany, 2011, pp. 22–31.

[5] S. Mubeen, E. Lisova, and A. Vulgarakis Feljan, "Timing predictability and security in safety-critical industrial cyber-physical systems: A position paper," *Applied Sciences*, vol. 10, no. 9, 2020.

[6] P. Beckman, J. Dongarra, N. Ferrier, G. Fox, T. Moore, D. Reed, and M. Beck, *Harnessing the Computing Continuum for Programming Our World*, 2020, pp. 215–230.

[7] L. Baresi, D. F. Mendonça, M. Garriga, S. Guinea, and G. Quattrocchi, "A unified model for the mobile-edge-cloud continuum," *ACM Trans. Internet Technol.*, vol. 19, no. 2, 2019.

[8] A. M. J. Ferrer, S. Becker, F. Schmidt, L. Thamsen, and O. Kao, "Towards a cognitive compute continuum: An architecture for ad-hoc self-managed swarms," *2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, pp. 634–641, 2021.

[9] A. Ullah, H. Dagdeviren, R. Ariyattu, J. DesLauriers, T. Kiss, and J. Bowden, "Micado-edge: Towards an application-level orchestrator for the cloud-to-edge computing continuum," *Journal of Grid Computing*, vol. 19, 12 2021.

[10] G. P. Mattia and R. Beraldi, "Leveraging reinforcement learning for online scheduling of real-time tasks in the edge/fog-to-cloud computing continuum," in *2021 IEEE 20th International Symposium on Network Computing and Applications (NCA)*, 2021.

[11] Y. Gan, Y. Zhang, K. Hu, D. Cheng, Y. He, M. Pancholi, and C. Delimitrou, "Leveraging deep learning to improve performance predictability in cloud microservices with seer," *SIGOPS Oper. Syst. Rev.*, vol. 53, no. 1, p. 34–39, 2019.

[12] T. Nylander, M. Thelander Andrén, K.-E. Årzén, and M. Maggio, "Cloud application predictability through integrated load-balancing and service time control," in *2018 IEEE International Conference on Autonomic Computing (ICAC)*, 2018, pp. 51–60.

[13] M. Chardet, H. Coullon, and C. Perez, "Predictable efficiency for reconfiguration of service-oriented systems with concerto," in *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*, 2020, pp. 340–349.

[14] M. Gutiérrez, A. Ademaj, W. Steiner, R. Dobrin, and S. Punnekkat, "Self-configuration of ieee 802.1 tsn networks," in *IEEE International Conference on Emerging Technologies and Factory Automation*, 2017.

[15] P. Pop, M. L. Raagaard, M. Gutierrez, and W. Steiner, "Enabling Fog Computing for Industrial Automation Through Time-Sensitive Networking (TSN)," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 55–61, 2018.

[16] M. Ashjaei, L. Lo Bello, M. Daneshtalab, G. Patti, S. Saponara, and S. Mubeen, "Time-sensitive networking in automotive embedded systems: State of the art and research opportunities," *Journal of Systems Architecture*, vol. 117, p. 102137, 2021.

[17] Z. Satka, D. Pantzar, A. Magnusson, M. Ashjaei, H. Fotouhi, M. Sjödin, M. Daneshtalab, and S. Mubeen, "Developing a Translation Technique for Converged TSN-5G Communication," in *18th IEEE International Conference on Factory Communication Systems*, 2022.

[18] S. Mubeen, T. Nolte, M. Sjödin, J. Lundbäck, and K.-L. Lundbäck, "Supporting timing analysis of vehicular embedded systems through the refinement of timing constraints," *Software & Systems Modeling*, vol. 18, pp. 39–69, 2019.

[19] S. Mubeen, J. Mäki-Turja, and M. Sjödin, "Support for end-to-end response-time and delay analysis in the industrial tool suite: Issues, experiences and a case study," *Computer Science and Information Systems*, vol. 10, no. 1, 2013.

[20] M. Becker, D. Dasari, S. Mubeen, M. Behnam, and T. Nolte, "End-to-end timing analysis of cause-effect chains in automotive embedded systems," *Journal of Systems Architecture*, vol. 80, pp. 104 – 113, 2017.