Edge Computing in Autonomous and Collaborative Assembly Lines

Dominik Urbaniak Jan Rosell Raúl Suárez

Institute of Industrial and Control Engineering
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain
dominik.urbaniak@upc.edu, jan.rosell@upc.edu, raul.suarez@upc.edu

***Abstract*—Industry 4.0 demands interconnected production lines that consist of modular assets. Recent advances of wireless communication technologies allow a large connectivity of devices and approach the performance of wireline communication, specifically regarding throughput, latency and reliability. As a result, more and more time critical connections can be performed wirelessly. Both attributes foster the emergence of edge computing, a concept that can efficiently utilize distributed computational resources. This is particularly beneficial for modular assets that have limited energy supply and capacity of computation hardware. Autonomous mobile robots offer high potential for object transportation, inspection and manipulation in shared workspaces with human operators. With edge computing, heavy computations can then be offloaded to more powerful computers or edge data centers to speed up the decision-making process and increase the productivity. For an efficient orchestration strategy of computation and communication resources, various task requirements in terms of latency, bandwidth, cost and energy must be considered. To this end, we aim at evaluating the requirements in autonomous and collaborative assembly lines, a use case that comprises diverse tasks including latency-sensitive ones in dynamic, uncertain, multi-agent environments. This work focuses on discussing latency requirements on the basis of a collaborative safety mode and autonomous robotic insertion.**

***Index Terms*—edge computing, collaborative robotics, 5G, Wi-Fi 6, Industry 4.0, automation, manipulation planning**

I. Introduction

Conventional industrial robots are most suitable when performing repetitive tasks of high volumes in a workspace separated from human workers by safety barriers. However, customized products and short product life cycles [1] demand more adaptability of the manufacturing system to avoid an increase of costly downtimes for reprogramming the robots and reconfiguring the production line [2]. Replacing wired connections by wireless communication networks represents a major opportunity to increase flexibility. Figure 1 illustrates a collaborative assembly line where mobile robot, cameras and edge server communicate wirelessly via a router. This concept of edge computing is enabled by developments in wireless technologies that keep narrowing the performance gap compared to wireline communication. The *International Telecommunications Union* (ITU-R) defined three categories of communication services for 5G cellular networks to address the requirements for Industry 4.0: ultra-reliable low-latency communication (URLLC), enhanced mobile broadband (eMBB) and massive machine-type communication (mMTC) [3]. Similarly, the *Wi-Fi Alliance* is working on increasing throughput and decreasing latency, also for "mission critical" [4] operations. This evolution is particularly important for autonomous mobile robots (AMRs). The market for AMRs is expected to increase from \$1.61 billion in 2021 to \$22.15 billion in 2030 [5] which could be explained by their flexible application in transport, inspection and manipulation tasks. However, these operations require the quick and proper reaction to uncertain and dynamic changes which pose significant challenges to current autonomous systems. Approaches that utilize deep learning promise to improve their "adaptability and resilience" [1] in such complex settings. On the other hand, acting reliably in uncertain and dynamic environments is a major strength of humans. To this end, collaborative robotics aims at combining the human adaptability and intuition with the robot repeatability and precision to increase delivery rate and product quality [6].

This work analyzes promising applications for edge computing in collaborative assembly lines requiring powerful wireless communication. Recent advances in manipulation planning and human tracking utilize deep learning approaches

This work was funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska - Curie grant agreement No 956670.

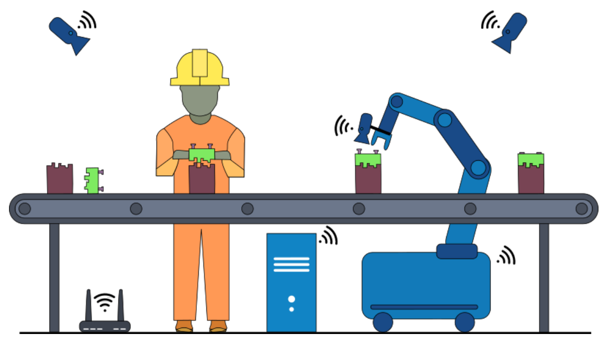

Fig. 1. Edge computing in a collaborative assembly line: External cameras observe the collaborative workspace. Expensive computations, e.g. for tracking the operator and performing manipulation planning, is performed on the edge server via wireless communication networks.

to handle the uncertainty of contact-rich interactions and variations in human appearance and poses [7]–[10]. However, they do not consider a limited availability of resources. For AMRs, however, high computational resources are costly and energy-demanding. Hence, an edge computing solution could present a desirable alternative. To this end, section II discusses related work that utilize edge computing for robotic applications. Section III compares this concept to the alternatives, cloud and device computing. Section IV analyzes the resource requirements for a collaborative safety mode and robotic assembly tasks on the basis of state-of-the-art work arguing that these applications demand edge computing in combination with 5G or Wi-Fi 6. Further progress of this work (Sec. V) will assess the theoretical expectations by experimentally evaluating the performance variations to enable the most suitable allocation of operations for edge, cloud and device computing, and orchestration of cellular and WLAN network resources.

II. Related work

Robotics is mentioned as an application area for powerful communication technologies like 5G [11], [12]. However, there is a lack of real-world experiments [13]. The targets of communication development to reduce latency, increase connectivity and throughput aligns well to the concept of edge computing. Voigtländer et al. [12] perform closed-loop control for a balancing task using an eight-DoF mobile robot. The balance controller and the inverse kinematics are offloaded to an external computer using a 5G prototype. In a subsequent work [14], the authors explore the capabilities of 5G communication in an industrial demonstrator platform, specifically considering safety requirements and heterogeneous data transmissions of a mobile robot for transportation and inspection tasks. The *3rd Generation Partnership Project* (3GPP), that developed a 5G standard, also outlines a number of use cases [11]. The authors of [14] refer to the use case "mobile control panels with safety functions". Raunholt et al. [15] show that the navigation planning and docking control of an AMR can be reliably performed using 5G-based edge computing. Even low-level control can be performed on an edge server in a navigation task using 5G [16] and a custom-build AMR. The authors justify the offloading of such small computations with the "ease of maintenance" and "improved resiliency to software and hardware failures".

In contrast to the related work, we are particularly interested in comparing the performance with different communication technologies and computational resources. To this end, we aim at exploiting the recent advancements in deep learning for high-DoF, high-frequency and closed-loop robotic manipulation and collaborative robotics which regards the 3GPP use case of a "flexible, modular assembly area" [11].

III. Comparison of computing solutions

Cloud, device and edge computing mainly differ in terms of the available size of computational resources and their distance to the location where the data is generated. One computing solution could be more suitable than others depending on the use case requirements, especially latency, bandwidth, cost and energy [17]. In this work, we concentrate on the latency requirements which accumulate from communication and computations. Figure 2 illustrates a schematic overview of the considered computing solutions. The bottom of the figure represents the edge devices that have the least computational power, and therefore, are expected to require the largest computation times. Such devices have the largest deployment scale [18] which makes the communication paths more likely to be short resulting in smaller communication latencies.

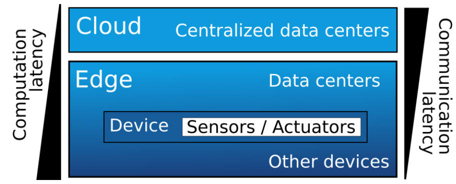

Fig. 2. Schematic expectations of computation and communication latency using cloud and edge computing. In comparison, the device that senses or acts can have fixed limited resources and has the lowest communication latency.

a) Cloud computing: Cloud computing offers "on-demand access to a shared pool of computing resources" [18] located at large data centers. This solution provides the most computational resources and the easiest scalability. However, large distances to the data centers and passing several networks increase the communication delay [19].

b) Device computing: We refer to device computing as the solution that utilizes the limited computational resources on-board the edge device, e.g. the AMR or the cameras, without requiring any communication with other devices or external computing sources.

c) Edge computing: Edge computing utilizes distributed computational resources between centralized data centers and the devices to improve "the performance, operating cost and reliability of applications and services" [18]. The computational resources vary in size and can be provided by other edge devices or edge data centers that are usually located in "close proximity to the last mile network" [18] and therefore, reduce "the latency and bandwidth constraints of today's internet" [18]. In combination with powerful wireless communication like 5G or Wi-Fi 6 that enables data transmission in a few milliseconds [15], [20], edge computing can provide resource-constrained mobile devices the computational performance of small data centers at the cost of minor communication latency.

IV. Towards edge-native applications

Small computations, such as low-level control of robots, might be most efficiently performed on the device, when the time gain from more computational resources does not compensate the communication latency. On the other hand, for tasks that require expensive computations, such as sampling-based motion planning, the increased communication latency with cloud computing might not be significant. Both of these examples can be explored in a collaborative assembly line, however, this use case is characterized by a highly uncertain and dynamic environment due to contact-rich interactions and human presence which demand proper reactions at low latency. Edge-native applications are built to “leverage edge computing capabilities” [18] such as resource and latency constraints [18] and to dynamically allocate “application logic to other edge locations depending on environmental conditions” [21]. In the following, the need and potential of edge computing solutions utilizing powerful wireless communication is illustrated on the basis of collaborative robotics and autonomous robotic assembly.

A. Speed and separation monitoring in collaborative robotics

Human safety in a collaborative environment is of utmost importance. The ISO/TS 15066 defines four safety modes for collaborative robotics [22]. Currently, the power and force limiting (PFL) mode is mostly considered for autonomous robots in collaborative environments [23]. However, in this mode the robot speed is limited. To allow faster motion, the speed and separation monitoring (SSM) mode is required which is suitable for parallel assembly lines [6], as illustrated in Fig. 1. To ensure human safety, a “protective separation distance” S between a robot and a human operator is continuously calculated. Assuming constant speed for the robot v_R and the human v_H towards each other, the equation can be simplified as

$$S = T_R(v_H + v_R) + T_S v_H + B + U, \tag{1}$$

where T_R is the reaction time of the system, T_S and B the stopping time and distance, respectively. The term U accounts for the system’s measurement uncertainty. The reader is referred to [23] for more details about the implementation of the SSM mode. The reaction time T_R consists of the communication and computation latency of the system and can have significant effect on the usability of the SSM mode in a collaborative setting. For instance, in [23], the authors assume maximum velocities $v_H = 2$ m/s, $v_R = 1.6$ m/s and measure $T_R = 0.113$ s. Hence, 0.4 m of the required separation distance is attributed to the latency alone. The authors in [24] use $v_H = 1.6$ m/s and $T_R = 0.21$ to reduce the robot speed from a maximum of $v_R = 5$ m/s while approaching the robot linearly with a human dummy. In this case, the reaction time contributes 1.6 m of the separation distance at full speed. This work utilizes laser scanner to track the human position in 2D. More sophisticated pose tracking can be performed with RGB-D cameras, e.g. with the Azure Kinect Body Tracking SDK, which can run at 15 Hz [10]. In the context of edge computing, we will consider cameras that also provide computational resources which could allow data preprocessing to reduce the transmission rate. In any case, observing collaborative workspaces will require prioritized access to significant computational and communication resources. With edge computing, the availability of these resources can be ensured by migrating less critical operations to less powerful edge hardware or to centralized clouds.

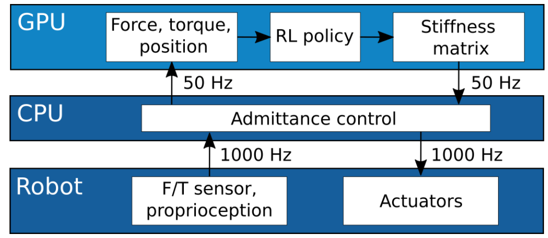

Fig. 3. System configuration in [7] that performs low-level control on a CPU and high-level control on a GPU for a robotic insertion task. In an edge computing scenario, the high-level control could be offloaded to the edge cloud.

B. Robotic insertion based on visual and haptic feedback

Insertion represents the most prevalent operation in an assembly task [2]. Small control offsets and inaccuracies can lead to collisions and jamming that potentially harms the robot and its environment. Force and torque (F/T) sensors can be used to avoid harmful motion. Additionally, they enable informed decision-making during contact. Oikawa et al. [7] propose such a solution for two high-precision insertions by combining reinforcement learning (RL) and engineered stiffness matrices for admittance control. They design a set of stiffness matrices each perturbing a linear trajectory in a different direction. A RL policy is then trained to select one matrix based on the F/T input and the current tip position. The use of haptic information requires an initial overlap of the peg and the hole. The F/T sensor returns a small six-dimensional vector that small neural networks can handle which is beneficial for training and inference times. The selection process is performed at 50 Hz on a NVIDIA GTX 1080 GPU, the admittance control runs on a Intel i7 CPU and updates the robot torques at 1000 Hz. This control loop is depicted in Fig. 3. The authors do not explain their choice and allocation of computational resources, hence, it is unknown if this is the most efficient setup. With edge computing, the resources can be allocated flexibly depending on the requirement of each operation within a task.

More resources are required for tasks that utilize more sensor data. Visual input can complement the haptic feedback particularly when there is no contact or overlap between the matching parts [9]. Lee et al. [8] utilize a fixed RGB-D camera (with image size of $128 \times 128 \times 4$), a wrist-mounted F/T sensor and proprioceptive data to generate a multi-modal input representation. A RL agent is trained with that representation to output 4D Cartesian displacements of the end-effector at 20 Hz which are subsequently interpolated to generate joint torques at 500 Hz. Vecerik et al. [25] take visual feedback from a wrist-mounted RGB camera (with image size of $128 \times 128 \times 3$), and joint positions, velocities and torques to run their RL policy at 5 Hz to generate 7D joint velocities which are “passed through an admittance layer” at 100 Hz. Due to the large size of images, more computational effort is required for data processing. At the same time, the communication latency may increase with the larger packet size. Both applications, the collaborative SSM mode and the robotic assembly, appear to benefit from minimal latency. However, to allow efficient resource planning and allocation, a more detailed analysis is required that compares the experimental performance of varying settings.

V. Ongoing work

We aim at implementing edge-native applications that utilize most recent wireless technologies to increase the robot productivity and human safety in autonomous and collaborative assembly lines. At the same time, we want to compare the performance of robotic manipulation tasks and workspace observation systems given varying computation and communication resources to enable cost-efficient resource orchestration. Since the applications consist of multiple operations with varying requirements we expect to find the most suitable configuration and best alternatives to define general criteria for prioritizing and allocating available resources dynamically. In a first step, we consider a visual servoing task where the robot manipulator follows an unknown trajectory based on visual input. Comparing the computation times of an object detector, image processing and inverse kinematics on a CPU versus a GPU, and introducing communication delays of diverse communication technologies based on [20] the precision of the trajectory can be assessed and recommendations for real-world requirements inferred. In the next steps, we consider reproducing insertion tasks based on haptic and visual feedback to investigate the impact of the reaction time on the insertion accuracy. To cope with unavoidable latency, the reaction time could be "negated" [23] by anticipating future motion.

VI. Conclusion

This work in progress presents a promising robotics use case that has the potential to fully leverage the benefits of edge computing and 5G or Wi-Fi 6 communication. Elaborate algorithms for human tracking and contact-rich interaction demand large computational resources that are limited on mobile assets such as AMRs. However, the lowest possible latency is desired to maximize productivity, e.g. by decreasing the minimal distance between humans and robots and increasing the robot motion speed. To this end, edge computing allows an efficient allocation of the required computational resources in close proximity to the edge device, and the required bandwidth of 5G or Wi-Fi 6 networks to wirelessly transmit the data with minimal communication latency.

Bibliography

[1] Q. Chen, B. Heydari, and M. Moghaddam, "Leveraging task modularity in reinforcement learning for adaptable Industry 4.0 automation," *Journal of Mechanical Design*, vol. 143, no. 7, p. 071701, 2021.

[2] K. Kimble *et al.*, "Benchmarking protocols for evaluating small parts robotic assembly systems," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 883–889, 2020.

[3] International Telecommunications Union Radiocommunication Sector (ITU-R), "IMT vision — framework and overall objectives of the future development of IMT for 2020 and beyond," Sep 2015, recommendation ITU-R M.2083-0. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.2083-0-201509-I!!PDF-E.pdf

[4] Wi-Fi Alliance, "Current work areas - Wi-Fi 7," 2022. [Online]. Available: https://www.wi-fi.org/who-we-are/current-work-areas

[5] Next Move Strategy Consulting (NMSC), "Autonomous mobile robot (amr) market by type, by application, by end-user - global opportunity analysis and industry forecast 2022-2030," Apr 2022. [Online]. Available: https://www.nextmsc.com/report/autonomous-mobile-robot-amr-market

[6] F. Vicentini, "Collaborative robotics: a survey," *Journal of Mechanical Design*, vol. 143, no. 4, p. 040802, 2021.

[7] M. Oikawa, T. Kusakabe, K. Kutsuzawa, S. Sakaino, and T. Tsuji, "Reinforcement learning for robotic assembly using non-diagonal stiffness matrix," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2737–2744, 2021.

[8] M. A. Lee *et al.*, "Making sense of vision and touch: Learning multimodal representations for contact-rich tasks," *IEEE Transactions on Robotics*, vol. 36, no. 3, pp. 582–596, 2020.

[9] O. Spector and D. Di Castro, "Insertionnet-a scalable solution for insertion," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5509–5516, 2021.

[10] W. Yang, C. Paxton, A. Mousavian, Y.-W. Chao, M. Cakmak, and D. Fox, "Reactive human-to-robot handovers of arbitrary objects," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 3118–3124.

[11] 3rd Generation Partnership Project (3GPP), "Study on communication for automation in vertical domains," 2020, 3GPP TR 22.804, Release 16 (V16.3.0). [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3187

[12] F. Voigtländer, A. Ramadan, J. Eichinger, C. Lenz, D. Pensky, and A. Knoll, "5G for robotics: Ultra-low latency control of distributed robotic systems," in *2017 International Symposium on Computer Science and Intelligent Controls (ISCSIC)*. IEEE, 2017, pp. 69–72.

[13] P. Varga *et al.*, "5G support for industrial iot applications—challenges, solutions, and research gaps," *Sensors*, vol. 20, no. 3, p. 828, 2020.

[14] F. Voigtländer *et al.*, "5G for the factory of the future: Wireless communication in an industrial environment," *arXiv preprint arXiv:1904.01476*, 2019.

[15] T. Raunholt, I. Rodriguez, P. Mogensen, and M. Larsen, "Towards a 5G mobile edge cloud planner for autonomous mobile robots," in *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*. IEEE, 2021, pp. 01–05.

[16] M. Balogh *et al.*, "Cloud-controlled autonomous mobile robot platform," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2021, pp. 1–6.

[17] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[18] P. Marshall, J. Davis, P. Shih, and A. Marcham, "State of the edge report 2021 - a market and ecosystem report for edge computing," 2021.

[19] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[20] I. Rodriguez *et al.*, "5G swarm production: Advanced industrial manufacturing concepts enabled by wireless automation," *IEEE Communications Magazine*, vol. 59, no. 1, pp. 48–54, 2021.

[21] D. Sprinzen, "Edge-native applications: What are they and where are they used?" Sep 2021. [Online]. Available: https://www.edgeir.com/edge-native-applications-what-are-they-and-where-are-they-used-20210912

[22] ISO/TS 15066:2016, "Robots and robotic devices - collaborative robots," 2016, International Organization for Standardization.

[23] J. A. Marvel and R. Norcross, "Implementing speed and separation monitoring in collaborative robot workcells," *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 144–155, 2017.

[24] C. Byner, B. Matthias, and H. Ding, "Dynamic speed and separation monitoring for collaborative robot applications–concepts and performance," *Robotics and Computer-Integrated Manufacturing*, vol. 58, pp. 239–252, 2019.

[25] M. Vecerik, O. Sushkov, D. Barker, T. Rothörl, T. Hester, and J. Scholz, "A practical approach to insertion with variable socket position using deep reinforcement learning," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 754–760.