A Vendor-Independent Distributed Supervisory Control Architecture for Vision-Based and SME-Oriented Robotic Sorting

Davide Galli[1], Chiara Nezzi[1], Dylan Morelato[1], Matteo De Marchi[1]

***Abstract*— Modern manufacturing systems increasingly demand flexible and interoperable automation solutions capable of integrating heterogeneous and legacy devices. This requirement is particularly relevant for small and medium-sized businesses and research environments, where vendor lock-in, system rigidity, and high demand for integration resources often hinder the adoption of advanced automation. This paper presents a vendor-independent distributed control architecture for a vision-based robotic sorting cell, designed to support modularity, adaptability, and eventual device replacement. The proposed approach models the robotic cell as a cyber-physical system composed of loosely coupled control nodes coordinated through a supervisory control layer. An event-driven asynchronous communication strategy enables interoperability among heterogeneous sensing and actuation subsystems without relying on proprietary automation platforms. The architecture has been instantiated on a laboratory-scale robotic cell integrating legacy robots, feeding systems, and multi-vendor vision devices. Late-breaking results highlight how the proposed control strategy supports flexible system evolution and simplifies integration, while trading strict quasi real-time guarantees for adaptability and scalability. The presented work contributes an architectural and control-oriented perspective on distributed automation systems, providing a foundation for future developments toward intelligent supervision, fault tolerance, and digital twin-enabled control.**

***Index Terms*— Distributed control, edge computing, vendor-independent automation, robotics, manufacturing systems**

I. Introduction

Modern industrial automation increasingly relies on the integration of advanced sensing, control, and communication technologies to enhance efficiency, accuracy, and flexibility of production systems. Vision-based optical sorting represents a key enabling technology in several industrial domains, where the classification of objects based on visual features supports downstream robotic operations such as pick-and-place. In this context, Selective Compliance Assembly Robot Arms (SCARAs) are widely adopted due to their high speed and repeatability, while artificial vision systems provide the necessary information to enable adaptive and precise manipulation within constrained environments [1], [7], [5].

Recent advances in optical sorting have demonstrated how vision systems, machine learning, and optical manipulation techniques can significantly improve the precision and adaptability of automated particle and component handling systems. However, most existing research primarily focuses on the sensing and manipulation principles themselves, while comparatively limited attention has been devoted to the supervisory control architectures required to integrate heterogeneous sensing, actuation, and legacy industrial devices into flexible and evolvable automation systems [8].

Despite these technological advances, Small and Medium-sized Enterprises (SMEs) still face significant difficulties in adopting high levels of automation. System complexity, high lifecycle costs, and strong dependence on proprietary vendor ecosystems often limit modern solution integration, especially when legacy devices are involved. As a result, existing automation systems tend to be rigid, difficult to update, and poorly suited for incremental evolution. Commercial automation platforms rarely support obsolete or heterogeneous components in a modular manner, making system upgrades costly and technically demanding [2], [6].

This work addresses these challenges, with a contribution that is primarily architectural. The proposed approach models the robotic cell as a set of loosely coupled control nodes coordinated by a supervisory controller through event-driven communication. The focus is on system modularity, device abstraction, and ease of integration of heterogeneous and discontinued hardware, rather than focusing on (quasi) real-time motion control.

II. Methodology

A. Control architecture and system modelling

From a control perspective, the proposed approach is framed within a distributed supervisory control paradigm for cyber-physical manufacturing systems. The architectural design follows a control-oriented abstraction in which each physical subsystem is associated with a logical control node, which can be interpreted as an event-driven state machine encapsulating device-specific behaviour. These nodes operate autonomously with respect to their internal control logic and interact with the rest of the system exclusively through high-level events and commands (see Figure 1). The supervisory controller operates at a discrete-event level, coordinating the global system behaviour by reacting to events generated by sensing and actuation nodes and by issuing high-level commands that trigger state transitions. This approach enables functional decoupling between sensing, decision-making, and actuation, a key requirement for flexible and evolvable manufacturing systems.

At a system level, the robotic cell is described as a set of interacting control nodes: a supervisory control node, one or more sensing nodes, and multiple actuation nodes.

[1]Faculty of Engineering, Industrial Engineering and Automation, Free University of Bozen-Bolzano, Bolzano, Italy dgalli@unibz.it, Chiara.Nezzi@unibz.it, Matteo.DeMarchi@unibz.it

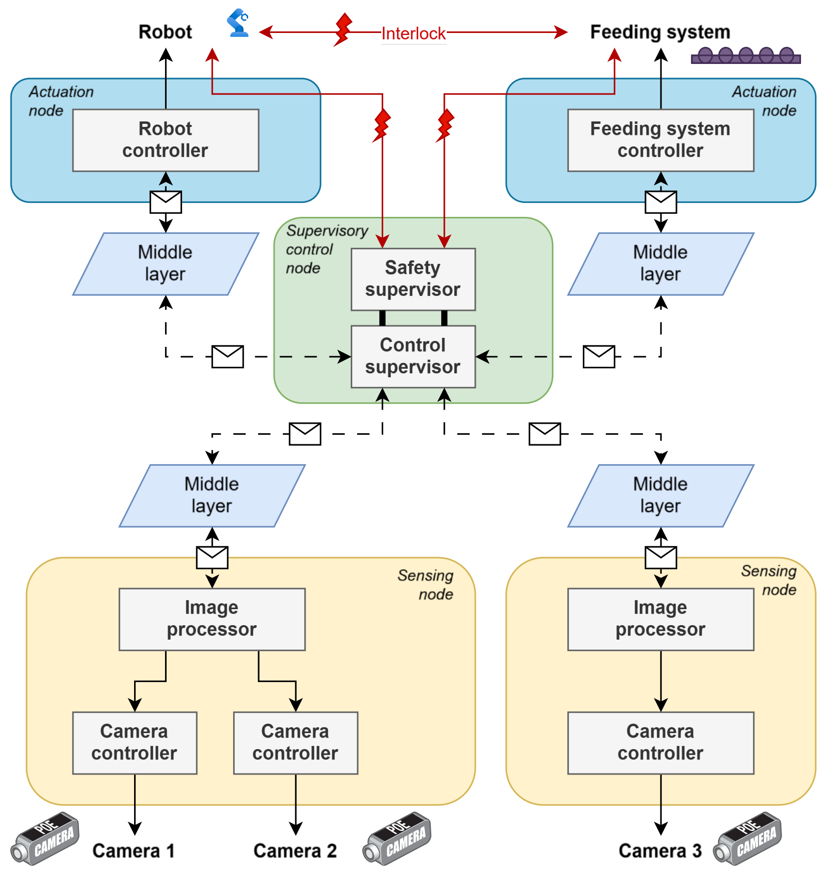

Fig. 1. Architectural design at system level.

The supervisory layer is responsible for controlling the overall machine cycle, managing process states, enforcing safety interlocks, and coordinating information exchange among subsystems. Sensing nodes encapsulate vision-based sensing and local data processing, while actuation nodes interface directly with physical devices such as robots or feeding systems. Each node operates autonomously with respect to its internal control logic, exposing only high-level commands and events to the rest of the system. From a control perspective, the overall behaviour of the robotic cell combines continuous-time execution at the device level (e.g., robot motion) with discrete-event coordination at the supervisory level. State transitions of the global system are triggered by events generated by perception or actuation nodes, enabling an event-driven control flow rather than a rigid cyclic scan. This approach is aligned with modern policies for manufacturing control systems, where flexibility and resilience are prioritized. This approach abstracts legacy and vendor-specific devices behind standard interfaces, reducing subsystem coupling and avoiding integrated control structures. By separating high-level sequencing from low-level execution, it enables easier reconfiguration-subsystems can be replaced or upgraded without redesigning overall logic. The model prioritizes modularity and adaptability in diverse environments, rather than formal optimal or real-time control.

B. Distributed control strategy

To enable distributed supervision and loose coupling among heterogeneous subsystems, the proposed control strategy adopts an asynchronous, event-driven communication model. This choice is motivated by the need to coordinate devices characterized by different dynamics, update rates, and communication capabilities, while avoiding tight synchronization constraints that typically arise in centralized or cyclic control architectures.

Communication between the supervisory controller and the distributed control nodes follows a publish–subscribe paradigm, which naturally supports supervisory control by decoupling event producers from event consumers in both time and execution context. Control commands, state updates, and perception results are exchanged as discrete events, allowing each control node to operate autonomously with respect to its internal logic while remaining coordinated at the system level. This interaction model enables scalable and modular integration of heterogeneous subsystems without enforcing a global control cycle. An explicit abstraction layer is introduced to isolate device-specific communication protocols from the supervisory logic. Legacy or vendor-specific devices are encapsulated within dedicated control nodes that translate high-level commands into device-specific actions and publish standardized feedback events.

As a result, the supervisory controller interacts exclusively with abstracted control interfaces, ensuring that changes at the hardware or middleware level do not propagate to the overall control strategy. This abstraction is a key enabler of vendor independence and system evolvability. The proposed distributed control strategy does not aim at hard real-time synchronization or deterministic motion control. Instead, it targets robust supervisory coordination, where bounded communication delays and asynchronous event handling are acceptable and explicitly accounted for at the control level. This trade-off allows the system to prioritize modularity, interoperability, and ease of reconfiguration over strict timing guarantees, which are typically enforced at lower control layers. From a control perspective, sensing and perception subsystems can be interpreted as edge-level control entities that perform local processing and decision-making before notifying the supervisory layer through events. This approach reduces unnecessary data transmission, supports scalability, and facilitates the integration of additional sensing or processing nodes without altering the core coordination mechanisms.

Overall, the adopted distributed, event-driven control strategy supports the integration of heterogeneous and legacy devices within a unified supervisory framework. By prioritizing loose coupling and architectural flexibility over strict determinism, the proposed approach is particularly suited for SME-oriented robotic cells and experimental environments characterized by frequent system evolution and mixed-generation hardware.

III. Preliminary validation and discussion

A. Robotic cell implementation

The proposed control architecture has been instantiated on a laboratory-scale robotic cell (see Figure 2) developed to investigate flexible optical sorting and vision-guided manipulation under heterogeneous and legacy constraints. The physical setup comprises a 4-degrees-of-freedom SCARA manipulator, a vibratory feeding system, and a multi-camera vision subsystem, intentionally selected from different vendors and generations to reflect realistic integration challenges commonly encountered in SMEs and research laboratories.

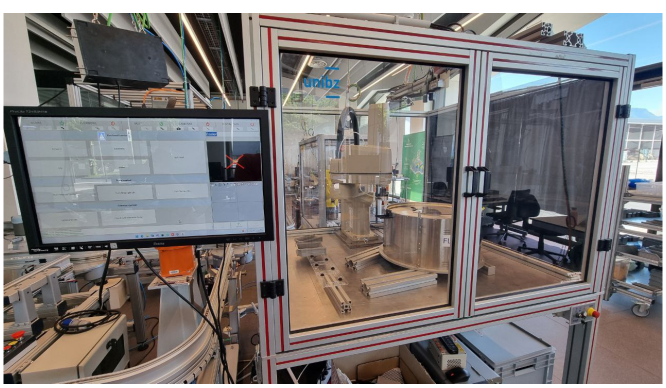

Fig. 2. Laboratory-scale robotic cell used for this study.

The actuation subsystem is centred on the Omron Cobra600 robot, originally manufactured by Adept and no longer in production. The robot exposes a TCP/IP-based communication interface and relies on a vendor-specific runtime environment to execute motion commands. A FlexiBowl 500 feeding system is used to randomly orient components prior to visual inspection; this device supports only UDP-based communication and offers limited feedback capabilities. The sensing subsystem consists of three heterogeneous cameras, including two industrial GigE Vision cameras, a Basler scA1300-32gm and a IDS uEye UI-5240CP-M-GL, and one stereo vision device, a Luxonis OAK-D SR Power-over-Ethernet (PoE), each characterized by distinct optical properties, drivers, and configuration interfaces. To allow the communication with the main computing unit, e.g. Lenovo ThinkCentre M75q Gen 2 with an AMD Ryzen 5 processor and 8 GB of RAM, all devices are connected to a PoE switch, e.g. D-Link DGS-1100-08P. This diversity motivated the adoption of a unified control abstraction rather than direct integration through commercial automation platforms.

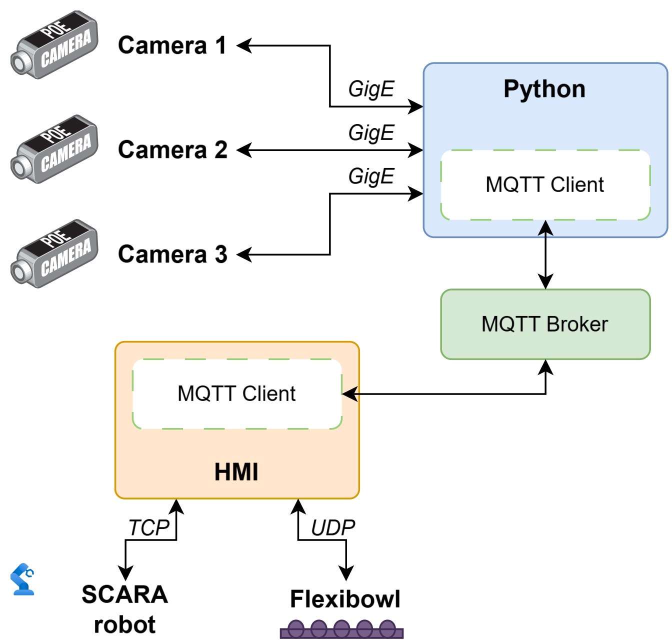

Fig. 3. Communication protocols between the nodes.

On the software side, the system is implemented as a distributed control solution composed of two main control nodes. A supervisory controller, implemented on a general-purpose computing platform, coordinates the overall machine cycle, manages process states, and interfaces with actuation devices [4]. A separate perception node handles camera management, image acquisition, and vision-based feature extraction. The latter was developed to allow image segmentation and object detection without machine learning, namely by using traditional computer vision techniques that rely on pixel properties, mathematical operations, or manual feature engineering. This choice was motivated by the predictable patterns of the objects to be recognised and the limited computational resources. The two nodes communicate exclusively through asynchronous message exchange, enabling a clear separation of concerns and facilitating independent development and testing of each subsystem.

B. Software development and control flow

The supervisory controller acts as the master coordination entity of the robotic cell. It is responsible for sequencing the automated process, enforcing interlocks, handling mode transitions (manual, automatic, configuration), and integrating feedback from the perception subsystem into control decisions. Rather than executing low-level control loops, the supervisor operates at a discrete-event level, issuing high-level commands such as acquisition requests, pick-and-place triggers, or feeder reconfiguration events.

The perception subsystem is implemented as an autonomous control node that encapsulates all vision-related functionalities. It manages the initialization and configuration of heterogeneous cameras, performs image acquisition, and executes machine vision pipelines to detect, segment and classify components. Internally, this node is structured as a state machine, ensuring that commands are executed only when the subsystem is in a valid operational state. This design prevents competing conditions and inconsistent configurations, which are common risks in distributed automation systems.

Communication between the control nodes is realized through a lightweight publish/subscribe messaging protocol, which enables event-driven coordination without tight synchronization requirements. Vision results, system states, and control commands are exchanged as discrete messages, allowing each node to operate asynchronously. From a control perspective, this realizes a networked supervisory control scheme in which sensing, decision-making, and actuation are decoupled yet coherently orchestrated.

C. Qualitative evaluation and practical implications

The practical realization of the robotic cell leads to an early validation, which is primarily qualitative and focuses on the architectural properties of the proposed solution. Flexibility is achieved by abstracting each physical device behind a logical control interface. Legacy hardware is integrated through dedicated middle-layer that translate standardized commands into device-specific instructions. This prevents vendor lock-in at the system level and enables straightforward replacement or upgrade of individual components without modifying the supervisory logic.

An early experimental characterisation was conducted to verify that the asynchronous coordination introduces acceptable delays in the considered application. Observed end to end event to command latencies were on the order of tens of milliseconds and compatible with the non-time-critical nature of the sorting process, while cycle times were dominated by robot execution rather than communication overhead, staying however in the range of tens of seconds. From a practical SME-oriented perspective, the proposed approach reduces integration effort and vendor lock-in compared to conventional PLC-centred solutions, at the cost of giving up strict timing determinism at the supervisory level. This trade-off is considered acceptable for flexible robotic cells operating under heterogeneous and legacy constraints.

A main area of concern is failure handling. In the implemented solution, it is managed at the supervisory level with explicit state machines, interlocks, and connection monitoring, rather than hard real-time fault tolerance. Camera management uses an event-driven state machine to block invalid commands and halt on disconnections or initialization failures. Message Queuing Telemetry Transport (MQTT) communication failures do not cause unsafe actions, as all critical operations require explicit confirmation. Robot communication timeouts pause automatic sequences for operator input.

Finally, the implementation demonstrates how such an architecture naturally supports future extensions toward advanced control functionalities. The existing modular structure can accommodate additional perception nodes, multiple robots, or higher-level optimization layers without altering the fundamental coordination mechanisms.

D. Integration to a digital shadow

The use of modern and standard communication protocols between the middle layers and the nodes, allows to expose useful information by intercepting the data traffic, that can be redirected to third party software for data storage or digital shadows. In order to test this capability, the robotic cell was integrated with an industrial simulation platform (industrialPhysics) to generate a functioning digital shadow of the SCARA robot (see Figure 4). It allows a real-time, one-way projection of a physical system into its virtual counterpart. This primarily serves for visualization, monitoring, and basic validation purposes without influencing the physical system. The advantages of implementing a digital shadow include enhanced situational awareness, improved diagnostic capabilities, and the ability to analyse the behaviour of the robot under different simulated scenarios without disrupting real-world operations [3].

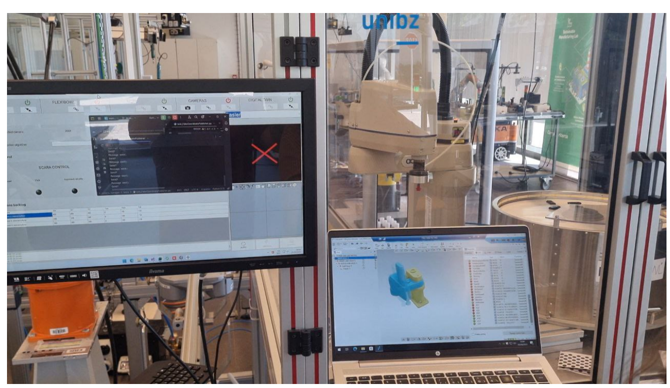

Fig. 4. Working digital shadow of the SCARA robot with industrialPhysics.

IV. Conclusion

This paper presents a distributed control architecture for flexible robotic cells aimed at overcoming interoperability and rigidity issues commonly associated with vendor-dependent automation solutions. By modelling the system as a set of loosely coupled control nodes coordinated through a supervisory layer, the proposed approach enables the integration of heterogeneous and legacy devices within a unified control framework. The event-driven communication strategy supports modularity and device abstraction, allowing system components to be replaced or extended with minimal impact on overall control logic.

The validation of the architecture is primarily qualitative and demonstrates the feasibility of the proposed approach under realistic conditions characterized by mixed-generation hardware and heterogeneous communication protocols. Experimental observations indicate that the asynchronous event-driven coordination provides sufficient responsiveness for supervisory tasks in non time-critical sorting applications, while preserving modularity and ease of reconfiguration. The integration of a digital shadow further highlights how the proposed communication and abstraction mechanisms can support monitoring and future extensions without modifying the core control logic.

Some limitations of the proposed approach must be acknowledged. First, the architecture does not target hard real-time guarantees and is therefore not suitable for low-level motion control or high-speed industrial sorting applications requiring deterministic timing. Second, fault handling is managed at the supervisory level through state machines, interlocks, and operator intervention, rather than through fully automated fault-tolerant recovery mechanisms. While this design choice improves transparency and safety when dealing with legacy hardware, it limits autonomous recovery capabilities. Third, the quantitative performance characterization is intentionally limited, as the focus of this work is on architectural feasibility rather than exhaustive benchmarking.

Future work will start by addressing the previously mentioned limitation, e.g. architecture benchmarking, as well as by focusing on extending the proposed framework toward more advanced control functionalities, including predictive supervision, fault detection, and multi-robot coordination. Additionally, the existing digital shadow can evolve into a bidirectional digital twin to enable closed-loop optimization and intelligent decision-making at the supervisory control level.

Acknowledgment

This research was funded by the European Union Next Generation EU, Mission 4 Component 1 (CUP I52B23000570003).

References

[1] Nicola Ivan Giannoccaro, Giuseppe Rausa, Roberta Rizzi, Paolo Visconti, and Roberto De Fazio. An innovative vision-guided feeding system for robotic picking of different-shaped industrial components randomly arranged. *Technologies*, 12(9):153, 2024.

[2] Silvia Moreira, Henrique S. Mamede, and Arnaldo Santos. Business Process Automation in SMEs: A Systematic Literature Review. *IEEE Access*, 12:75832–75864, 2024.

[3] Chiara Nezzi, Matteo De Marchi, Renato Vidoni, and Erwin Rauch. Modeling and simulation of mechatronics equipment for a Digital Twin-enabled demonstrator. In *2024 10th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 2526–2529, Vallette, Malta, July 2024. IEEE.

[4] Chiara Nezzi, Matteo De Marchi, Renato Vidoni, and Erwin Rauch. A multi-purpose simulation layer for digital twin applications in mechatronic systems. *Machines*, 13(8), 2025.

[5] van-Truong Nguyen, Phu-Tuan Nguyen, Shun-Feng Su, Phan Xuan Tan, and Thanh-Lam Bui. Vision-Based Pick and Place Control System for Industrial Robots Using an Eye-in-Hand Camera. *IEEE Access*, 13:25127–25140, 2025.

[6] Choudary Muthu Sumanth and Nithin Rama Reddy. Exploring levels of automation related to smes. Master's thesis, Jönköping Univeristy, 2020.

[7] Yuanlong Xie, Shuting Wang, Shiqi Zheng, and Zhaozheng Hu. Advanced sensing and control technologies for autonomous robots. *Sensors*, 24(17), 2024.

[8] Meng Yang, Yuzhi Shi, Qinghua Song, Zeyong Wei, Xiong Dun, Zhiming Wang, Zhanshan Wang, Cheng-Wei Qiu, Hui Zhang, and Xinbin Cheng. Optical sorting: past, present and future. *Light: Science & Applications*, 14(1):103, February 2025.