Integrating ROS 2 and Physical AI: Architecture and Challenges

1st Sanghoon Lee
Electrical Engineering and Computer Science (EECS)
Daegu Gyeongbuk Institute of Science and Technology (DGIST)
Daegu, Republic of Korea
leesh2913@dgist.ac.kr

2nd Jiyeong Chae
EECS
DGIST
Daegu, Republic of Korea
cowldud3@dgist.ac.kr

3rd Kyung-Joon Park
EECS
DGIST
Daegu, Republic of Korea
kjp@dgist.ac.kr

2025 Sixteenth International Conference on Ubiquitous and Future Networks (ICUFN) | 979-8-3315-2487-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICUFN65838.2025.11170038

***Abstract*—Physical Artificial Intelligence (Physical AI) embeds sensors, actuators, and AI algorithms to enable robots to learn and adapt in real-world settings. ROS 2, with its DDS-based publisher–subscriber model and advanced Quality of Service (QoS) features, forms a robust platform for such real-time, distributed applications. Yet, integrating AI within a Cyber-Physical System (CPS) framework raises concerns about resource constraints, safety, and uncertainties. This paper explores how to incorporate Physical AI into ROS 2 by fully automating or partially augmenting robot functionalities, focusing on topic structures, message formats, and key QoS considerations. We highlight major challenges—data availability, data loss, inference latency, and security vulnerabilities—and emphasize the importance of a holistic approach to ensure reliability and safety.**

***Index Terms*—Physical AI, ROS 2, Cyber-Physical System**

I. Introduction

Physical Artificial Intelligence (Physical AI) has recently emerged as a novel approach for equipping robots with AI-driven capabilities [1]. In this paradigm, sensors enable AI software to perceive real-world conditions, while actuators allow it to interact with or modify the physical environment. By embedding AI directly into a robot's hardware and control loops, Physical AI seeks to create systems that learn from and adapt to their surroundings in real time.

To ensure Physical AI solutions remain robust and practical, developers increasingly rely on standardized software platforms for managing sensor data, actuator commands, and AI-based decisions. ROS has become the de facto standard for modern robotics, offering key enhancements for real-time and distributed systems [2]. Unlike ROS 1, which centers on a master node, ROS 2 adopts a Data Distribution Service (DDS)–based publisher–subscriber model with advanced Quality of Service (QoS) options. These features make ROS 2 especially well-suited for scalable, low-latency applications—critical components of next-generation Physical AI.

When considering the combined architecture of AI software, communication infrastructure, and robot hardware, it aligns naturally with the concept of a cyber-physical system (CPS). In CPS, computational (cyber) and mechanical (physical) components work together over networks to achieve monitoring, control, and adaptive behaviors [3]. Thus, a Physical AI robot that relies on ROS 2 for software orchestration neatly fits into the CPS framework, where AI algorithms interact closely with physical processes under real-world constraints.

The Cyber-Physical AI (CPAI) domain has emerged to address the unique challenges that arise when integrating AI into CPS environments [4]. Rather than focusing solely on performance gains, CPAI emphasizes resource constraints, uncertainty, and safety—crucial factors for real-world deployments. By examining how AI and CPS elements intersect, CPAI seeks to ensure that intelligent behaviors remain both robust and efficient in complex physical settings.

Building on this perspective, our study explores how to integrate Physical AI into ROS 2 for automating (automation) or enhancing (augmentation) robot capabilities. Although ROS 2 has effectively become the de facto standard in modern robotics, efficiently managing AI operations and data flow under CPS constraints—such as safety, real-time requirements, and limited resources—remains an unresolved challenge. Consequently, from the standpoint of CPAI, we investigate conceptual approaches and technical obstacles to deploying AI effectively on ROS 2-based robots. This investigation emphasizes how AI nodes should comply with or transform existing topic structures and message formats, the distinctions between complete functionality replacement (automation) and functionality enhancement (augmentation), as well as the comprehensive consideration of data availability, reliability, and security issues.

The remainder of this paper is organized as follows. Section 2 provides a detailed introduction to adding or replacing AI modules while preserving the existing ROS 2 node, topic, and message architecture, illustrating how Physical AI can be implemented through automation and augmentation. Section 3 delves into representative challenges encountered during actual integration. Finally, Section 4 presents the conclusions of this study.

II. Physical AI in ROS 2

In the context of CPS, ROS 2 serves as a core platform that links the robot's hardware (physical system) to the software (cyber system) responsible for its operation. A robot system is divided into multiple nodes for management, and

This work was supported in part by the National Research Foundation of Korea (NRF) and in part by Korea government (MSIT) under Grant 2023R1A2C2003901.

data exchange among nodes is handled through the DDS. This mechanism is not limited to communication within a single robot; it also includes data transmission among multiple robots or between robots and servers. Meanwhile, Physical AI refers to the concept of replacing (automation) or assisting (augmentation) part or all of the robot's control and decision-making functions with AI algorithms. From a CPAI viewpoint, automation typically entails a complete replacement of existing functionality, whereas augmentation preserves existing functionality while adding AI-based assistance.

To introduce Physical AI into the ROS 2 platform, developers must accurately identify the input and output topics, as well as the message structures, of the existing functional nodes they intend to replace or augment. In this study, we assume that users will not modify the internal logic of existing functional nodes or change their existing topic structures; in other words, the newly added AI inference node will adhere to the topic architecture to which the existing nodes both publish and subscribe.

A. Automation

When Physical AI fully replaces (automation) an existing robot function, the AI node must subscribe and publish to the same topics with identical message types. For example, if the existing node uses `/scan` for LiDAR data and publishes motion commands to `/cmd_vel`, the AI node should also subscribe to `/scan` and publish `Twist` messages to `/cmd_vel`. If new sensor data is needed, additional topics and messages can be introduced, taking care to match or properly convert QoS settings. If the message format is different, a bridge node or an internal conversion process should be implemented to preserve compatibility.

B. Augmentation

When Physical AI only partially assists (augmentation) an existing robot function, there are two main approaches: *perception-oriented* and *control-oriented*. The AI node processes sensor input (e.g., `/scan`) to provide new or enriched data. If the existing node expects certain message types—such as `Twist` or a custom detection message—the AI node can publish using the same format on the same or a newly created topic, enabling direct interpretation by the existing node. The AI node subscribes to the control commands published by the existing node (e.g., `/cmd_vel`), modifies or refines them (for obstacle avoidance, speed limits, etc.), then republishes the adjusted commands back to `/cmd_vel` using the same `Twist` format. This seamlessly integrates advanced AI control logic without altering the overall robot control framework.

To integrate Physical AI into a ROS 2 environment, it is crucial to understand existing message types, topic names, and QoS settings [5]. The AI node should maintain the same structure or provide suitable conversions for compatibility. Figure 1 shows the conceptual structure of how Physical AI is integrated with ROS 2.

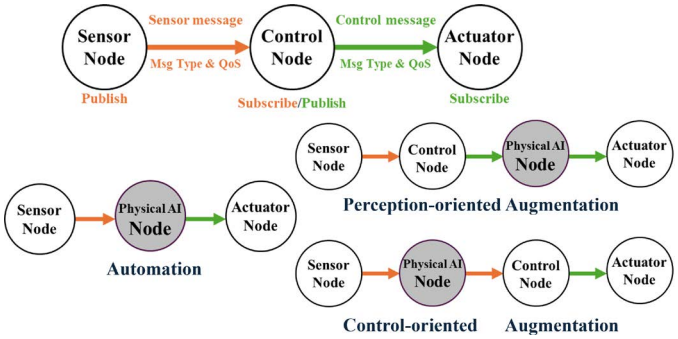

Fig. 1. Structure of Physical AI in ROS 2.

III. Key Challenges

Following the approach described in the previous section, various challenges may arise during the integration of Physical AI into ROS 2. This section discusses the potential issues that may occur when Physical AI is introduced into ROS 2.

A. Data Availability

The first challenge pertains to data availability. Because AI generally demands a broader range and greater volume of data than existing robot functional nodes, such data must be provided as topics at regular intervals for real-time processing. If data must be gathered from multiple robots or external sensor networks rather than local sensors alone [6], [7], synchronization and integration procedures may become bottlenecks or even fail to provide sufficient data for the inference node, resulting in incomplete inputs. In scenarios involving large-scale data such as video feeds or LiDAR point clouds, it is easy to exceed network bandwidth and potentially disrupt the transmission itself. To address these issues, one could adopt a suitable feature extraction algorithm [8] or employ dynamic network optimization [9], [10].

B. Data Loss

The second challenge concerns data loss. Because AI algorithms can be highly sensitive to missing or incomplete inputs, any loss of data may render inference impossible or lead to inaccurate results. Data loss can arise at the ROS 2 process level if the inference node's execution timing does not align with topic publication, or at the network level due to packet loss or delay. While applying ROS 2's Reliable QoS can increase reliability, the retransmission overhead may cause additional latency [11], thereby preventing timely inference results in applications requiring real-time performance. To address these challenges, employing a dynamic optimization strategy for the control node in response to data latency or loss can be effective [12].

C. Inference Time Constraints

The third challenge relates to inference time constraints. AI, especially deep neural network algorithms, typically requires more computational resources than conventional robot functionalities [13]. Consequently, on-device processing may

fail to meet strict deadlines or may accumulate latency due to hardware performance limitations. This latency does not remain confined to a single inference node; rather, it propagates through interconnected nodes, influencing scheduling and execution order system-wide [14]. Although leveraging a server for high-performance computing is an option, communication delays or network bottlenecks may still impede real-time responsiveness.

D. Lack of Inference Reliability

A fourth challenge is the reliability of inference results themselves. Overly aggressive commands or infeasible trajectories generated by an AI inference node can lead to physical collisions or task failures. Even minor perception errors or noise, if accumulated, may cause the robot to behave unpredictably. Because safety is at stake, logic that verifies or constrains AI outputs (for example, ignoring results that exceed predefined thresholds or issuing warnings) becomes essential. To address these challenges, one can adopt an AI training strategy that accounts for potential system failures in advance [15], or measure the uncertainty of the AI inference and avoid using the results when uncertainty is high [16].

E. Malicious Attacks

The final challenge is vulnerability to malicious attacks. AI nodes may have a larger attack surface than traditional robot systems if they rely on extensive network-based data exchange, thereby increasing the likelihood of data breaches or malicious data injection. Although ROS 2 supports multiple security protocols (e.g., SROS [17]), there remain few definitive measures to defend deep learning models against adversarial attacks or data tampering in real-world applications. Consequently, robotic systems may malfunction, or sensitive data may be exposed [18]. Adequate security measures must be put in place to address these risks.

IV. Conclusion

In this paper, we examined how Physical AI can enhance or replace robot functionalities within ROS 2, emphasizing both automation and augmentation strategies under CPS constraints. Maintaining existing topic structures and message formats was shown to be crucial for smooth operation, while AI-driven modules must address data availability, timing requirements, and security threats.

Our analysis underscores the need for robust QoS settings, thorough validation of inference results, and carefully designed data flows to safeguard system performance and reliability. Looking ahead, future research should standardize interfaces, refine scheduling mechanisms, and strengthen security protocols for AI nodes. By tackling these issues, ROS 2–based Physical AI can evolve into a more scalable, reliable, and secure platform for the next generation of intelligent robotic systems.

References

[1] Y. Li, Z. Li, Y. Duan, and A.-B. Spulber, "Physical Artificial Intelligence (PAI): the next-generation Artificial Intelligence," *Frontiers of Information Technology & Electronic Engineering*, vol. 24, no. 8, pp. 1231–1238, 2023.

[2] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, *et al.*, "ROS: an open-source Robot Operating System," in *In Proceedings of the ICRA Workshop on Open Source Software*, vol. 3, p. 5, Kobe, 2009.

[3] J. Chae, S. Lee, J. Jang, S. Hong, and K.-J. Park, "A survey and perspective on industrial Cyber-Physical systems (ICPS): From ICPS to AI-augmented ICPS," *IEEE Transactions on Industrial Cyber-Physical Systems*, vol. 1, pp. 257–272, 2023.

[4] S. Lee, J. Chae, H. Jeon, T. Kim, Y.-G. Hong, D.-S. Um, T. Kim, and K.-J. Park, "Cyber-physical artificial intelligence: Systematic research domain for integrating artificial intelligence and cyber-physical systems," *ACM Transactions on Cyber-Physical Systems*, Mar. 2025. Just Accepted.

[5] J. Chae, H. Seo, S. Lee, Y. Park, H.-S. Park, and K.-J. Park, "PINMAP: A Cost-efficient Algorithm for Glass Detection and Mapping Using Low-cost 2D LiDAR," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–14, 2025.

[6] S. Moon, S. Lee, and K.-J. Park, "Graph-based reinforcement learning for flexible job shop scheduling with transportation constraints," in *In Proceedings of the IEEE IECON 2023–49th Annual Conference of the IEEE Industrial Electronics Society*, pp. 1–6, 2023.

[7] S. Moon, S. Lee, and K.-J. Park, "Learning-enabled flexible job-shop scheduling for scalable smart manufacturing," *Journal of Manufacturing Systems*, vol. 77, pp. 356–367, 2024.

[8] S. Lee, J. Chae, S. Moon, S.-C. Lee, and K.-J. Park, "False alarm prevention through domain knowledge-driven machine learning: Leakage detection in water distribution networks," *IEEE Sensors Journal*, vol. 24, no. 19, pp. 31538–31550, 2024.

[9] S. Moon, S. Lee, W. Jeon, and K.-J. Park, "Learning-enabled network-control co-design for energy-efficient industrial internet of things," *IEEE Transactions on Network and Service Management*, vol. 21, no. 2, pp. 1478–1489, 2024.

[10] H.-S. Park, S. Moon, J. Kwak, and K.-J. Park, "CAPL: Criticality-aware adaptive path learning for industrial wireless sensor–actuator networks," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 8, pp. 9123–9133, 2023.

[11] H.-S. Park, S. Lee, and K.-J. Park, "An analytical latency model of the data distribution service in ROS 2," in *In Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, May 2025. to appear.

[12] S. Kim, S. Lee, and K.-J. Park, "Real-time controller reconfiguration for delay-resilient cyber-physical systems," *IEEE Access*, vol. 10, pp. 101220–101228, 2022.

[13] R. Agyeman and B. Rinner, "Resource-efficient pervasive smart camera networks," in *In Proceedings of the 2022 IEEE International Conference on Pervasive Computing and Communications Workshops and Other Affiliated Events (PerCom Workshops)*, pp. 503–508, IEEE, 2022.

[14] C. Randolph, "Improving the predictability of event chains in ROS 2," *PhD diss., Delft University of Technology*, 2021.

[15] S. Lee, J. Kim, G. Wi, Y. Won, Y. Eun, and K.-J. Park, "Deep reinforcement learning-driven scheduling in multijob serial lines: A case study in automotive parts assembly," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, pp. 2932–2943, 2024.

[16] X. Gu and A. Easwaran, "Towards safe Machine Learning for CPS: Infer uncertainty from training data," in *In Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*, ICCPS '19, (New York, NY, USA), pp. 249–258, ACM, 2019.

[17] R. White, D. H. I. Christensen, and D. M. Quigley, "SROS: Securing ROS over the wire, in the graph, and through the kernel," *arXiv preprint arXiv:1611.07060*, 2016.

[18] S. Kim, K.-J. Park, and C. Lu, "A survey on network security for cyber–physical systems: From threats to resilient design," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 3, pp. 1534–1573, 2022.