POLITECNICO DI TORINO
Repository ISTITUZIONALE

UAV Model-based Flight Control with Artificial Neural Networks: A Survey

Original
UAV Model-based Flight Control with Artificial Neural Networks: A Survey / Gu, W., Valavanis, K.P., Rutherford, M.J., Rizzo, A.. - In: JOURNAL OF INTELLIGENT & ROBOTIC SYSTEMS. - ISSN 0921-0296. - 100:3-4(2020), pp. 1469-1491. [10.1007/s10846-020-01227-8]

Availability:
This version is available at: 11583/2855210 since: 2020-12-09T10:50:17Z

Publisher:
Springer Science and Business Media B.V.

Published
DOI:10.1007/s10846-020-01227-8

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
Springer postprint/Author's Accepted Manuscript

This version of the article has been accepted for publication, after peer review (when applicable) and is subject to Springer Nature's AM terms of use, but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record is available online at: http://dx.doi.org/10.1007/s10846-020-01227-8

(Article begins on next page)

19 November 2026

Journal of Intelligent & Robotic Systems (2026) 112:96
https://doi.org/10.1007/s10846-026-02432-7

RESEARCH

Digital Twin-Based Endurance Prediction and Health Monitoring for Heterogeneous Mobile Manipulator

Yunhan Zhang[1] · Xiaolang Yang[2] · Khurshid Aliev[1] · Dario Antonelli[1]

Received: 17 March 2026 / Accepted: 1 July 2026
© The Author(s) 2026

Abstract

Digital Twin (DT) has been increasingly implemented to improve reliability, transparency, and decision-making in industrial robotics. The MObile MAnipulator (MOMA), which combines an autonomous mobile robot with a collaborative manipulator, provides high mobility and manipulability to meet the growing demand for flexibility and agility in intelligent manufacturing. Although the heterogeneous robots in MOMA perform different functions, they are driven by a shared energy source. Therefore, the Prognostics and Health Management (PHM) of MOMA remains challenging, stemming not only from the coupled dynamics of navigation-manipulation functions but also from the heterogeneity of sensing modalities involved. To address these challenges, this paper proposes a DT-based endurance prediction and health monitoring approach for heterogeneous MOMA. First, a systematic architecture, including the physical layer, DT layer, machine learning layer, and dashboard layer, is proposed, and the intra- and inter-layer data flows are defined in detail. Subsequently, a continuous bidirectional telemetry data pipeline is established to consolidate heterogeneous real-time data from the physical MOMA and maintain data synchronization between physical and virtual spaces. A data-driven prediction method based on the CatBoost model is employed to provide remaining useful cycle estimates, enabling mission-level endurance assessment. Furthermore, a virtual reality-assisted health monitoring dashboard is developed to support human-in-the-loop execution supervision and reactive control for MOMA operations. Finally, the proposed approach is experimentally validated with a MOMA in the laboratory. The results demonstrate the proposed framework is a practical solution for MOMA PHM in intelligent manufacturing.

Keywords Digital twin · Endurance prediction · Health monitoring · Mobile manipulator · Intelligent manufacturing

1 Introduction

The MObile MAnipulator (MOMA), which integrates an autonomous mobile robot with a collaborative manipulator, is designed for logistics, handling, and assembly tasks that require both mobility and dexterity [1]. However, real-time Prognostics and Health Management (PHM) for MOMA is more challenging than for fixed-base manipulators or standalone mobile robots because the mobile base and manipulator differ in sensing modalities, communication protocols, sampling frequencies, control schemes, power-consumption patterns, and motion dynamics, while often sharing the same battery source [2]. Their coupled behavior, including load transfer, motion interaction, and energy sharing, makes the operational state of a MOMA difficult to characterize by monitoring either subsystem independently. This differs from many existing industrial-robot PHM studies, which mainly focus on component-level degradation or single-robot operating conditions [3].

Digital Twin (DT) technology provides a promising way to connect physical robots with their virtual representations for real-time state synchronization, data-driven prediction, and human-in-the-loop supervision [4, 5]. This capability is increasingly relevant to human-centric smart manufacturing and Industry 5.0, where industrial robots are expected to achieve higher autonomy, resilience, and collaborative capability [6, 7]. However, existing DT-based robotic monitoring studies are still mainly designed for single-platform systems, such as fixed-base manipulators, standalone mobile robots, or unmanned aerial systems [8–10]. Many frameworks also remain visualization- or simulation-oriented, with limited

✉ Dario Antonelli
dario.antonelli@polito.it

[1] Department of Management and Production Engineering, Politecnico Di Torino, 10129 Turin, Italy

[2] School of Mechanical Engineering, Tongji University, Shanghai 201804, China

Published online: 10 July 2026

support for integrating heterogeneous telemetry, predictive endurance information, abnormal-state perception, and supervisory intervention into a deployable end-to-end PHM system.

This limitation becomes more critical for Industrial Robots (IRs), as existing DT-based robotic monitoring studies have not sufficiently addressed how heterogeneous robot telemetry can be synchronized and transformed into a human-interpretable, mission-level endurance representation [11]. In particular, there remains a lack of an endurance metric that describes residual operational capability in terms of executable task cycles, rather than only battery percentage, remaining operating time, or component-level lifetime [12]. Meanwhile, the heterogeneity between the mobile base and manipulator requires a continuous bidirectional data channel between the physical MOMA layer and its DT-based virtual layer to support multi-source telemetry synchronization and real-time state mapping. In addition, an operator-oriented dashboard is also needed to integrate robot-state visualization, predicted endurance, abnormal-state perception, and supervisory intervention.

To address these challenges, this paper proposes a DT-based endurance prediction and health monitoring framework for heterogeneous MOMA, which includes a MiR100 mobile robot and a UR3 collaborative manipulator. A continuous bidirectional data channel is first established between physical robots and their digital counterparts, enabling real-time detection, synchronization, and computation of key states such as navigation status, mechanical metrics, mission progress, and energy status. Subsequently, based on the synchronized telemetry data, a dedicated machine learning module is designed to estimate the Remaining Useful Cycle (RUC) as a metric of endurance. Furthermore, the abnormal operational modes of MOMA are defined, and a Virtual Reality (VR)-assisted dashboard is developed to support responsive human-in-the-loop interventions under fault or performance degradation conditions.

The paper is organized as follows. Section 2 discusses the latest work on DT for IRs and PHM for IRs to situate the scope of this study. The DT-based multi-layer framework, as well as the intra- and inter-layer data flows, are presented in Sect. 3. In Sect. 4, the key implementation technologies are introduced in detail. A research case in the laboratory is implemented and demonstrated with the proposed approach in Sect. 5. Finally, in Sect. 6 the conclusion and outlook are discussed.

Nomenclature

Abbreviation	
ANN	Artificial Neural Network
API	Application Programming Interface
AR	Augmented Reality
CDT	Cognitive digital twin
CV	Cross Validation
DRL	Deep Reinforcement Learning
DT	Digital Twin
EOL	End of Line
HRC	Human–Robot Collaboration
HRI	Human–Robot Interaction
HTTP	HyperText Transfer Protocol
IR	Industrial Robot
IRs	Industrial Robots
JSON	JavaScript Object Notation
LLM	Large Language Model
MAE	Mean Absolute Error
MOMA	MObile MAnipulator
MQTT	Message Queuing Telemetry Transport
MR	Mixed Reality
PC	Personal Computer
PHM	Prognostics and Health Management
P&P	Pick-and-Place
RF	Random Forest
RMS	Robotic Machining System
RMSE	Root Mean Squared Error
ROS	Robot Operating System
RTDE	Real-Time Data Exchange
RUC	Remaining Useful Cycle
RUL	Remaining Useful Life
SHAP	SHapley Additive exPlanations
SoC	State of Charge
SVR	Support Vector Regression
VR	Virtual Reality
XR	eXtended Reality

2 Related Works

This section summarizes recent advancements of DT for IRs and PHM for IRs in intelligent manufacturing as follows.

2.1 DT for IRs

Based on the differences in participants and the number of robots involved in the intelligent manufacturing process, IR-enabled manufacturing systems can be categorized into three types: single-robot systems, human–robot systems, and multi-robot systems [13]. At present, a substantial body of research has focused on DT-empowered robotic

manufacturing systems. However, due to variations in system composition and application scenarios, the research priorities and architectural designs of DT differ significantly across these system types.

Among them, for single-robot systems, Ni et al. developed a DT-driven virtual commissioning system for the Robotic Machining System (RMS) to improve machining accuracy and reduce the difficulty of commissioning. The framework of the developed system included the interaction, data preprocessing, DT model of RMS, and optimization service [14]. Li et al. developed an integrated embodied control system for robots to support a unified perception-decision-action loop. The framework integrated the DT, machine vision and Deep Reinforcement Learning (DRL), and operated through a DT-based co-training and online optimization mechanism [15]. Jin et al. proposed a DT-based smart assembly framework for cable-driven parallel robots with a large workspace, high load capacity, and strong dynamic performance. The framework consisted of the physical assembly space, virtual-physical connection layer, and virtual assembly space [16]. Yang et al. proposed a multi-level multi-domain DT framework for IRs. The framework systematically integrated the physical space, digital space and their bidirectional interactions, and explicitly defined hierarchical structures and cross-domain mechanisms [17].

Unlike single-robot systems, the DT for human–robot systems must additionally account for the digital representation of human, as well as Human–Robot Interaction (HRI) and perception [18]. Han et al. proposed a model-based approach for human DT development, and the proposed four-layer structure included the physical layer, data layer, inference layer and implementation layer [19]. Feng et al. developed a vision-based human DT method for rapid and accurate human behavior modeling. The architecture of the proposed approach consisted of the Mixed Reality (MR), object detection, human detection, motion planning and digital twins [20]. Li et al. proposed a DT-based Human–Robot Collaboration (HRC) safety control framework for the distance calculation between humans and robots. The proposed framework included the physical system, virtual simulation system, VR system and data interaction [21]. Gkournelos et al. addressed communication complexities in HRC by introducing an execution system integrating with DT. The framework of the proposed system consists of three components: human interface, Large Language Model (LLM)-based execution system, and DT. This system served to provide a natural language interface for operators to facilitate the intuitive HRI [22]. Nguyen et al. recently proposed a DT-empowered framework for interactive manufacturing using wearable devices, which further demonstrates the value of DT for real-time HRI and manufacturing transparency [23]. Pham et al. investigated social perception and formation control for human–robot co-working systems, highlighting the importance of safety- and comfort-aware interaction in shared human–robot workspaces [24].

Furthermore, compared to the other two types, research on multi-robot systems places emphasis on the integration of models across different robots and their collaborative control. Xu et al. presented a Cognitive digital twin (CDT) framework for multi-robot collaborative manufacturing, which integrates virtual models of DT and knowledge/data-driven cognition models. The CDT framework encompasses three interrelated spaces, namely the physical-virtual space, cognition space, and data space [25]. Zhou et al. developed a digital twin model integrated with dynamics for an industrial MOMA, accounting for the physical interaction of robot-humans/environments. The framework included the resource module, robot controller module, digital twin module and function module [26]. Oo et al. introduced a DT framework including the physical world, digital world, and information integration. The framework was designed to enhance the flexibility and efficiency of multi-robot operating systems by collaborating with operator skills [27]. Antonelli et al. proposed a CDT design and implementation approach to enhance the capabilities of autonomous industrial MOMA in industrial settings. The conceptual framework of CDT was composed of the physical system, sensors&data acquisition, control system, DT model, and decision support [28].

Based on the reviewed literature, Table 1 provides an overview of the system type, research purpose, DT architecture and application scenario addressed in recent academic publications. Existing single-robot DT studies have mainly focused on virtual commissioning, path optimization, assembly synchronization, or embodied control for individual robotic platforms. Human–robot DT studies have primarily emphasized human representation, safety monitoring, interaction interfaces, and collaborative task execution. Multi-robot DT studies have further extended DT toward collaborative manufacturing, dynamic simulation, decision support, and autonomous capability enhancement. However, compared with these state-of-the-art studies, the integration of DT with PHM-oriented endurance prediction for heterogeneous MOMA remains insufficient. In particular, existing DT-enabled MOMA studies mainly emphasize virtual representation, dynamic simulation, collaboration, or decision support, but do not sufficiently address the combined requirements of heterogeneous multi-rate telemetry synchronization, shared-battery endurance prediction, fault-aware health monitoring, and human-in-the-loop supervisory response within a deployable end-to-end pipeline. This gap is especially critical for MOMA systems, where the mobile base and manipulator exhibit different sensing modalities, control schemes, sampling frequencies, and behavioral patterns while jointly contributing to task execution and energy consumption. Therefore, a DT framework for heterogeneous MOMA should not only maintain a virtual representation of the physical system, but also establish a stable telemetry pipeline and predictive monitoring mechanism to support mission-level PHM and operator decision-making.

Table 1 The overview about the application of DT for IRs in intelligent manufacturing in academic publications

System type	Refs	Research purpose	DT architecture	Application scenario
Single-robot systems	[14]	Machining path prediction and error compensation	Interaction, data pre-processing, DT model of RMS, and optimization service	RMS
	[15]	Integrated embodied control for robot	Real word site, status perception, DT, and DRL agent	Robotic arm
	[16]	Coordinating dynamic assembly processes and ensuring digital-physical synchronization	Physical assembly space, virtual-physical connection layer, and virtual assembly space	Cable-driven parallel robot
	[17]	Modeling inherent multi-domain complexities of IRs	Physical space, digital space, and their bidirectional interactions	SD3/500 IR
Human–robot systems	[19]	Human DT development	Physical layer, data layer, inference layer, and implementation layer	Human
	[20]	Promoting the evolution of human worker's safe operation cognition and the evolution of robot's safe motion planning strategy	MR, object detection, human detection, motion planning, and DT	HRI system
	[21]	Ensuring the safety of the HRC process in real-time	physical system, virtual simulation system, VR system and data interaction	HRC system
	[22]	Addressing communication complexities in HRC	Human interface, LLM-based execution system, and DT	HRC system
	[23]	Aiming to further productivity, automation, and safety	Perception and sensing layer, digital twin and synchronization layer, control and execution layer	HRI system
	[24]	Delivering flexible behaviors in front of human or a group of humans	Perception layer, pre-processing layer, dynamic risk potential estimation layer, Actuation layer	Human–robot co-working
Multi-robot systems	[25]	Enhancing the intelligence and cognition capabilities of multi-robot collaborative manufacturing	Physical-virtual space, cognition space, and data space	Multi-robot collaborative manufacturing system
	[26]	Real-time dynamic simulation analysis and optimization design	resource module, robot controller module, digital twin module and function module	MOMA
	[27]	Enhancing the flexibility and efficiency of multi-robot operating systems	Physical world, digital world, and information integration	Multi-robot collaborative manufacturing system
	[28]	Enhancing the autonomous capabilities of MOMA in industrial settings	Physical system, sensors&data acquisition, control system, DT model, and decision support	MOMA

2.2 PHM for IRs

PHM, a form of condition-based maintenance, is evolving from a reactive to a proactive and even autonomous paradigm with the progress of enabling technologies such as edge computing, high-speed communication, and machine learning [29]. Its core value lies in the early prediction of impending failures and the anticipation of system degradation mechanisms, thereby contributing to minimizing system risks and operational costs [30]. PHM for IRs can be categorized into three main aspects: process control and monitoring, fault diagnosis, and Remaining Useful Life (RUL) prediction.

Beyond robotic systems, Kang et al. developed a multimedia knowledge-based bridge health monitoring method using DT, showing that physical-virtual synchronization can support health state assessment and maintenance decision-making [31]. In terms of manufacturing process control and monitoring, real-time data-driven DT has achieved satisfactory performance [32]. Li et al. extended the DT concept to the description of robot-environment interaction and proposed a novel semantic-enhanced DT system to implement more accurate and fine-grained real-time monitoring of robot-environment interaction. The exteroceptive and proprioceptive modalities of robots were used as mechanisms for state monitoring and semantic inference associated with interactions [33]. Touhid et al. constructed a cloud-based DT for remote monitoring and control of a robotic assembly system, which featured the integration of modularized cloud intelligence, DT modeling, and DT visualization to achieve enhanced remote accessibility [34]. Zhu et al. proposed and developed an intelligent monitoring system for the robotic milling process based on transfer learning and DT to address the practical challenges, including complex and time-consuming robot trajectory planning, insufficient monitoring, and lack of three-dimensional visualization [35]. Furthermore, the close integration of eXtended Reality (XR) technologies with DT has also been employed to enhance the effectiveness of manufacturing process control and monitoring [36]. Williams et al. presented an Augmented Reality (AR)-assisted DT solution. The solution enabled users to predict the battery charge of the mobile robot using historical data as input, and provided a visual representation of the mobile robot's movements through an AR device that displayed the digital data [37]. Li et al. proposed a novel multi-robot collaborative manufacturing system with human-in-the-loop control by leveraging the cutting-edge AR and DT techniques. The system supported three interactive AR-assisted DT modes, including the real-time motion control, planned motion control, and robot monitoring mode [38]. Calandra et al. presented a DT and XR-based telepresence for collaborative robot programming [39].

With respect to the fault diagnosis for IRs, data-driven models and algorithms are widely used for health assessment and anomaly detection [40]. Wang et al. implemented a DT framework for robot health assessment by embedding physical information neural networks into the DT and using virtual signals from the twin to verify real signals for anomaly detection [41]. Yin et al. introduced a parallel serial belief rule base model for assessing the health status of IR systems to ensure IR function correctly and implemented adequate safety strategies [42]. Long et al. proposed an unsupervised anomaly detection framework that fuses statistical analysis and graph neural networks for IR clusters. The method balances global collaborative modeling and local behavior recognition, and is particularly suitable for anomaly monitoring scenarios in large-scale robotic systems [43]. Kayan et al. proposed an IoT-based cyber-physical anomaly detection system that can detect motion-based behavioral changes in an industrial robotic arm. The machine learning methods were applied to real-time data collected from an edge development board attached to the robotic arm [44]. Yang et al. proposed a novel method merging imputation and feature collaboration for robotic anomalies, where collaborative and temporal graph attention were introduced for nuanced robotic analysis [45]. Yun et al. proposed the autoencoder-based anomaly detection framework for IRs using an internal sound sensor. The reconstruction errors of the autoencoder were compared to distinguish the abnormal status from normal status [46].

In addition, as a key element of PHM, RUL prediction assesses the remaining operational time before a system requires maintenance or replacement. This term RUL is widely applied to IR systems and their components. Gao et al. proposed a degradation-aware RUL prediction scheme for IRs using multiscale temporal memory transformer framework [47]. Ren et al. presented a novel RUL prediction method for robot rotary vector reducers with multi-depth network and multi-kernel support vector data description [48]. Yuan et al. proposed an in-situ RUL prediction method via a lightweight multiscale attention deep network and current signal for the harmonic reducer of industrial robots [49]. Kolvig-Raun et al. presented a knowledge-based predictive model aimed at approximating the end of life for a robotic joint, enabling the prediction of its RUL with respect to the designated load case [50]. Additionally, considering that mobile robots rely on onboard batteries, the term State of Charge (SoC), which is similar to RUL, provides information about the remaining energy in the battery and is essential for battery management [51]. In terms of SoC prediction, Shokri et al. deployed a novel approach based on the moving horizon estimation algorithm [52]. Rico-Melgosa et al. developed and optimized a feed-forward neural network to predict relative energy consumption, which provided accurate, repeatable, and interpretable energy predictions suitable for integration into DT platforms [53].

Based on the above, data-driven models and algorithms have significantly advanced the field of PHM [54]. The acquisition

and effective analysis of status data serve as the foundation for state prediction and stable operation of manufacturing systems [55]. A recent HRC review also emphasizes trust, explainability, and transparent decision support in human-in-the-loop operation, reinforcing the need to present PHM outputs in an operator-interpretable form [56]. However, most existing PHM studies focus on individual robots, specific components, or isolated degradation modes, and they do not directly address mission-level remaining executable capacity for heterogeneous MOMA systems. In particular, for a shared-battery MOMA, navigation workload, manipulation workload, task phase, battery behavior, and fault states are coupled during mission execution. Therefore, conventional RUL, SoC, or energy-estimation metrics are not sufficient to provide an operator-interpretable endurance indicator for human-in-the-loop monitoring. To address this gap, this work introduces RUC as a mission-level endurance metric and integrates telemetry synchronization, data-driven prediction, fault-aware monitoring, and human-in-the-loop decision support within a DT-based PHM framework.

2.3 Motivation

Given the heterogeneous characteristics of MOMA, we propose a DT-based endurance prediction and health monitoring framework for MOMA. Our research differs from existing studies in the following aspects:

Unlike conventional DT frameworks designed for single-robot platforms, the proposed framework addresses challenges in hybrid navigation-manipulation environments, where heterogeneous sensing frequencies, multi-functional behavior coupling, and shared energy consumption must be jointly considered to maintain a coherent system-level representation. A multi-layer DT architecture is presented to integrate telemetry data and edge computing, human-in-the-loop interactive control and task monitoring, as well as fault awareness and endurance prediction. Meanwhile, compared with traditional methods of using historical data for analysis [38], a real-time telemetry data synchronization pipeline is designed to align and aggregate heterogeneous sensing streams from the mobile robot and collaborative manipulator into a coherent, time-consistent representation.

In addition, considering that heterogeneous MOMA perform different functional roles while sharing the same battery source, conventional indicators such as SoC, RUL, or remaining operating time are insufficient to describe task-level residual capability. Therefore, this study introduces the RUC metric, which represents the number of executable operational cycles remaining from the current cycle until the battery level first reaches the low-battery abort threshold. Based on this definition, a mission-level endurance prediction model is constructed to estimate residual task-execution capacity under different MOMA task loads. The detailed formulation of RUC is provided in Sect. 4.2.1.

3 The Proposed Framework

3.1 Components of the Proposed Framework

The proposed framework comprises four functional layers: DT layer, physical layer, dashboard layer, and machine learning layer. As shown in Fig. 1, these layers integrate data acquisition, edge computing, pre-trained predictive models, VR-based visualization, and human-in-the-loop supervisory control. The framework advances DT technology beyond visualization-oriented monitoring toward a PHM-guided paradigm. Through multi-type, multi-frequency, and time-synchronized telemetry streams, it enables integrated mission-level endurance prediction and health monitoring for the MOMA platform. The following will describe the four-layer framework in detail.

(1) DT Layer

The DT layer serves as the integration and decision-making core of the framework. It consolidates heterogeneous telemetry streams from the mobile robot and manipulator, synchronizes multi-rate sensing data, and maintains a time-consistent virtual representation of the MOMA system. In addition, the DT layer establishes a unified digital space for health monitoring data and the endurance prediction model, facilitating the fusion of physical execution with predictive state analysis. By leveraging bidirectional communication between the virtual and physical spaces, the DT layer transmits predictive alerts and anomaly notifications to the visualization interface, thereby supporting reactive control of the MOMA via human-in-the-loop decision-making.

(2) Physical Layer

The physical layer comprises the MiR100 mobile robot and the UR3 collaborative manipulator operating in the physical space. It serves as the foundation for functional mission execution and telemetry data acquisition. During mission execution, the physical layer continuously exposes operational and safety-related information required for monitoring, prediction, and supervisory control.

(3) Dashboard Layer

The dashboard layer provides the human–robot interaction interface. Through synchronized visualization components, operators access real-time telemetry data and endurance metrics such as RUC, system alarms, and mission status information. It shares output information from the DT layer with operators, who can issue commands through a validated control pipeline, thereby enabling human-in-the-loop decision-making.

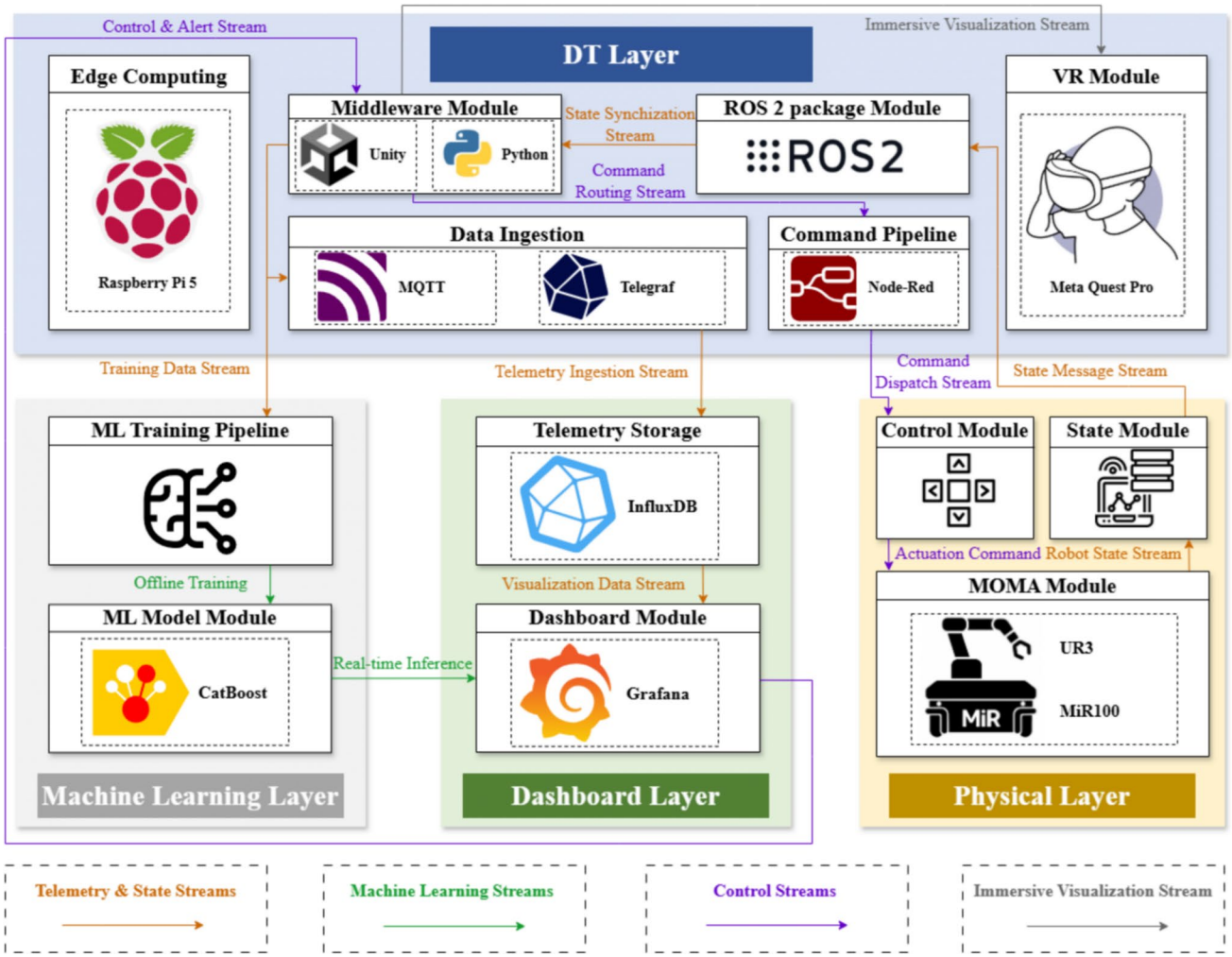


Fig. 1 DT-based framework of endurance prediction and health monitoring for MOMA

(4) Machine Learning Layer

The machine learning layer acts as the predictive analytics engine of the framework. It supports data preprocessing, feature engineering, model training, and real-time inference for mission-level endurance estimation. A validated CatBoost regression model is deployed online to continuously generate RUC predictions.

3.2 Data Flow Analysis in the Proposed Framework

With the overall conceptual organization of the proposed framework established, the intra-layer components, and the intra- and inter-layer data flows are shown in Fig. 2.

At the physical layer, navigation states, joint dynamics, battery status, and mission information from the MiR100 mobile robot and UR3 collaborative manipulator are transmitted to the DT layer and machine learning layer via standardized communication protocols and ROS 2-based interfaces. Conversely, validated control commands are sent back to the robots through the communication channels, forming a closed-loop interaction between the virtual and physical spaces.

In the DT layer, edge-level nodes interface directly with the MiR100 and UR3 robots to normalize heterogeneous messages and coordinate device-level communication. Telemetry streams are managed via a lightweight publish-subscribe mechanism and ingested into a time-series storage backend for downstream visualization and analytics. Meanwhile, before dispatching verified commands to the appropriate robot interfaces, a structured command processing pipeline handles operator inputs by extracting parameters, validating commands, and applying safety filters.

In the machine learning layer, raw telemetry is processed by specialized training and inference modules that incorporate navigation trajectories, manipulation forces, mission descriptors, and battery-related signals. Following the evaluation

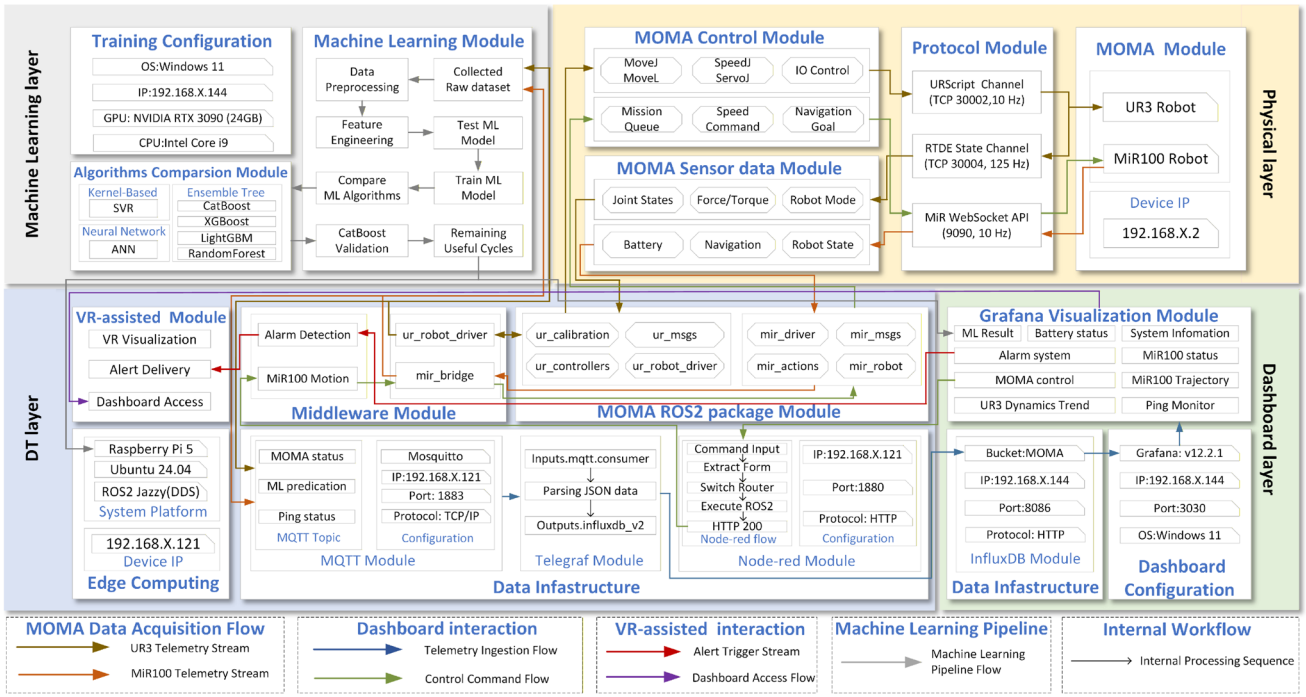


Fig. 2 Data flow analysis in the proposed framework

of multiple candidate algorithms during model selection, the optimal model is deployed for real-time inference. This deployed model continuously produces mission-level predictive endurance metrics, including RUC, which are integrated into the DT layer and forwarded to the dashboard layer.

At the dashboard layer, the time-series database is integrated with adaptive visualization components to present real-time robot states, endurance prediction metrics, anomaly events, and mission transitions in an intuitive manner. These functionalities are further enhanced by a VR-assisted interface, which provides immersive DT visualization and context-aware alerting.

Compared with the compositional overview presented in Fig. 1, Fig. 2 elaborates on the runtime workflows and module-level interactions, including communication drivers, routing logic, telemetry parsing, and cross-layer coordination. The conceptual architecture and implementation mechanisms enable the framework to function as a fully deployable end-to-end DT system.

4 Endurance Prediction and Health Monitoring for MOMA

This section outlines the technical workflow for developing a DT-based endurance prediction and health monitoring framework tailored to a heterogeneous MOMA platform, as shown in Fig. 3. The workflow begins with the construction of a synchronized data acquisition and ingestion pipeline that collects heterogeneous telemetry data from the MOMA platform. Next, leveraging the extracted telemetry features, a data-driven endurance prediction model is developed to estimate mission-level RUC. Finally, the validated prediction model is integrated into a VR-assisted virtual environment, enabling human-in-the-loop decision-making. In addition, the meanings for the defined notations are listed in Table 2 for reference.

4.1 Telemetry Data Acquisition and Processing

A unified and temporally consistent telemetry data stream is constructed, as shown in Fig. 4. The pipeline fuses heterogeneous robot telemetry at the edge into a unified MOMA feature set and forwards the synchronized stream to downstream modules for machine learning, on-edge inference, and visualization.

4.1.1 Telemetry Data Acquisition

All telemetry data acquisition is centralized on an edge node based on a Raspberry Pi 5 running Ubuntu 24.04 and ROS 2 Jazzy. The node acts as the system's aggregation point by hosting the required MOMA communication endpoints, enabling heterogeneous telemetry streams to be accessed and forwarded through a single interface. To enable cross-device fusion, the edge node enforces a common time base. Incoming packets are timestamped using a synchronized Unix clock, so that data

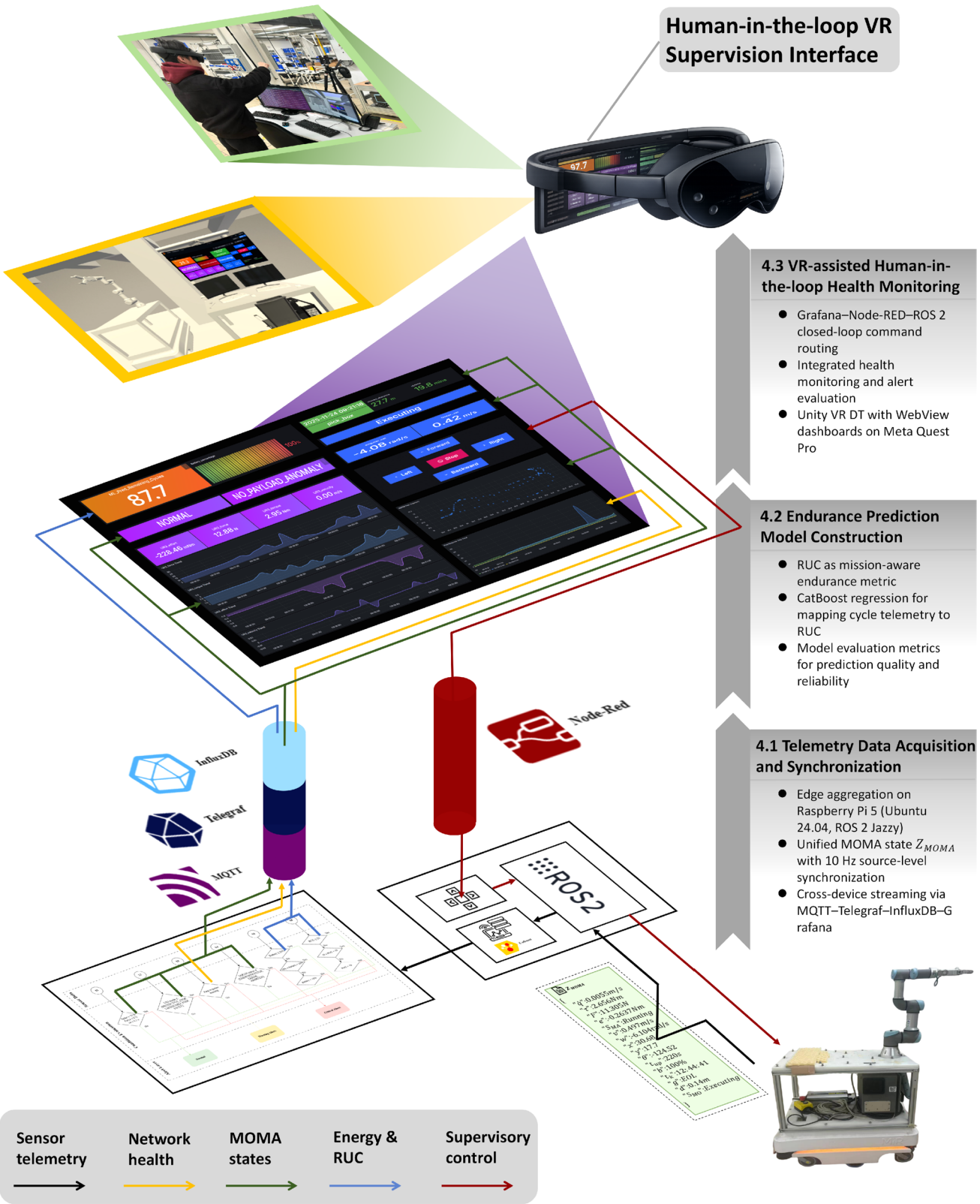


Fig. 3 Technical development workflow

Table 2 Notation definitions

Symbol	Definition
$a_{i,g}$	A normalized temporal indicator that quantifies the relative position of mission level g within cycle i
$B(t_B)$	Battery state of charge (SoC) evaluated at time instant t_B
$B_{i,g}$	Battery state of charge at the beginning of mission g within cycle i
B_{abort}	Low-battery abort threshold
c	Constant scalar prediction used in the initialization step
$e_{i,g}$	UR3 joint effort during mission g of cycle i
F_0	Constant initialization of the model, obtained by minimizing the empirical loss over the training set
$F_{i,g}$	UR3 joint force during mission g of cycle i
$F(x_{i,g})$	Final ensemble predictor mapping the feature vector to the estimated RUC
$F_t(x_{i,g})$	Prediction of the ensemble after t iterations for sample $x_{i,g}$
$f_t(x_{i,g})$	Output of the $t-th$ tree for input $x_{i,g}$
$I_{i,gbon}$, $I_{i,glin}$	Binary indicator defined as $B_{i,g} = B_{i,g+1}/B_{i,g} \neq B_{i,g+1}$
i_{end}	Cycle index at which the abort condition is met
J	Number of leaf regions (leaves) in the $t-th$ decision tree
$\dot{m}_{i,g}$	Moved distance during mission g of cycle i
N	Number of training samples
P	Total number of cycles in the dataset
$\dot{q}_{i,g}$	UR3 joint velocity during mission g of cycle i
Q	Total number of mission stages per cycle
$\mathbb{R}^{22}$	22-dimensional real vector space
$r_{t,(i,g)}$	Pseudo-residual (negative gradient) for the sample of mission g of cycle i at boosting iteration t
$R_{t,j}$	The $j-th$ leaf region of the $t-th$ tree
$\{R_{t,j}\}_{j=1}^{J}$	Partition of the feature space induced by the $t-th$ tree into J leaf regions
$RUC_{i,g}$	Remaining Useful Cycles at mission g of cycle i (ground-truth label)
$\widehat{RUC}_{i,g}$	CatBoost-predicted RUC produced on the edge node
t	Boosting iteration index
t_B	Time instant at which the battery SoC is evaluated
$t_{i,g}^{start}$, t_i^{end}	Start/End time (uptime) of cycle i
$t_{i,g}^{start}$	Start time (uptime) of mission level g within cycle i
$t_{i,g}$	System uptime at the start of mission g within cycle i
T	Total number of boosting iterations
$\|v_{i,g}\|$	Euclidean norm of combined linear and angular velocities during mission g of cycle i
$v_{i,g}$	Linear velocity of the MiR100 during mission g of cycle i
$w_{t,j}$	Optimal leaf value assigned to region $R_{t,j}$ obtained by minimizing the loss over samples
$x_{i,g}$	Telemetry-derived feature vector constructed for mission level g within cycle i from the MOMA
$y_{i,g}$	Regression target for mission g of cycle i
$\overline{y}$	The sample mean of the ground-truth targets
$\widehat{y}_{i,g}$	The predicted value for the sample $x_{i,g}$
η	Learning rate scaling the contribution of each newly added tree
$\omega_{i,g}$	Angular velocity of the MiR100 during mission g of cycle i
$\tau_{i,g}$	UR3 joint torque during mission g of cycle i

originating from different protocols and sampling rates can be aligned to the same temporal reference. Beyond aggregation, the edge node runs the endurance-prediction model and produces real-time predictions of the MOMA RUC directly at the edge, reducing end-to-end latency compared with cloud-based inference. In parallel, the enforced common time base provides a consistent time reference for downstream synchronization, predictive inference, and DT integration.

The MiR100 mobile robot and the UR3 collaborative manipulator interface with the system through the

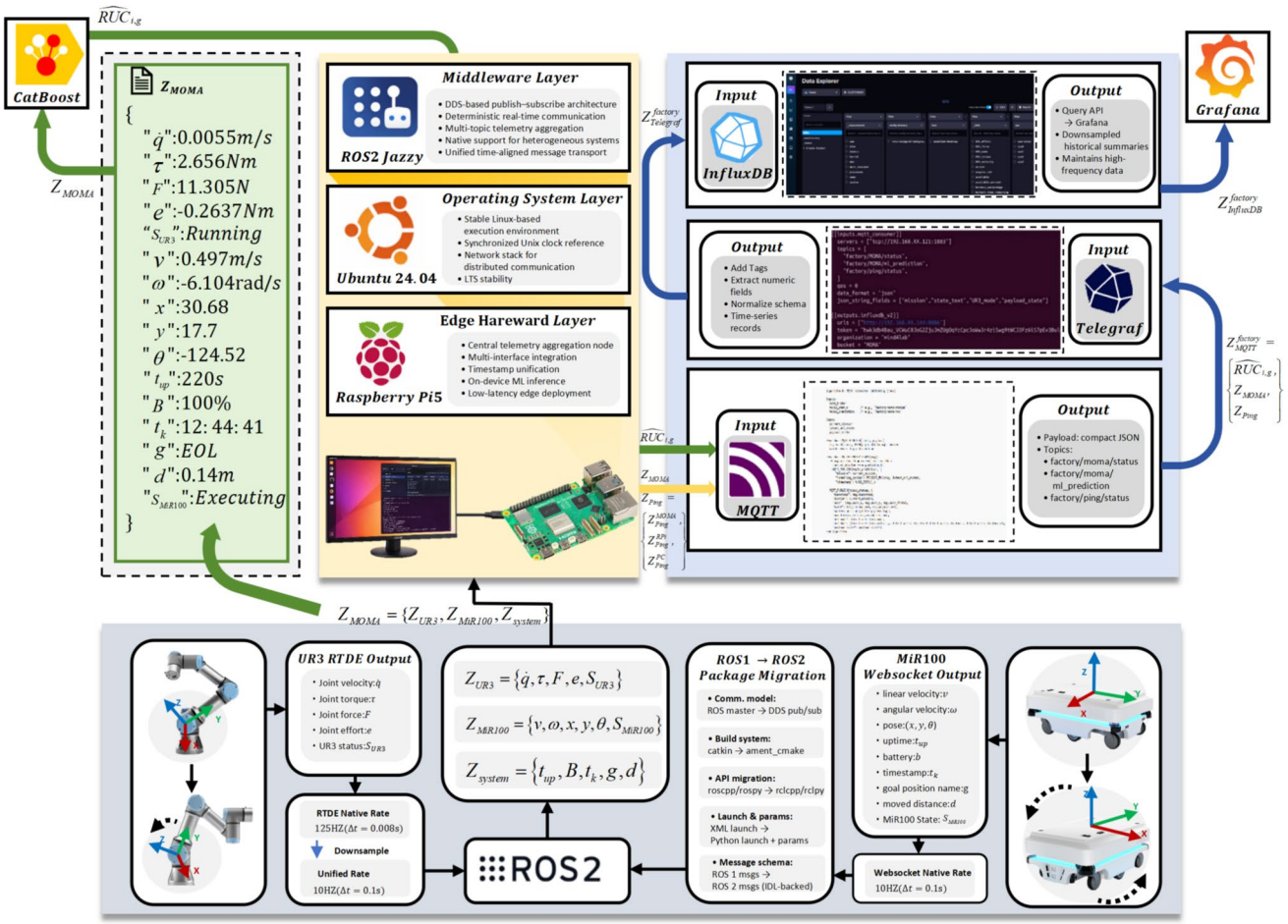


Fig. 4 Telemetry data acquisition and processing from heterogeneous MOMA

edge-computing middleware. Their heterogeneous telemetry data are organized into a unified MOMA feature set $Z_{MOMA} = \{Z_{UR3}, Z_{MiR100}, Z_{system}\}$, which serves as the common input to subsequent data processing.

Since the MiR100 does not provide an official ROS 2 driver, we ported the original MiR100 ROS 1 packages to ROS 2 to maintain a consistent ROS 2-based communication framework on the edge device. The MiR100 publishes navigation- and energy-related telemetry data at 10 Hz via its WebSocket API. On the edge device, a dedicated ROS 2 node parses the received JSON payloads and republishes them as structured ROS 2 messages. The MiR100 sensor data are represented as Z_{MiR100} and Z_{system}, as shown below:

$$Z_{MiR100} = \{v, \omega, x, y, \theta, S_{MiR100}\} \tag{1}$$

where, v and ω are the linear and angular velocities, x, y and θ denote the planar pose of the mobile base, S_{MiR100} summarizes the status of MiR100.

$$Z_{system} = \{t_{up}, B, t_k, g, d\} \tag{2}$$

where, t_{up} is the uptime,B is the battery percentage, t_k is the synchronized Unix timestamp, g denotes goal position name and d denotes current moved distance.

In parallel, the UR3 manipulator exposes an RTDE channel streaming high-rate signals at 125 Hz. Z_{UR3} is defined as:

$$Z_{UR3} = \{\dot{q}, \tau, F, e, S_{UR3}\} \tag{3}$$

where, $\dot{q}$ is the joint velocity, τ is the joint torque, F is the joint force, e is the joint effort and S_{UR3} denotes the robot status. ROS 2 nodes on the edge device parse both streams and map them into the unified schema, enabling a coherent representation across the mobility and manipulation subsystems.

4.1.2 Telemetry Data Synchronization

A coherent telemetry data pipeline requires the MiR100 and UR3 measurements to share a common sampling rate. To avoid multi-rate buffering and downstream aggregation on

the edge node, we enforce sampling-rate alignment at the source by down-sampling the UR3 stream to 10 Hz within the ROS 2 UR3 driver running on the Raspberry Pi 5. Specifically, the driver's publishing rate is configured to publish UR3 data at a fixed 10 Hz interval, matching the MiR100 rate. As a result, the two streams can be fused into a single time-aligned sample $Z_{MOMA}(t_k)$ without additional resampling or interpolation. This design keeps the DT visualization responsive and reduces the preprocessing overhead for learning-based modules, while maintaining the acquisition stack lightweight enough to run on the Raspberry Pi 5. The logged dataset therefore forms a temporally consistent stream that can be used directly for DT visualization and machine learning.

4.1.3 Telemetry Data Stream

To stream synchronized telemetry data to the visualization layers, we deploy a cross-device pipeline spanning the Raspberry Pi 5 and the Personal Computer (PC) workstation. The edge node packages selected ROS 2 telemetry data, on-edge inference outputs and device network-health metrics, then forwards them to the monitoring stack (MQTT-Telegraf-InfluxDB-Grafana) for real-time dashboarding.

On Raspberry Pi 5, MQTT ingests three synchronized streams: $\widehat{RUC}_{i,g}$, Z_{MOMA} and Z_{Ping}. Here, $\widehat{RUC}_{i,g}$ is the RUC estimate produced by the CatBoost model deployed on the edge node, Z_{MOMA} denotes the unified robot telemetry data vector and $Z_{Ping} = \left\{ Z_{Ping}^{MOMA}, Z_{Ping}^{RPi}, Z_{Ping}^{PC} \right\}$ aggregates network-health measurements for MOMA, the Raspberry Pi 5 and the PC. These inputs are encoded into compact JSON payloads and published under a structured topic namespace yielding the MQTT stream $Z_{MQTT}^{factory}$.

The MQTT stream $Z_{MQTT}^{factory}$ is then ingested by Telegraf on Raspberry Pi 5. Telegraf subscribes to the above topics, attaches tags, extracts numeric fields from the JSON payloads, normalizes them into a consistent measurement schema, and converts the messages into time-series records. The resulting output, denoted $Z_{Telegraf}^{factory}$, represents schema-normalized telemetry data ready for database ingestion.

Finally, $Z_{Telegraf}^{factory}$ is written to InfluxDB on the PC workstation, producing $Z_{InfluxDB}^{factory}$. InfluxDB exposes a query API for Grafana, retains high-frequency data to support real-time diagnostics, and stores downsampled historical summaries for long-term analysis. Grafana queries $Z_{InfluxDB}^{factory}$ to render synchronized dashboards and DT-relevant views, thereby completing the edge-to-workstation telemetry data pipeline. The Grafana-based dashboards are further streamed to the VR interface through a Unity WebView application deployed on the Meta Quest Pro, enabling immersive access to DT information without modifying the underlying data pipeline.

By deploying MQTT and Telegraf on the edge device while hosting the time-series database InfluxDB and Grafana on the workstation, the pipeline keeps acquisition lightweight on the robot side and reserves storage and rendering for the monitoring side.

4.2 Endurance Prediction Model Construction

4.2.1 Problem Formulation and RUC Definition

The objective of this study is to estimate how many operational cycles the MOMA can still complete before reaching the low-battery abort threshold under the current battery state and workload condition. Therefore, the proposed RUC is positioned as a mission-level and task-aware endurance metric.

As summarized in Table 3, conventional indicators describe battery state, time-to-threshold, energy consumption, or battery health, but they do not directly answer how many mission cycles the MOMA can still complete. RUC addresses this gap by converting endurance prediction into a mission-level task-completion estimate. Since MOMA endurance depends on both SoC and navigation-manipulation workloads, RUC provides a more intuitive decision variable for human-in-the-loop supervision, task scheduling, and charging decisions.

The MOMA is powered by an industrial-grade lithium-ion battery module manufactured by Smart Battery Solutions GmbH. The battery model 7INR19/66–12 has a nominal voltage of 25.2 V and a nominal capacity of 31.2 Ah, corresponding to a total energy of 785 Wh.

Under an idealized operating assumption, the theoretical upper bound of the system uptime can be derived from the nominal battery energy and the average power consumption of the main subsystems. E_{bat} denotes the nominal battery energy and P_{tot} is the total average power consumption of the MOMA system. The ideal operating time T_{ideal} is given by:

$$T_{ideal} = E_{bat}/P_{tot} \tag{4}$$

According to the UR3 technical specifications, the typical power consumption of the UR3 during standard operation is $P_{UR3} = 125$ W. Prior work on the MiR100 reports an average power consumption of $P_{MiR100} = 34.45$ W at a nominal speed of 1.5 m/s under typical internal logistics conditions. Therefore, the total average power consumption is $P_{tot} = P_{UR3} + P_{MiR100} = 159.45$ W.

Using the nominal battery energy $E_{bat} = 785$ Wh originated from the manual provided by the manufacturer, the idealized maximum operating time is obtained as $T_{ideal} = E_{bat}/P_{tot} \approx 4.9$ h.

This value represents an upper bound on the achievable uptime and is computed under the assumption of a fresh battery without capacity degradation. Dynamic power

Table 3 Comparison of endurance-related indicators

Metric	Purpose
SoC	How much battery charge remains?
RUL	How long before the system reaches its end-of-life or failure threshold?
Remaining operating time	How much operating time remains?
Energy throughput	How much energy has been consumed?
state-of-health-aware energy estimation	What is the usable battery capacity considering aging?
RUC	How many mission cycles can be completed before threshold?

fluctuations, auxiliary loads, and battery aging effects are not considered in this idealized estimate. In practice, however, the actual uptime of the MOMA system can deviate significantly from this idealized upper bound.

To further analyze this impact, battery discharge trajectories under four representative operating conditions are collected, as shown in Fig. 5. The results indicate that even at similar battery charge levels, different task configurations can lead to significantly different discharge rates.

Moreover, the longest uptime observed in all experiments was approximately 3 h, which is noticeably shorter than the ideal uptime of about 4.9 h estimated based on the battery's rated specifications. This decrease is mainly attributed to battery aging, as the battery packs used in the MOMA system have been in service for many years. These observations suggest that relying on SoC alone is insufficient to characterize the remaining operational endurance.

To address this limitation, we define a mission-level updated RUC metric (in cycle units) that decouples prediction from specific tasks by fusing heterogeneous telemetry dynamics with system context, enabling task-agnostic endurance assessment across diverse task definitions.

We define missions as the smallest execution units, and assume that one case is indexed by a mission level label $g \in \{g_1, g_2, g_3, g_4\}$, where g_1 denotes the designated cycle-boundary stage, as shown in Fig. 6.

A cycle is constructed as an ordered mission sequence that starts and ends at g_1, each return to g_1 marks a cycle boundary. An operational cycle is mission-defined: its duration depends on the selected task pattern and is determined by the time required to complete one closed route and manipulation loop. We index consecutive operational cycles by $i \in \{1,2,...,P\}$, and denote respectively the start and end uptime of a cycle t_i^{start} and t_i^{end}. Within cycle i, let $t_{i,g}^{start}$ denote the start uptime of a specific mission-level g within cycle i. To quantify the temporal evolution within a cycle, the mission-level g within cycle i progress factor $a_{i,g}$ is defined as:

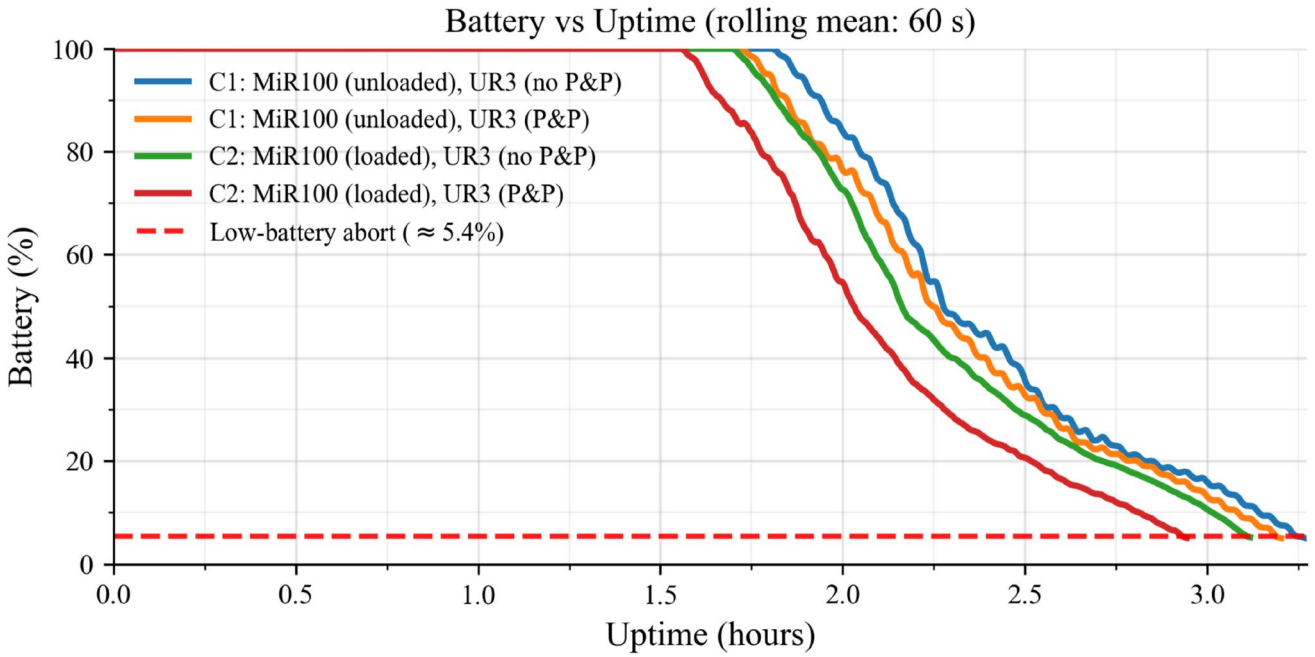


Fig. 5 Battery discharge profiles of MOMA under four mission configurations

$$a_{i,g} = \frac{t_{i,g}^{start} - t_i^{start}}{t_i^{end} - t_i^{start}} \in [0,1) \tag{5}$$

A task specifies the mission set and ordering used to form the cycle, and different tasks may adopt different mission categories and sequences. As shown in Fig. 6, Task1 repeats the loop $g_1 \rightarrow g_2 \rightarrow g_3 \rightarrow g_4 \rightarrow g_1$, whereas Task2 follows $g_1 \rightarrow g_4 \rightarrow g_2 \rightarrow g_3 \rightarrow g_1$. Based on these definitions, we predict mission-level RUC, updated at each mission and thus potentially fractional, enabling generalization across different task definitions.

$B(t_B)$ denotes the battery state of charge at time t_B, and $B(t_B) \in [0,100\%]$. For two time instants $t_B = 1$ and $t_B = 2$, represent the first and second time instants used to illustrate identical SoC can correspond to different remaining durations, the same battery level $B(t_B = 1) = B(t_B = 2)$ may correspond to substantially different remaining mission durations, depending on motion intensity and manipulation load. This ambiguity indicates that battery percentage alone does not provide a reliable estimate of the remaining operational capability and motivates the introduction of a cycle-based endurance metric that reflects mission-level execution capacity.

The index i_{end} represents the cycle at which the abort condition is reached, as follow:

$$i_{\text{end}} = \min\left\{ i \mid \min_g B_{i,g} \le B_{\text{abort}} \right\} \tag{6}$$

where, $B_{i,g}$ denotes the smoothed battery percentage evaluated at the start of mission level g within cycle i. B_{abort} denotes the low-battery abort threshold, and $B_{\text{abort}} \approx 5.4\%$. When the battery level reaches this threshold, the current discharge process is considered terminated and the RUC is set to zero.

The $RUC_{i,g}$ at mission level g within cycle i is then defined as:

$$RUC_{i,g} = i_{end} - i - a_{i,g} \tag{7}$$

Based on the above definitions, at each mission-level g, a telemetry data-derived feature vector $x_{i,g} \in \mathbb{R}^{22}$ from synchronized MOMA measurements is constructed. Each feature is computed over the duration of one mission. The feature space is designed to fuse heterogeneous telemetry dynamics with system context, and is organized as follow:

$$x_{i,g} = \left[x_{i,g}^{system}, x_{i,g}^{MiR100}, x_{i,g}^{UR3}, x_{i,g}^{task} \right] \tag{8}$$

where, the system-context $x_{i,g}^{system} = [B_{i,g}, t_{i,g}, \dot{m}_{i,g}]$ captures the operating baseline at the mission start. The telemetry-dynamics summarize mobility and manipulation behaviors over the mission duration:

$$x_{i,g}^{MiR100} = [\mu(v_{i,g}), \mu(\omega_{i,g}), \mu(\|v_{i,g}\|), \sigma(v_{i,g}), \sigma(\omega_{i,g})] \tag{9}$$

$$\begin{aligned} x_{i,g}^{UR3} = [&\mu(\tau_{i,g}), \mu(e_{i,g}), \mu(F_{i,g}), \mu(\dot{q}_{i,g}), \\ &\sigma(\tau_{i,g}), \sigma(e_{i,g}), \sigma(F_{i,g}), \sigma(\dot{q}_{i,g}), max(\tau_{i,g}), \\ &max(e_{i,g}), max(F_{i,g}), max(\dot{q}_{i,g})] \end{aligned} \tag{10}$$

Finally, $x_{i,g}^{task} = [I_{i,gbon}, I_{i,glin}]$ provides two binary phase indicators.

4.2.2 CatBoost-Based RUC Prediction Model

Before deploying the CatBoost, six candidate regression algorithms are benchmarked, and the results identified CatBoost as the best-performing model for RUC prediction (see Sect. 5.3.1).

Based on Sect. 4.2.1, each mission stage g yields a telemetry data feature vector $x_{i,g} \in \mathbb{R}^{22}$ and a regression target $y_{i,g} = RUC_{i,g}$. Figure 7 illustrates the internal architecture of the CatBoost-based RUC prediction model used to learn a non-linear mapping from $x_{i,g}$ to $y_{i,g}$ under the gradient boosting framework.

The telemetry data features are first organized into the input vector $x_{i,g}$ (Step ①), while the corresponding label $y_{i,g} = RUC_{i,g}$ is provided as the regression target (Step ①).

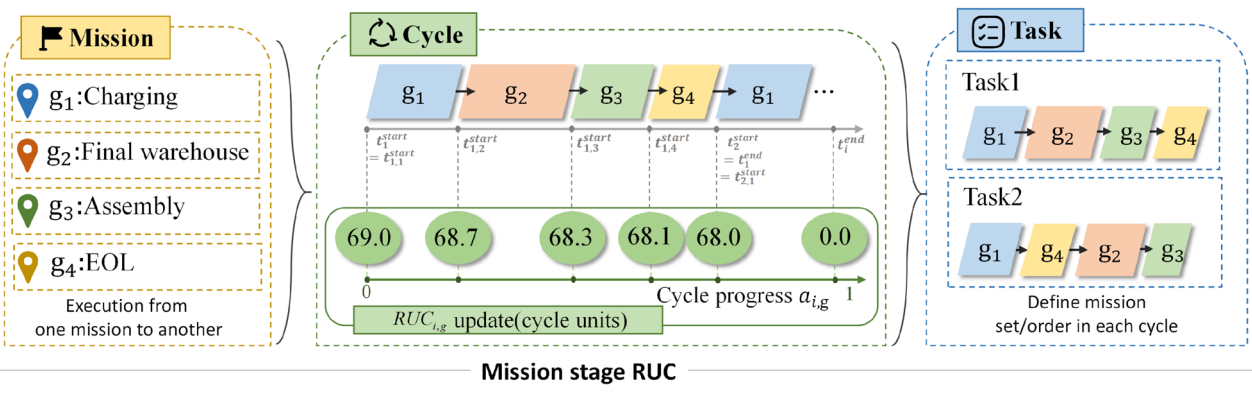


Fig. 6 Definitions of mission, cycle and task

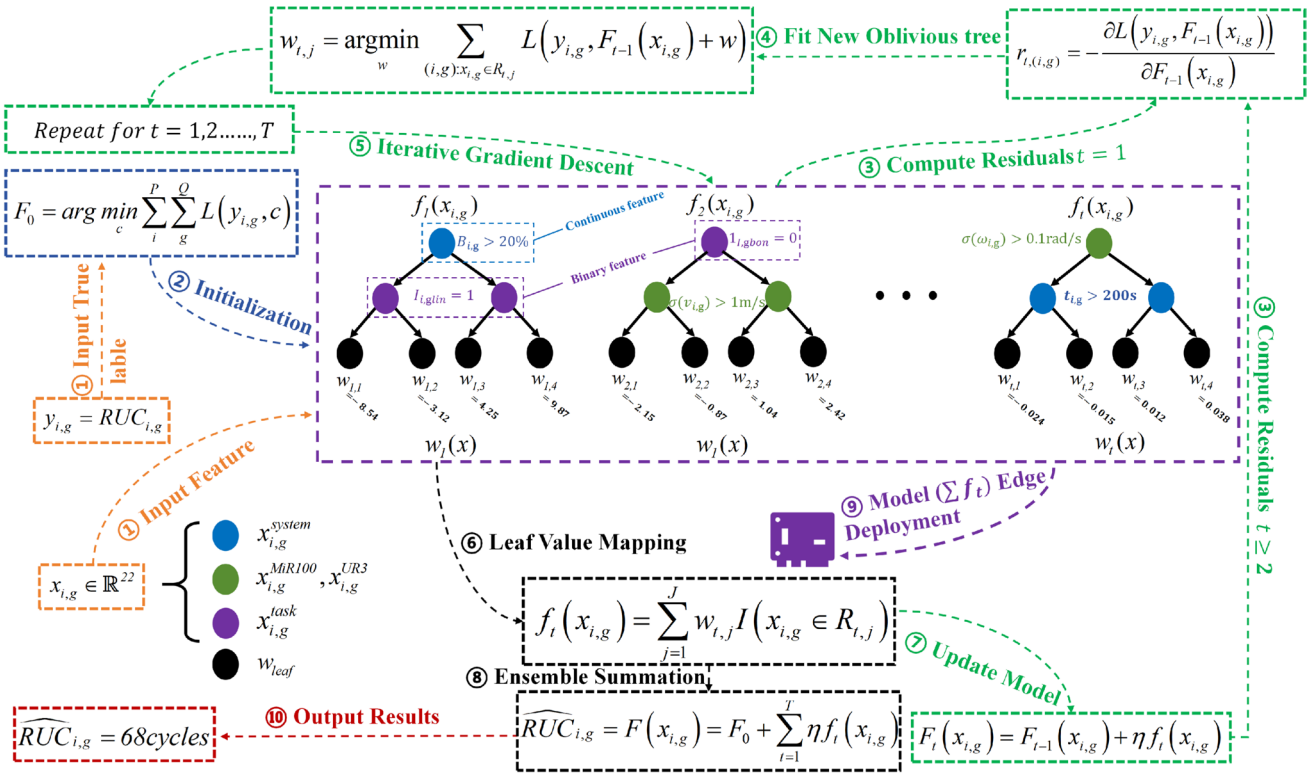


Fig. 7 Internal architecture of the CatBoost-based RUC prediction model

The model training starts with a constant initialization F_0 (Step ②), where the initial predictor is obtained by minimizing the empirical loss over the training set.

$$F_0 = \underset{c}{argmin} \sum_{i}^{P} \sum_{g}^{Q} L(y_{i,g}, c) \tag{11}$$

where, $N = P \times Q$ denotes the number of training samples and c is a constant prediction (scalar bias term). For the RUC regression task, the squared error loss is adopted and as shown in Eq. (12):

$$L(y_{i,g}, F(x_{i,g})) = (y_{i,g} - F(x_{i,g}))^2 \tag{12}$$

An additive ensemble of regression trees under the iterative boosting loop is constructed by CatBoost (Step ⑤). At each boosting iteration t, the current model prediction $F_{t-1}(x_{i,g})$ is used to compute the pseudo-residual $r_{t,(i,g)}$ (negative gradient), which guides the construction of the next tree (Step ③), as shown in Eq. (13):

$$r_{t,(i,g)} = -\frac{\partial L(y_{i,g}, F_{t-1}(x_{i,g}))}{\partial F_{t-1}(x_{i,g})} \tag{13}$$

A new oblivious decision tree is fitted based on this residual information (Step ④), resulting in a partition of the feature space into J leaf regions $\{R_{t,j}\}_{j=1}^{J}$. J denotes the number of leaves in the $t-th$ tree and $R_{t,j}$ is the $j-th$ leaf region of that tree. Given the learned tree structure, the optimal leaf value $w_{t,j}$ is obtained by minimizing the loss within each region as follows:

$$w_{t,j} = \underset{w}{\text{argmin}} \sum_{(i,g):x_{i,g} \in R_{t,j}} L(y_{i,g}, F_{t-1}(x_{i,g}) + w) \tag{14}$$

4.2.3 (14)

The resulting $t-th$ regression tree $f_t(x_{i,g})$, which outputs a real-valued prediction in the same scale as the target variable, can be expressed via the leaf-value mapping (Step ⑥):

$$f_t(x_{i,g}) = \sum_{j=1}^{J} w_{t,j} I(x_{i,g} \in R_{t,j}) \tag{15}$$

where, I denotes if the condition holds, the indicator function equals 1, otherwise equals 0.

The ensemble is updated by adding the newly learned tree scaled by the learning rate η (Step ⑦). After the update, F_t becomes the current predictor for the next iteration. The pseudo-residuals are then recomputed using $F_t(x_{i,g})$ (Step ③), as shown in Eq. (16), i.e., $r_{t+1,(i,g)}$ is computed with

respect to $F_t(x_{i,g})$ and Steps ④, ⑥, and ⑦ are repeated for $t = 2,3,...,T$ (Step ⑤).

$$F_t(x_{i,g}) = F_{t-1}(x_{i,g}) + \eta f_t(x_{i,g}) \tag{16}$$

After training, the final CatBoost model $F(x_{i,g})$ is obtained by aggregating all trees (Step ⑧), as shown in Eq. (17):

$$F(x_{i,g}) = F_0 + \sum_{t=1}^{T} \eta f_t(x_{i,g}) \tag{17}$$

The predicted remaining useful cycles $\widehat{RUC}_{i,g} = F(x_{i,g})$ is generated for each operational mission (Step ⑩). The learned ensemble can then be deployed at the edge for online inference (Step ⑨). Model performance is evaluated using the metrics defined in Sect. 4.2.3, i.e., the coefficient of determination (R^2), the Root Mean Squared Error (RMSE), and the Mean Absolute Error (MAE).

In addition to the standard gradient boosting procedure illustrated in Fig. 7. We specify binary indicators as categorical features in CatBoost (*cat_feature*) and are encoded using CatBoost's permutation-based target statistics within the ordered boosting scheme, which helps mitigate target prediction shift and improves training robustness.

4.2.4 Evaluation Metrics

Based on the formulation in Sect. 4.2.1, the prediction of RUC is cast as a supervised regression problem, where a learning model estimates the remaining number of operational cycles from telemetry data-derived features. To quantitatively evaluate the performance of the proposed RUC prediction models, three commonly used regression metrics are employed: R^2, RMSE, and MAE.

The R^2 is defined as follow:

$$R^2 = 1 - \sum_{i=1}^{P}\sum_{g=1}^{Q}(y_{i,g} - \hat{y}_{i,g})^2 / \sum_{i=1}^{P}\sum_{g=1}^{Q}(y_{i,g} - \bar{y})^2, \bar{y} = \frac{1}{PQ}\sum_{i=1}^{P}\sum_{g=1}^{Q} y_{i,g} \tag{18}$$

where, $y_{i,g}$ denotes the ground-truth $RUC_{i,g}$ at cycle i mission level g, $\hat{y}_{i,g}$ is the corresponding predicted value, and N (where $N = P \times Q$) is the total number of evaluation samples. $\bar{y}$ denotes the mean of the ground-truth RUC values. The R^2 score measures the proportion of variance in the target variable that is explained by the regression model, with higher values indicating better predictive performance.

The RMSE is defined as follow:

$$RMSE = \sqrt{\frac{1}{PQ}\sum_{i=1}^{P}\sum_{g=1}^{Q}(y_{i,g} - \hat{y}_{i,g})^2} \tag{19}$$

Equation (19) penalizes large prediction errors more strongly and reflects the overall deviation between predicted and true RUC values.

The MAE is as shown in Eq. (20):

$$MAE = \frac{1}{PG}\sum_{i}^{P}\sum_{g}^{Q}\left|y_{i,g} - \hat{y}_{i,g}\right| \tag{20}$$

Equation (20) provides a more interpretable measure of the average absolute prediction error in terms of the number of remaining cycles.

Together, these metrics offer complementary perspectives on model performance. While R^2, captures the goodness of fit and explanatory power of the model, RMSE and MAE quantify the absolute prediction accuracy in the RUC domain. All three metrics are reported in the following sections to comprehensively assess the effectiveness of the proposed machine learning models for RUC prediction.

4.3 VR-Assisted Human-in-the-Loop Health Monitoring

4.3.1 Supervisory Command Routing

The human-in-the-loop supervisory control mechanism enables safe and controlled manual intervention during mission execution. Through this mechanism, operators can issue navigation commands to the MiR100 directly from the dashboard. The supervisory control mechanism further supports remote operator intervention when the MiR100 is temporarily halted due to obstacle detection or safety triggers, allowing the mission state to be recovered and execution to be resumed without manual on-site intervention. The complete supervisory control chain spans the dashboard, Node-RED command processing, and ROS 2 actuation on the edge device.

The dashboard serves as the primary interface for operators and exposes a set of mission-level control commands, including forward, backward, left, right, and stop. Each interaction triggers an HTTP POST request that is transmitted to a Node-RED endpoint running on the Raspberry Pi 5 edge device.

Node-RED implements a streamlined command routing pipeline that transforms dashboard operations into

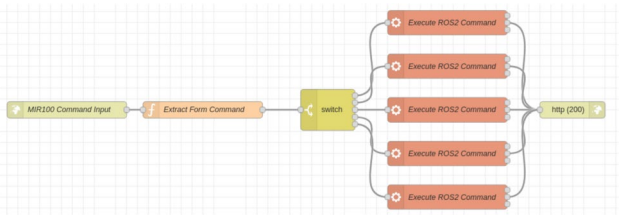


Fig. 8 Node-RED command-routing workflow

executable ROS 2 commands. The routing logic deployed on the edge device is illustrated in Fig. 8.

4.3.2 Alert Evaluation

The system incorporates an integrated monitoring and alerting workflow that continuously evaluates the operational health of MOMA by combining real-time telemetry data analysis and state evaluation. The overall alert-generation logic is illustrated in Fig. 9.

Multiple heterogeneous sensor data streams (S1-S6) are fed into the condition-evaluation rules. Specifically, the evaluation considers six categories of inputs, as summarized in Table 4. Based on the outcome of this evaluation, the system assigns one of three alert levels: *Normal, Warning,* or *Critical*. The resulting alert status is propagated to the dashboard embedded into VR interface, enabling operators to make timely interventions. The specific criteria defining each alert level are summarized in Table 5.

4.3.3 VR-Assisted Visualization

The VR-assisted visualization provides the primary interface through which operators perceive the real-time state, predicted endurance, and operational behavior of the MOMA. Grafana dashboards, the virtual space in Unity 3D environment, and a VR deployment on the Meta Quest Pro headset are integrated to form a continuously synchronized virtual representation of the physical system and enable immersive, human-in-the-loop supervision for health monitoring.

Grafana serves as the primary dashboard interface, as illustrated in Fig. 10. As summarized in Table 6, the dashboard is organized into multiple functional blocks corresponding to different aspects of the MOMA. Each functional block is implemented as a dedicated visualization panel linked to InfluxDB time-series queries, with configured retention policies and refresh intervals. This design enables low-latency visualization while preserving historical data for long-term trend analysis and degradation assessment.

To further enhance operator perception and spatial understanding, the visualization dashboard is extended to the Unity 3D environment with VR deployment, as shown in Fig. 11. The virtual scene reproduces the experimental workspace layout, operator station, and robot deployment areas, providing a spatially coherent context in which system states and operational information can be interpreted. Within this VR environment, selected Grafana dashboards are embedded via a WebView module, allowing real-time telemetry data, predictive endurance metric, and fault alerts to be visualized directly within the immersive scene.

By integrating Grafana-based dashboards, Unity 3D, and VR interfaces, the proposed framework closes the loop between perception, prediction, and supervision. Operators can visualize real-time telemetry data, monitor MOMA endurance, inspect force anomalies, and initiate corrective actions through a unified interface.

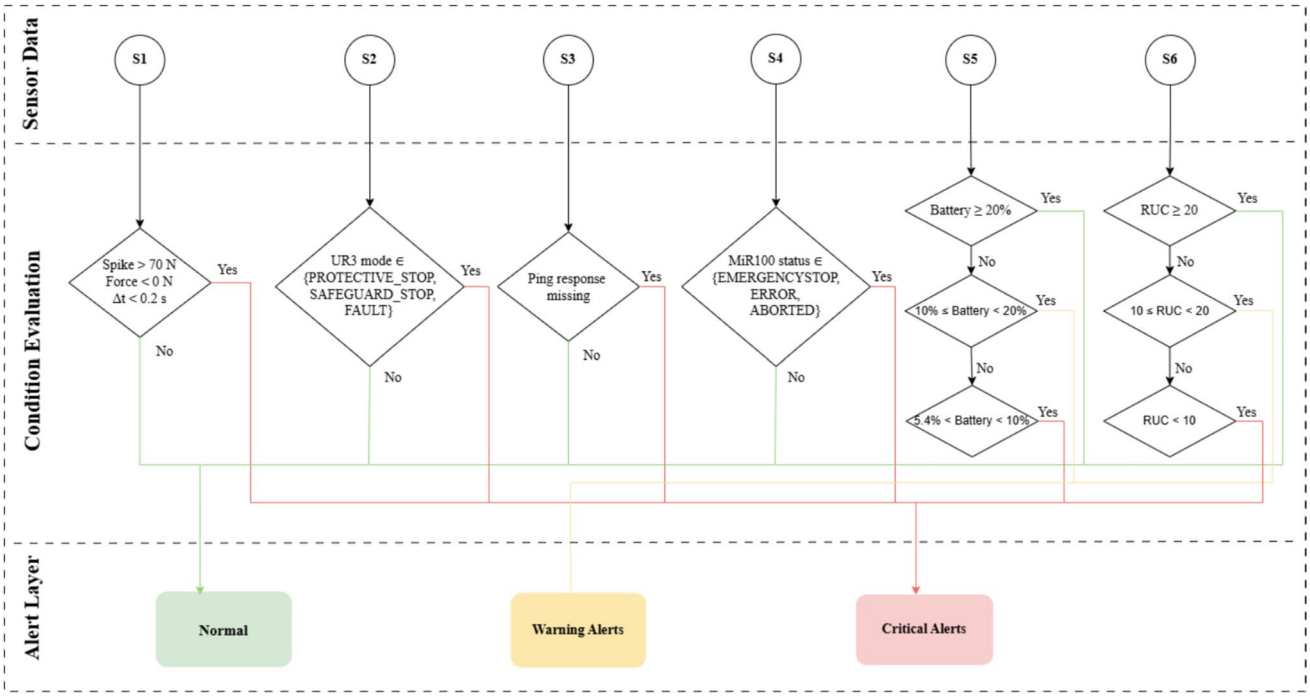


Fig. 9 Alert generation flow for MOMA health monitoring

Table 4 Input definitions for condition evaluation

ID	Sensor data streams	Description
S1	UR3 force (Fz)	Detects payload-drop events using spike-negative logic on the UR3 force signal
S2	UR3 safety mode	Reports UR3 safety status, including {PROTECTIVE STOP, SAFEGUARD STOP, FAULT}
S3	Network health	Monitors network connectivity and latency to identify communication anomalies
S4	MiR100 safety state	Reports MiR100 safety state, including {EMERGENCY STOP, ERROR, ABORTED}
S5	Battery level	Defines multi-level battery thresholds for alert severity classification
S6	Predicted RUC	Defines multi-level RUC thresholds for alert severity classification

Table 5 Alert level definitions

Alert level	Condition
Normal	All inputs within safe range
Warning	Battery 10–20% or RUC 10–20 cycles
Critical	Battery 5.4–10%, RUC < 10 cycles, or any safety stop

5 Experimental Results

5.1 Dataset Construction and Processing

This section provides a detailed explanation of the data preparation workflow, including the definition of experimental scenarios, dataset construction, data preprocessing, and feature engineering, which collectively form the foundation for subsequent model training and evaluation.

5.1.1 Experimental Scenarios

To ensure that the endurance prediction model captures realistic variations in energy consumption, two tasks are conducted on the MOMA platform. As illustrated in Fig. 12, the experiments follow a representative intralogistics workflow connecting the final warehouse, assembly station, EOL inspection area, and the charging station.

The overlaid trajectories represent a single continuous discharge cycle, starting from a fully charged battery (100%) and ending when the remaining capacity drops to about 5.4%. The execution of MOMA covers the full mobility-manipulation process in a continuous manner, rather than separate or segmented actions. During the cycle, the robot repeatedly follows the predefined routes across successive missions, which results in the densely overlapped trajectories shown in Fig. 12.

In this continuous operation, two navigation tasks are defined to represent typical shop-floor mobility patterns: a long straight trajectory and a route characterized by frequent turns and stop–go behavior. For each task, the MiR100 is evaluated under two load conditions: no payload (Case 1) and an approximately 42.5 kg payload (Case 2). In parallel, the UR3 manipulator remains powered on throughout the mission but is activated only at the designated pick-and-place stations, where it performs grasping and placement either without payload or with a 3 kg payload. Under payload conditions, the increased system mass leads to more pronounced

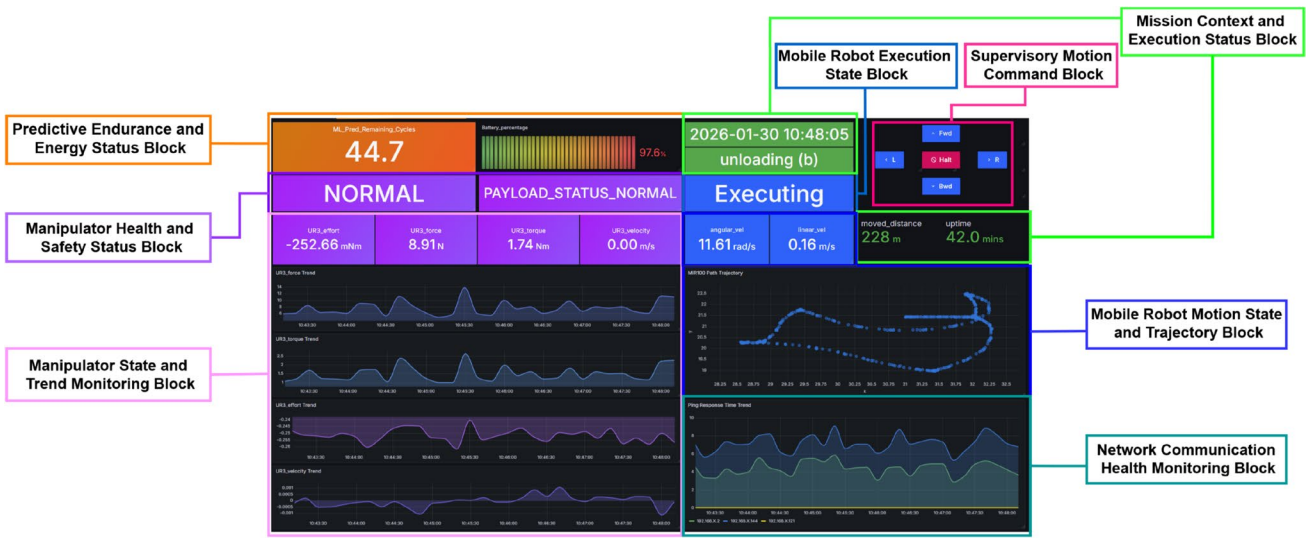


Fig. 10 The dashboard for MOMA health monitoring

Table 6 Descriptions of functional blocks in the dashboard

Functional block	Description
Predictive endurance and energy status	Combines machine learning-based RUC estimation with real-time battery state monitoring to support mission-level endurance awareness
Manipulator health and safety	Displays the real-time safety state of the UR3 manipulator and fault-related conditions such as payload drop detection
Manipulator state and trends	Visualizes instantaneous manipulation signals and their temporal evolution, including force, torque, effort, and joint velocity profiles
Mission context and execution	Summarizes mission-level information, including execution state, timestamp, moved distance, and system uptime
Mobile robot execution and safety	Reports the operational and safety state of the MiR100 mobile robot, including paused and safety-interrupted conditions
Supervisory motion commands	Provides high-level supervisory motion primitives to support safe human-in-the-loop recovery actions during abnormal situations
Mobile robot motion and trajectory	Visualizes instantaneous motion indicators together with the two-dimensional navigation trajectory within the DT
Network communication health	Monitors connectivity and latency between the MOMA system, edge device, and host PC to ensure reliable DT synchronization

transient deviations during turning and path transitions, resulting in greater trajectory dispersion than in the unloaded case, as well as higher traction demand and battery discharge.

The datasets generated by these experimental configurations can capture motion-navigation dynamics, load-related energy consumption, MOMA's battery percentage, system uptime, and moved distance. They are used for mission-level data aggregation and feature extraction. The final dataset contains 613 operational cycles and 5,157 mission-level samples, including 263 cycles and 2,756 samples in Case 1, and 350 cycles and 2,401 samples in Case 2.

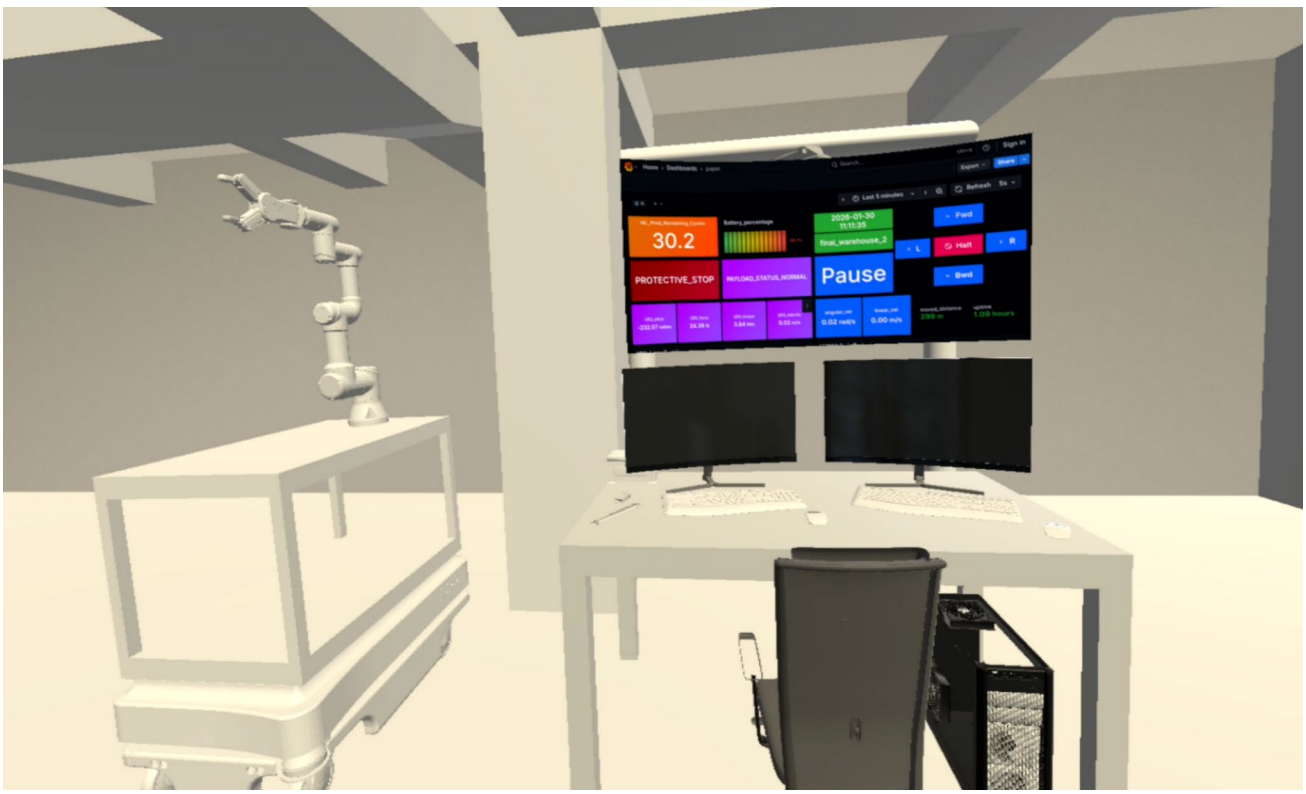


Fig. 11 Dashboard embedded in the DT environment on the Meta Quest Pro

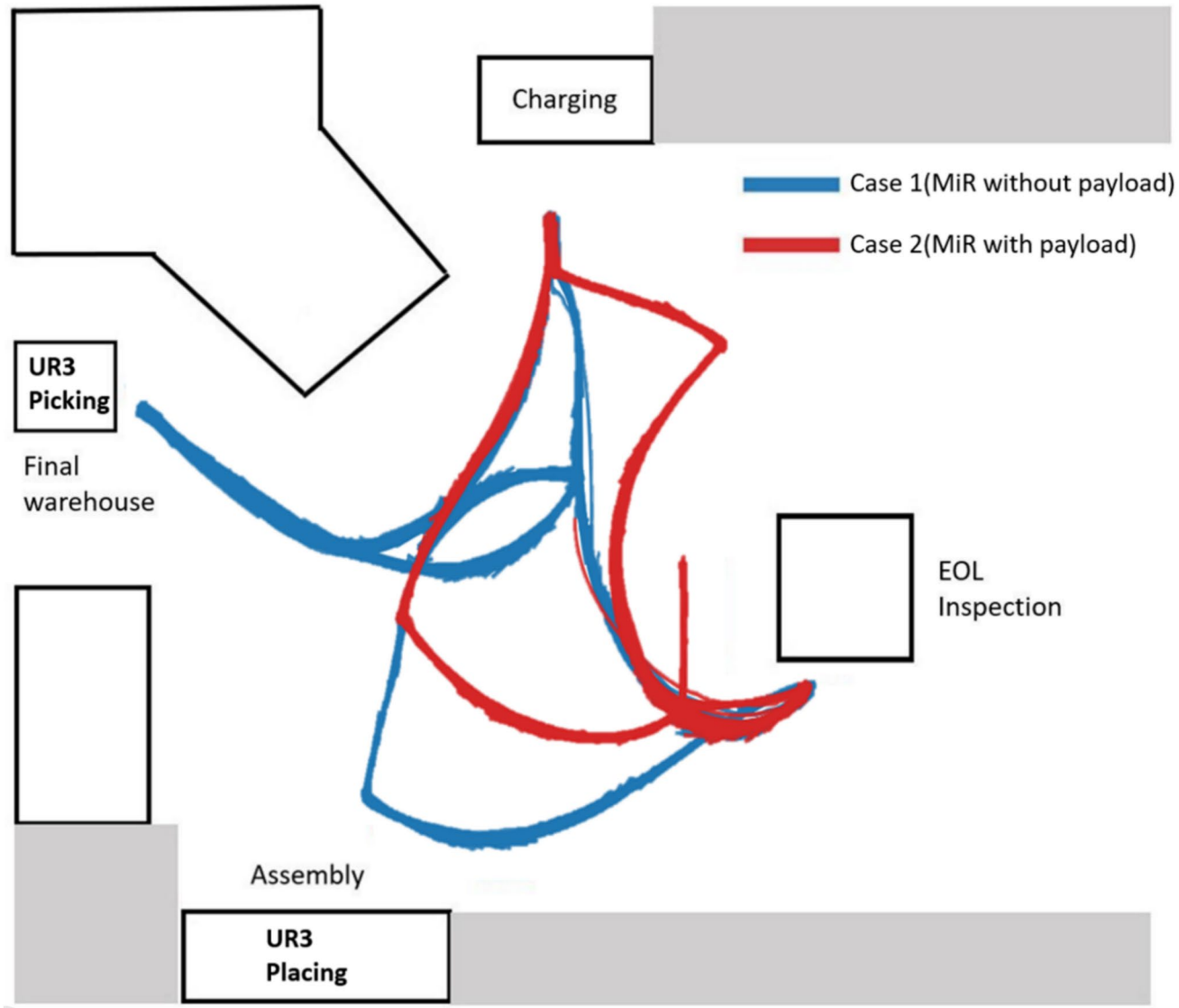


Fig. 12 MOMA experimental trajectory for model training

5.1.2 Data Preprocessing and Feature Engineering

Raw telemetry data collected from the MOMA platform is recorded at the timestamp level, resulting in time-series streams across multiple sensing modalities. To provide structured inputs suitable for regression modeling, the raw telemetry data is preprocessed and featurized.

Figure 13 illustrates the end-to-end pipeline. Within the raw telemetry data set module, heterogeneous measurements are logged at each timestamp as a unified stream Z_{MOMA}. During data preprocessing, the timestamped records are first segmented by the mission index g, producing mission-level segments within each operational cycle. Cycle boundaries are defined by the completion of a closed mission loop, and the ground-truth endurance target is assigned each cycle-mission pair as $y_{i,g} = RUC_{i,g}$.Based on these segmented records, the feature engineering step converts heterogeneous timestamped data into aligned, mission-indexed training instances. Then it maps each segmented instance into a compact representation $x_{i,g} \in \mathbb{R}^{22}$, represented by four groups of features. System and task context includes battery state, uptime, and moved distance, together with binary phase indicators. Mobile-base motion is summarized by statistical descriptors of MiR100 kinematics, including the mean and standard deviation of linear/angular velocities as well as the mean norm of the combined velocity. Manipulator workload is quantified using mean, standard deviation, and peak values of UR3 torque, effort, force, and velocity. The resulting feature vector $x_{i,g}$ is then used as input to the CatBoost regressor to predict the corresponding $\widehat{RUC}_{i,g}$.

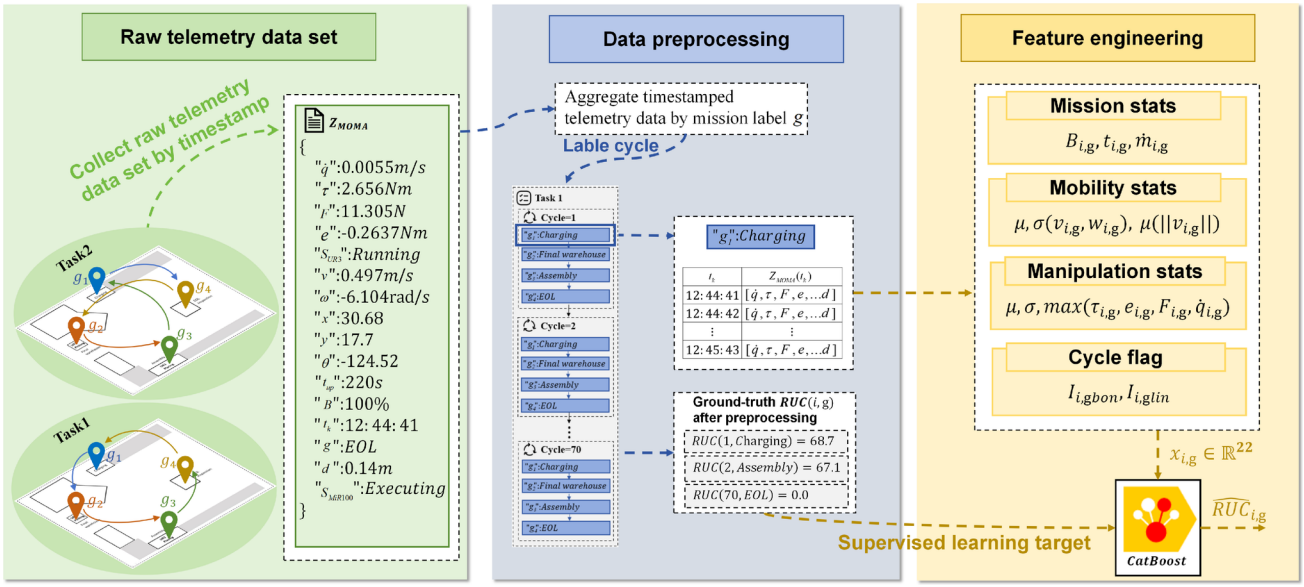


Fig. 13 Data preprocessing and feature engineering pipeline

5.2 Analysis of Real-Time Performance

This section evaluates the real-time feasibility of the proposed DT framework from two aspects: dashboard-level data update latency and edge-node computation cost. In the implemented Grafana-based monitoring system, telemetry data are queried from InfluxDB with a refresh interval of 5 s, corresponding to an effective dashboard update rate of 0.2 Hz. Therefore, the end-to-end monitoring latency perceived at the dashboard level is mainly bounded by this 5 s refresh interval. The proposed system is designed for soft real-time PHM, endurance prediction, alarm visualization, and human-in-the-loop supervision. For these monitoring-oriented functions, a 5 s update interval is considered acceptable because the target variables, such as battery state, predicted RUC, mission status, and health alarms, evolve at the mission or system-operation level rather than at the servo-control level. However, this refresh interval is not suitable for safety–critical reactive control. The VR dashboard is intended only for soft real-time monitoring and supervisory decision-making and does not replace the certified robot safety systems.

To evaluate the edge computation cost of the proposed DT framework, an additional resource-consumption experiment is conducted on the Raspberry Pi 5 edge node. Four operating modes are evaluated: M0, idle baseline; M1, MOMA telemetry acquisition only; M2, telemetry acquisition with MQTT-Telegraf-InfluxDB transmission and storage; and M3, the full DT pipeline with CatBoost-based RUC inference. For each mode, CPU usage, memory usage, CPU temperature, and network throughput are recorded during continuous operation. Each mode is executed for approximately 30 min, and the first 60 s are excluded as warm-up. The monitoring script records system-resource indicators at 1 Hz.

The results in Table 7 show a progressive increase in edge-node workload as more DT functions are enabled. As shown in Table 7, CPU mean and CPU P95 represent the average and near-peak processor utilization, RAM Mean indicates average memory usage, Temp mean denotes average device temperature, and Net RX/TX correspond to the average network receiving/transmitting throughput. Telemetry acquisition alone did not impose a substantial computational burden, as the CPU usage in M1 remained close to the idle baseline. The major increase occurred when the MQTT-Telegraf-InfluxDB pipeline is activated, indicating that continuous telemetry transmission,

Table 7 Edge computation cost of the proposed DT pipeline on the Raspberry Pi 5

Mode	CPU mean (%)	CPU P95 (%)	RAM Mean (%)	Temp mean (°C)	Net RX (kb/s)	Net TX (kb/s)
M0	9.74	17.99	30.50	57.19	1.44	0.33
M1	8.60	20.51	35.05	57.30	3.50	1.16
M2	24.10	34.12	39.03	61.04	40.99	17.95
M3	34.29	48.28	36.53	63.93	67.56	26.77

message forwarding, and database storage are the main contributors to the edge workload. When the CatBoost-based RUC inference module is further enabled in M3, the CPU load increased but remained within a moderate range.

Overall, the resource-consumption results indicate that the complete DT pipeline can be deployed on the Raspberry Pi 5 without exhausting CPU, memory, thermal, or network resources. The measured workload is therefore acceptable for the intended soft real-time PHM, endurance prediction, and human-in-the-loop monitoring tasks.

5.3 Analysis of Endurance Prediction

5.3.1 Benchmarking

The subsection focuses on systematically evaluating multiple regression algorithms to identify the most suitable model for mission-level endurance prediction. Six representative regression approaches are trained and compared: Support Vector Regression (SVR), Random Forest (RF), XGBoost, LightGBM, CatBoost and a PyTorch-based Artificial Neural Network (ANN). Comparison of baseline regression models is listed in Table 8.

To determine the most suitable regression approach for RUC prediction, all six candidate models are benchmarked using an identical mission-level dataset and evaluated using a 5-fold Cross Validation (CV) protocol to ensure statistical robustness and mitigate overfitting. The CV was performed at the mission-sample level after random shuffling. Performance is assessed using three standard regression metrics to ensure a fair and consistent comparison. The performance evaluation results are summarized in Fig. 14.

Among the evaluated approaches, gradient-boosting models perform the most stable and accurate predictions, indicating their strong ability to capture the nonlinear coupling between navigation dynamics, manipulator workload, and battery-state evolution. As illustrated in Fig. 14, CatBoost achieves the best overall predictive performance among the evaluated algorithms, obtaining an R^2 of 0.974, an RMSE of 3.592, and an MAE of 2.160. LightGBM and XGBoost produce comparable results but exhibit slightly larger residual errors, whereas RF, SVR, and ANN show noticeably lower accuracy and weaker generalization across heterogeneous mission scenarios.

RMSE is treated as the primary reliability metric because large RUC deviations can lead to incorrect maintenance decisions and suboptimal supervision. CatBoost obtains the lowest RMSE, suggesting superior robustness for endurance forecasting, and its low MAE further indicates consistently accurate predictions across varying payload conditions, trajectory configurations, and mission durations. The high R^2 values achieved by CatBoost models also confirm their effectiveness in capturing the nonlinear interaction between MiR100 mobility and UR3 manipulation workloads.

5.3.2 Prediction Analysis

Given its consistent superiority across all benchmark comparisons, CatBoost is selected as the final regression model for RUC prediction. Its main hyperparameters are determined through controlled tuning experiments. The final deployed model consists of 1,500 oblivious decision trees with a maximum depth of 10. The learning rate was set to 0.03, the L2 leaf regularization coefficient to 5, and the random-strength parameter to 0.2. The model was trained using the RMSE loss function.

Figure 15 summarizes the CatBoost regression performance across four progressively expanded datasets under two evaluation settings: full-data and 5-fold CV training. As the dataset is augmented with different routes and payload configurations in Fig. 12, the data distribution becomes increasingly heterogeneous and the prediction task becomes more challenging. The smallest dataset in Fig. 15a shows near-ceiling performance, which is consistent with overfitting to a limited and relatively homogeneous distribution. After introducing new task conditions shown in Fig. 15b, the CV error increases markedly, reflecting the increased variance induced by the expanded operating space, while full-data training remains highly accurate. With further data augmentation shown in Fig. 15c and d, the CV performance stabilized and slightly improved.

For the final expanded dataset shown in Fig. 15d, CatBoost achieved an R^2 of 0.974 ± 0.002, an RMSE of

Table 8 Comparison of baseline regression models

Category	Model name	Prediction function
Ensemble tree	CatBoost	$\hat{y}(x) = F_0 + \sum_{t=1}^{T} \eta f_t(x_i)$
	XGBoost	
	LightGBM	
	RF	$\hat{y}(x) = \frac{1}{T} \sum_{t=1}^{T} \eta f_t(x_i)$
Kernel	SVR	$\hat{y}(x) = \sum_{t=1}^{T} (a_i - a_i^*) K(x_i, x) + b$
Neural network	ANN	$\hat{y}(x) = W_3 \sigma(W_2 \sigma(W_1 x + b_1) + b_2) + b_3$

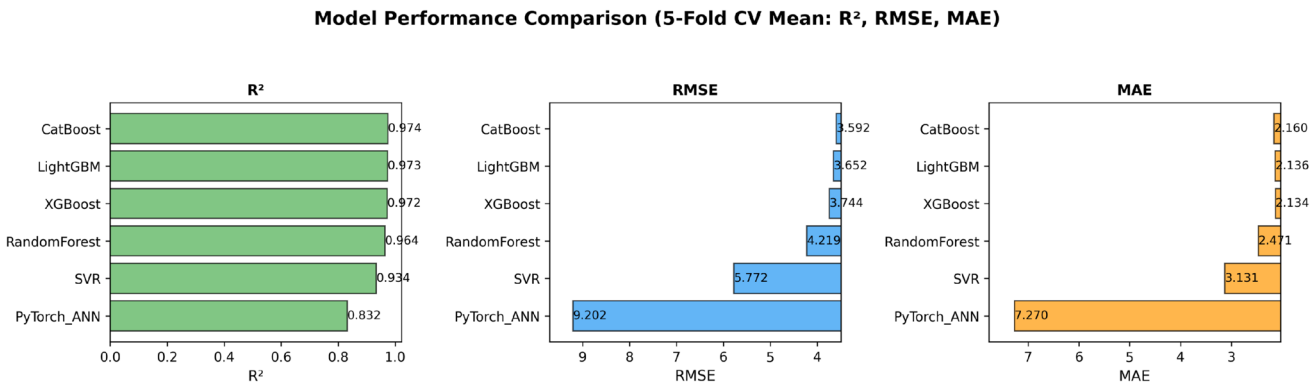


Fig. 14 Model performance comparison across six machine learning algorithms

3.592 ± 0.149 cycles, and an MAE of 2.160 ± 0.061 cycles across the five folds. The relatively small standard deviations indicate consistent predictive performance across the adopted cross-validation folds.

The discrepancy between CV and full-data training also decreased as the dataset grew, indicating reduced sensitivity to the increasing heterogeneity of the investigated operating conditions. In both settings, the predicted RUC values aligned closely with the ground truth and clustered around the ideal reference line. These results indicate that CatBoost captures the relationships between the mission-level operational features and RUC and provides stable prediction performance within the investigated operating conditions and the adopted mission-level validation protocol.

Figure 16 presents the residual analysis for the CatBoost regressor. In Fig. 16a, most residuals are concentrated around zero, and the distribution is roughly symmetric, indicating no obvious systematic overestimation or underestimation. However, a small number of larger residuals are still present on both sides. Figure 16b shows that the Q-Q plot follows the normal reference line in the central quantiles, whereas noticeable departures occur in the tails. This long-tail behavior is consistent with the 5-fold CV scatter in Fig. 15d, where a small fraction of samples exhibits larger errors near the beginning and end of the RUC range. Such deviations are physically plausible for MOMA operations because the battery trajectory is strongly nonlinear. At the beginning of the cycle, the discharge curve can be nearly flat. Near the end of life, the trajectory is further affected by the abort threshold of about 5.4% SoC. These characteristics make the signal less informative for RUC estimation and increase the likelihood of extreme residuals.

Figure 17 summarizes the optimized CatBoost model using feature-importance rankings and learning-curve behavior. As shown in Fig. 17a, the feature-importance analysis indicates that battery- and time-related variables, such as *battery_start*, *uptime_start*, and *moved_dist_start* dominate RUC prediction. These variables capture the initial energy state and mission progression, making them strong indicators of discharge dynamics over successive cycles. The notable contributions of *is_bonus_cycle* and *mean_UR3_effort* further highlight the role of operational phase and manipulator workload, suggesting that both mission context and mechanical stress influence energy consumption and cycle longevity. Several motion-variability descriptors, including *std_linear_vel* and *std_angular_vel*, also contribute to the prediction. Their influence implies that velocity fluctuations and frequent acceleration-deceleration transitions are associated with higher energy demand, particularly under irregular or highly dynamic mission profiles. As shown in Fig. 17b, the learning curve provides additional insight into training dynamics. RMSE decreases smoothly with boosting iterations, with rapid gains early on and gradual convergence after roughly 800–1000 iterations. The behavior of the learning curve suggests stable optimization, while the late-stage plateau indicates diminishing returns from adding more trees and reflects a reasonable balance between learning rate and regularization.

Figure 18 reports the SHAP analysis used to interpret the CatBoost RUC predictor. As shown in Fig. 18a, the global summary shows that feature influence is dominated by energy- and time-related variables. *battery_start* contributes positively to RUC, whereas *uptime_start* and *moved_dist_start* contribute negatively, consistent with the effect of accumulated runtime and prior travel on the remaining capacity. *mean_UR3_effort* also has a clear impact, indicating that higher manipulation workload increases energy demand and shortens endurance. Motion-variability terms (e.g., *std_linear_vel*, *std_angular_vel*) have smaller but non-negligible effects, suggesting that irregular velocity profiles reduce efficiency. The long SHAP tails for several features imply that uncommon missions (e.g., high payload or highly dynamic motion) can drive larger prediction shifts.

To examine these effects in detail, SHAP dependence plots are shown for *battery_start*, *uptime_start*, and *is_bonus_cycle*, as shown in Fig. 18b-d. *battery_start* exhibits a monotonic

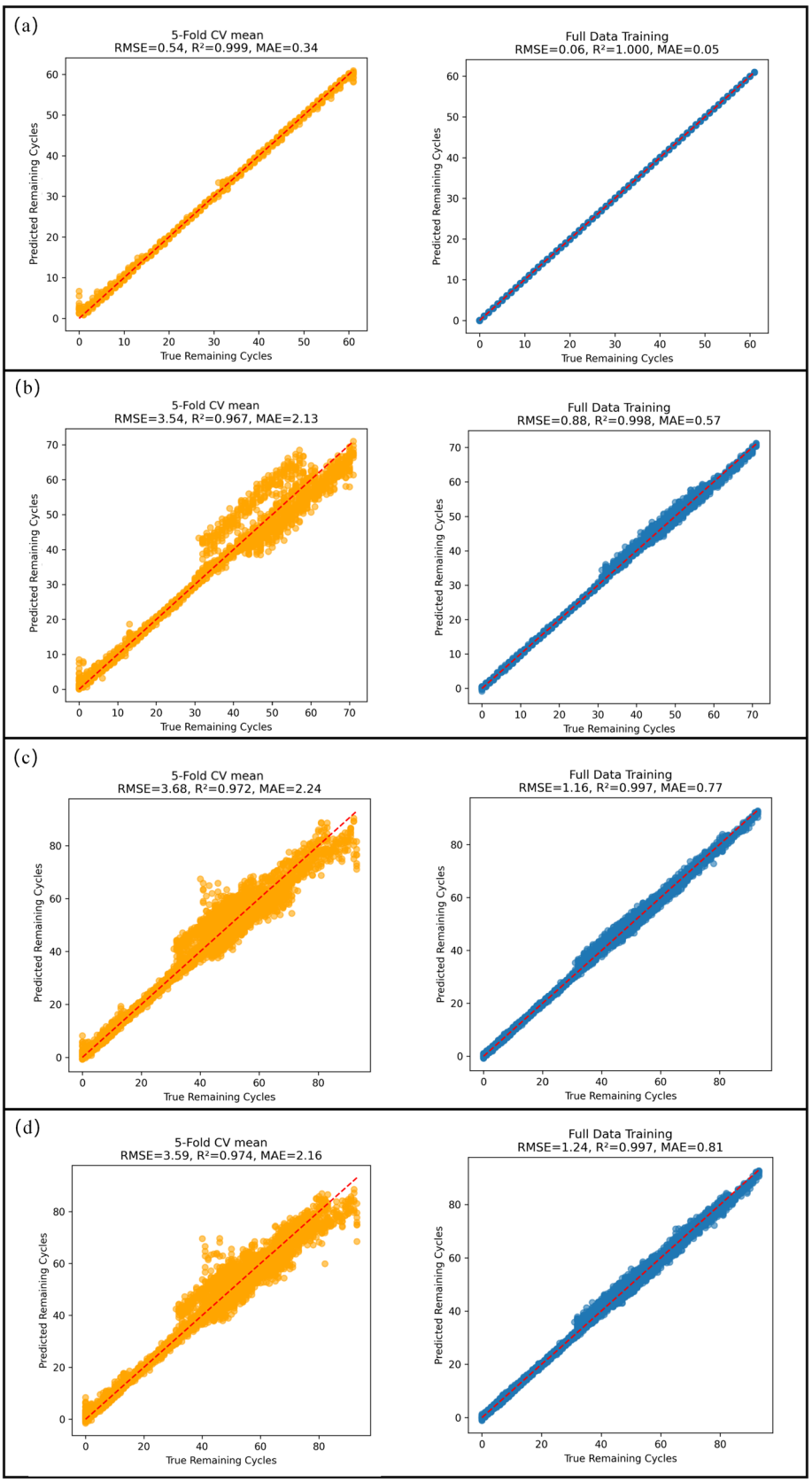


Fig. 15 Prediction performance of the CatBoost model

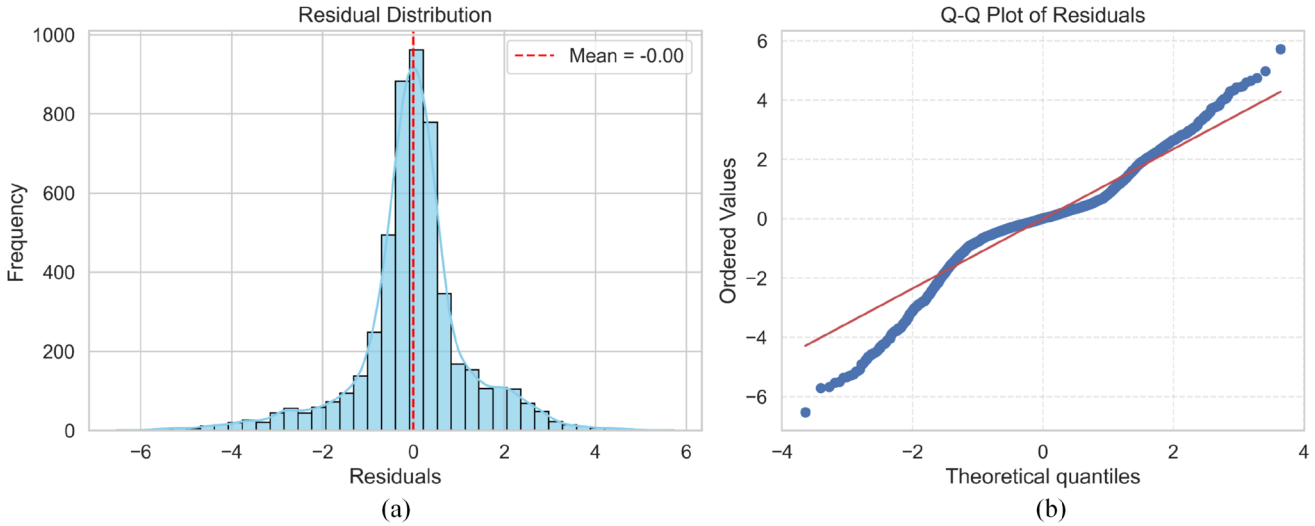


Fig. 16 Residual analysis for the CatBoost model

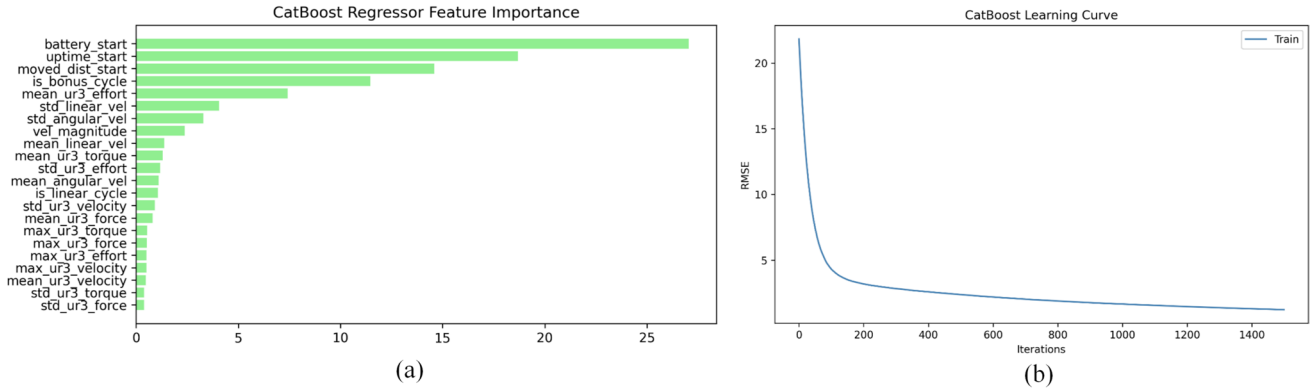


Fig. 17 CatBoost model characteristics

relationship: higher initial charge yields larger positive contributions to predicted RUC. *uptime_start* shows increasingly negative contributions beyond roughly 4000-6000s, while interaction coloring indicates that higher initial charge partially mitigates this penalty. *is_bonus_cycle* separates early-stage operation from the main discharge regime: cycles near full-charge conditions produce positive contributions, whereas normal discharge phases tend to contribute negatively, reflecting higher energetic cost later in the discharge curve.

CatBoost is selected as the final regressor for RUC prediction, as it outperforms other baseline models on various metrics. The predicted RUC shows an almost linear relationship with the actual RUC, and residual diagnostics indicates low bias with stable error behavior across missions. Feature importance and learning curve analysis suggests good optimization behavior, while SHAP explanations are consistent with physical intuition: the initial battery state determines longer endurance, whereas accumulated distance and higher robotic arm load reduce remaining cycles, and motion variability provides additional effects. These results support CatBoost as a reliable and interpretable DT ensemble endurance prediction model.

5.4 Analysis of Health Monitoring

To demonstrate human-in-the-loop execution supervision and reactive control for MOMA health monitoring, an experimental case study is conducted using the Grafana-based supervisory dashboard deployed in the VR environments. Figure 19 demonstrates the VR-assisted event awareness and operator response under representative conditions. Three views are shown in Fig. 19, including the relevant VR dashboard block in the left, the corresponding physical situation in the middle, and the reactive action of operator in the right.

In Fig. 19a, the manipulator health and safety block in Fig. 10 reports a UR3 payload-drop alarm during grasping,

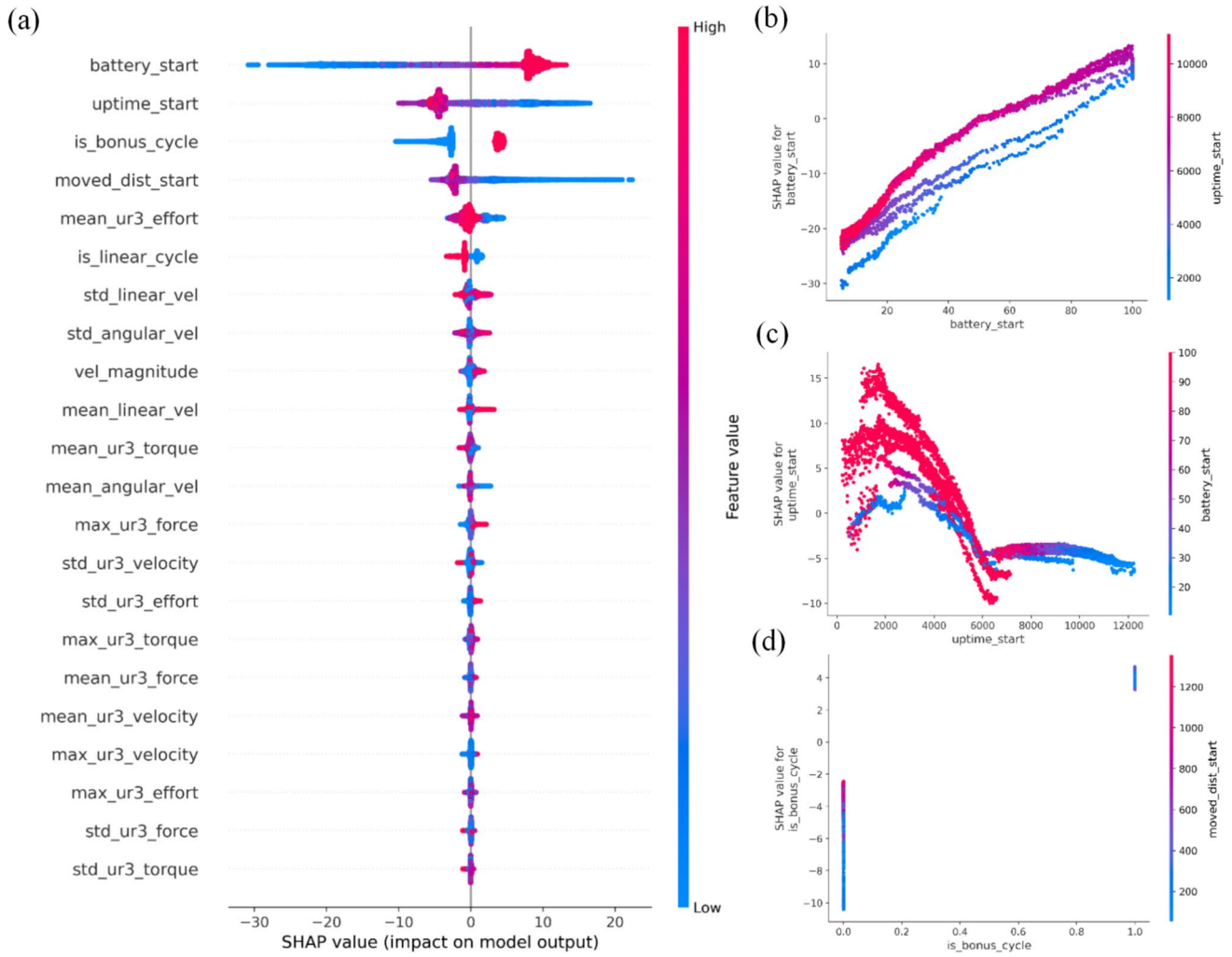


Fig. 18 SHAP interpretability: (**a**) global summary plot; (**b**) dependence plot for *battery_start*; (**c**) dependence plot for *uptime_start*; and (**d**) dependence plot for *is_bonus_cycle*

which matches a confirmed drop in the laboratory and requires direct human intervention to recover the task. In Fig. 19b, the same block used in Fig. 19a indicates a UR3 protective stop. The manipulator is observed in a safety-stop condition and is restored manually via the teach pendant. These two cases are human-intervention-required events.

In Fig. 19c, the mobile robot execution and safety block in Fig. 10 flags a MiR100 emergency stop caused by an obstacle, and the operator recovers operation by issuing avoidance maneuvers through the supervisory motion command block in Fig. 10. In Fig. 19d, the predictive endurance and energy-status block issues an endurance warning due to low RUC and battery level, prompting the operator to command a return to the charging station via the supervisory motion command blocks. Unlike Fig. 19a and b, c and d are supervisory-recoverable events that can be resolved through VR-assisted command intervention.

The case study shows that the framework supports operator awareness and supervisory intervention during health monitoring. The VR-assisted human-in-the-loop control method supports operators to handle MOMA safety events and endurance-related alerts, and validates the end-to-end feasibility of the proposed endurance prediction and health monitoring framework for MOMA.

6 Discussion

The proposed DT-based framework demonstrates the feasibility of integrating telemetry synchronization, endurance prediction, and human-in-the-loop monitoring for a heterogeneous MOMA platform. However, several limitations and practical deployment issues should be discussed to clarify the boundary of the proposed approach.

The practical performance of the framework is influenced by the reliability of both robot execution and data communication. During operation, events such as obstacle-induced stops,

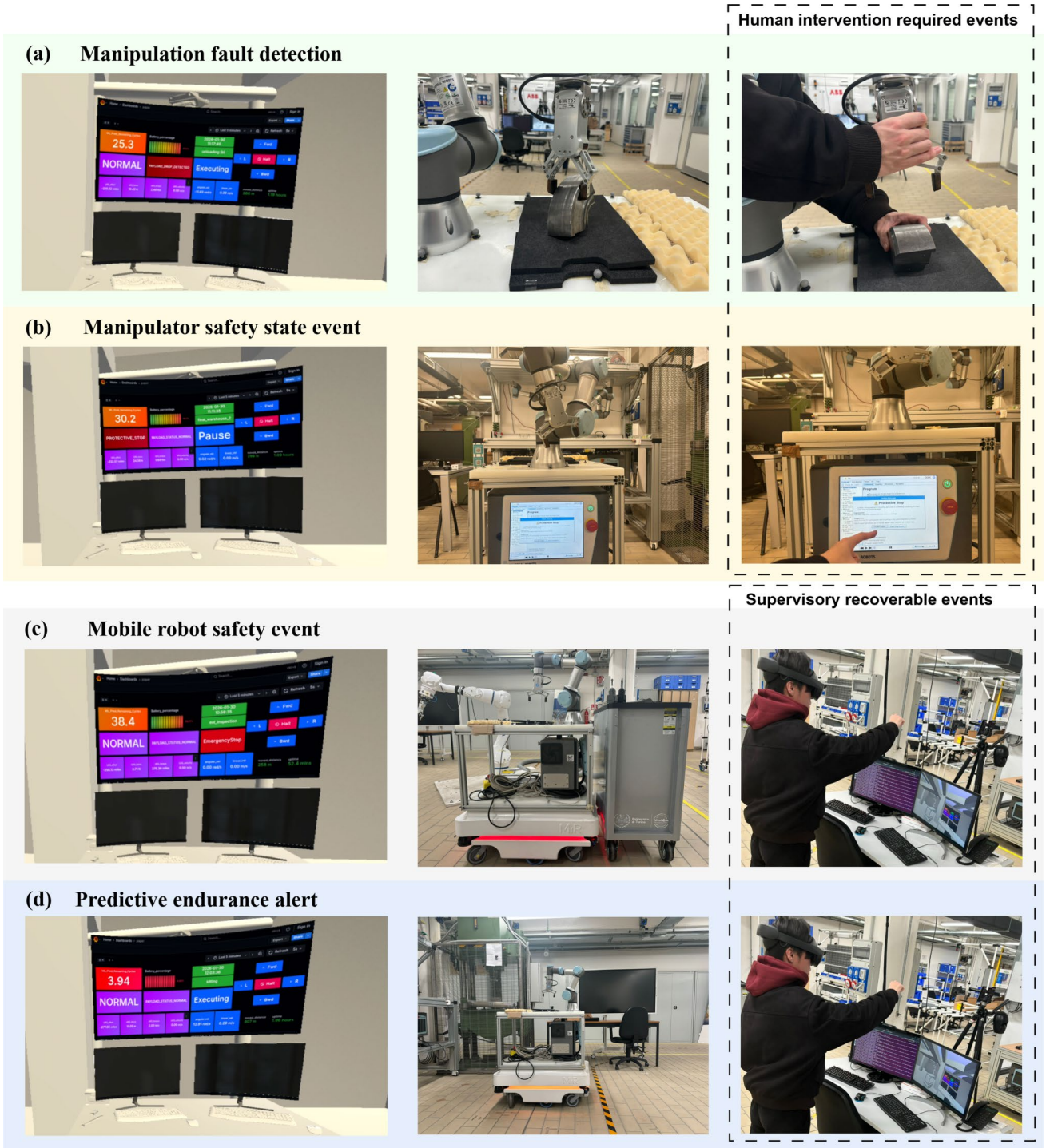


Fig. 19 VR-assisted human-in-the-loop supervision and intervention during operation

mission interruptions, payload drops, manipulator safety stops, abnormal battery discharge, delayed messages, or missing telemetry packets may temporarily reduce the completeness of the DT representation. Since the RUC model is built on synchronized mission-level telemetry, unstable battery, velocity, force, torque, effort, or mission-state signals may also affect feature extraction and prediction accuracy. Therefore, more robust filtering, outlier detection, missing-data compensation, and uncertainty-aware prediction methods should be considered in future implementations. For example, Kalman filtering, moving-average smoothing, interpolation-based missing-data handling, anomaly detection algorithms, or probabilistic

confidence estimation could be integrated to improve robustness against noisy or incomplete telemetry.

Another limitation is related to deployment generalization. The current implementation is based on a MiR100-UR3 platform, and transferring the framework to other MOMA configurations may require adaptation of communication interfaces, telemetry schemas, mission definitions, battery conditions, and payload settings. Model recalibration or retraining may also be necessary when the task structure or robot configuration changes significantly. In addition, the proposed DT framework should be understood as a decision-support tool for PHM, endurance prediction, alert visualization, and human-in-the-loop monitoring. It is not intended to replace certified robot safety mechanisms, emergency-stop systems, or industrial safety procedures in safety–critical operation.

7 Conclusion and Outlook

This paper proposes a multi-layer DT framework for endurance prediction and health monitoring of a heterogeneous MOMA composed of a MiR100 mobile robot and a UR3 collaborative manipulator, addressing the challenges of heterogeneous sensing frequencies, nonlinear navigation-manipulation function coupling, and shared power consumption inherent in MOMA. The proposed architecture integrates telemetry data synchronization, endurance prediction with fault-aware monitoring, and VR-assisted human-in-the-loop supervisory control within a deployable system. A telemetry data synchronization pipeline is designed to align and aggregate heterogeneous sensing streams from the mobile robot and manipulator into a coherent, time-consistent representation. To accurately assess mission-level endurance of MOMA, the RUC metric is defined. Subsequently, a mission-level CatBoost-based endurance prediction model is established to extract RUC from telemetry data and detect anomalous navigation-manipulation coupling behaviors by incorporating fault-aware monitoring. A VR-assisted HRI system is developed to provide real-time state visualization and human-in-the-loop control for responsive interaction. Experimental results demonstrate that the framework achieves an RMSE of 3.59 cycles and responsive supervisory interaction across diverse mission scenarios, confirming its suitability for PHM in MOMA.

Overall, MOMA with integrated mobile-manipulation functions will be better adapted to intelligent manufacturing systems evolving toward autonomous intelligent systems, such as matrix manufacturing systems. MOMA can meet the requirements of inter-station part transportation and intra-station part pick-and-place. Therefore, although this study focuses on MOMA, its potential contribution to advancing intelligent manufacturing is significant. However, it should be noted that this work represents only a foundational step toward MOMA PHM. Future work will focus on two aspects. On the one hand, it is to explore the research depth. Constructing a system fault diagnosis and maintenance strategy recommendation system combining LLMs and knowledge engineering technologies, and achieving a comprehensive and reliable intelligent PHM platform based on DT. On the other hand, it is to expand the research breadth by extending the current framework to multi-robot collaborative manufacturing systems and HRC manufacturing systems to support the intelligent manufacturing and Industry 5.0 paradigm.

Author Contributions Y.Z. conceived the research idea, designed the methodology, implemented the algorithms, conducted experiments, analyzed the results, and drafted the manuscript. X.Y. analyzed the results, drafted the manuscript, and revised the manuscript. K.A., and D.A. provided supervision, contributed to the research idea, and revised the manuscript. All authors reviewed and approved the final manuscript.

Funding Open access funding provided by Politecnico di Torino within the CRUI-CARE Agreement. X.Y. has received research support from the China Scholarship Council Program (Grant No. 202506260086).

Data Availability No datasets were generated or analysed during the current study.

Declarations

Ethics Approval This study did not require ethics approval.

Consent to Participate Informed consent was obtained from all individual participants included in the study.

Consent for Publication All participants in this study consented to publication.

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Tao, Y., Wan, J., Song, Y., Li, X., Wang, B., Wang, T., Wang, Y.: A safety posture field framework for mobile manipulators based on human-robot interaction trend and platform-arm coupling motion. Robot. Comput.-Integr. Manuf. **93**, 102903 (2025). https://doi.org/10.1016/j.rcim.2024.102903
2. Sandakalum, T., Ang, M.H.: Motion planning for mobile manipulators—a systematic review. Machines **10**(2), 97 (2022). https://doi.org/10.3390/machines10020097
3. Antonelli, D., Aliev, K., Soriano, M., Samir, K., Monetti, F.M., Maffei, A.: Exploring the limitations and potential of digital twins

for mobile manipulators in industry. Procedia Comput. Sci. **232**, 1121–1130 (2024). https://doi.org/10.1016/j.procs.2024.01.110
4. Yang, X., Liu, X., Zhang, H., Fu, L., Yu, Y.: Meta-model-based shop-floor digital twin architecture, modeling and application. Robot. Comput.-Integr. Manuf. **84**, 102595 (2023). https://doi.org/10.1016/j.rcim.2023.102595
5. Cui, Z., Yang, X., Yue, J., Liu, X., Tao, W., Xia, Q., Wu, C.: A review of digital twin technology for electromechanical products: evolution focus throughout key lifecycle phases. J. Manuf. Syst. **70**, 264–287 (2023). https://doi.org/10.1016/j.jmsy.2023.07.016
6. Shi, Y., Lao, L., Zheng, X., Chen, P.: Real-time spatiotemporal synchronization for human-robot collaborative systems: an integrated digital twin framework with adaptive data processing. J. Intell. Robot. Syst. **112**(1), 12 (2026). https://doi.org/10.1007/s10846-025-02346-w
7. Li, W., Liu, X., Du, X., Shao, H., Zhu, T., Yang, S., Yang, X.: Matrix-structured manufacturing systems: literature review and future directions. Int. J. Prod. Res. (2026). https://doi.org/10.1080/00207543.2026.2669883
8. Ayankoso, S., Kaigom, E., Louadah, H., Faham, H., Gu, F., Ball, A.: A hybrid digital twin scheme for the condition monitoring of industrial collaborative robots. Procedia Comput. Sci. **232**, 1099–1108 (2024). https://doi.org/10.1016/j.procs.2024.01.108
9. Malik, A.A., Brem, A.: Digital twins for collaborative robots: a case study in human-robot interaction. Robot. Comput.-Integr. Manuf. **68**, 102092 (2021). https://doi.org/10.1016/j.rcim.2020.102092
10. Hollenbeck, D., Chen, Y.: A digital twin framework for environmental sensing with sUAS. J. Intell. Robot. Syst. **105**(1), 1 (2022). https://doi.org/10.1007/s10846-021-01542-8
11. Qin, Q., Liu, Z., Zhong, R., Wang, X.V., Wang, L., Wiktorsson, M., Wang, W.: Robot digital twin systems in manufacturing: technologies, applications, trends and challenges. Robot. Comput.-Integr. Manuf. **97**, 103103 (2026). https://doi.org/10.1016/j.rcim.2025.103103
12. Antonelli, D., Aliev, K.: Intelligent energy management for mobile manipulators using machine learning. FME Trans. **50**(4), 752–761 (2022). https://doi.org/10.5937/fme2204752A
13. Wang, G., Zhang, C., Liu, S., Zhao, Y., Zhang, Y., Wang, L.: Multi-robot collaborative manufacturing driven by digital twins: advancements, challenges, and future directions. J. Manuf. Syst. **82**, 333–361 (2025). https://doi.org/10.1016/j.jmsy.2025.06.014
14. Ni, H., Hu, T., Deng, J., Chen, B., Luo, S., Ji, S.: Digital twin-driven virtual commissioning for robotic machining enhanced by machine learning. Robot. Comput.-Integr. Manuf. **93**, 102908 (2025). https://doi.org/10.1016/j.rcim.2024.102908
15. Li, H., He, X., Wu, Y., Liu, G., Wang, H., Wen, X., Li, L.: Digital twin and AI-driven robotic embodied control system: a novel adaptive learning and decision optimization method. Robot. Comput.-Integr. Manuf. **98**, 103138 (2026). https://doi.org/10.1016/j.rcim.2025.103138
16. Shi, J., Liu, X., Su, M., Yi, Y., Yang, H., Cai, W.: A digital twin-based smart assembly for cable-driven parallel robots. Adv. Eng. Inform. **68**, 103623 (2025). https://doi.org/10.1016/j.aei.2025.103623
17. Yang, K., Liu, Y., Tuo, B., Pan, Y., Wang, X., Zhang, L., Wang, L.: A multi-level multi-domain digital twin modeling method for industrial robots. Robot. Comput.-Integr. Manuf. **95**, 103023 (2025). https://doi.org/10.1016/j.rcim.2025.103023
18. Wang, B., Zhou, H., Li, X., Yang, G., Zheng, P., Song, C., Yuan, Y., Wuest, T., Yang, H., Wang, L.: Human digital twin in the context of industry 5.0. Robot. Comput.-Integr. Manuf. **85**, 102626 (2024). https://doi.org/10.1016/j.rcim.2023.102626
19. Han, S., Li, F., Lee, C., Wang, T., Diaconeasa, M.A.: Mirror the mind of crew: maritime risk analysis with explicit cognitive processes in a human digital twin. Adv. Eng. Inform. **62**, 102746 (2024). https://doi.org/10.1016/j.aei.2024.102746
20. Feng, B., Wang, Z., Yuan, L., Zhou, Q., Chen, Y., Bi, Y.: Towards safe motion planning for industrial human-robot interaction: a co-evolution approach based on human digital twin and mixed reality. Robot. Comput.-Integr. Manuf. **95**, 103012 (2025). https://doi.org/10.1016/j.rcim.2025.103012
21. Li, H., Ma, W., Wang, H., Liu, G., Wen, X., Zhang, Y., Yang, M., Luo, G., Xie, G., Sun, C.: A framework and method for human-robot cooperative safe control based on digital twin. Adv. Eng. Inform. **53**, 101701 (2022). https://doi.org/10.1016/j.aei.2022.101701
22. Gkournelos, C., Konstantinou, C., Makris, S.: An LLM-based approach for enabling seamless human-robot collaboration in assembly. CIRP Ann. **73**(1), 9–12 (2024). https://doi.org/10.1016/j.cirp.2024.04.002
23. Nguyen, T.P., Nguyen, H., Cao, H., Bui, T.T., Ngo, H.Q.T.: A digital twin-empowered framework for interactive consumers in manufacturing using wearable device. IEEE Access **13**, 146557–146568 (2025). https://doi.org/10.1109/ACCESS.2025.3598864
24. Pham, C.D., Nguyen, H., Nguyen, T.P., Ngo, H.Q.T.: Social perception and formation control for human–robot co-working systems. IEEE Access **14**, 68597–68609 (2026). https://doi.org/10.1109/ACCESS.2026.3687098
25. Xu, W., Yang, H., Ji, Z., Ba, M.: Cognitive digital twin-enabled multi-robot collaborative manufacturing: framework and approaches. Comput. Ind. Eng. **194**, 110418 (2024). https://doi.org/10.1016/j.cie.2024.110418
26. Zhou, Z., Yang, X., Wang, H., Zhang, X.: Digital twin with integrated robot-human/environment interaction dynamics for an industrial mobile manipulator. In: IEEE, pp. 5041–5047 (2022). https://doi.org/10.1109/ICRA46639.2022.9812004.
27. Oo, K.H., Koomsap, P., Ayutthaya, D.H.N.: Digital twin-enabled multi-robot system for collaborative assembly of unorganized parts. J. Ind. Inf. Integr. **44**, 100764 (2025). https://doi.org/10.1016/j.jii.2024.100764
28. D. Antonelli, K. Aliev, F.M. Monetti, A. Maffei, Enhancing industrial mobile manipulators through cognitive digital twins, in, Berlin, Germany, pp. 25–36 (2025). https://doi.org/10.1007/978-3-031-94484-0_3
29. Vogl, G.W., Weiss, B.A., Helu, M.: A review of diagnostic and prognostic capabilities and best practices for manufacturing. J. Intell. Manuf. **30**(1), 79–95 (2019). https://doi.org/10.1007/s10845-016-1228-8
30. Gomes, I.P., Wolf, D.F.: Health monitoring system for autonomous vehicles using dynamic Bayesian networks for diagnosis and prognosis. J. Intell. Robot. Syst. **101**(1), 19 (2021). https://doi.org/10.1007/s10846-020-01293-y
31. Kang, J., Chung, K., Hong, E.J.: Multimedia knowledge-based bridge health monitoring using digital twin. Multimed. Tools Appl. **80**(26), 34609–34624 (2021). https://doi.org/10.1007/s11042-021-10649-x
32. Wang, X., Yan, J., Xu, X.: A novel dynamic spatio-temporal graph based condition monitoring framework for consistency retention of digital twin. J. Manuf. Syst. **79**, 455–465 (2025). https://doi.org/10.1016/j.jmsy.2025.01.006
33. Li, X., He, B., Wang, Z., Zhou, Y., Li, G., Jiang, R.: Semantic-enhanced digital twin system for robot–environment interaction monitoring. IEEE Trans. Instrum. Meas. **70**, 1–13 (2021). https://doi.org/10.1109/TIM.2021.3066542
34. Touhid, M.T.B., Marne, M., Oskroba, T., Mirahmadi, S.A., Zhu, E., Mehrabian, A., Defersha, F., Yang, S.: Building a cloud-based digital twin for remote monitoring and control of a robotic assembly system. Int. J. Adv. Manuf. Technol. **129**(9–10), 4045–4057 (2023). https://doi.org/10.1007/s00170-023-12611-7
35. Zhu, Z., Zhu, W., Huang, J., He, B.: An intelligent monitoring system for robotic milling process based on transfer learning and digital twin. J. Manuf. Syst. **78**, 433–443 (2025). https://doi.org/10.1016/j.jmsy.2024.12.009
36. Feddoul, Y., Ragot, N., Duval, F., Havard, V., Baudry, D., Assila, A.: Exploring human-machine collaboration in industry:

a systematic literature review of digital twin and robotics interfaced with extended reality technologies. Int. J. Adv. Manuf. Technol. **129**(5–6), 1917–1932 (2023). https://doi.org/10.1007/s00170-023-12291-3
37. Williams, R., Erkoyuncu, J.A., Masood, T., Vrabic, R.: Augmented reality assisted calibration of digital twins of mobile robots. IFAC-PapersOnLine **53**(3), 203–208 (2020). https://doi.org/10.1016/j.ifacol.2020.11.033
38. Li, C., Zheng, P., Li, S., Pang, Y., Lee, C.K.M.: AR-assisted digital twin-enabled robot collaborative manufacturing system with human-in-the-loop. Robot. Comput.-Integr. Manuf. **76**, 102321 (2022). https://doi.org/10.1016/j.rcim.2022.102321
39. Calandra, D., Pratticò, F.G., Cannavò, A., Casetti, C., Lamberti, F.: Digital twin- and extended reality-based telepresence for collaborative robot programming in the 6g perspective. Digit. Commun. Netw. **10**(2), 315–327 (2024). https://doi.org/10.1016/j.dcan.2022.10.007
40. Li, J., Liu, X., Wu, X., Wang, D., Xu, K., Li, Y.: A time and frequency convolutional autoencoder for anomaly detection in industrial robots based on inertial measurement unit error calibration. Eng. Appl. Artif. Intell. **162**, 112269 (2025). https://doi.org/10.1016/j.engappai.2025.112269
41. Wang, S., Tao, J., Jiang, Q., Chen, W., Qin, C., Liu, C.: A digital twin framework for anomaly detection in industrial robot system based on multiple physics-informed hybrid convolutional autoencoder. J. Manuf. Syst. **77**, 798–809 (2024). https://doi.org/10.1016/j.jmsy.2024.10.016
42. Yin, X., He, W., Wang, J., Peng, S., Cao, Y., Zhang, B.: Health state assessment based on the parallel–serial belief rule base for industrial robot systems. Eng. Appl. Artif. Intell. **142**, 109856 (2025). https://doi.org/10.1016/j.engappai.2024.109856
43. Long, W., Yang, B., Zhang, Y., Wang, S., He, Y., Li, Y.: Dynamic anomaly detection in industrial robot clusters: a statistical-deep learning hybrid approach. Mech. Syst. Signal Proc. **234**, 112863 (2025). https://doi.org/10.1016/j.ymssp.2025.112863
44. Kayan, H., Heartfield, R., Rana, O., Burnap, P., Perera, C.: CASPER: context-aware IoT anomaly detection system for industrial robotic arms. ACM Trans. Internet Things (2024). https://doi.org/10.1145/3670414
45. Yang, B., Long, W., Zhang, Y., Xi, Z., Jiao, J., Li, Y.: Multivariate time series anomaly detection: missing data handling and feature collaborative analysis in robot joint data. J. Manuf. Syst. **75**, 132–149 (2024). https://doi.org/10.1016/j.jmsy.2024.06.006
46. Yun, H., Kim, H., Jeong, Y.H., Jun, M.B.G.: Autoencoder-based anomaly detection of industrial robot arm using stethoscope based internal sound sensor. J. Intell. Manuf. **34**(3), 1427–1444 (2023). https://doi.org/10.1007/s10845-021-01862-4
47. Gao, Z., Wang, C., Wu, J., Wang, Y., Jiang, W., Dai, T.: Degradation-aware remaining useful life prediction of industrial robot via multi-scale temporal memory transformer framework. Reliab. Eng. Syst. Saf. **262**, 111176 (2025). https://doi.org/10.1016/j.ress.2025.111176
48. Ren, G., Wang, Z., Liu, X., Song, F.: Remaining useful life prediction of industrial robot RV reducer with multiple deep networks and multicore support vector data description. J. Mech. Sci. Technol. **38**(8), 4037–4051 (2024). https://doi.org/10.1007/s12206-024-0703-y
49. Yuan, Y., Han, Y., Xiao, K., Xu, Z., Jiang, X.: Remaining useful life prediction for the harmonic reducer of industrial robots via in-situ current signal and lightweight multiscale attention deep networks. J. Manuf. Syst. **83**, 322–336 (2025). https://doi.org/10.1016/j.jmsy.2025.09.008
50. Kolvig-Raun, E.S., Kjærgaard, M.B., Brorsen, R.: Joint stress estimation and remaining useful life prediction for collaborative robots to support predictive maintenance. IEEE Robot. Autom. Lett. **9**(4), 3554–3561 (2024). https://doi.org/10.1109/LRA.2024.3368296
51. Monsalve, G., Acevedo-Bueno, D., Cardenas, A., Martinez, W.: An improved ECM-based state-of-charge estimation for SLA and LFP batteries used in low-cost agricultural mobile robots. IEEE Access **12**, 146265–146276 (2024). https://doi.org/10.1109/ACCESS.2024.3473896
52. Shokri, M., Lyons, L., Pequito, S., Ferranti, L.: Battery identification with cubic spline and moving horizon estimation for mobile robots. IEEE Trans. Control Syst. Technol. **32**(5), 1944–1951 (2024). https://doi.org/10.1109/TCST.2024.3380950
53. Rico-Melgosa, D., Zulueta, E., Rodriguez-Guerra, J., Inziarte-Hidalgo, I., Aramendia, I.: Cross-validated neural network optimization for explainable energy prediction in industrial mobile robots. Appl. Sci.-Basel **15**(23), 12644 (2025). https://doi.org/10.3390/app152312644
54. Lee, J.D., Seo, D., Shin, J., Bang, H.: Fast real-time state-of-health estimation method for lithium-ion battery using sparse identification of nonlinear dynamics. J. Intell. Robot. Syst. **111**(3), 93 (2025). https://doi.org/10.1007/s10846-025-02274-9
55. Kibira, D., Shao, G., Weiss, B.A.: Buiding a digital twin for robot workcell prognostics and health management. In: IEEE, pp. 1–12 (2021). https://doi.org/10.1109/WSC52266.2021.9715501
56. Elgohr, A.T., Rashad, M., El-Gendy, E.M., Shaaban, W., Saafan, M.M.: Trust as a design principle in human–robot collaboration: a review of explainable and adaptive control. Artif. Intell. Rev. **59**(5), 134 (2026). https://doi.org/10.1007/s10462-026-11554-1

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Yunhan Zhang received the Bachelor of Engineering degree from Guangdong Ocean University in 2024. He is currently pursuing the M.Sc. degree at Politecnico di Torino, where he is conducting his master's thesis research. His current research interests include human-robot collaboration, multimodal learning and perception for robotics, digital twins, and intelligent robotic systems.

Xiaolang Yang received his Bachelor's degree in Mechanical Manufacturing in 2019 and his Master's degree in Mechanical Engineering in 2022, both from Tongji University, China. He is currently pursuing a Ph.D. thesis at the School of Mechanical Engineering, Tongji University, China, and is a visiting Ph.D. student at the Department of Management and Production Engineering, Politecnico di Torino, Italy (2025–2026). His research interests include intelligent manufacturing, shop-floor digital twin, knowledge engineering, and Model-Based Systems Engineering (MBSE).

Khurshid Aliev received his Ph.D. degree in Electronics from Politecnico di Torino, Italy, in 2018. He is currently a Postdoctoral Researcher at the Department of Management and Production Engineering at Politecnico di Torino. He is also the scientific coordinator of the Mind-4Lab laboratory. His current research interests include collaborative robotics, industrial automation, and the integration of artificial intelligence and Digital Twin technologies into manufacturing systems.

Dario Antonelli received the M.S. degree with honours in Mechanical Engineering from the Politecnico di Torino in 1990. During his thesis, he studied the optimization of the mechanical performances of robotic manipulators. He is currently Associate Professor with the Politecnico di Torino, Italy, where he works on optimizing human-robot collaboration in manufacturing systems and designing production planning and control architectures.