Intelligent Energy Management for Mobile Manipulators Using Machine Learning

Dario Antonelli
Associate Professor
Politecnico di Torino – Torino
Department of Management and Production Engineering
Italy

Khurshid Aliev
Research Assistant
Politecnico di Torino – Torino
Department of Management and Production Engineering
Italy

Integrated robotic systems combining manipulators with mobile robots provide outstanding improvement opportunities for semi-automatic assembly processes leveraged by Industry 4.0. Factory operations are released from the rigid layout constraints imposed by conventional fixed robots. Thus, they introduce new challenges in managing the recharge cycles as the energy consumption of mobile manipulators is not simply related to the travelled distance but to the overall tasks executed. Its estimation requires a systemic approach. In the proposed solution, an intelligent monitoring system is implemented on board. Data gathered online, and Key Performance Indicators (KPIs) calculated during the working tasks are exploited by Machine Learning (ML) to optimize energy recharging cycles. Although the development of an intelligent monitoring framework for a mobile manipulator was the original objective of the research, the monitoring system is exploited here for energy management only, leaving space for other future applications.

Keywords: *Mobile robots, Collaborative manipulators, Machine Learning, Energy consumption, Online Monitoring, Industry 4.0.*

1. INTRODUCTION

Currently, robotic technologies such as collaborative and mobile robots are considered enabling technology for deploying Industry 4.0 (I4.0) [1]. Multi-robot systems in Figure 1 composed of one or more mani–pulator arms mounted on a mobile robot can offer new employment possibilities for industry automation, gre–atly increasing the flexibility of factory layout definition [2]. Unfortunately, there are also negative fallouts. Among them autonomy of the mobile robot is lowered by the energy consumption of the manipulator; there–fore, it is no more related to the travel distance (or the operating time, as mobile robots usually travel at constant speed).

The limited autonomy of the batteries mounted onboard mobile robots is the main constraint limiting the operational time of the robot inside the factory. It negatively affects potential employment on the line. When the battery charge is low, the robot must quit work, locate the nearest charging station and dock to it.

The estimate of energy level in order to schedule the time and frequency of recharging cycles is still an open problem, as it is scarcely correlated with the operating time of either of the two robots. Furthermore, the common method of estimation of the State Of Charge (SOC) by integrating the ampere-hour was demon–strated to be both inaccurate and unreliable [3].

It is possible to gather a huge amount of data to monitor the mobile manipulator through the application of the Internet of Things (IoT) [4]. Many of them are directly or indirectly related to energy consumption [5] and can be exploited to estimate the need for recharging. In small production, the future tasks of the robot could be unknown. It is difficult to build a consumption model analytically as it depends on uncountable parameters. On the contrary, ML presents advanced analytical capa–bilities for processing and analyzing large amounts of production data without requiring an underlying con–sumption model. ML allows predicting with reasonable accuracy the energy consumed based on an approximate description of the working tasks described only in terms of their effect on some Key Performance Indexes (KPIs).

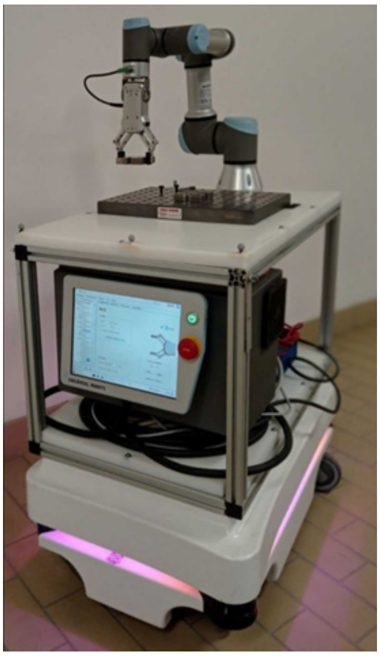

Figure 1. Mobile manipulator

Energy management belongs to the production control as far as robot halts during recharging can have

Received: October 2022, Accepted: November 2022
Correspondence to: Prof. Dario Antonelli
Dept. Of Management and Production Engineering
Politecnico di Torino - 10129 Torino, Italy
Email: dario.antonelli@polito.it
doi: 10.5937/fme2204752A

a significant impact on production performances. The recharging stops should be chosen and scheduled to optimize the production throughput. In turn, optimal scheduling relies on an accurate and reliable estimate of SOC. To provide such an estimate, the necessary actions include the selection of appropriate indicators of the impact of each work task on energy consumption and the continuous monitoring of the mobile manipulator. Simulating the future missions of the mobile manipulator, it is possible to validate the optimized recharging sequence and to predict the future energy demand. The present research aims at providing a robust and effective estimate of SOC for mobile manipulators.

In conclusion, the research questions the study proposes to answer are:

RQ1: classify robot work tasks according to energy consumption by using measurable KPIs.

RQ2: measure the mentioned KPIs on a mobile manipulator, during its work, by continuous monitoring.

RQ3: provide a reliable estimation of battery con–sumption, given a programmed work sequence.

2. STATE OF THE ART

The above-defined RQs have already been considered in the literature. The answer to RQ1 requires defining meaningful KPIs that are used not only to monitor and display energy consumption but also for planning, scheduling, predictive maintenance, quality control, etc. ISO 22400 defines standardized KPIs, of which 34 KPIs may be utilized in manufacturing ([6] and [7]). Among them, there are 5 that significantly affect the working performance of mobile and collaborative robots emp–loyed in the factory ([8]). They are reported in Table 1.

Table 1. Implemented KPIs on the monitoring framework

KPI	Formulation	Description
Cycle time	sequence start time - previous sequence start time	-
Cycles completed	Number of cycles completed	Increment every time a cycle finish
Wait time	Sum of robot wait times	Robot idle or waiting
Utilization	use time / total time	How long is the robot used against the potential use time
Efficiency	cycle time/ use time	% of productive work

The answer to RQ2 requires setting up a remote monitoring system providing the data needed to cal–culate the aforementioned KPIs. In [9], an intelligent monitoring framework has been implemented on a mobile robot, allowing the measurement of the consi–dered KPIs [10]. Mobile manipulators could be con–nected with the factory network through WIFI once the connectivity is guaranteed by adopting proper strategies as described in [4].

Therefore, the first two RQs have found a solution in the literature and require only the implementation of the present case study. On the contrary, to the author's knowledge, RQ3 has not yet been answered.

Estimation of battery consumption is necessary to adjust the scheduling of work tasks with battery management. The scheduling problem is not trivial and requires global optimization, but it already has a number of solutions in the literature.

As an example, [11] recurs to the Theory of Inventive Problem Solving (TRIZ) and the multiagent system (MAS). Other authors find the optimal travel route that considers additional charging stops. [12] propose a general constrained optimization algorithm by modeling the problem as an extension of the classic Travelling Salesman Problem.

In the present case study of the mobile manipulator, none of the existing solutions can be adopted straightforwardly because all of them rely on an estimate of SOC as a function of the length of traveled routes. On the contrary, in the present case, the battery could run out while the mobile robot is stationary at a workstation and only the manipulator arm is working. This prevents access to a charging point. Therefore, it is necessary to monitor the battery level and predict in advance when robotic arm operations will lead to a low battery situation. SOC and temperature for the battery of mobile robots can be monitored with the Internet of Things (IoT), as described by [13]. In literature already exist efficient predictive models of energy consumption for mobile robots, given the known state of charge, like the one employed by [14]. No model applicable to mobile manipulators was found.

Machine Learning (ML) was already used in mobile robotics to assist navigation [15] or to control the trajectories [16,17], while regression analysis assists predictive maintenance of the robot [18,19].

The advantage of ML is the possibility to include numerous input data from production in the learning phase, leading to a quality prediction even in the absence of a reliable analytical model of the process. Therefore, in the present study, ML assists in deve–loping a predictor for energy consumption in a mobile manipulator.

3. DEVELOPED METHODS

In former research [10], KPIs of the robotic cell and robot data were measured in real-time on the mobile manipulator and made accessible through a dashboard, as can be seen at the top of Fig. 2.

The framework's application layer integrates and visualizes ML results, status and battery data, and KPIs. A Node-RED program calculates KPI metrics from the onboard robot's data by user-defined functions. To com–pute KPIs, Node-red requests data (status, start time, uptime, downtime, etc.) using the MODBUS TCP/IP and RTDE protocols. The raw data received are trans–formed into human-readable data, and robot KPIs are computed.

Starting from this monitoring framework, a set of experiments was conducted to classify the robot's ope–rative tasks in terms of impact on energy consumption.

3.1 Experiment with a monitoring framework

The monitoring framework is composed of several interconnected pieces, including data acquisition from robots, communication layers, ML and KPI deployment, and dashboard integration. Figure 2 depicts the propo–sed framework of the ML-based monitoring system.

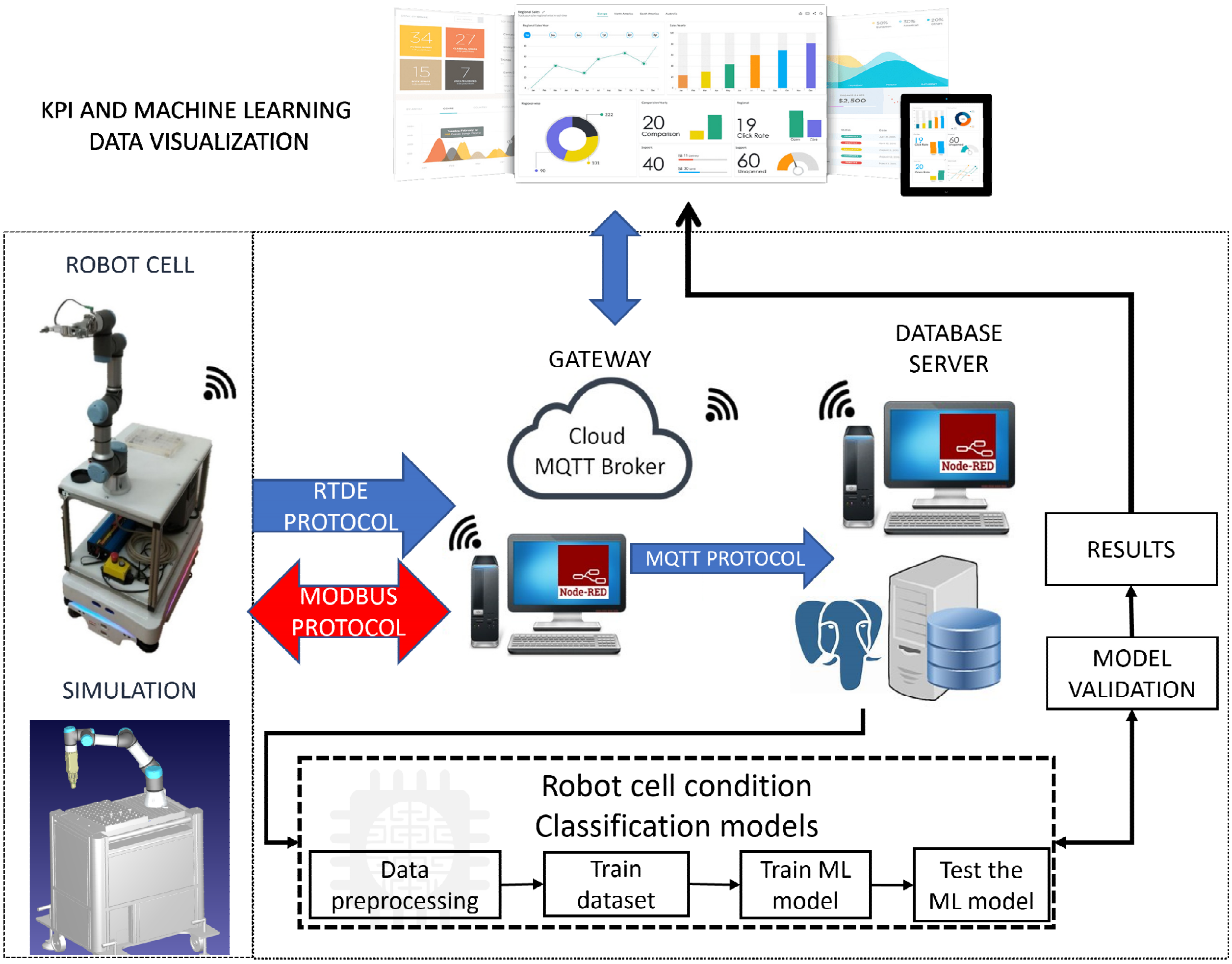

Figure 2. Intelligent Manufacturing Monitoring Framework

Table 2. Evaluation metrics and descriptions of the classification models

Metric	Formulation	Description
Log loss	$Logloss = -\frac{1}{N}\sum_{i=1}^{N}\omega_i\left(y_i \ln \ln\left(p_i\right)+\left(1-y_i\right)\ln\left(1-p_i\right)\right)$	N is the total number of observations in the equation; ω is the per-row user-defined weight; p is the predicted value, and y is the actual target value.
MSE	$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}\right)^2$	The mean squared error averages the squares of the mistakes or variances. N, the total number of observations; y_i actual target value; $\hat{y}$ predicted target value
RMSE	$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}\right)^2}$	The root means square error measures a model's ability to predict a continuous value. N, the total number of observations; $\hat{y}$ actual target value; $\hat{y}$ predicted target value
VAR	$VAR = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}\right)^2$	The statistical significance of each variable in the dataset in terms of its effect on the model. The variables are presented in descending order of relevance.

In the proposed framework, the ML best-trained model classifies robots' different conditions, such as if the mobile robot is "ON". The cobot is "OFF" if the mobile robot moves with the cobot "ON" and if the mobile robot is moving. The cobot program is running if the multi-robot system is moving with different weights, such as 30kg, 60 kg, and 90 kg, and to visualize the condition of the intelligent monitoring system in real-time.

The main components of the framework are the UR3-collaborative robot and MIR 100 mobile robot, which are connected through an ethernet cable. The MIR100 has WiFi and, therefore, can connect to gateways to get access to the Internet network. It supports the MQTT protocol since it allows direct access to the MQTT broker. MODBUS and RTDE protocols and interfaces are used for communication with robots. The RTDE protocol is used to acquire UR3 status data such as POWER OFF/ON, Emergency Stop, Protective Stop, the status of program-is it running, paused, or stopped, and other parameters necessary to compute KPIs. The received data are delivered to the cloud using an MQTT broker. Mobile robot registers contain discrete variables such as the On/Off status, emergency, battery status, distance run, uptime, length of missions, and PLC registers are used to calculate cycle, average times, etc...

3.2 Machine learning model

The Automatic machine learning (AutoML) approach is used to identify the robots' behavior and condition based on the data acquired. According to recent research, H2O AutoML outperforms other competing automated ML systems [20]. AutoML's robustness and efficiency were examined by [21]. According to them, in contrast to other automated models such as TPOT [22] and AutoKeras [23], AutoML is the quickest tool for training machine learning algorithms to generate a large number of ML models in a short period of time.

H2O AutoML provides supervised training of regression, binary classification, and multi-class classi–fication models on datasets [24]. The H2O AutoML platform key models include Generalized Linear Models (GLM), Distributed Random Forests (DRF), XGBoost, Gradient Boosting Machines (GBM), and Deep Learning. The H2O AutoML platform chooses one of three different models. The assessment metrics used for the classification models are listed and described in Table 2. Logarithmic loss (Log loss), mean squared error (MSE), root mean square error (RMSE), and variable importance (VAR) metrics were used to assess the performance of multinomial classification models.

ML classification models predict multi-robot system behavior and condition according to the battery, status data, and KPIs.

In the multi-robot system, the mobile robots' default dashboard does not provide information about manipulator power consumption. For this reason, ML models classify multi-robot systems' conditions and status according to the battery data of the mobile robot. Predicted ML class names are MIR_0, MIR_30, MIR_60, MIR_90, MIR_ON, MIR UR_P_R (mobile robot is steady while manipulator is doing some tasks), MIR_UR_ON (both mobile robot and manipulator are actively performing tasks).

3.3 Robotic cell monitoring and KPI integration

The dashboard deployment and integration constitute the application layer (top layer in Fig.2), presenting robot data, KPIs, and ML prediction results.

The dashboard is intended for a wider utilization apart from energy management. It may be utilized to analyze robot performance in production lines and for predictive maintenance applications. Furthermore, pro–duction managers can receive remote alarm signals for emergency stops, warnings, protective stops, etc.

KPI metrics are implemented utilizing user-defined functions in the Node-RED program. To compute KPIs, Node-red requests data (status, start time, uptime, down–time, etc.) using the MODBUS TCP/IP and RTDE proto–cols.

The raw data received are transformed into human-readable data and integrated into KPIs. The following results are provided: Figure 3 shows the dashboard of the mobile robot, whereas Figure 4 depicts the dashboard of the manipulator's arm.

The dashboard displays basic information (battery level, robot condition, mission/task distance, and dura–tion), cycle time (number of completed tasks, previous and average cycle time, beginning mission time), and selected KPIs (utilization, efficiency, and wait time).

3.4 Robotic cell simulation

The robotic cell simulation was developed to evaluate the mobile robot's energy consumption during the task execution and to provide the same simulation environment for the case study. MATLAB, Simulink, and the Robotics System Toolbox are used to simulate a robotic cell and case study environment.

As illustrated in Figure 5, five main blocks have been defined in the simulation environment (Simulink): Robot Scheduler, Planning, Control, Plant Model, and Visualization.

Robot Scheduler is the initial block. In this block, the robot's position on the map is updated, and the robot's mission is controlled using a Finite State Machine. The scheduler's input data are the positions of the Charging, Loading, and Unloading stations. The goal of the control block is to determine whether or not the robot is at the target position. The output data consists of the mobile robot's start and final positions, as well as a stop signal indicating that the robot is in the charging station position. The Finite State Machine describing the logic of charging management is integrated into the robot scheduler. The Planning block is a roadmap path planner object for the supplied environment map. The map of the mobile robot is used to produce a roadmap in the shape of a graph of feasible pathways based on free and occupied areas in the map.

The Planner block receives three inputs, start position, target position, and mobile working map, and generates a set of waypoints on the trajectory. The Control block utilizes the Pure Pursuit algorithm (PPA) [26] to simulate the trajectory of the mobile robot on the map. The control block calculates linear and rotational velocity signals based on the waypoints and the robot's current position.

If the robot reaches the goal, the zero velocity at the goal simulation block will halt it. The PPA simulation block has two inputs and two outputs. The pose indicates the robot's location on the xy-plane of the simulation map. The Lookahead distance is set in the simulation to fine-tune how closely the robot follows the trajectory. Path tracking of the robot is improved with a decreased Lookahead distance, which in our case study is 0.3 meters.

The plant model block consists of the Differential Drive Kinematic Model (DDKM), which is used to generate a vehicle model that may be used to simulate reduced mobile robot kinematics. Using the Differential

Drive Kinematics object, the differential drive kine–matics equations simulate a vehicle in which the wheels on the left and right may spin separately, as shown in Figure 6.

Finally, the robot visualizer simulation block receiv–es as inputs the robot position (from the plant model simulation block) and waypoints (from the Control block) and visualizes robot movements on the map.

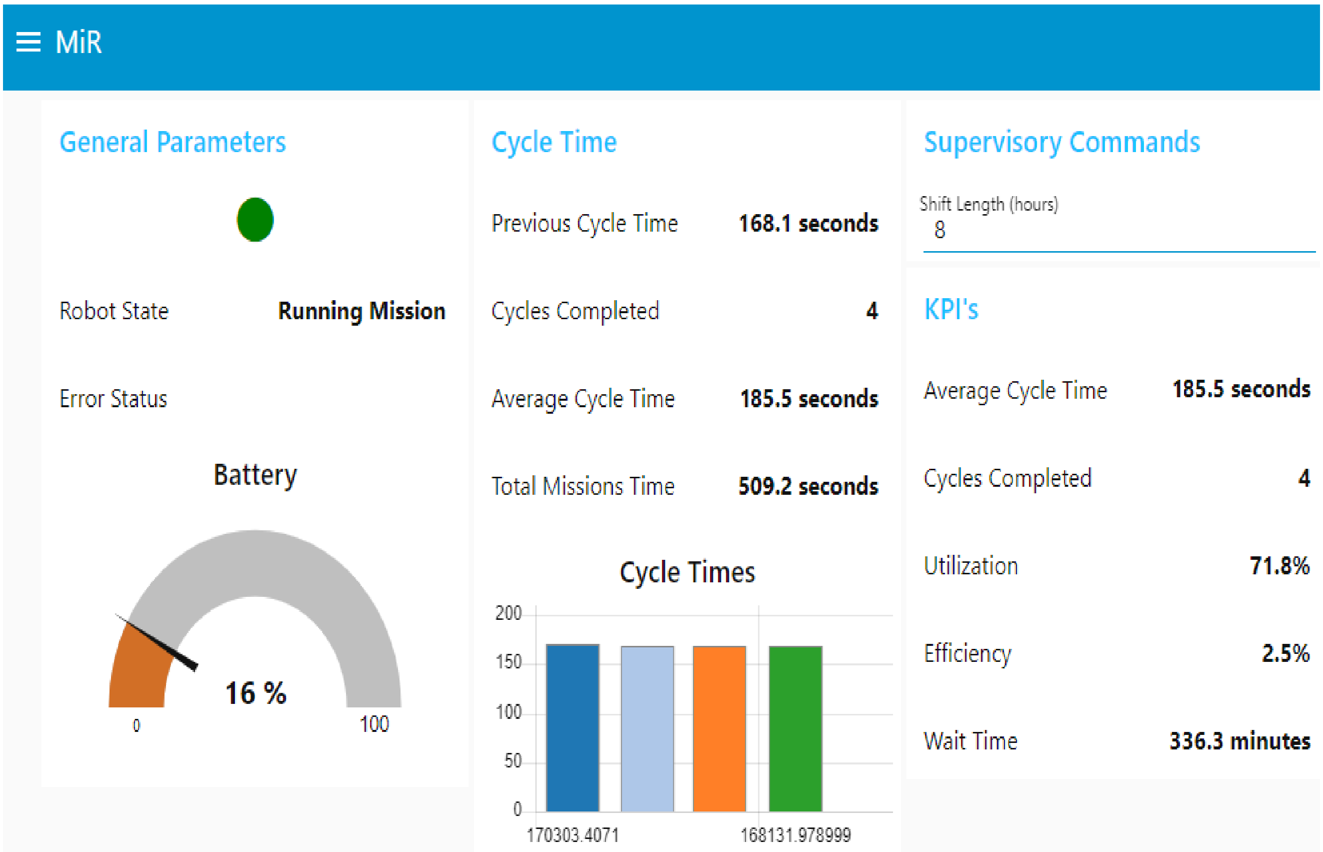

Figure 3. Mobile robot's dashboard

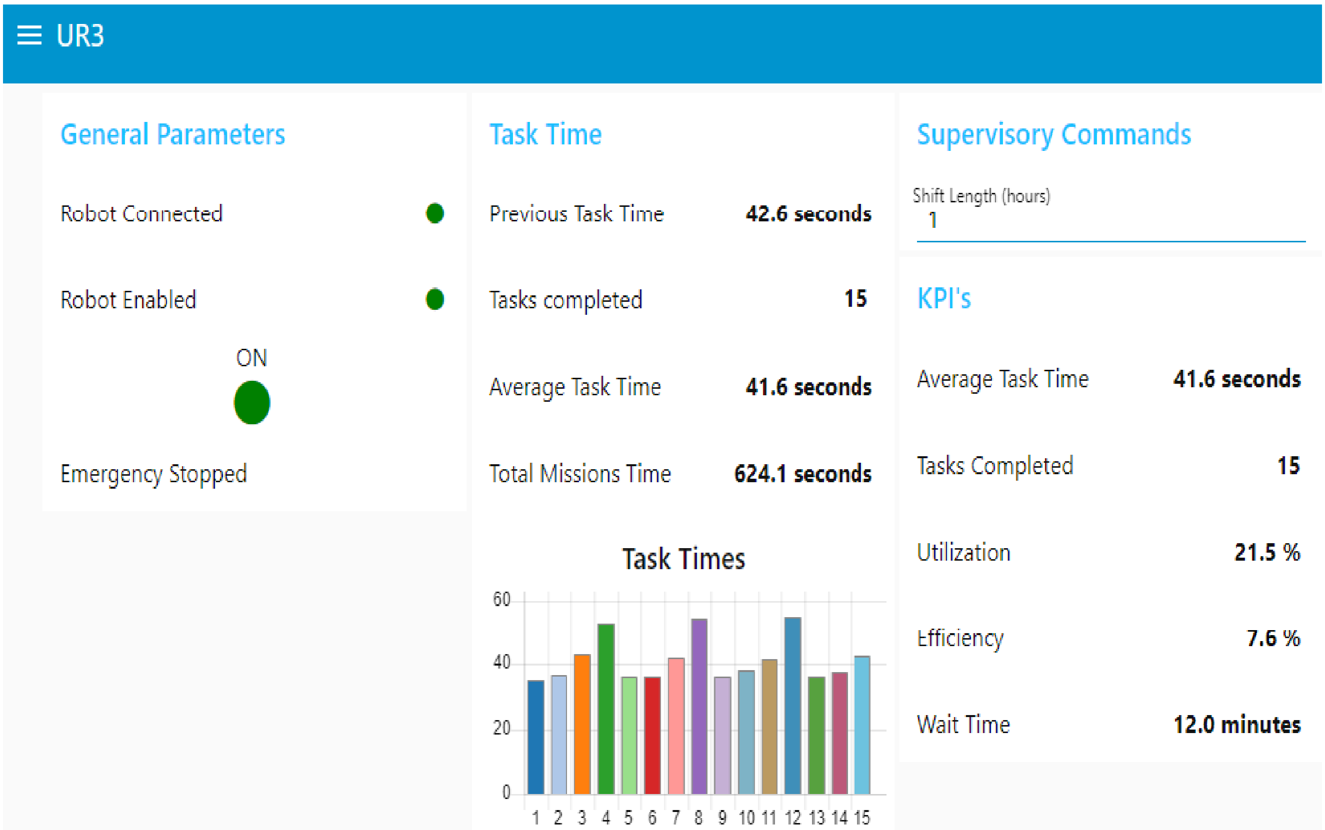

Figure 4. Manipulator's dashboard

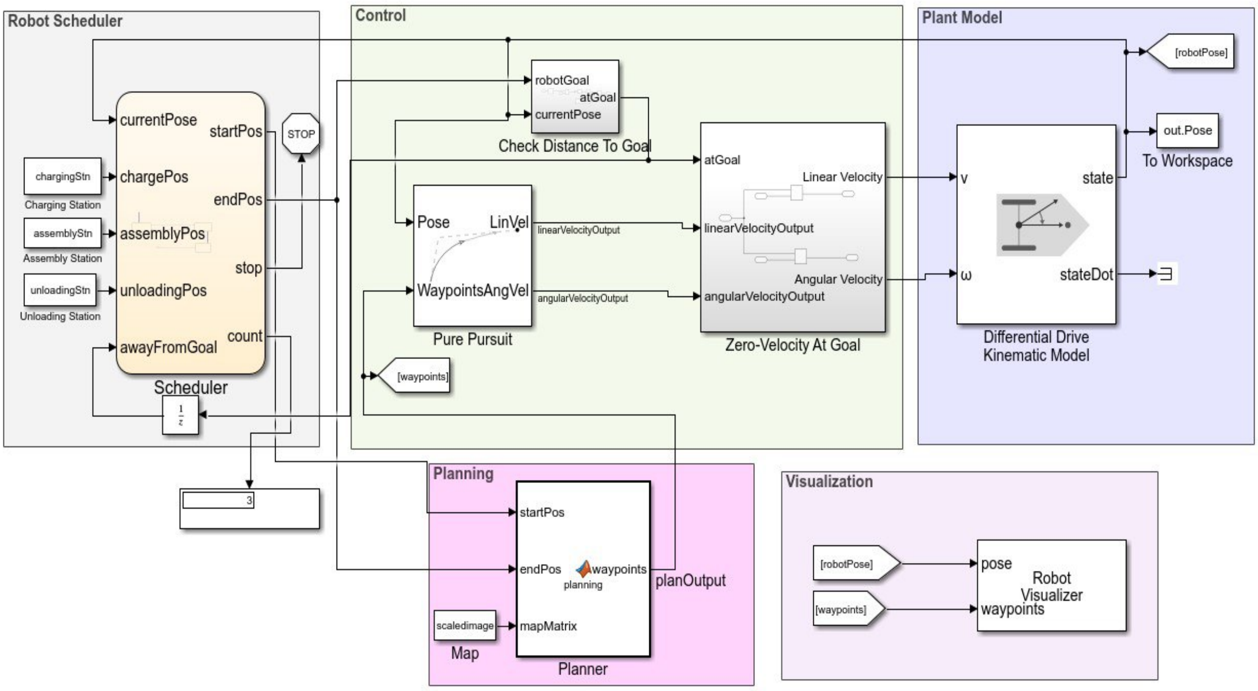

Figure 5. Energy management simulation blocks

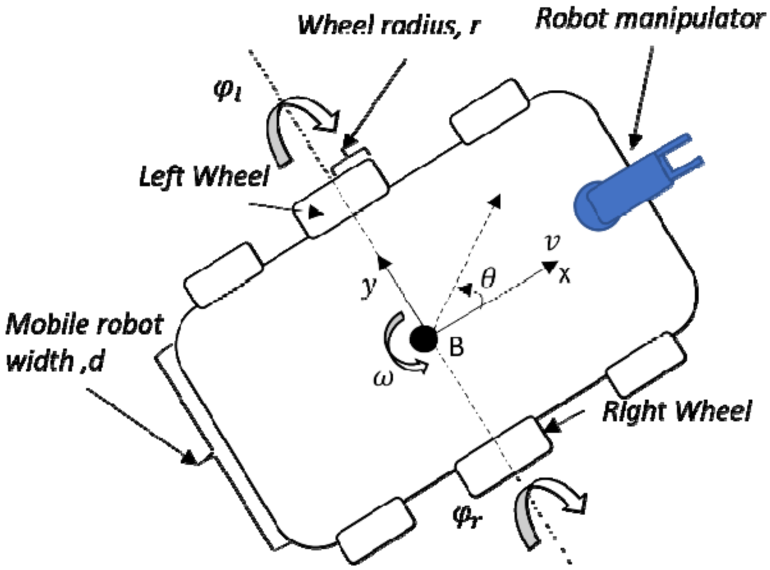

Figure 6. A simplified view of differential drive

4. EXPERIMENTAL SETUP

The robotic cell is located inside the Mind4Lab labora–tory of the Turin Polytechnic. MIR learns the map of Mind4Lab during preliminary navigations. Forbidden areas are added by hand. Figure 7 shows the final map of the experimental area.

The use case consists of an assembly operation on a desk followed by the transportation of assembled parts to a target unloading position.

The process is repeated until the battery of MIR decreases below 5%. After that point, the MIR is prog–rammed to go to the charging point and wait until it is fully charged. The mobile robot repeats the same work–flow for the loading/unloading tasks of 30kg, 60kg, and 90 kg weights. During the experiment, different condi–tional data of the MIR battery were acquired in the database. The acquired dataset is driven by ML classification.

The same case study is simulated using Matlab Simulink. In the simulation case, the Charging state is the robot's initial state on the map. MIR's battery is fully charged at the beginning. In the simulation, the battery level is managed by a Finite State Machine to be reduced by a constant value as it moves between states. Transitions between the Loading and Unloading states occur until the battery goes below 5%.

The Planner MATLAB function block uses the mobileRobotPRM route planner, which receives three inputs: a start location, a goal position, and the Mind4Lab laboratory map. The Pure Pursuit controller block employs the scheduled waypoints downstream by generating linear and rotational velocity signals based on the waypoints and the robot's current position. The Differential Drive Kinematic Model block creates a mobile robot model that is used to simulate simple vehicle kinematics.

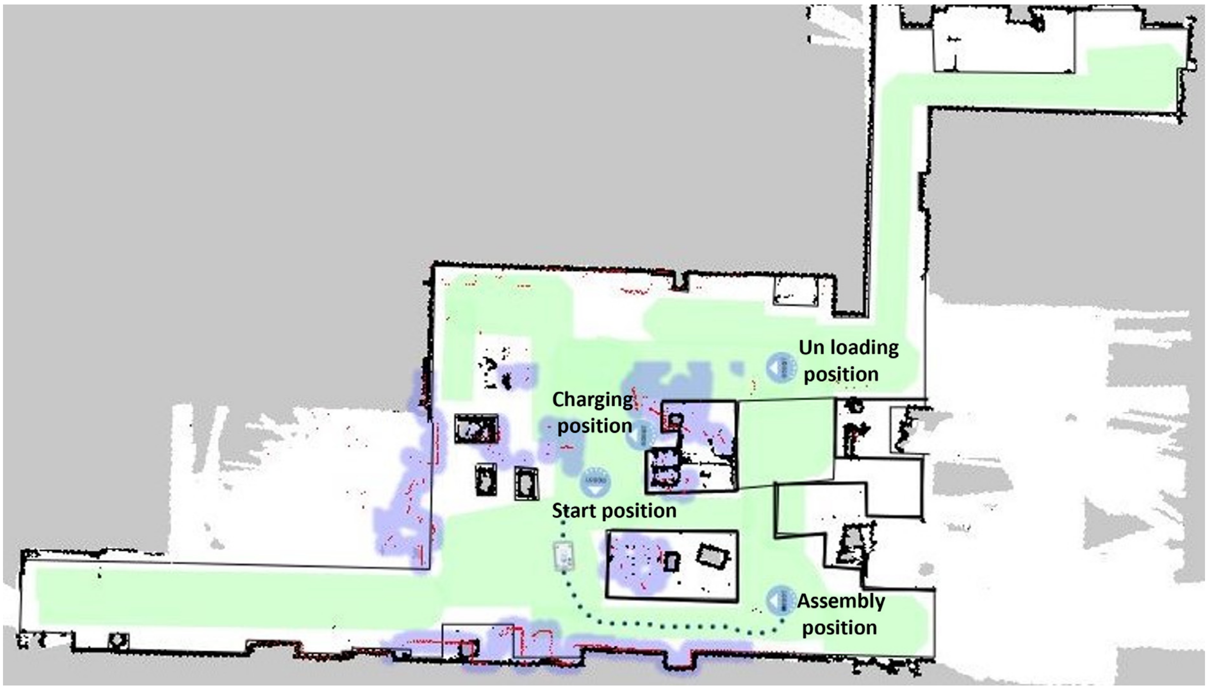

Figure 7. Map of the experimental area

5. RESULTS

In the proposed framework, H2O AutoML is used to determine which robot is operating based on the battery and status data of the robots. The AutoML function in H2O automates the process of identifying the best suitable models for a given dataset.

As the dataset is categorical, the multinomial distri–bution technique is utilized for training it. To assess the best performing models, error metrics were selected. Table 3 displays the results of the top ML models used to identify different robot conditions.

According to Table 3, the best models for the present dataset are GBM 2 and DRF 1. Gradient Boosting Machine (GBM) and Distributed Random Forest (DRF) models are powerful classification and regression techniques that forward-learn ensemble models and progressively construct regression trees on all aspects of the dataset in a completely distributed manner - each tree is constructed in parallel.

Table 3. Results of metrics on different ML models

Model id	Mean error	Log loss	RMSE	MSE	Training time(ms)
GBM_2	0	1.46	0.0145	2.11×10^{-4}	149
DRF_1	0	2.53	0	7.13×10^{-8}	74
XRT_1	0	2.48	0	6.85×10^{-8}	110
Stacked Ensemble	0	1.03	0.01	4.62×10^{-8}	4664
GLM_1	0	9.94	0.03	1.28×10^{-8}	5900
DL_1	0.09	4.75	0.17	3.04×10^{-8}	41280

The models' performance is excellent with low to null values for RMSE. The variables that impact more on the GBM model prediction more are robot battery capacity (rbcap), battery voltage (battvolt), and robot battery discharge time in seconds (rbtimesec). Figure 8 displays the variable importance determined by AutoML.

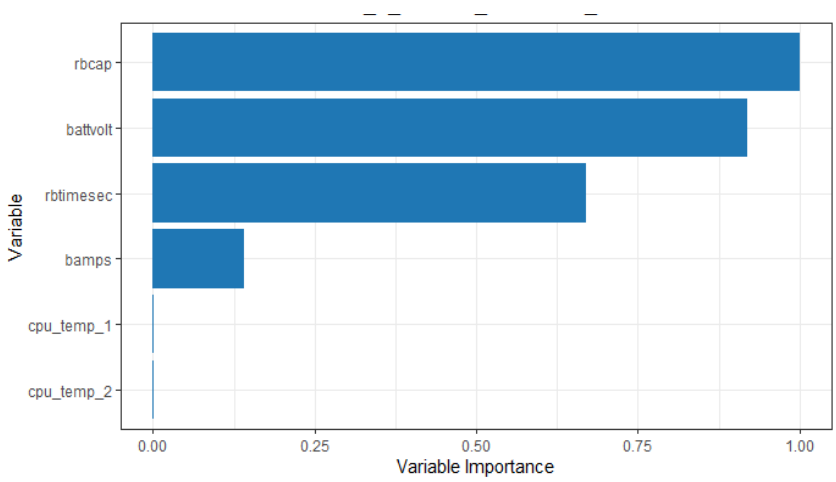

Figure 8. Variable importance for the GBM prediction model

In the case study, the majority of battery consum–ption occurred when the mobile robot was performing transportation procedures and the cobot software was operating. As a result of the case study, the battery goes from 100% to less than 5% after 4 hours and 13 minutes. The robotic cell completes 60 assembly opera–tions with a fully charged battery during this timeframe. During the real case study, the robot runs a total distance of 2.61 km.

Most of the battery consumption occurred during the transportation processes rather than assembly opera–tions. Furthermore, because the mobile robot's battery depletes quicker than 5%, it is suggested that a higher threshold level be used to recharge the battery, depending on the distance between the charging and unloading positions.

The simulation model provides the battery consum–ption rate as a constant number. The map of the real case study is uploaded to Simulink. In the simulation scenario of the robotic cell, the robot performed 58 assembly operations in 4 hours and 11 minutes. The simulation distance of the robot for the given setup is 2.25 km. The simulation and case study experiment results are quite similar. In the simulation, the battery consumption is reduced constantly. In the actual case, the battery consumption differs throughout the processes and operations. This might explain why the number of assembly procedures differs in the two scenarios. Figure 9 shows a comparison between the case study and simulation results.

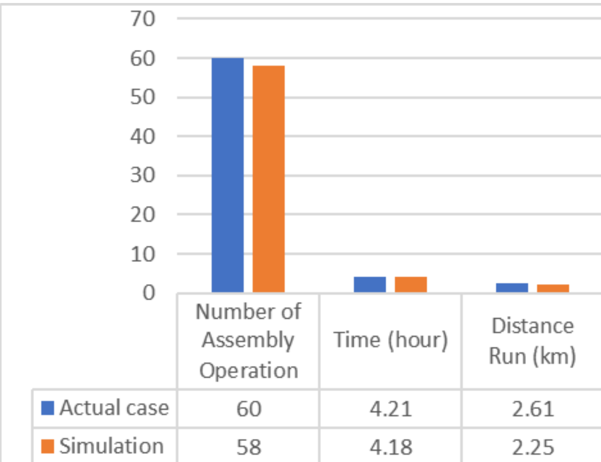

Figure 9. Comparison simulation vs. experiment

6. DISCUSSION AND CONCLUSIONS

The paper describes the possible exploitation of an intelligent monitoring system based on machine lear–ning, as recommended by [27]. The framework is organized into four layers: the smart devices (robots) layer, the network layer, the cloud layer, and the application layer. On the application layer, machine learning algorithms are applied to classify various conditions and behaviors of the robots.

The simulation model controls the SOC and predicts the battery consumption of the mobile robot when the manipulator is connected. The prediction model, trained by ML, guarantees the reliability of the system. The proposed system can be advantageous to manufacturers who are willing to integrate mobile and manipulator robotic cells in their production lines, specifically in logistics. Data acquisition systems and data sets acquired from the experiments can be utilized for predictive maintenance tools and algorithms. To prove the applicability and reliability of the framework experiment has been conducted on a case study in the laboratory.

ML model predicts robotic cell conditions if a mobile robot is working alone or with a manipulator based on the data coming from the robotic cell. The automatic ML platform tool AutoML H2O is used to classify different conditions, and according to the multinomial classification models, the DRF model is the best-performed model for our case study.

During the experiment, most of the battery consumption occurred while the mobile robot was executing transportation tasks, and the manipulator was in use. The robotic cell completed 60 assembly operations with a fully charged battery in this period. In the simulation scenario, the robotic cell performed 58 assembly operations. Current research fills a gap in modeling mobile manipulators, specifically addressing energy management. Nevertheless, there are several significant limits to this study, which are as follows: in terms of the ML monitoring framework for robotic cells, the framework is quite limited. Future investigations are required to add more labeling to the robotic cell's conditional dataset and integrate other ML tools such as unsupervised or reinforcement learning, as well as evaluate the system's integration with other production lines to assess robotic cell reliability, repeatability, robustness, and ease of use. Future research will focus on developing a digital twin of the mobile manipulator to adapt the production manage–ment to the actual working conditions continuously.

REFERENCES

[1] Putnik, G. D., & Ferreira, L. G. M. (2019). Industry 4.0: Models, tools and cyber-physical systems for manufacturing. *FME Transactions*, 47(4), 659-662.

[2] Aleotti, J., Baldassarri, A., Bonf`e, M., Carricato, M., Chiaravalli, D., Di Leva, R., ... & Zaccaria, F. (2021). Toward Future Automatic Warehouses: An Autonomous Depalletizing System Based on Mobile Manipulation and 3D Perception. *Applied Sciences*, 11(13), 5959.

[3] Rao, M. S., & Shivakumar, M. (2018). Overview of Battery Monitoring and Recharging of Autonomous Mobile Robot [J]. *International Journal on Recent and Innovation Trends in Computing and Communication*, 6(5), 174-179.

[4] Khateri, K., Pourgholi, M., Montazeri, M., & Sabattini, L. (2019). A comparison between decentralized local and global methods for connectivity maintenance of multi-robot networks. *IEEE Robotics and Automation Letters*, 4(2), 633-640.

[5] Chowdhary, R. R., Chattopadhyay, M. K., & Kamal, R. (2020). IoT-Based State of Charge and Temperature Monitoring System for Mobile

Robots. In *Innovations in Electronics and Communication Engineering* (pp. 401-413). Springer, Singapore.

[6] ISO22400-1 (2014). *Automation systems and integration -Key performance indicators (KPIs) for manufacturing operations management* – Part 1: Overview, concepts and terminology.

[7] ISO22400-2 (2014). *Automation systems and integration - Key performance indicators (KPIs) for manufacturing operations management* – Part 2: Definitions and descriptions.

[8] Top 5 Cobot Key Performance Indicators: HOW TO MEASURE AND IMPROVE THE PERFORMANCE OF COLLABORATIVE ROBOTS - https://info.universal-robots.com/how-to-measure-your-cobotsperformance-cobot-kpi

[9] Aliev, K., Traini, E., Asranov, M., Awouda, A., & Chiabert, P. (2021). *Prediction and estimation model of energy demand of the AMR with cobot for the designed path in automated logistics systems*. Procedia CIRP, 99, 116-121.

[10] Aliev, K., Antonelli, D., Awouda, A., & Chiabert, P. (2019). *Key Performance Indicators Integrating Collaborative and Mobile Robots in the Factory Networks*. In Working Conference on Virtual Enterprises (pp.635-642). Springer, Cham.

[11] Petrović, M., Miljković, Z., & Babić, B. (2013). Integration of process planning, scheduling, and mobile robot navigation based on TRIZ and multiagent methodology. *FME Transactions*, 41(2), 120-129.

[12] De Ryck, M., Versteyhe, M., & Shariatmadar, K. (2020). Resource management in decentralized industrial Automated Guided Vehicle systems. *Journal of Manufacturing Systems*, 54, 204-214.

[13] Djuric, A. M., Urbanic, R. J., & Rickli, J. L. (2016). A framework for collaborative robot (CoBot) integration in advanced manufacturing systems. *SAE International Journal of Materials and Manufacturing*, 9(2), 457-464.

[14] Yacoub, M. I., Necsulescu, D. S., & Sasiadek, J. Z. (2016). Energy consumption optimization for mobile robot motion using predictive control. *Journal of Intelligent & Robotic Systems*, 83(3), 585-602.

[15] Vuković, N., & Miljković, Z. (2009). New hybrid control architecture for intelligent mobile robot navigation in a manufacturing environment. *FME Transactions*, 37(1), 9-18.

[16] Mitić, M., Miljković, Z., & Babić, B. (2011). Empirical control system development for intelligent mobile robot based on the elements of the reinforcement machine learning and axiomatic design theory. *FME Transactions*, 39(1), 1-8.

[17] Hofbaur, M., K¨ob, J., Steinbauer, G., & Wotawa, F. (2007). Improving robustness of mobile robots using model-based reasoning. *Journal of Intelligent and Robotic Systems*, 48(1), 37-54.

[18] Aliev, K., & Antonelli, D. (2021). Proposal of a Monitoring System for Collaborative Robots to Predict Outages and to Assess Reliability Factors Exploiting Machine Learning. *Applied Sciences*, 11(4), 1621.

[19] Cardoso, D., & Ferreira, L. (2021). Application of predictive maintenance concepts using artificial intelligence tools. *Applied Sciences*, 11(1), 18.

[20] Halvari, T., Nurminen, J. K., & Mikkonen, T. (2020). Testing the Robustness of AutoML Systems. arXiv 2020. *arXiv preprint* arXiv:2005.02649.

[21] Feurer, M., Klein, A., Eggensperger, K., Spring–enberg, J., Blum, M., & Hutter, F. (2015). Efficient and robust automated machine learning. *Advances in neural information processing systems*, 28.

[22] Olson, R. S., & Moore, J. H. (2016). *TPOT: A tree-based pipeline optimization tool for automating machine learning*. In Workshop on automatic machine learning (pp. 66-74). PMLR.

[23] Jin, H., Song, Q., & Hu, X. (2019, July). *Auto-keras: An efficient neural architecture search system*. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 1946-1956).

[24] LeDell, E., & Poirier, S. (2020, July). *H2o automl: Scalable automatic machine learning*. In Proceedings of the AutoML Workshop at ICML (Vol. 2020).

[25] H2O Available online: https://docs.h2o.ai/h2o/lateststable/h2o-docs (accessed on 16/02/2022).

[26] Coulter, R. C. (1992). Implementation of the pure pursuit path tracking algorithm. Carnegie-Mellon UNIV Pittsburgh PA *Robotics* INST.

[27] Shah, V., & Putnik, G. D. (2019). Machine learning based manufacturing control system for intelligent cyber-physical systems. *FME Transactions*, 47(4), 802-809.

ИНТЕЛИГЕНТНО УПРАВЉАЊЕ ЕНЕРГИЈОМ ЗА МОБИЛНЕ МАНИПУЛАТОРЕ КОЈИ КОРИСТЕ МАШИНСКО УЧЕЊЕ

Д. Антонели, К. Алијев

Интегрисани роботски системи који комбинују манипулаторе са мобилним роботима пружају изванредне могућности побољшања за полуаутоматске процесе склапања које користи Индустрија 4.0. Фабричке операције су ослобођене ригидних ограничења распореда које намећу конвенционални фиксни роботи. Дакле, они уводе нове изазове у управљању циклусима пуњења јер потрошња енергије мобилних манипулатора није повезана само са пређеном раздаљином већ и са укупним извршеним задацима. Њена процена захтева системски приступ. У предложеном решењу, на броду је имплементиран интелигентни систем праћења. Подаци прикупљени на мрежи и кључни

индикатори учинка (КПИ) израчунати током радних задатака се користе од стране Машинског учења (МЛ) за оптимизацију циклуса пуњења енергије. Иако је развој интелигентног оквира за праћење за мобилни манипулатор био првобитни циљ истраживања, систем за надзор се овде користи само за управљање енергијом, остављајући простор за друге будуће апликације.