



Key Performance Indicators Integrating Collaborative and Mobile Robots in the Factory Networks

Khurshid Aliev, Dario Antonelli^(✉), Ahmed Awouda,
and Paolo Chiabert

Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Turin, Italy
{khurshid.aliev, dario.antonelli, ahmed.awouda,
paolo.chiabert}@polito.it

Abstract. Measuring performances of collaborative robots in Industry 4.0 applications is an open research area since the emergence of collaborative and mobile robots as a support for semi-automatic manufacturing processes. A compelling management problem is the definition of convenient performance measures on which to assess the new generation of robots, to improve process performances both at the robotic cell design stage and at the production stage. A consequent problem is to gather the required data to measure performances. Data must be obtained automatically and in real time. Different levels of communication protocols have to be harmonized in order to transfer data from robots and other factory machines to the cloud on the internet and eventually to the production control system. A case study allows to demonstrate the operation of data acquisition system for collaborative and mobile robots and the real-time monitoring dashboard. The outcome of the study is the gathering of data at field level, the evaluation of robot performances at machine level in order to execute the real time production control at factory level.

Keywords: KPI · ISO 22400 · Cobots · Mobile robots · Factory network protocols

1 Introduction

Collaborative robots (Cobots) and mobile robots are being introduced massively in the smart factory. Indeed, they have been considered as an advanced manufacturing solution and are an enabling technology in Industry 4.0 (I4.0) [1]. The main reason for their success is the combination of robot strength and endurance with the dexterity and the flexibility of human operator.

Going in detail, there are several assets of Cobots with respect to traditional robots: easy to program, safe interaction with human, possibility to share the workspace with other humans and robots, no need for fixed workplace surrounded by fences. This last feature means that cobots are easily movable, while mobile robots, by design, are able to move in the plant without following fixed trajectories. In the framework for workcell architecture design proposed by [2], this feature addresses specifically the operative perspective.

During the operation, it is practice to monitor the Key Performance Indicators (KPIs) of the process. Unfortunately, as the robot is no more constrained to stay in a specific cell, the measure of production performances is made both difficult to execute and to assign to a specific process. The choice of suitable indicators and their monitoring on a mobile robot or on a robot that does not belong to a specific workcell poses new and unprecedented problems [3] that are the target of this paper.

Transfer of robot data from the field level to the factory internet allows to leverage the exploitation of collaborative networks. In this network collaboration is the process of various agents working together on a voluntary basis by respecting a set of behavior rules [16]. Evaluation of Cobots' performance in such a collaborative network (CN) plays significant role in the domain of Industry 4.0 and can be supportive tool for a successful business.

Industrial machines and sensors from different vendors have different standardized protocols which can be challenging for data exchange and to align them in the CN. For that reason, this paper proposes a framework of connecting different robots and sensors with different protocols to increase the overall performance of the CN and to monitor performance of the robots [15].

In Sect. 2, the related literature is presented. In Sect. 3, the implementation of the data transfer in order to measure cobot performance is discussed. In Sect. 4, the KPIs suitable for cobot operations are defined and calculated for the case study of Sect. 5. Eventually, in Sect. 6 the study's outcomes are discussed and the future work is introduced.

2 Related Works

Low cost and precise way of measurement of the machine performance in industry grasped the attention of managers and researchers for many years. The integration of performance monitoring with maintenance systems can assist the manufacturer in achieving the goal of maintaining the performance of machines and support the control strategies [3].

Therefore, there are various proposed frameworks and systems for performance monitoring. Authors of [4] implemented and visualized KPIs according to ISO 22400 standards within a discrete manufacturing web-based interface to monitor and control an assembly line at runtime. In [5] review has been made by focusing on aspects of KPIs management, unification of taxonomy gathers relevant aspect highlighted by the literature and which captures the unique characteristics of KPIs in a more fully way. Observed characteristics can help researchers to decide about the most suitable solution for their requirements. General definition of the KPI scheme for monitoring on-line production process is described with 8-step iterative closed-loop model by authors of [6]. Real time monitoring energy consumption using five selected KPIs of the work cell that composed: robot, cabinet and conveyor successfully implemented in [7]. All above mentioned reviews mainly focused on models and simulations of the models. Moreover, proposed models and frameworks are not low cost which can be suitable for education and Industrial IoT (IIoT) applications. In next sections, we describe the open-source and reduced cost implementation of KPIs for collaborative and mobile robots and sensors within the factory networks.

3 High Level and Low Level Protocols in Factory Networks to Connect Machines into Cloud

Developing Industrial IoT networks which can connect and exchange data between sensors and machines to the backend systems are very challenging. Protocols provide device-to-device or device-to-server communications. There is a big variety of protocols used in today's industrial scene and the choice among these different protocols is application dependent and device specific. Industrial communication protocols are classified as Ethernet and non-Ethernet protocols: Non-Ethernet-Fieldbus protocols (Modbus RTU, Profibus DP), and Industrial Ethernet protocols (such as Modbus TCP and Profinet). Industrial communication protocols are low level or device management protocols. High level open source IoT domain data exchange protocols are Message Queuing Telemetry Transport (MQTT), CoAP (Constrained Application Protocol), Advanced Message Queuing Protocol (AMQP), Extensible Messaging and Presence Protocol (XMPP). In this section the implemented ones are discussed.

Modbus is an application layer protocol that defines the rules for organizing and interpreting data independent of the underlying communication layers and the transmission medium used [8]. *Modbus TCP/IP* is an alternative of the *Modbus* protocol where the message frame is encapsulated in a TCP/IP wrapper. TCP/IP refers to the Transmission Control Protocol and Internet Protocol, which provides the transmission medium for Modbus TCP messaging. The primary function of TCP is to ensure that all packets of data are received correctly, while IP makes sure that messages are correctly addressed and routed.

Modbus TCP combines a physical network (Ethernet), with a networking standard (TCP/IP), and a standard method of representing data which results in Modbus TCP being fully compatible with the already installed Ethernet infrastructure of cables, connectors, network interface cards, hubs, and switches. Simply stated, Modbus TCP shares the same physical and data link layers of traditional IEEE 802.3 Ethernet and uses the same TCP/IP suite of protocols.

Another protocol used in this work and resident to the IOT domain is *MQTT* and *REST*. *Message Queuing Telemetry Transport (MQTT)* is a protocol designed to connect the physical world devices and networks, with applications and middleware to develop Web applications in IT area. It is designed to minimize network bandwidth and device resource requirements whilst also attempting to ensure reliability and some degree of assurance of delivery, also MQTT protocol is a good choice for wireless networks that experience varying levels of latency [9].

MQTT uses the Publish/Subscribe Model which consists of three main components: publishers, subscribers, and a broker. Publishers are the lightweight sensors and devices that connect to the broker to send their data and go back to sleep whenever possible. Subscribers are applications or devices that are interested in a certain topic, or sensory data, so they connect to brokers to be informed whenever new data is received. The brokers classify sensory data in topics and send them to subscribers interested in those topics only. A device can behave as a publisher and a subscriber at the same time by publishing to specific topics and subscribing to others, the term MQTT client is used to distinguish publishers/subscribers from brokers.

Representational State Transfer (REST) or RESTful web services describes a set of architectural principles by which data can be transmitted over a standardized interface such as Hypertext Transfer Protocol (HTTP). REST does not contain an additional messaging layer and focuses on design rules for creating stateless services. A client can access the resource using the unique URI (Uniform Resource Identifier) and a representation of the resource is returned. With each new resource representation, the client is said to transfer state. While accessing RESTful resources with HTTP protocol, the URL of the resource serves as the resource identifier and GET, PUT, DELETE, POST and HEAD are the standard HTTP operations to be performed on that resource.

In next section we describe KPIs and most suitable KPIs for collaborative and mobile robots according to the standards.

4 Selection of KPIs for Cobots and Mobile Robots

Key performance indicators (KPIs) gives possibility to measure the performance and progress of the manufacturing machines and systems. Acquired data from different machines and things in Industry can serve to monitor and visualize energy consumption, planning and scheduling, maintenance, product quality, inventory, machines capability and etc. Standardized KPIs are defined in ISO 22400 and can be used in different fields of industry. On the other hand, implementation of those KPIs in real industrial applications are challenging. The title of the ISO 22400 is “Automation systems and integration—Key performance indicators (KPIs) for manufacturing operations management” that is collection of 34 KPIs. ISO 22400 standards composed of two parts: ISO 22400-1 describes basic overview and terminologies of the KPIs framework in the manufacturing [10]. ISO 22400-2 is dedicated to definitions and descriptions of 34 possible KPIs which can be used in industry [11]. In [12] five top KPIs that can improve the performance of cobots and robots are defined and possible measurement techniques are described. Moreover, importance of standard metrics such as Overall Equipment Effectiveness (OEE) that is standard for measuring manufacturing productivity is explained. Taking account above mentioned standards, following five KPIs has been selected to implement for cobots and mobile robots in our case study:

Cycle time – Summation of the steps time of the robot while executing sequence of tasks

$$\text{Cycle time} = \sum \text{Time to execute a single process} \quad (1)$$

Cycles completed – increment a variable every time a cycle is completed

$$\text{Cycle completed} = \sum_i^n C_i = C_1 + C_2 + C_3 + \dots + C_n \quad (2)$$

Wait time – percentage of time that the robot is waiting or time that not executing productive tasks. Wait time is defined as sum of all individual wait times:

$$\text{Wait time} = \sum \text{Robot static time} = \text{Planned use time} - \sum C_i \quad (3)$$

Planned Use Time is defined from the Cycle Time per unit, multiplying by the number of products that are going to be processed.

Utilization – measures how long a robot is being used compared to how long it could be used. Utilization for robots can be defined as following formula:

$$\text{Utilization} = \frac{\text{Total Use time}}{\text{Total time}} \quad (4)$$

Efficiency – percentage of time that the robot performs productive work while running a program.

$$\text{Efficiency} = \frac{\sum C_i(\text{Total Cycle Times})}{\text{Total Use time}} \quad (5)$$

5 Implementation and Application to a Case Study

5.1 Collaborative Network Integrating Data from Cobots and Mobile Robots

The proposed CN to collect data from industrial machines and to transfer them to the cloud is presented in Fig. 1. The framework is composed by three layers. In the bottom layer all industrial machines are located (here the mobile robot MIR100 and the cobotUR3) together with external sensors like temperature, proximity.

Data acquisition from MIR100 and UR3 has been executed using the Modbus TCP/IP protocol where the robots are a Modbus Server and the RPi is a Modbus Client the client sends requests to read specific registers available on the robot's internal memory, the robot responds by providing the value of the requested register. Both UR3 and MIR100 registers can hold discrete variables such as On/Off status. Moreover, UR3 has access to analog values such as joint velocities, angles and also robot temperature and input current. At the same time, it was possible to retrieve battery level, positions, orientation data and control the mobile robot through MODBUS TCP/IP.

The middle layer is responsible for gateway and networking the machines and things using open source hardware and software. Raspberry Pi is used as operating system that runs open source software such as Node-red (used to create and visualize live data on browsers), Mosquitto (message broker that implements MQTT protocol) and NGROK (reverse proxy software that creates a secure tunnel on a local machine along with a public URL). NGROK is used to allow the Node-Red Dashboard to be accessible via a public URL on the internet. The ESP8266 is Wi-Fi module with a full TCP/IP Stack and microcontroller capability. The module has been operated as an MQTT client that acquires sensor data from the sensors connected to it and then transmits the data to the Raspberry Pi via MQTT protocol.

On the top layer of the framework, all data is transmitted to the Thingspeak platform [13] to store data and perform analytics. Moreover, Node-red dashboard is used to

visualize KPIs on the browsers. The implementation and resulting graphs of this component are presented in further sections.

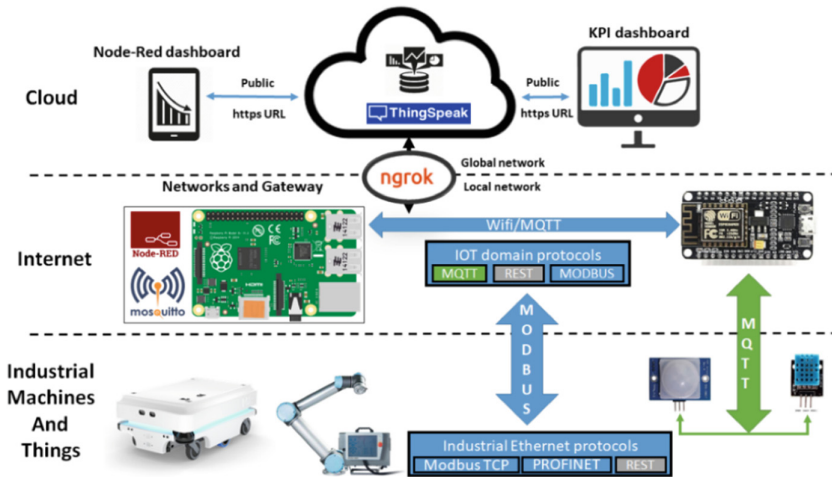


Fig. 1. Architecture of the CN for sharing information among factory machines.

5.2 Calculation KPIs from Robot Data and Their Visualization on the Dashboard

In case study, KPIs of the mobile robot has been evaluated and implemented on the dashboard. KPIs of the mobile robot was implemented for the transportation tasks of the metallic pieces to the UR3 manipulator [14]. Basically, MIR100 has 10 missions to transport metallic piece and the aim of the experiment was to estimate KPIs of the MIR100 in this demonstrator. The sequence of the mission on the dashboard of the MIR100 is shown in Fig. 2.

The experiment was conducted in the following way: to demonstrate KPIs of the MIR100 on the dashboard and on the cloud, we considered that MIR100 was working for an hour. Total time is 60 min but the robot was On for 30.78 min which results in a total use time of 30.78 min. The robot spent 6.3 min to transport all workpieces and completes its task with a total cycle time of 6.3 min. Taking account values coming from MIR100 and using above mentioned KPIs formulas, Utilization and Efficiency of the MIR100 have been estimated as 51% and 20.5% respectively.

$$Utilization = \frac{Total\ Use\ time}{Total\ time} = \frac{30.78\ min}{60\ min} * 100 = 51\%$$

$$Efficiency = \frac{\sum C_i(Total\ Cycle\ Times)}{Total\ Use\ time} = \frac{6.3\ min(378.1\ s)}{30.78\ min} = 20.5\%$$

Cycle time, cycle completed and wait time, utilization and efficiency have been calculated triggering registers of the MIR100 and programmed on the node-red using user

defined functions (formulas has been converted into functions) and transferred to the dashboard.

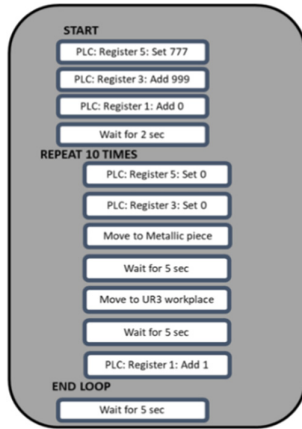


Fig. 2. Sequence of the missions of the mobile robot on the control dashboard.

On the top level of the proposed framework, KPIs are integrated and visualized. KPI formulas are implemented on the Node-red software with user defined functions. Node-red requests necessary data (state, start time, uptime, downtime etc.) through MODBUS TCP/IP to calculate KPIs. Received raw data are converted in human readable data and robot KPIs are calculated. Results are sent to the Node-red dashboard (Fig. 3). Dashboard reports general information (battery level, state of the robot, distance and length of the mission), cycle time (number of CT completed, previous and average cycle time, initial mission time), Supervisory Commands (pause, play, cancel, clear mission queue) and selected KPIs (utilization, efficiency and wait time).

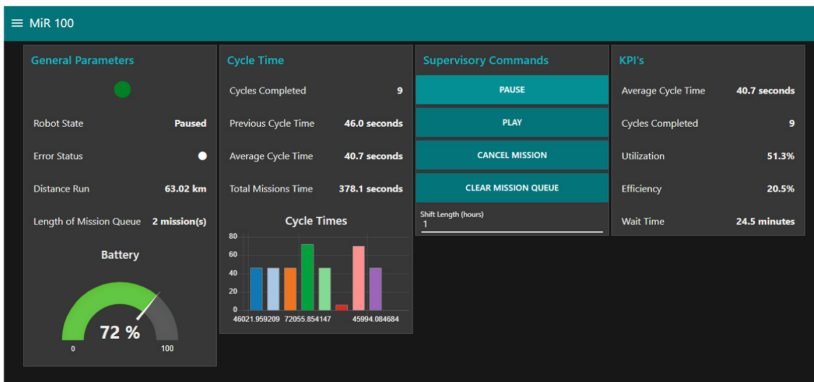


Fig. 3. KPIs of the mobile robot on the Node-red dashboard. (Color figure online)

6 Conclusions and Future Works

The paper demonstrates the practical integration and implementation of KPIs inside a CN by gathering data from field level and using them to measure performance at factory level. Developed system is composed by open access tools and protocols that are a suitable demonstrator of Industry 4.0 for academics and manufacturers. Future work will be the application of the measuring system in an actual industrial contest to assess the robustness of the system in factory conditions.

References

1. Lorenz, M., et al.: Man and Machine in Industry 4.0: How will Technology Transform the Industrial Workforce Through 2025. The Boston Consulting Group, Boston (2015)
2. Djuric, A.M., et al.: A framework for collaborative robot (CoBot) integration in advanced manufacturing systems. *SAE Int. J. Mater. Manuf.* **9**(2), 457–464 (2016)
3. Weiss, B.A., Qiao, G.: Hierarchical decomposition of a manufacturing work cell to promote monitoring, diagnostics, and prognostics. In: ASME 2017 12th International Manufacturing Science and Engineering Conference. ASME (2017)
4. Ramis Ferrer, B., Muhammad, U., Mohammed, W., Martínez Lastra, J.: Implementing and visualizing ISO 22400 key performance indicators for monitoring discrete manufacturing systems. *Machines* **6**(3), 39 (2018)
5. Domínguez, E., Pérez, B., Rubio, Á.L., Zapata, M.A.: A taxonomy for key performance indicators management. *Comput. Stand. Interfaces* **64**, 24–40 (2019)
6. Rakar, A., Zorzut, S., Jovan, V.: Assessment of production performance by means of KPI. In: Proceedings of the Control, pp. 6–9 (2004)
7. Zhang, B., Postelnicu, C., Lastra, J.L.M.: Key performance indicators for energy efficient asset management in a factory automation testbed. In: IEEE 10th International Conference on Industrial Informatics, pp. 391–396. IEEE (2012)
8. Modicon, I.: Modicon Modbus protocol reference guide, pp. 28–29. North Andover, Massachusetts (1996)
9. OASIS Standard: MQTT ver. 3.1.1 (2014). <https://docs.oasis-open.org/mqtt/mqtt/v3.1>
10. ISO22400-1: Automation systems and integration - key performance indicators (KPIs) for manufacturing operations management - Part 1: overview, concepts and terminology (2014)
11. ISO22400-2: Automation systems and integration - key performance indicators (KPIs) for manufacturing operations management - Part 2: definitions and descriptions (2014)
12. Top 5 Cobot key performance indicators: how to measure and improve the performance of collaborative robots. <https://info.universal-robots.com/how-to-measure-you-cobots-performance-cobot-kpi>
13. Official website of the Thingspeak platform. https://thingspeak.com/pages/learn_more
14. Aliev, K., Antonelli, D.: Analysis of cooperative industrial task execution by mobile and manipulator robots. In: Trojanowska, J., Ciszak, O., Machado, J.M., Pavlenko, I. (eds.) MANUFACTURING 2019. LNME, pp. 248–260. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-18715-6_21
15. da Piedade Francisco, R., Azevedo, A., Bastos, J., Almeida, A.: Using key alignment indicators for performance evaluation in collaborative networks. In: Camarinha-Matos, L.M., Pereira-Klen, A., Afsarmanesh, H. (eds.) PRO-VE 2011. IAICT, vol. 362, pp. 159–166. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23330-2_18. {hal-01569982}
16. Talukdar, S.N.: Collaboration rules for autonomous software agents. *Decis. Support Syst.* **24** (3–4), 269–278 (1999)