

Cloud-based Industrial Cyber-Physical System for Data-driven Reasoning. A Review and Use Case on an Industry 4.0 Pilot Line

A. Villalonga, G. Beruvides, *Member, IEEE*, F. Castaño and R. Haber, *Member, IEEE*

Abstract:

Nowadays, reconfiguration and adaptation by means of optimal re-parametrization in industrial cyber-physical systems (ICPS) is one of the bottlenecks for the digital transformation of the manufacturing industry. This work proposes a cloud-to-edges-based ICPS equipped with machine learning techniques. The proposed reasoning module includes a learning procedure based on two reinforcement learning techniques, running in parallel, for updating both the data-conditioning and processing strategy and the prediction model. The presented solution distributes computational resources and analytic engines in multiple layers and independent modules increasing the smartness and the autonomy for monitoring and control the behavior at shop floor level. The suitability of the proposed solution, evaluated in a pilot line, is endorsed by fast time response (i.e., 0.01s at the edge level) and the appropriate setting of optimal operational parameters for guaranteeing the desired quality surface roughness during macro and micro milling operations.

Index Terms— Cloud-to-edge-based concept, reinforcement learning, Industrial Cyber-Physical Systems, Machine-learning library, Condition-based monitoring, smart manufacturing, Industry 4.0.

I. INTRODUCTION

THE digitalization roadmap is a top priority in the global manufacturing industry. Information Technology (IT)-based solutions in the industrial domain have been for decades employed for monitoring and control of industrial processes [1, 2]. Factories can now be visualized and monitored via web browsers and other interfaces from remote locations. The R&D community is continuously reviewing state-of-the-art approaches that support the operational efficiency of large-scale systems [3, 4].

Generally, IT-based approaches are in greater demand, due to the impact and the requirements of the well-known Industry 4.0 (I4.0) paradigm [5]. Revolutionary changes have been evolving against the backdrop of Industrial Cyber Physical Systems (ICPS) [6, 7], the Industrial Internet of Things (IIoT), and smart-manufacturing solutions, in real-time/non-real-time communication, data exchange from multiple devices, factory integration, information flow (traceability), system monitoring, anomaly detection and prediction, and actions in support of knowledge-based decision-making [8].

Within that area, smart solutions for different manufacturing industries have been identified, to increase connectivity, interactivity, resource sharing, and data-collection from machines, robots, and other devices, as well as, the digitalization of processes, production systems, and product traceability [9, 10]. The I4.0-based solutions have introduced new paradigms into manufacturing environments such as: Service-oriented Architecture (SOA) [11], Reference Architectural Model Industrie 4.0 (RAMI 4.0) [3], and cloud-based service-oriented architectures [12, 13]. Nevertheless, there are still several difficulties over creating, consuming, representing, and inferring information for data inputs from heterogeneous sources [14]. Standards such as Asset Administrator Shell [15] are focused on describing the attribute of each asset in an optimal (plug and work) way, in order to interconnect layers in the factory of the future. Finally, communications security and data-transfer safety between systems and assets operating down the industrial chain should all be part of these new digital solutions [16]. New strategies such as blockchain lay the foundations of the new paradigm for safe data exchange in ICPS ecosystems [17].

At the same time, other emergent fields have, over past decades, been related with Machine-Learning (ML) techniques with practical applications. Rapid advances of the newly developed algorithms, computing resources, and open source community make ML-based solutions a key player for the digital transformation of industry [18, 19]. Following the digital transformation roadmap, it can be assumed that the factory of the future, will not only be fully-connected and digitalized, but will also be smarter than the current manufacturing environments [20].

Trends found in many industrial sectors reveal that sensor reliability, condition-based monitoring, anomaly detection, and prediction, prescriptive actions (anticipation), and knowledge-based decision-making will play an important role in smart manufacturing scenarios [21, 22]. Furthermore, the combination of unsupervised and supervised learning, clustering, and metaheuristic techniques, and the new self-functionalities will yield a new set of methods and know-how for advancing our understanding of these complex cutting-edge industrial manufacturing processes [23, 24].

In this context, ML-based solutions can transform the human perspective towards the interpretation of variable relations,

physical phenomena, cause-effect analysis, and decision-making effectiveness beyond the current state of the art [25]. The expectations will be faster, customizable, more efficient and clean (green manufacturing), but at the same time, more accessible and better aligned with the demands of society [26].

This paper will present a data-driven reasoning method comprising learning and the optimization of computational procedures, for enabling re-parametrization of the edge modules, based on the available process knowledge. The unsupervised learning procedure consists of two Q-learning strategies for simultaneously updating both the data conditioning and processing strategy and the prediction model. A cloud-to-edge-based industrial cyber-physical system (ICPS) for smart manufacturing is also introduced for supporting the data-driven reasoning approach. The industrial target is to improve the prediction of the surface roughness quality in macro and micro-milling operations by means of re-parametrization of the data-conditioning and processing strategy, and updating the prediction model.

The paper is structured as follows. After the introduction, the role of machine-learning for the digital transformation of manufacturing is analyzed. Then, the proposed cloud-based architecture is described in section III. Subsequently, in section IV, the Industry 4.0 pilot line validation is introduced. Finally, the conclusions are presented in section V.

II. THE ROLE OF MACHINE-LEARNING IN THE DIGITAL TRANSFORMATION OF MANUFACTURING

In the previous section, several approaches to the digital transformation of the industry were described. Even so, the common misconception in many industrial environments that “Industry 4.0 is only related with a fully-automated (digitalized) factory” must be dismissed. The question that should be asked is: “whether factory A, which has integrated the full vertical value chain of information that is collected from the production line (even in real-time), and transmitted to a cloud-based platform, is a highly digitalized factory?” The answer is “Yes” in the opinion of the authors, because one main function of the digitalization paradigm is the digitalization of information from multiple sources, and formats based on standards and open protocols, to connect the entire manufacturing value chain [27]. However, with these actions alone, the answer to the question “Is it a smart factory?”, will be probably “No”. Fig 1. attempts to clarify the above affirmations in a simple way. Indeed, Fig. 1 shows the digitalization degree of manufacturing companies in the Y axis. Nevertheless, this is only a conceptual representation of the company maturity level and the required steps to be addressed according to the Industry 4.0 roadmap for the digital transformation. Furthermore, this degree of digitalization (“full-digitalized vertical company”) is a medium-term target for manufacturing companies.

Nowadays, several companies have adopted Industry 4.0-based platforms to increase product traceability, access to information, and real-time visualization of the current status of machines, factory production, and assembly lines [28]. Factory A represents a full-digitalized vertical company, able to provide the above-mentioned functionalities. However, the conclusions

from line operators to their executives will only be based on what they see. To do so, actions will only be taken on the basis of expertise acquired from the management of issues and trends displayed on the UX/UI dashboards.

On the other hand, factory B introduces horizontal growth in the form of ML core solutions, which learn new patterns from the system, correlate events (root cause analysis), predict a potential status, support (based on data-driven decisions) the decision-making process, and optimize the sequence of actions to improve the manufacturing process [29].

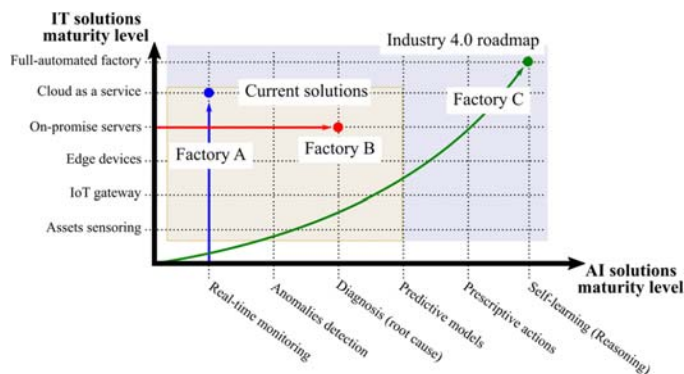


Fig 1 . Maturity level of IT and AI-based solutions in the manufacturing scope.

Despite the potential benefits of ML-based solutions there are some limitations such as limited amounts of data, appropriate and structured data formats, limited time-stamped or event-stamped data for a proper description of the process, and the consideration of other key performance indicators (e.g., spare-part delivery, logistics, maintenance scheduling). There is, therefore, a clear need for new methods to fill in computational platforms, whether provided through web services or onsite, for adding new capabilities to smart manufacturing (see Factory C).

In that sense, Table I summarizes some reported ICPS architectures for smart manufacturing. A complete review of all the approaches developed to date goes beyond the scope of this paper, due to the large amount and the diversity of methods and operating systems, and the main frameworks and techniques, etc. [30].

TABLE I. REVIEW OF STATE-OF-THE-ART IN ICPS ARCHITECTURES FOR SMART MANUFACTURING. FIT AND GAPS.

Project acronym	Main features	Main approaches	Reference
SOCRADES	Factory automation, service-oriented infrastructure	Collaborative and heuristic methods at device and application levels	[31]
CONMICRO	Control, self-learning, middleware.	Artificial cognitive systems.	[32, 33]
PLANTCockpit	Automation, web-services, production and logistics cockpit	Statistics methods.	[34, 35]
COGNETCON	Control, Self-optimization, federate architecture	Evolved neurofuzzy systems.	[36, 37]

IMC-AESOP	Automation, Services-oriented, cloud services, service mediators design,	Modular collaborative systems.	[38]
ARROWHEAD	engineering, and operation of large automation system, web-services	Cooperation as system of systems, interoperability, verification.	[39]
PRIME	Multi-agent technology	Self-organization.	[40]
GRACE	Quality control, Service-oriented architecture, Multi-agent technology	self-adaptation method, quality control method.	[41]
IDEAS	Evolvable assembly architecture, distributed control applications	Evolvable systems, self-reconfiguring.	[42]
REBORN	Flexible assembly architecture, machine automation	Symbiotic assembly method, evolutionary method.	[43]
PERFORM	Distributed control systems, industry-oriented system architecture, middleware	plug-and-produce concept, human factor in the loop.	[44]

A cloud-based ICPS configuration is proposed, in order to fill this gap, designed and validated on an Industry 4.0 pilot line in which multiple edge-to-cloud solutions are introduced. In addition, another important motivation behind this ICPS proposal are the current solutions such as Enterprise Resource Planning (ERP) systems, dynamic manufacturing scheduling, and logistics, amongst others, that are currently migrating to the cloud as a services platform, to deploy real-time control for all stakeholders involved in end-to-end business values [45].

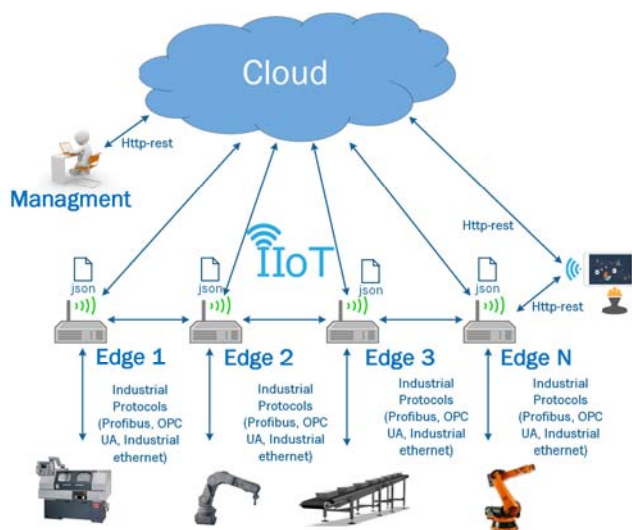


Fig 2. Overall diagram of the cloud-based ICPS

III. CLOUD-BASED ICPS DESCRIPTION

The proposed cloud-to-edges-based ICPS is based on a generic cloud-edge system (see Fig 2.).

Each edge has monitoring functions, based on signals and events analysis, the operational features, and the performance of the production chain elements. Process monitoring is based on three fundamental subsystems: (i) signal processing; (ii) predictive model; and, (iii) local decision-making embedded in the local edges. The location of these subsystems into the architecture is shown at the bottom of the edge device in Fig. 3.

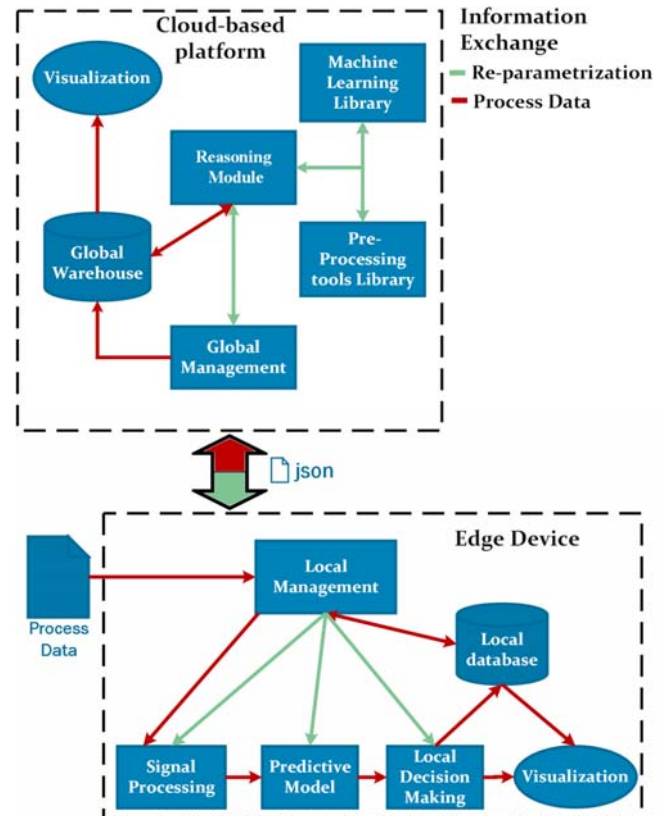


Fig 3. Scheme of the cloud-edge interaction and internal module interconnections

A. Cloud-based platform

The cloud layer takes charge of supervising the distributed set of edges, based on the configurations at plant level. The edge parametrization can be done at any time, providing the system with a dynamic re-parametrization mechanism, based on optimal key performance indicators estimated by the reasoning layer. In this way, the cloud-based platform will modify the configuration of the different edge parameters, depending on the behavior of the process. For this purpose, the edge modules will require continuous information flows on the process, from the cloud, thereby activating the reasoning module that will compute the most appropriate set of parameters for every recommended action. The cloud is composed of six main subsystems, as shown in Fig. 3.

- *Global warehouse*: the global data, supplied by each process/machine component connected to the edges, are

stored in a structured database that is used for the reasoning and the management phases. The same database also stores data to validate the training and the performance indices of the different predictive models.

- *Machine learning library*: consists of a set of modeling techniques based on machine learning methods to generate predictive models of each component of the plant.
- *Pre-processing library*: composed of different time-domain methods such as peak-to-peak value, mean square root, kurtosis, and frequency domain techniques, such as Fast-Fourier transform and Wavelet transform for data pre-processing.
- *Reasoning module*: supported by the hybridization of a reinforcement learning procedure (i.e., Q-learning method) and an evolutionary algorithm in charge of yielding predictive modules with optimal parametrization. The reasoning process is triggered by the arrival of new data from the production line.
- *Global management module*: in charge of the synchronization of all the different subsystems that constitute the cloud platform.
- *Visualization*: based on a web service, a customized visualization can be streamed of each machine component in action, as well as, their overall performance.

B. Edge

The local edge is responsible for monitoring and diagnosis on the basis of sensory information gathered from different subsystems on the shop-floor, such as spindle temperature, lubricant level, working time of main components, etc. Each edge is equipped with different communication protocols such as OPC UA, Profibus, Modbus, and Ethernet. This sensory information is processed, and then evaluated through a predictive model provided by the cloud. The model output is analyzed for the decision-making module and shown to the users through a web service.

The edge is composed of six main blocks, as depicted in Fig. 3:

- *Signal processing*: contains the techniques defined by the cloud for data-processing in each subsystem.
- *Model*: the predictive model defined by the cloud that is triggered to obtain performance rates according to the key performance indicators defined in the global module for each subsystem.
- *Local decision-making*: receives data on representative variables and manages possible alarms or events that can occur in a certain period. It uses a decision-making procedure with an adaptive threshold using the weighted sum of squared residuals (WSSR).
- *Visualization*: consists of a web-based user interface and facilitates the interpretation of the information that is displayed. In this interface, the user can track the behavior of the different components of the machine and the history of alarms, events, and failures of the corresponding components, and parts.

- *Local database* contains the local information of the machine. The information on alarms, failures, and process variables.
- *Local Management module*: synchronizes the operation of all the subsystems that comprise the edge and that control cloud communications.

C. In-process decision-making support

A model-based decision-making module consists of three essential parts: the residuals generation (difference between the actual output and the estimated output by the model), the residuals evaluation and decision-making relating to processes condition, or state. The threshold can be set using different statistical criteria (variance, standard deviation, mean), deterministic criteria (based on distance measurements in vector spaces), or using methods based on artificial intelligence techniques. Among the simplest strategies is the weighted sum of the square of the residuals (WSSR) [46]. The WSSR method is based on the sequence of residuals:

$$e_M(t) = y(t) - \hat{y}(t) \quad (1)$$

In the edge decision-making module, two criteria were considered by combining the influence of the residuals vector and its derivative. In this way, the data from the residuals vector and its derivative are used, not only to evaluate the degree of process-model matching, but also to use the residual vector trend for the detection of the process state.

$$\varepsilon_1(t) = \frac{1}{\|\mathbf{e}_M\|_2 + \|\dot{\mathbf{e}}_M\|_2} \quad (2)$$

$$\varepsilon_2(t) = \frac{1}{\|\mathbf{e}_M\|_\infty + \|\dot{\mathbf{e}}_M\|_\infty} \quad (3)$$

where $\|\cdot\|_\infty$ and $\|\cdot\|_2$ are the infinite norm and the Euclidean norm respectively; \mathbf{e}_M and $\dot{\mathbf{e}}_M$ are the residual vector and its derivative in the window $[t-NT-1, t]$.

D. Knowledge-based re-parameterization in the cloud

The reasoning module performs the re-parameterization in the cloud. Based on a reinforcement learning procedure, it is possible to capture new knowledge on the process, based on new data and the accumulated information extracted from previous data.

With the aim of implementing the reinforcement learning in the reasoning module, a Q-learning method is selected. Q-learning is one of the most frequently used approaches to find optimal policy in finite Markov decision processes. The main goal of the method is to maximize the total reward obtained from a set of actions executed from the initial state. The quality of each state-action is measured by the function Q (equation 4).

$$Q(s, a) \leftarrow Q_t(s, a) + \alpha[R(s, a) + \gamma \max_{a_{t+1}} Q_{t+1}(s_{t+1}, a_{t+1}) - Q(s, a)] \quad (4)$$

where, s_t is the state in time t ; a_t is the action taken in time t ; R is the reward received after performing action a_t ; α is the learning rate; and, γ is the discount factor which trades off the importance of sooner versus later rewards.

In each step, the Q function is updated based on the selected policy. From among several policies the ϵ -greedy policy was used in this particular implementation to select an action.

Two reward functions were defined, in order to identify certain central aspects of the learning process through the selection of the best fit model with the greatest accuracy and with the lowest computational load. The first reward was designed to select the best predictive model. The second one was designed to choose the best pre-processing tools, to reduce irrelevant background noise in the data.

On the one hand, three main aspects were considered to define the first reward function, i.e., model accuracy, generalization capability, and computational load. The first and the second are well-known model-quality parameters. The last one was chosen, because those models process data in real time where the computational load is a critical factor for these sorts of applications. The determination coefficient (R^2) and the Relative Absolute Error (RAE) were selected as performance indices, to take into consideration the accuracy and the generalization capability, defining a reward matrix illustrated in table II.

TABLE II. REWARD MATRIX FOR DECISION-MAKING MODEL.

R2	RAE				
	0 – 10%	10 – 20%	20 – 40%	40 – 70%	> 70%
90 – 100%	1	0.95	0.78	0.5	0.2
80 – 90%	0.88	0.82	0.68	0.4	0.15
70 – 80%	0.72	0.61	0.55	0.3	0.1
30 – 60%	0.55	0.4	0.32	0.2	0.05
0 – 30%	0.3	0.22	0.15	0.1	0.01

A penalty coefficient in the reward function was included, in order to consider the computational load factor, as shown in equation 5.

$$R_{model}(s, a) = R(s, a) - \lambda(m) \quad (5)$$

where, $R_{model}(s, a)$ is the total reward calculated for the corresponding model, $R(s, a)$ is the reward from the matrix (see table 1), defined by the performance indices of the model (R^2 , RAE), and the penalty coefficient, $\lambda(m)$. The penalty coefficient was defined on the basis of the computational load of each ML algorithm included in the library.

On the other hand, the second reward function was designed to select the most appropriate data pre-processing methods, to generate model outputs of greater accuracy. Two main characteristics were considered to define this second reward function. Firstly, the Mean Absolute Error (MAE) of the data that were processed to train the models was considered for assessing the accuracy of the models. Secondly, a penalty coefficient was defined that was linked to the computational load of each signal processing tool in the library. Equation 6 represents the reward function for the selection of the pre-processing tool.

$$R_{p-tool}(s, a) = \alpha(1 - MAE) - \eta \quad (6)$$

where, R_{p-tool} is the total reward; α is the learning rate coefficient; and, η is the penalty coefficient.

Feature extraction of measured signals in industrial processes extracts information from the signal that is difficult to interpret, due to background noise and irrelevant information. The application of those techniques is therefore of great importance and even essential for some systems.

Therefore, as two reward functions are available, there are also two Q-learning functions that will be updated in parallel to determine the optimal model, considering both the best processing strategy and the best prediction model. Taking into account these modifications, the functions to update the Q-values will be:

$$Q_{model}(s, a) \leftarrow Q_t(s, a) + \alpha[R_{model}(s, a) + \gamma \max_{a_{t+1}} Q_{t+1}(s_{t+1}, a_{t+1}) - Q(s, a)] \quad (7)$$

$$Q_{p-tool}(s, a) \leftarrow Q_t(s, a) + \alpha[R_{p-tool}(s, a) + \gamma \max_{a_{t+1}} Q_{t+1}(s_{t+1}, a_{t+1}) - Q(s, a)] \quad (8)$$

As a final step, if the optimal model, selected from the interpretation of the update process, yields different results for both Q functions, they will be downloaded to the edge. This procedure is part of the learning algorithm, shown below in Fig. 4.

Learning algorithm

```

1 Initialize, train and optimize all models
2 Initialize Q matrices
3 repeat
4   for each step do
5     receive data from shop floor
6     evaluate all models
7     calculate performances indices
8     update Q_model
9     update Q_p-tool
10    choose best model
11    choose best pre-processing tool
12    update edge model if different
13    update edge pre-processing tool if different
14  end
15 until reach learning epochs

```

Fig 2 . Learning algorithm running the reasoning module.

IV. INDUSTRY 4.0 PILOT LINE: A USE CASE ANALYSIS

A. Pilot line for smart manufacturing. Description.

The use case demonstrates cloud-based reasoning applied to a pilot line for smart manufacturing that represents both a challenge and an opportunity for Industry 4.0. The ICPS framework serves to address some key issues related with the Industry 4.0 paradigm. Fig. 5 illustrates the configuration of the pilot line.

The milling process is the most frequent machining operation in any manufacturing industry. Milling is the process for removing material by advancing a rotation cutter into a work piece. Milling covers a wide variety of different operations and

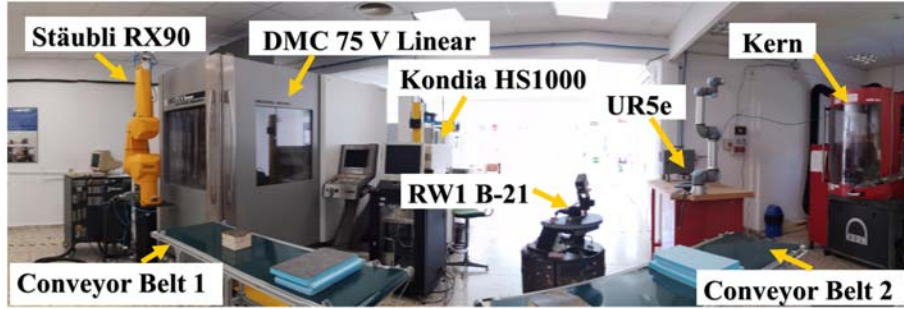


Fig 5. Overall view of Industry 4.0 pilot line for ICPS: testing and applications.

machines, on scales from micro scale (less than 0.5 mm of cutting tool diameter) up to macro scale (more than 0.5 mm of cutting tools diameter). This process is essential for producing high quality parts with complex shapes including ramps, contours, pocket, holes, etc. at high removal material rate. Nowadays, several manufacturing industries in aeronautics, health and space sectors demand high productivity with very high quality of the products. The surface roughness is one of the most important key performance indicators related surface quality and the overall quality of components. However, the setting of parameters for guaranteeing the required surface roughness is very cumbersome and challenging and the industry lacks systematic procedures for predicting surface roughness either at micro and macro scale milling processes.

The pilot line has three machine tools fully equipped with sensors, two manipulator robots, two conveyor belts, and an autonomous robot for inspection. The machine tools are a Deckel Maho DMC 75V Linear high-speed machining centre with a CNC Siemens 840D, a Kondia HS 1000 with a CNC Siemens 820D, and an ultra-precision micromachining centre equipped with a laser control Nano NT. In addition, two manipulator robots, Universal Robots UR5e and Stäubli RX90, are located next to the machines, for operational handling of the work pieces. Finally, two conveyor belts are in charge of transporting the work pieces between machines that perform different cutting operations. Further down the two conveyor belts, an autonomous robot, RWI B-21 is responsible for inspection of the manufactured parts.

Furthermore, it is important to highlight that the communication link, between the edges and the machine tools are implemented by two industrial standard protocols, i.e., Profibus to communicate with the Deckel Maho, and industrial Ethernet to Kondia HS. The edge application was embedded in two raspberry pi 3B+ models with 16 Gb storage card. The cloud application was running on a virtual machine linked to a remote central server in another town, at some distance from the pilot line location. The cloud and edge application were implemented in C/C++ with Qt library version 5.10, using an Integrated Development Environment Qt Creator 4.5.0. The visualization dashboard was implemented in Python 3.6.2.

B. Manufacturing shop-floor validation

The validation of the proposed Cloud-based ICPS and more specifically, the reasoning module, was performed for a condition-based monitoring process of two machining centers out of the three connected in parallel on the pilot line, as

previously described. Specifically, the representative variable to be monitored is surface roughness [47].

In the first manufacturing process (process A), a micro-milling operation was selected using a Kern Evo machine (see Fig. 6 A)) [48]. A set of piezoelectric accelerometers were also attached to measure vibrations on each (x, y, and z) axis.

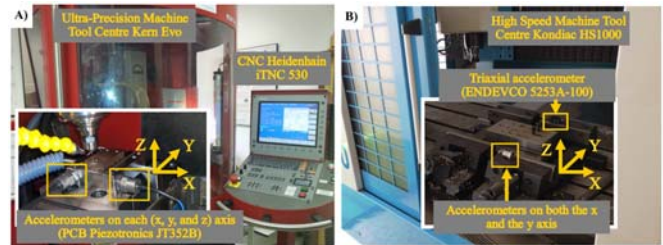


Fig 6. Detail of the micromachining center and devices.

The second manufacturing process (process B) was a milling operation using a Kondia HS1000 machine (see Fig. 6 B) [36]. In addition, an external sensory system that consisted of three piezoelectric accelerometers was also added, for measuring the vibration signals of each (x, y and z) axis.

1) Dataset generation and model training

The main goal was to generate an on-production dataset capable of initial training and then, validation of the corresponding machine-learning-based models implemented in the Cloud-based ICPS, as described below [49]. The process monitoring consisted of the measurement of feed per tooth (f_z), tool diameter (D), radial depth of cut (ae), spindle speed (rpm) and the resulting vibration signals. Subsequently, the pieces were subjected to quality control, measured through their surface quality and more specifically, considering the arithmetic average roughness value, R_a . The models should therefore predict in-process the R_a value using as inputs, the vibration signal and the f_z . Table III illustrates the structure of both datasets.

TABLE III. DATA SETS FOR TRAINING AND VALIDATION.

Micro machining (Samples)		Macro machining (Samples)	
Training	Valid	Training	Valid
241	137	5890	2000

Five strategies for developing the predictive models using the ML library were used, in order to demonstrate the advantages of the reasoning module in the cloud-based ICPS configuration: (i) k-Nearest Neighbors (kNN); (ii) Support Vector Machine (SVM); (iii) Ordinary Least Square Regression (OLS); (iv)

Multilayer Perceptron (MLP); and, (v) Hybrid Incremental Modeling (HIM). Moreover, the pre-processing and signal processing library was populated with ten techniques based on temporal, frequency and time-frequency analysis.

2) Evaluation of the overall Cloud-based ICPS

For the sake of clarity and in order to show the advantage of the proposed cloud-based ICPS for data-driven reasoning, different studies were conducted. Firstly, the overall assessment was carried out taking into account computational performance indices (i.e., computational load and cost) of the model and the reasoning module. Secondly, an error-based performance index, i.e., the relative absolute error (RAE) of the estimated R_a in relation to the actual measured R_a , was computed and the comparative analysis with and without the application of the reasoning module was performed.

Having generated the dataset necessary for processes A and B, the experimental evaluation of the overall cloud-based ICPS was performed. Firstly, the cloud begins the reasoning process by training and optimal parametrization of the ML-based models, selecting the most appropriate initial model and the pre-processing data methods as required.

Initially, as a default configuration in the first iteration, the reasoning module selects well-known time-domain techniques, i.e., the RMS value, for conditioning the input data of both processes. In addition, the two models with the best performance indices were selected: HIM-based model and MLP-based model. They were chosen from among the five ML-based models for predicting surface roughness in the evaluation procedure for processes A and B, respectively.

After yielding the initial models, the configuration parameters and the models were downloaded to the corresponding edges and the supervision process was therefore ready to start. Each time either of the two processes, A or B, sent a new set of data to the cloud through its edge, the corresponding reasoning process was triggered. The epochs of the learning algorithms were set to 100 for both processes.

Computational load is an important performance index, to evaluate the behavior of the Cloud-based ICPS. Table IV shows the computational load in terms of the latency times of some of the main events at the edge.

TABLE IV. COMPUTATIONAL LOAD.

Event	Time (s)
Downloading models from the cloud	4.5
Signal processing in edge	0.003
Model evaluate in the edge	0.008
Initial model training and optimization	2546

With the aim of evaluating the improvements introduced to the cloud-based ICPS by the proposed data-driven reasoning module, a comparative study using two simulations was performed. The first comparison was focused on the analysis of the model fitting error in the edge for the best model selected with brute-force search algorithm and the best model selection by the reasoning module (i.e., the re-parametrized model). The second comparison was centered on the computational load of the reasoning procedure and the brute force algorithm. A brute-force search or exhaustive search refers to the fact that all

models, once trained, are evaluated in each reasoning iteration with datasets obtained from each signal processing strategy.

Table V shows the computational load, in terms of the latency of the best model selection at the end of the reasoning process in comparison with brute-force search or exhaustive search. Shorter latency times are indicative of improvements to the reasoning procedure, i.e., latency time 13 times less than the latency time for brute force. It is important to note that the brute force selection algorithm can only take into account one of the selected performance indices used by the reasoning process in the best algorithm selection.

TABLE V. REASONING VS BRUTE-FORCE SEARCH OF COMPUTATIONAL LOAD.

Event	Time (s)
Best model selection by the reasoning module	0.085
Best model selection with brute-force search	1.10

In order to illustrate the actual performance of the cloud-based reasoning module, Fig. 7 shows the results of re-parametrization and updating of predictive models of processes A and B, during the run of the edges. There is not updating in the predictive model for the process B. This predictive model, based on MLP, yields a RAE of 1.06%, and therefore with this accuracy no further re-parametrization and updating are required. On the contrary, there are two clear transitions in the behavior of the predictive model corresponding to the process A. The increase of the error detected by the reasoning module triggers two updates of predictive models of the process A. This updating of the predictive model from HIM-based to SVM-based model, and vice versa decreased RAE in about 7% which is very significant from industrial informatics viewpoint.

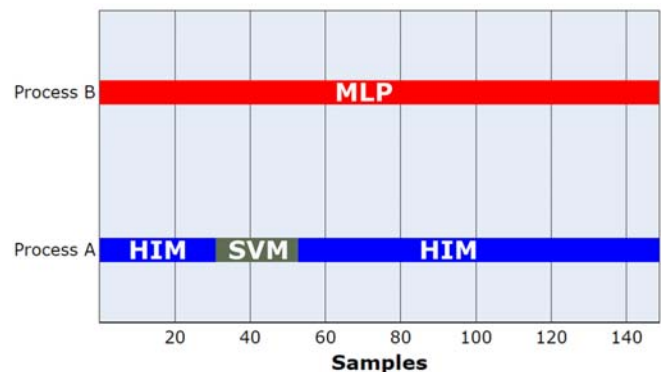


Fig 7. Best model for processes A and B selected by the reasoning module

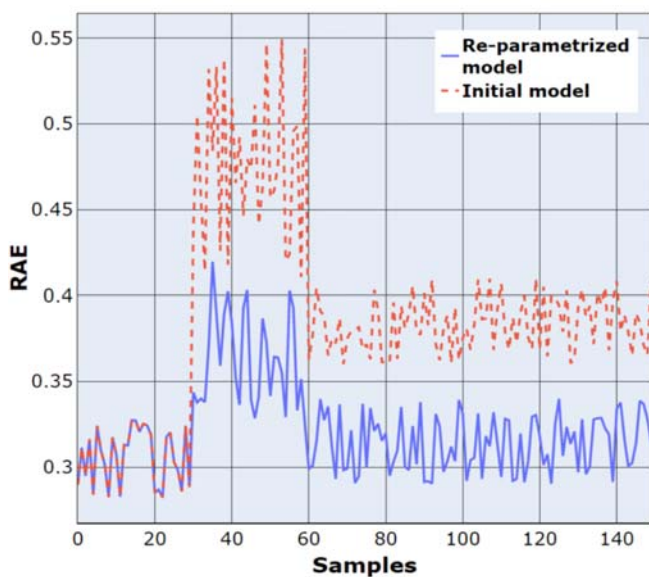


Fig 8. RAE in re-parametrized model vs initial model in process A.

For the sake of coherence and due to the high fidelity of the MLP-based predictive model for the process B, no further details of the corresponding RAE are shown. On the contrary, the contribution of the reasoning module to improve RAE of the process A is detailed in Fig. 8. The behavior of the predictive model of the process A clearly depicts a remarkable decrease in RAE of the best model selected from the reasoning module with regard to the brute force selected model (i.e., HIM-based model). At the beginning, both models depict similar behavior in relation to the corresponding RAE. However, after some iterations, the good performance of the reasoning module, by improving the accuracy in predicting surface roughness, is clearly shown in Fig. 8.

V. CONCLUSION

A cloud-based data-driven reasoning process is presented that enables re-parametrization of predictive models in a set of edge modules based on the new knowledge extracted from data in an industrial cyber-physical system. The application of unsupervised learning procedures, represented by a Q-learning method, updates both the data-processing strategy and the prediction model. The reasoning module was therefore integrated into a cloud-to-edge industrial cyber-physical system that can visualize, monitor, predict, and deploy the main parameters in smart manufacturing scenarios. During the execution phase, a machine-learning-based model library is run to pre-process the decision-making actions and to enable the re-parameterization of models, using information collected by sensors on the factory floor.

Finally, all modules were tested and validated on an Industry 4.0 pilot line, outperforming the results of single predictive models. The reasoning module is able to select the best fitting model for multiple manufacturing processes running in parallel. Future works will be conducted to add more functionalities to the current cloud-based framework and to explore other machine learning-based procedures, in order to increase the

degrees of freedom of the reasoning module.

REFERENCES

- [1] R. Haber-Haber, R. Haber, M. Schmittiel, and R. M. del Toro, "A classic solution for the control of a high-performance drilling process," *International Journal of Machine Tools and Manufacture*, Article vol. 47, no. 15, pp. 2290-2297, 2007.
- [2] A. Gajate, R. E. Haber, P. I. Vega, and J. R. Alique, "A transductive neuro-fuzzy controller: Application to a drilling process," *IEEE Transactions on Neural Networks*, Article vol. 21, no. 7, pp. 1158-1167, 2010, Art. no. 5492227.
- [3] S. Iarovyi, W. M. Mohammed, A. Lobov, B. R. Ferrer, and J. L. M. Lastra, "Cyber-Physical Systems for Open-Knowledge-Driven Manufacturing Execution Systems," *Proceedings of the IEEE*, Article vol. 104, no. 5, pp. 1142-1154, 2016, Art. no. 7430247.
- [4] B. R. Ferrer *et al.*, "Towards the Adoption of Cyber-Physical Systems of Systems Paradigm in Smart Manufacturing Environments," in *Proceedings - IEEE 16th International Conference on Industrial Informatics, INDIN 2018*, 2018, pp. 792-799.
- [5] R. C. Schl pfer, M. Koch, and P. Merkofer, "Industry 4.0 challenges and solutions for the digital transformation and use of exponential technologies," *Deloitte, Zurich*, 2015.
- [6] A. Villalonga, G. Beruvides, F. Castano, and R. Haber, "Industrial cyber-physical system for condition-based monitoring in manufacturing processes," in *Proceedings - 2018 IEEE Industrial Cyber-Physical Systems, ICPS 2018*, 2018, pp. 637-642.
- [7] A. Villalonga, F. Castano, G. Beruvides, R. Haber, S. Strzelczak, and J. Kossakowska, "Visual analytics framework for condition monitoring in cyber-physical systems," in *2019 23rd International Conference on System Theory, Control and Computing, ICSTCC 2019 - Proceedings*, 2019, pp. 55-60.
- [8] X. Xu and Q. Hua, "Industrial Big Data Analysis in Smart Factory: Current Status and Research Strategies," *IEEE Access*, Article vol. 5, pp. 17543-17551, 2017, Art. no. 8012376.
- [9] P. Zheng *et al.*, "Smart manufacturing systems for Industry 4.0: Conceptual framework, scenarios, and future perspectives," *Frontiers of Mechanical Engineering*, Review vol. 13, no. 2, pp. 137-150, 2018.
- [10] R. H. Guerra, R. Quiza, A. Villalonga, J. Arenas, and F. Castano, "Digital Twin-Based Optimization for Ultraprecision Motion Systems with Backlash and Friction," *IEEE Access*, Article vol. 7, pp. 93462-93472, 2019, Art. no. 8759853.
- [11] J. Puttonen, A. Lobov, and J. L. M. Lastra, "Semantics-based composition of factory automation processes encapsulated by web services," *IEEE Transactions on Industrial Informatics*, Article vol. 9, no. 4, pp. 2349-2359, 2013, Art. no. 6311467.
- [12] M.-H. Hung, Y.-Y. Li, Y.-C. Lin, C.-F. Wei, H.-C. Yang, and F.-T. Cheng, "Development of a novel cloud-based multi-tenant model creation service for automatic virtual metrology," *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 174-189, 2017.
- [13] Z. Yan, X. Li, M. Wang, and A. V. Vasilakos, "Flexible Data Access Control Based on Trust and Reputation in Cloud Computing," *IEEE Transactions on Cloud Computing*, vol. 5, no. 3, pp. 485-498, 2017.
- [14] A. Dennert, J. G. I. Montemayor, J. Krause, S. Hesse, J. L. M. Lastra, and M. Wollschlaeger, "Advanced Concepts for Flexible Data Integration in Heterogeneous Production Environments," *IFAC Proceedings Volumes*, vol. 46, no. 7, pp. 348-353, 2013/05/01/ 2013.
- [15] P. Adolphs *et al.*, "Structure of the administration shell. continuation of the development of the reference model for the industrie 4.0 component," *ZVEI and VDI, Status Report*, 2016.
- [16] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14-76, 2015.
- [17] P. Urien, "Blockchain IoT (BIoT): A new direction for solving Internet of Things security and trust issues," in *2018 3rd Cloudification of the Internet of Things (CIoT)*, 2018, pp. 1-4: IEEE.
- [18] I. La Fe-Perdomo, G. Beruvides, R. Quiza, R. Haber, and M. Rivas, "Automatic Selection of Optimal Parameters Based on Simple Soft-Computing Methods: A Case Study of Micromilling Processes,"

- IEEE Transactions on Industrial Informatics*, Article vol. 15, no. 2, pp. 800-811, 2019, Art. no. 8325494.
- [19] F. Castaño, G. Beruvides, R. E. Haber, and A. Artuñedo, "Obstacle recognition based on machine learning for on-chip lidar sensors in a cyber-physical system," *Sensors (Switzerland)*, Article vol. 17, no. 9, 2017, Art. no. 2109.
- [20] P. A. Sarvari, A. Ustundag, E. Cevikkan, I. Kaya, and S. Cebi, "Technology roadmap for Industry 4.0," in *Industry 4.0: Managing The Digital Transformation*: Springer, 2018, pp. 95-103.
- [21] F. Ansari, S. Erol, and W. Sihni, "Rethinking Human-Machine Learning in Industry 4.0: How Does the Paradigm Shift Treat the Role of Human Learning?," *Procedia Manufacturing*, vol. 23, pp. 117-122, 2018.
- [22] F. Castaño, S. Strzelczak, A. Villalonga, R. E. Haber, and J. Kossakowska, "Sensor reliability in cyber-physical systems using internet-of-things data: A review and case study," *Remote Sensing*, Article vol. 11, no. 19, 2019, Art. no. 2252.
- [23] B. Brik, B. Bettayeb, M. h. Sahnoun, and F. Duval, "Towards Predicting System Disruption in Industry 4.0: Machine Learning-Based Approach," *Procedia Computer Science*, vol. 151, pp. 667-674, 2019.
- [24] F. Matía, V. Jiménez, B. P. Alvarado, and R. Haber, "The fuzzy Kalman filter: Improving its implementation by reformulating uncertainty representation," *Fuzzy Sets and Systems*, Article 2019.
- [25] R.-E. Precup and R.-C. David, *Nature-Inspired Optimization Algorithms for Fuzzy Controlled Servo Systems*. Butterworth-Heinemann, 2019.
- [26] B. Salgues, *Society 5.0: Industry of the Future, Technologies, Methods and Tools*. John Wiley & Sons, 2018.
- [27] Z. Zhou, S. S. Xie, and D. Chen, "Theory System of Digital Manufacturing Science," in *Fundamentals of Digital Manufacturing Science*: Springer, 2011, pp. 19-55.
- [28] A. A. Sozinova, "Causal Connections of Formation of Industry 4.0 from the Positions of the Global Economy," in *Industry 4.0: Industrial Revolution of the 21st Century*: Springer, 2019, pp. 131-143.
- [29] S. Yin, J. Bao, J. Li, and J. Zhang, "Real-time task processing method based on edge computing for spinning CPS," *Frontiers of Mechanical Engineering*, pp. 1-12, 2019.
- [30] S. Iarovyi, J. L. M. Lastra, R. Haber, and R. Del Toro, "From artificial cognitive systems and open architectures to cognitive manufacturing systems," in *Proceeding - 2015 IEEE International Conference on Industrial Informatics, INDIN 2015*, 2015, pp. 1225-1232.
- [31] P. Leitão, A. W. Colombo, and S. Karnouskos, "Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges," *Computers in Industry*, Article vol. 81, pp. 11-25, 2016.
- [32] R. E. Haber, C. Juanes, R. Del Toro, and G. Beruvides, "Artificial cognitive control with self-x capabilities: A case study of a micro-manufacturing process," *Computers in Industry*, Article vol. 74, pp. 135-150, 2015.
- [33] G. Beruvides, R. Quiza, and R. E. Haber, "Multi-objective optimization based on an improved cross-entropy method. A case study of a micro-scale manufacturing process," *Information Sciences*, Article vol. 334-335, pp. 161-173, 2016.
- [34] V. Vasyutynskyy, C. Hengstler, D. Nadoveza, J. McCarthy, K. G. Brennan, and A. Dennert, "Layered architecture for production and logistics cockpits," in *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, 2012.
- [35] A. Dennert, J. G. I. Montemayor, J. Krause, S. Hesse, J. L. M. Lastra, and M. Wollschlaeger, "Advanced concepts for flexible data integration in heterogeneous production environments," in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2013, vol. 46, pp. 348-353.
- [36] G. Beruvides, F. Castaño, R. E. Haber, R. Quiza, and A. Villalonga, "Coping with complexity when predicting surface roughness in milling processes: Hybrid incremental model with optimal parametrization," *Complexity*, vol. 2017, 2017.
- [37] R. E. Haber, G. Beruvides, R. Quiza, and A. Hernandez, "A simple multi-objective optimization based on the cross-entropy method," *IEEE Access*, Article vol. 5, pp. 22272-22281, 2017, Art. no. 8070310.
- [38] A. W. Colombo, T. Bangemann, and S. Karnouskos, "IMC-AESOP outcomes: Paving the way to collaborative manufacturing systems," in *Proceedings - 2014 12th IEEE International Conference on Industrial Informatics, INDIN 2014*, 2014, pp. 255-260.
- [39] P. Varga *et al.*, "Making system of systems interoperable – The core components of the arrowhead framework," *Journal of Network and Computer Applications*, Article vol. 81, pp. 85-95, 2017.
- [40] L. Ribeiro, A. Ocha, A. Veiga, and J. Barata, "Collaborative routing of products using a self-organizing mechatronic agent framework - A simulation study," *Computers in Industry*, Article vol. 68, pp. 27-39, 2015.
- [41] P. Leitao, N. Rodrigues, C. Turrin, and A. Pagani, "Multiagent System Integrating Process and Quality Control in a Factory Producing Laundry Washing Machines," *IEEE Transactions on Industrial Informatics*, Article vol. 11, no. 4, pp. 879-886, 2015, Art. no. 7104129.
- [42] M. Onori, N. Lohse, J. Barata, and C. Hanisch, "The IDEAS project: Plug & produce at shop-floor level," *Assembly Automation*, Article vol. 32, no. 2, pp. 124-134, 2012.
- [43] P. Ferreira, S. Doltsinis, and N. Lohse, "Symbiotic assembly systems - A new paradigm," in *Procedia CIRP*, 2014, vol. 17, pp. 26-31.
- [44] P. Leitão *et al.*, "Instantiating the PERFORM system architecture for industrial case studies," in *Studies in Computational Intelligence* vol. 694, ed, 2017, pp. 359-372.
- [45] X. Huang and J. Dong, "Reliable cooperative control and plug-and-play operation for networked heterogeneous systems under cyber-physical attacks," *ISA Transactions*, 2019/10/24/ 2019.
- [46] M. Hatami, *Weighted residual methods: principles, modifications and applications*. Academic Press, 2017.
- [47] G. Beruvides, F. Castaño, R. Quiza, and R. E. Haber, "Surface roughness modeling and optimization of tungsten-copper alloys in micro-milling processes," *Measurement: Journal of the International Measurement Confederation*, Article vol. 86, pp. 246-252, 2016.
- [48] F. Castaño, R. M. del Toro, R. E. Haber, and G. Beruvides, "Conductance sensing for monitoring micromechanical machining of conductive materials," *Sensors and Actuators A: Physical*, vol. 232, pp. 163-171, 2015.
- [49] R. M. Del Toro, M. C. Schmittiel, R. E. Haber-Guerra, and R. Haber-Haber, "System identification of the high performance drilling process for network-based control," in *2007 Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC2007*, 2008, vol. 1 PART B, pp. 827-834.