Received 16 January 2022; revised 8 February 2022; accepted 13 February 2022. Date of publication 22 February 2022;
date of current version 7 March 2022. The review of this paper was arranged by Associate Editor Doriana Marilena D'Addona.

Digital Object Identifier 10.1109/OJIES.2022.3152725

A Fuzzy Logic Recommendation System to Support the Design of Cloud-Edge Data Analysis in Cyber-Physical Systems

JONAS QUEIROZ [1,2], PAULO LEITÃO [1] (Senior Member, IEEE), AND EUGÉNIO OLIVEIRA [2]

[1] Research Center in Digitalization and Intelligent Robotics, Instituto Politécnico de Bragança, 5300-253 Braganca, Portugal
[2] Artificial Intelligence and Computer Science Laboratory, University of Porto Faculty of Engineering, 4200-465 Porto, Portugal

CORRESPONDING AUTHOR: JONAS QUEIROZ (e-mail: jonas.queiroz@fe.up.pt)

This work was supported by FCT - Fundação para a Ciência e Tecnologia within the Project Scope: UIDB/05757/2020.

ABSTRACT The ongoing 4th industrial revolution is characterized by the digitization of industrial environments, mainly based on the use of Internet of Things, Cloud Computing and Artificial Intelligence (AI). Regarding AI, although data analysis has shown to be a key enabler of industrial Cyber-Physical Systems (CPS) in the development of smart machines and products, the traditional Cloud-centric solutions are not suitable to attend the data and time-sensitive requirements. Complementary to Cloud, Edge Computing has been adopted to enable the data processing capabilities at or close to the physical components. However, defining which data analysis tasks should be deployed on Cloud and Edge computational layers is not straightforward. This work proposes a framework to guide engineers during the design phase to determine the best way to distribute the data analysis capabilities among computational layers, contributing for a lesser ad-hoc design of distributed data analysis in industrial CPS. Besides defining the guidelines to identify the data analysis requirements, the core contribution relies on a Fuzzy Logic recommendation system for suggesting the most suitable layer to deploy a given data analysis task. The proposed approach is validated in a smart machine testbed that requires the implementation of different data analysis tasks for its operation.

INDEX TERMS Cyber-physical system design, distributed data analysis, fuzzy recommendation system.

I. INTRODUCTION

The digitization of industrial environments has been characterized by the increasing adoption of Internet of Things (IoT), Cloud Computing and Artificial Intelligence (AI) technologies [1]–[3]. In particular, data-driven Machine-Learning (ML) based algorithms allow to analyze the huge amount of produced data and build data models for monitoring, diagnosis, prediction and optimization. Besides to increase the autonomy of machines and processes towards truly smart autonomous systems, such algorithms can augment the capabilities of engineers and operators in decision-making and tasks execution [4]–[6].

Although the adoption of data analysis has increased and demonstrated great potential, it faces some challenges, especially when considering industrial scenarios, constrained by response time, data sensitiveness and network connectivity [7], [8]. In this context, the traditional IoT-based data analysis approaches, where all the data are sent to be processed at the Cloud, are not suitable. In spite of providing a mean to run powerful algorithms using huge batches of aggregated data, they lack responsiveness to condition change.

On the other hand, Edge Computing enables the local and decentralized data processing, i.e., at or close to the data sources [8]–[10], endowing end devices with data analysis capabilities to increase the intelligence and responsiveness of physical devices, and consequently the efficiency and autonomy of the whole system.

In this sense, a raising challenge is related to harmonize the trade-offs of distributing intelligence and performing data analysis among the Cloud-Edge computational layers in an industrial Cyber-Physical Systems (CPS) perspective, taking into consideration the scenario requirements and constraints, e.g., resource availability and costs, bandwidth, responsiveness and algorithm complexity. In fact, rather than an alternative to Cloud, the Edge should be viewed as a complementary approach, providing solutions for data and time sensitive applications, like in real-time monitoring and control, while Cloud should be responsible for the high level supervisory, planning and optimization tasks, also generating knowledge to dynamically support the Edge layer tasks.

Having this in mind, the objective of this paper is to propose a conceptual framework that supports the system engineers, during the design phase, to decide where to deploy data analysis capabilities in cyber and physical components, covering the Cloud-Edge layers. The proposed framework provides a guideline to identify the main aspects and concerns that should be considered in this engineering process to identify the data analysis requirements, and defines a multi-criteria Fuzzy Logic recommendation system to determine, during the design phase, the most suitable computational layer where a given data analysis capability should be deployed. This innovative recommendation method constitutes the core contribution of this work, allowing to reduce the complexity of the problem by using a formal approach that is suitable to deal with the uncertainty and vagueness from the perception and experience of engineers in the software design decisions. This way, this framework provides a suitable tool to support a lesser ad-hoc design of distributed data analysis in CPS.

A smart electrical machine testbed was used to illustrate and assess the proposed approach, demonstrating the use of some criteria, e.g., responsiveness, network bandwidth and algorithm processing time, to guide the selection of the most suitable computing layers to deploy the different data analysis capabilities presented in the case study. The data analysis tasks were deployed according to the recommendations provided by the Fuzzy Logic system, with the achieved performance allowing to validate the proposed approach.

The remaining of this paper is organized as follows. Section II discusses the issues raised with the distribution of data analysis capabilities among Cloud-Edge computational layers, as well as the existing approaches to support the design of such Cloud-Edge systems and Section III presents the framework to support the design of Cloud-Edge data analysis in industrial CPS, and particularly the Fuzzy Logic recommendation system. Section IV describes the case study and the application of the proposed approach to determine the computational layers where the data analysis tasks should be deployed, including the analysis of the experimental results after deploying the tasks according to the recommendations. Finally, Section V rounds up the paper with the conclusions and points out some future work.

II. CLOUD-EDGE DATA ANALYSIS

The related work regarding the design of Cloud-Edge applications will be discussed in this section, with special emphasis to the analysis of the existing frameworks to support the design and distribution of data analysis tasks among these computational layers.

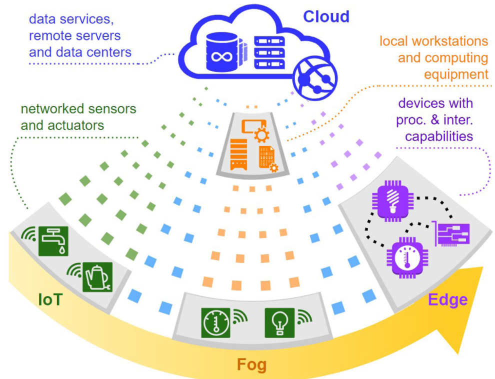

FIGURE 1. Traditional Cloud-centric IoT approach towards Edge Computing.

A. DECENTRALIZATION OF DATA ANALYSIS

In traditional Cloud-centric IoT approaches, networked sensing devices are able to monitor the environment and its objects by sending huge amounts of data to be processed by Cloud-based systems. In this context, Cloud solutions and data analysis contributed to leverage the impact that sensor data can have in different domains, promoting the development of a variety of data-driven services.

Furthermore, besides promoting several advances and opportunities in smart devices, communication and data analysis technologies, IoT drew the attention of more conservative sectors, like the production industry, where it is considered a key enabling technology in the 4th industrial revolution (4IR) [1], [8]. However, in such constrained scenarios, the adoption of IoT technologies faces some challenges, mainly regarding responsiveness (low latency for time critical tasks), data security (avoid sensitive data to be accessed by third parties) and network connectivity (high bandwidth and reliable connection to cope with the large number of devices and volumes of raw data) [7], [10].

In this context, Edge Computing emerged as a complementary paradigm, promoting approaches and strategies to perform data processing locally, close to the data sources. Such data analysis decentralization plays an important role in the realization of industrial CPS, promoting the development of more intelligent physical components. Although the terms Edge and Fog Computing are used interchangeably, the second is often used to designate the approaches where the data processing is performed by computing equipment at the local network, e.g., local servers or computers [10]–[12].

Fig. 1 illustrates the traditional Cloud-centric IoT approach towards the Fog/Edge Computing trends. The differences are mainly in the computing resources of end devices and the local where the data is processed. On the left, the traditional IoT is shown, where the end devices are directly connected to Cloud systems, sending the collected data and/or receiving commands. In the Fog Computing (Fig. 1, middle), the IoT devices communicate with equipment at local network. The Edge approach (Fig. 1, right) considers devices that have

FIGURE 2. **Edge-Cloud Computing trade-off.**

enough computational resources to perform some kinds of data processing, and also present horizontal communication capabilities.

Although there is a trend in the adoption of Edge Computing approaches towards the decentralization of data analysis [13], [14], this strategical decision implies several trade-offs, as illustrated in Fig. 2 (left). For instance, many aspects that are considered benefits of Cloud, may represent drawbacks in Edge solutions, and vice-versa, closing the trade-off cycle. This raises the importance to harmonize what can be performed by each computational layer. Indeed, in many scenarios these approaches can coexist. Fig. 2 (right) maps the strengths and weaknesses of these approaches, illustrating some of their main complementary aspects (where "5" means a very good aspect and "1" a very bad aspect).

Besides that, different kinds of data analysis capabilities and algorithms are suitable for specific computing layers. For instance, when considering (real-time) monitoring tasks, simple algorithms running at the Edge should be more appropriate than complex algorithms. But this should also consider the amount of data produced and the required computational resources. For instance, if the problem does not consider high sample rate and data size, the data can be sent to a remote system and still get the response inside the time constraints. On the other hand, if the data analysis algorithm requires significant computational resources, deploying it locally may not be feasible, thus some hybrid solution could be adopted. As example, some works propose the distribution of the neural network layers along Cloud to Edge that besides offload the computation in central servers can also provide a local fast and partial response [9], [15]. In general, Cloud should be the most suitable choice when considering tasks related to the system optimization or planning, which may require the analysis of historical data and/or from multiple sources.

B. DISTRIBUTION OF DATA ANALYSIS CAPABILITIES BY CLOUD-EDGE LAYERS

In this context, the definition of where to deploy data analysis capabilities is a major challenge in the specification and design of industrial CPS solutions, which is not straightforward and raises several architectural and technological concerns. Several frameworks are available for the design of Cloud-based data analysis systems, but in this work, the objective is the design of industrial CPS applications that span the data analysis capabilities among Cloud-Edge computational layers.

A general approach to develop CPS is a 3 layer architecture, where the cyber components are deployed on the Cloud (top) and Fog (middle) layers, while the physical components are at the Edge (bottom) [8], [12], [16]–[18]. Despite being seen as a layered architecture, it does not imply the existence of a direct hierarchy between the components, as in the ISA-95 architecture. In CPS, a network of distributed and cooperating intelligent components is considered, where the Edge nodes promote the decentralization of processing, control and decision making, enhancing the autonomy of end devices, system dynamics and self-reconfiguration.

This layered architecture represents a common consensus regarding the organization of the system components. However, the distribution of the data analysis capabilities is usually performed in an ad-hoc manner, only considering the benefits of some aspects (e.g., latency, computing resources, data type or data analysis algorithms), and paying minor attention to their specific features and constraints that can be different for each layer [19], [20]. The large number of aspects that can influence the distribution of such capabilities raises fundamental questions about where, when and how to choose the most suitable layer to deploy a given data analysis capability [10].

These aspects are also discussed in the domain of Computation Offloading that provides approaches to optimize the choice of where to execute a given task [21], [22]. However, these approaches are more focused on service-based solutions in the context of mobile Edge Computing to address dynamic nodes and service workloads through the dynamic task allocation and load balancing solutions, aiming to assure QoS, low latency and device energy efficiency [21]–[23]. Besides that, they consider the use of lightweight virtualization technologies that provide high level of scalability in exchange of extra overhead that are still not suitable for most limited resource and fast response applications [24], [25]. On the other hand, industrial CPS go beyond a service-based approach, also considering a collaborative and complementary interaction between components (cyber and physical). This requires customized and optimized solutions, where the data analysis capabilities should be defined during the system design.

In this context, some works propose general approaches and raise some main aspects and concerns that may support the proper distribution of data analysis functionalities in CPS. For instance, the 5 C Architecture [26] presents a functional organization of data analysis capabilities in 5 levels that goes from the basic data acquisition and processing to high level decision support tasks that can be abstracted along Cloud-Edge. A Fog Computing taxonomy is presented in [11], raising several general aspects and concerns that should be considered and evaluated to deploy a CPS solution. Similarly, some aspects,

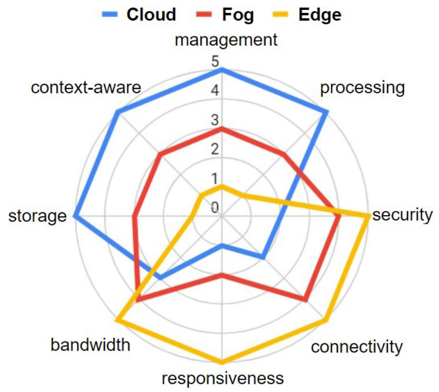

FIGURE 3. Edge-Cloud Computing complementary aspects.

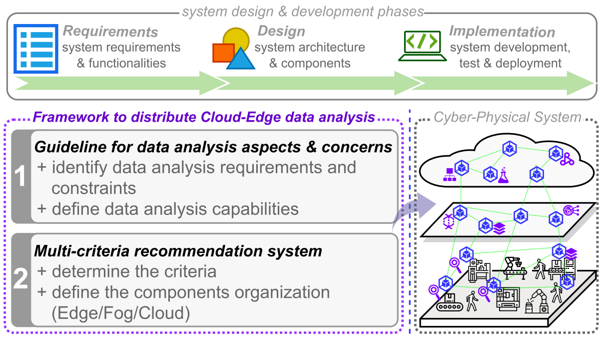

FIGURE 4. Framework to support the design of Cloud-Edge data analysis in industrial CPS.

called "architectural imperatives" (e.g., positioning of Fog nodes, numbers, types, topology, protocols, data bandwidth, hardware and software), are presented in [27]. They are used to support architectural decisions, regarding the adoption of Fog solutions. In the same manner, the work presented in [21] discusses 8 criteria to decide if the task offload is necessary, including security, affordability, feasibility, and maintenance aspects. Another approach is presented in [12], where a decision model, considering 6 parameters, is proposed to choose the most suitable layer for a given task. The importance of a decision support framework to support IoT designers to distribute application components along Edge-Cloud is discussed in [28], being analyzed several key attributes, e.g., response time, energy consumption, resource usage and accuracy, that can impact the design decisions through two real use cases.

In summary, the design of industrial CPS comprises a complex engineering process, where several tasks are still performed in an ad-hoc manner, mainly based on the experience of experts on similar projects. Moreover, most existing approaches only present very general and reference architectures, criteria and ad-hoc approaches to use Cloud or Edge separately, or to optimize the choice of where to execute a given task during the operation phase, as the computation offloading approach does. In spite of covering broad and high-level aspects that support several abstractions and concepts during the engineering process, they are missing the definition of guidelines and recommendation methods to support the decision on how to distribute and balance the deployment of data analysis capabilities among the different computational layers during the design of an industrial CPS.

III. FRAMEWORK TO DISTRIBUTE CLOUD-EDGE DATA ANALYSIS

The definition of where data analysis capabilities should be deployed comprises an important architectural decision in the development of industrial CPS, that should be handled during the requirement analysis and system design phases. In this context, the proposed framework, illustrated in Fig. 4, defines a guideline to support the identification of the requirements and constraints for the specification and the development of data analysis capabilities in cyber and physical components. These capabilities should be analyzed in the second phase, in order to design the system architecture in terms of the organization of the components. For this purpose, the framework defines a multi-criteria recommendation system to determine the most suitable computational layer, regarding Edge-Cloud, that a given data analysis capability could be deployed.

A. GUIDELINES TO IDENTIFY DATA ANALYSIS REQUIREMENTS IN CPS

The analysis of the system requirements, comprises one of the first tasks in a software engineering methodology. Therefore, it should be performed following the chosen methodology, where the guideline proposed here intends to support the execution of this phase, especially during the analysis of the system requirements and the definition of the use cases to develop the CPS solution. Considering that data analysis represents a mean to achieve intelligent systems, this guideline aims to help stakeholders, experts in the knowledge domain (system experts, engineers and operators) and software engineers (architects, analysts, developers, and data analysts) to answer the questions about how decentralize intelligence in CPS, but also to better understand why, when, where and most important what should or could be decentralized.

In order to design and develop distributed data analysis in industrial CPS, several aspects should be evaluated, with each one raising several concerns that engineers need to have in mind for the design and development of such approaches, in order to properly address or mitigate their implications and impacts. The proposed framework considers five main aspects to define a guideline for the identification of the data analysis requirements and the related concerns, as illustrated in Fig. 5. These aspects represent the most common and relevant types of concerns and requirements considered during the design of CPS architectures that were identified during the analysis of the literature review. Note that the proposed guideline does not intend to list and discuss all the possible aspects, since they vary with the application domain and scenario. Therefore, it should be considered as a general guideline, where its structure allows to be easily adapted or extended to support other scenarios and related aspects.

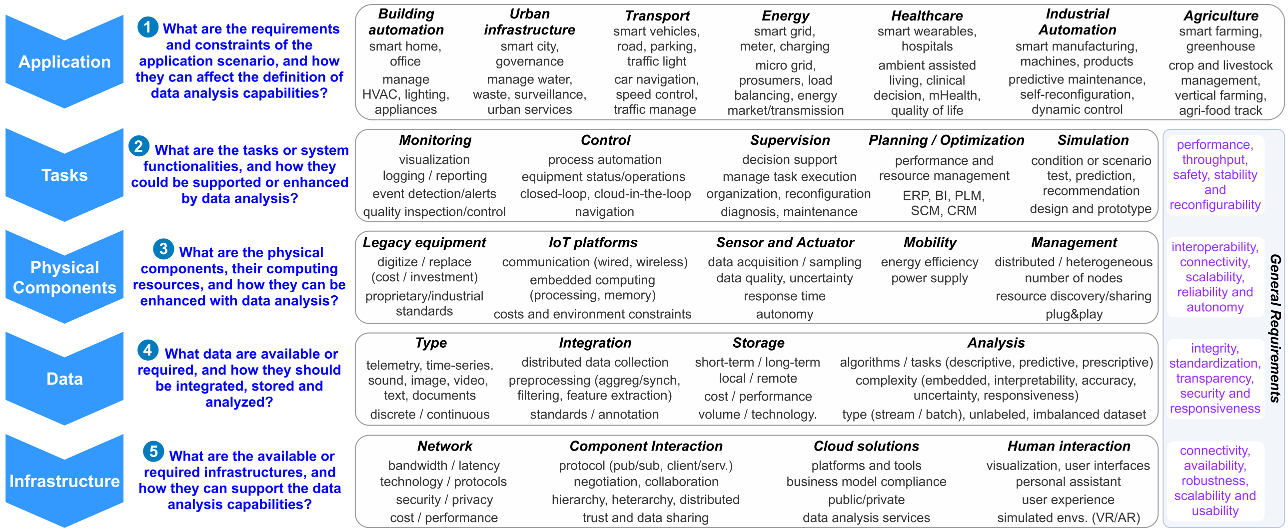

FIGURE 5. **General guideline to identify the data analysis capabilities for the design of CPS, comprising a five steps top-down approach supported by a set of questions that should be answered, considering several concerns and general requirements regarding the system's functionalities.**

Each aspect, further discussed in the following subsections, is supported by a specific question that should be answered, taking in consideration several concerns and general requirements regarding the system's functionalities. Based on the main aspects identified, the general guideline is organized in a five steps top-down approach, starting with the aspects related to the application domain to those related to the required infrastructure.

1) APPLICATION

The framework considers several Application domains where CPS can provide suitable solutions, each one with specific requirements and constraints that should be considered in the definition and specification of the data analysis capabilities. They are related to, e.g., the expected investment to build the application, the expected benefits and impacts for the business, society and/or environment, as well as the implications in the existing solution. For instance, while the application of data analysis in sectors like healthcare and building automation can directly affect people, their well-being, and should have a high concern with the privacy of user data, other sectors like urban infrastructures and transportation can affect the environmental conditions and may need to handle heterogeneous and social data issues. On the other hand, sectors like energy, industry automation and agriculture are more related to the economic impacts and need to manage the data uncertainty.

2) TASKS

The next step comprises the identification of the system tasks and functionalities and how they can be supported by data analysis. Monitoring and Control are two kinds of operational level functionalities. Although Monitoring encompasses a rich field for the application of data analysis, it may face barriers related to the data availability and reliability, but also regarding the dynamic nature of the system that may require the continuous adaptation of the data models. Control tasks usually operate together with Monitoring tasks in a closed or open loop, thus facing the same concerns. Other concerns include, e.g., the integration of data analysis in the control loops, and the trade-offs between the autonomy against the security and the proper operation of the system, given the nondeterministic behavior of some data analysis algorithms.

On the other hand, in tasks more related to business and strategical levels, e.g., Supervision, Planning and Optimization, the application of data analysis aims to enhance decision making, focusing on performance and costs improvements. Some main concerns include the access and integration of heterogeneous and distributed data from third parties and not completely digitized. Simulation is another category of task that can run in parallel with the real environment to evaluate hypothesis or system configurations, and provide recommendations. In this context, data analysis requires careful engineering to provide accurate and timely information.

3) PHYSICAL COMPONENTS

The Physical Components represent an important aspect in CPS. They are characterized by heterogeneous platforms with limited computing resources, leading to several concerns, e.g., the integration of legacy equipment with new solutions and the embodying of data analysis. The adoption of IoT platforms also raises concerns regarding the communication interfaces and protocols, computational capabilities, and the environment constraints. For instance, hostile environments may affect the operation of the sensors, requiring robust and costly platforms.

Additionally, sensors need to attend the application requirements, mainly regarding the data sampling and uncertainty aspects, while controllers need to follow the response time and autonomy constraints. On the other hand, mobile or isolated equipment require energy efficiency that can affect their sensing, communication and computing capabilities. The management of the physical components also raises concerns, e.g., regarding the number of nodes, their distribution and heterogeneity, as well as how they can be dynamically discovered and integrated.

4) DATA

In the definition of data analysis capabilities, the most important aspect is related to identify what is the available or required data, how it should be managed, as well as what kind of analysis should be performed. In this context, the identification of the data type determines, not only how it will be managed and analyzed, but also the issues that should be faced by the data analysis solution. For instance, while telemetry data can present significant noise, missing values and variable sample rate, video data may require huge storage space or data compressing strategies. Another aspect comprises the mechanisms to collect and integrate distributed data, especially in the cases of complex CPS, characterized by a large number of heterogeneous nodes and external data sources. How and what kind of data should be stored is another concern, where the volume of data, storage technology, as well as the costs and performance should also be considered.

The definition of the data analysis algorithms is a key concern that may require expertise and domain knowledge. In this context, the choice of the algorithm must comply with the system constraints. For instance, it is necessary to consider the characteristics of data, the accuracy and the responsiveness of the data analysis algorithm.

5) INFRASTRUCTURE

The last aspect concerns the available or required infrastructures to support the data analysis capabilities. This aspect includes the network infrastructure that besides the bandwidth and latency constraints, also raises concerns regarding the technology, transmission protocols and data security. The distributed nature of CPS components raises concerns about interaction protocols and topologies, as well as negotiation, collaboration and trust mechanisms. On the other hand, although Cloud solutions can provide several tools and data analysis services, the public Cloud should be compliant with the business model, while private Cloud can be an alternative.

Another aspect comprises the interfaces for the interaction with humans that will operate or use the system. In this sense, besides the design of tools to visualize the data analysis outputs and assist the users in their tasks, it is also necessary to consider interfaces to retrieve the user feedback, e.g., to obtain information to improve the data models. Although more advanced interfaces, e.g., personal assistants or virtual environments, can provide a more interactive and enhanced experience, they still face many limitations and challenges.

B. CRITERIA TO DETERMINE WHERE TO DEPLOY DATA ANALYSIS CAPABILITIES

After identifying the requirements and constraints of the CPS data analysis capabilities, the next phase comprises the design of the system architecture and components (Fig. 4). The guideline proposed in the previous section can support the identification of the data analysis tasks together with the most relevant concerns, requirements and constraints for a given application scenario. Although being aware of them, the effective design and development of data analysis capabilities in CPS still is a complex task, especially considering the number of criteria that need to be evaluated to decide the most suitable computing layer to deploy a given data analysis capability.

In this context, during the design phases, the choice of where data analysis capabilities should be deployed is usually performed in an ad-hoc manner, mainly based on the engineers and architects experience and intuition. In this sense, in order to have a more technical and systematic approach to better support such decisions, the proposed framework considers seven main quantitative criteria to create a multi-criteria decision support system, that is presented in Section III-C. These criteria, briefly described in the following sub-sections, were identified as the most relevant by considering the related works (described in Section II-B), where they are commonly used, representing those that most affect the decision to perform a given data analysis task at the Cloud or Edge Computing layers.

Although their choice is mainly influenced by the characteristics of the case study used in this work, additional criteria can be easily added to this approach, as well as the use of a partial number of these criteria.

1) RESPONSIVENESS

Responsiveness describes the time the system has to respond to an observed state without compromising its operation and reliability. In this sense, it is directly dependent on the system functionality requirements, and can range from few milliseconds (ms), like in the cases of hard real-time operational tasks, to seconds, more common in management and decision support tasks. High responsiveness requires Edge processing, when it is not possible to ensure the reliability of the network connection.

2) PROCESSING

This criterion represents the average time that a data analysis algorithm takes to process a given data sample, being dependent on the complexity of the algorithm, and evaluated in conjunction with the available computational resources. For instance, simpler algorithms can present fast response time, but may not provide the desired accuracy levels. In this context, the processing time can range from milliseconds in the case of simple algorithms, to seconds for complex algorithms. Note that, this criterion must consider the time from the moment the data arrives at the processing module until the result is obtained. This may include data transformation, feature extraction, and even (en/de)cryption. Given the limited resources of Edge platforms, high processing requirements need to consider the use of Cloud or Fog.

3) DATA PERSISTENCE

This criterion represents the average time interval that the data samples need to be kept, regarding the short or long-term memory. Although most algorithms dedicated to analyze data streams process the data sample once it arrives, in some cases and according to the data analysis strategies, the samples can be processed in small batches (e.g., a time window that can be averaged to determine a trend). On the other hand, some scenarios require more complex algorithms, e.g., regarding the descriptive or prescriptive data analysis that need to process large batches of data. In this sense, the data persistence can range from minutes to days. While in the first case the data can be kept in the memory (short-term), in the second case it may be necessary to use files or databases (long-term). This criterion is directly related to the amount of data the component needs to keep in memory or store, and also the algorithm processing time, both affecting the required computing resources. Therefore, Edge devices are suitable for short-term data persistence, otherwise Fog or Cloud are preferred.

4) BANDWIDTH

This criterion represents the available data transfer rate between Edge devices and Fog-Cloud systems, indicating the round-trip-time of the communication channel, i.e. the network throughput, directly affecting the responsiveness requirements. While the bandwidth between Edge and Fog depends on the local network infrastructure, where affordable equipment, like routers, can provide very high values, for the Cloud it is limited by the subscribed internet service. Moreover, these scenarios mainly consider the upload speed that can range from few kilobits per second (Kbps), like in some mobile network connections, to hundreds of megabits per second (Mbps) in the cases of broadband connections. High available bandwidth favors the use of Cloud solutions, but the reliability and quality of the network connection should be considered in scenarios that require high responsiveness.

5) NUMBER OF NODES

This parameter represents the number of nodes that will share the network, directly affecting the bandwidth and the quality of the connection. Moreover, the number of nodes and the data produced by them influence the computing resources requirements (e.g., storage and processing) of the remote systems that interact with them. According to the application, the number of nodes varies from few nodes, in simple environments, to hundreds of nodes, in more complex environments. A high number of nodes can overload the network and remote systems, which can be addressed by decentralizing the data processing to the Edge.

6) MESSAGE RATE

This parameter represents the average number of messages sent by each component per second, and like the number of nodes, can overload the network and the systems that need to process them. Usually, it is associated with the data sampling rate, but it should be defined based on the data analysis capability requirements, to optimize the quality of outputs and computing resource utilization. In this context, the average message rate can range from very few msgs/sec, in scenarios where the devices only inform status or events, to hundreds of msgs/sec in the cases of real-time monitoring systems. Note that, although in some scenarios there are sensors that can achieve sampling rates in the order of kHz, these scenarios assume by default the use of local devices to process the data (at least extracting features from these samples before sending any data to other systems), since it is unfeasible to send such amount of data over the network.

7) MESSAGE SIZE

This criterion represents the average amount of data transported by a message. Although it is associated with the collected data, it is important to note that, for instance, some devices can measure several parameters that can be aggregated and encoded in a structure before send them to other components. Like the number of nodes and message rate, the message size can range from few bytes, like in telemetry systems, to hundreds of kilobytes (Kb) in systems based on image or video.

C. FUZZY LOGIC DECISION SUPPORT SYSTEM TO DETERMINE WHERE TO DEPLOY DATA ANALYSIS

Although there are other criteria, including qualitative ones, e.g., regarding costs and security aspects, in most cases, the evaluation of the criteria discussed in the previous section must be enough to define the most suitable computing layer to deploy a given data analysis capability. However, the design of a detailed mathematical model that combines all these criteria, and also handles the uncertainty during the system design, is challenging. In this context, there are several multi-criteria decision-making methods [29], including AHP (Analytic Hierarchy Process), TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) and Fuzzy Logic.

While the first two are used for ranking alternatives, the Fuzzy Logic can also be used for mapping the experts knowledge into Fuzzy rules, creating a decision support system that allows to incorporate imprecise information in the form of linguistic terms, e.g., when the boundaries of the criteria or alternatives are not clearly defined. Fuzzy Logic has been used for decision support in several engineering domains for modeling problems involving imprecision, vagueness and subjective aspects [29], [30], matching perfectly the uncertainty of this design problem.

In this context, a Fuzzy system was designed to determine the most suitable computational layer where data analysis tasks should be deployed, considering as input variables the

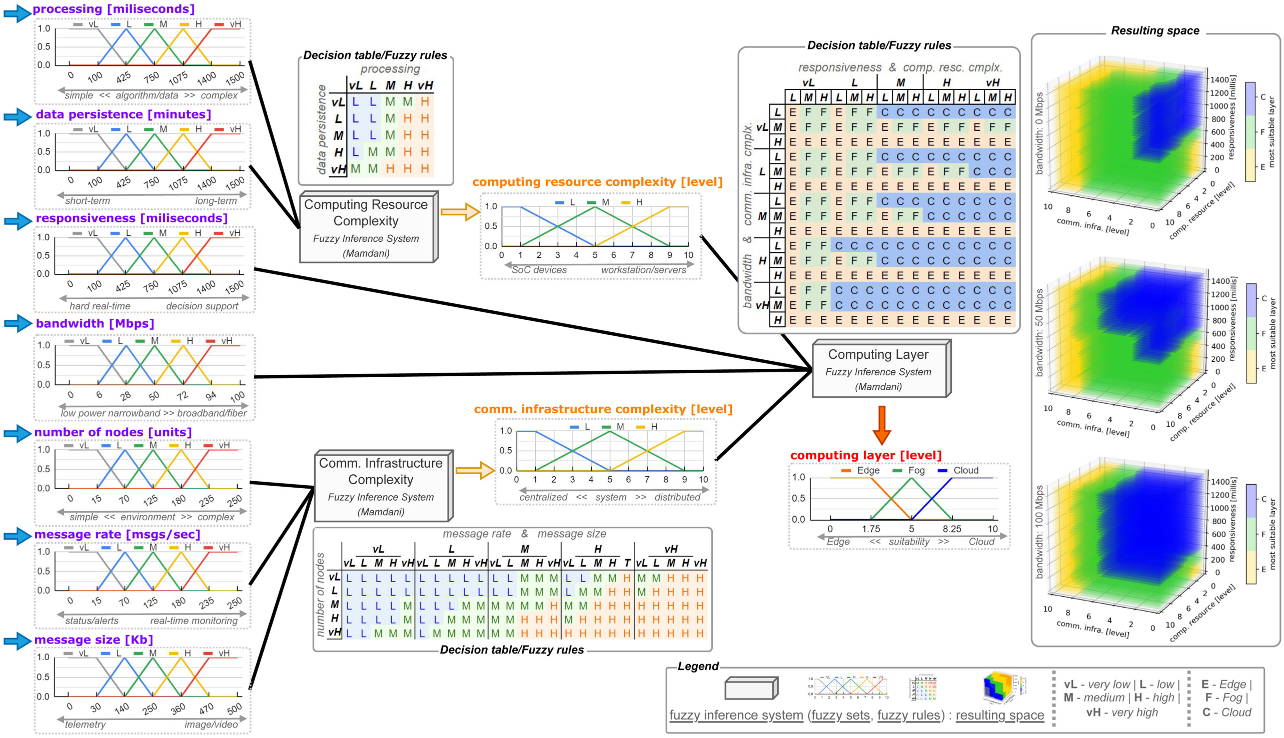

FIGURE 6. Fuzzy decision-support approach: fuzzy sets, membership functions, fuzzy rules, inference systems and the resulting spaces for 3 cases when the bandwidth criterion assumes values of very low (0), medium (50) and very high (100).

seven described criteria, being each data analysis capability characterized by a set of values of these criteria. For each input variable, a Fuzzy set was defined based on the following linguistic terms: *very low (vL)*, *low (L)*, *medium (M)*, *high (H)* and *very high (vH)* (see Fig. 6). In order to reduce the complexity of the Fuzzy system, in terms of the number of Fuzzy rules (that increases exponentially with the number of input variables and linguistic terms), this approach considers a combination of 3 Fuzzy systems, as illustrated in Fig. 6.

The first Fuzzy system considers as input variables the *processing* and *data persistence* criteria, and provides as output the level of *computing resource complexity*. The second Fuzzy system considers as input variables the *number of nodes*, the *message rate* and the *message size* criteria, and gives as output the level of *communication infrastructure complexity*. The third Fuzzy system considers as input variables the outputs of the previous two fuzzy systems, and also the *responsiveness* and the *bandwidth* criteria, providing as output the level of suitability of each *computing layer* to deploy a given data analysis capability.

The Fuzzy sets of the input variables were defined using trapezoidal (at the edges) and triangular (in the middle) Membership Functions (MFs), where their values range follow the characteristics of each criteria, as also illustrated in Fig. 6. The output variables were similarly defined, where their value ranges represent a level from 0 to 10. In the case of the third Fuzzy system, its Fuzzy sets were described by the linguistic terms Edge, Fog and Cloud. Their overlapping illustrates the scenarios where usually a solution can be deployed on different layers, e.g., an output value close to 0 suggests a high suitability of Edge, and likewise 5 to Fog and 10 to Cloud. In this sense, the defuzzyfied output value of the Fuzzy system is mapped to a suitability level ranging from 0 to 1, for each layer (defined in equations (1), (2), (3) and (4)).

$$x = L(r, b, R(p, d), I(n, m, s)) \tag{1}$$

$$S(Edge) = \begin{cases} 1 & \text{for x} \leq 1.75 \\ \frac{-4x+20}{13} & \text{for x} \in [1.75,5] \\ 0 & \text{for x} \geq 5 \end{cases} \tag{2}$$

$$S(Fog) = \begin{cases} 0 & \text{for x} \leq 1.75 \text{ and x} \geq 8.25 \\ \frac{4x-7}{13} & \text{for x} \in [1.75,5] \\ \frac{-4x+33}{13} & \text{for x} \in [5,8.25] \end{cases} \tag{3}$$

$$S(Cloud) = \begin{cases} 0 & \text{for x} \leq 5 \\ \frac{4x-20}{13} & \text{for x} \in [5,8.25] \\ 1 & \text{for x} \geq 8.25 \end{cases} \tag{4}$$

where:

$R(p, d)$: is the *Computing Resource Complexity* Fuzzy inference system

$I(n, m, s)$: is the *Comm. Infrastructure Complexity* Fuzzy inference system

$L(r, b, R, I)$: is the *Computing Layer* Fuzzy inference system

r, b, p, d, n, m, s	: are the input criteria: *responsiveness*, *bandwidth*, *processing*, *data persistence*, *number of nodes*, *message rate* and *message size*
x	: is the crisp value output of the *Computing Layer* Fuzzy Inference system
$S(layer)$	: is the suitability of a given x for each $layer \in [Edge, Fog, Cloud]$

The Fuzzy inference system uses a set of IF-THEN rules that map the MFs of the input variables to the MFs of the output variables. The decision tables, illustrated in Fig. 6, present the rules defined for each Fuzzy system. Each cell represents the result of a logical operation AND between the input variables (rows and columns). In this case, the IF-THEN rules for a given scenario can be interpreted as the following, where 1) evaluates the Computing Resource Complexity (top-left decision table of Fig. 6), 2) evaluates the Comm. Infrastructure Complexity (bottom-left decision table of Fig. 6), and 3) evaluates the Computing Layer (top-right decision table of Fig. 6).

1) **IF** *data persistence* = `vH` **AND** *processing* =`M` **THEN** *computing resource complexity* `is H`
2) **IF** *nodes* = `vL` **AND** *message rate* = `vL` **AND** *message size* = `H` **THEN** *comm. infrastructure complexity* `is L`
3) **IF** *responsiveness* = `L` **AND** *computing resource complexity* = `H` **AND** *bandwidth* = `L` **AND** *comm. infrastructure complexity* = `L` **THEN** `most suitable` *computing layer* `is Fog`

The Fuzzy rules considered in the decision tables define the resulting space for each Fuzzy system. For instance, in the first Fuzzy system, the level of *computing resource complexity* increases with the increase of processing and/or data persistence. Similarly, the level of *comm. infrastructure complexity* will be higher the greater the number of nodes, message size and rate, and lower in the opposite scenario. In the case of the *computing layer*, a higher bandwidth favors the Cloud, except if the communication infrastructure complexity is high or if a low responsiveness is required, when the Edge is more suitable. This can be observed in the three 3D charts, included in Fig. 6 that illustrate the resulting spaces of the Fuzzy decision-support approach for the 3 scenarios when the bandwidth criterion assumes values of very low (0), medium (50) and very high (100).

The configuration of the MFs, the Fuzzy sets and Fuzzy rules were defined based on the experience and knowledge of the authors and the feedback from an interview with a group of experts, where the best configuration was found empirically, after several computational experiments.

IV. CASE STUDY AND EXPERIMENTAL RESULTS

The proposed framework was conceptually validated using a testbed case study, illustrated in Fig. 7, that is based on a set of smart electric machines. The design of the CPS solution requires to determine how the data analysis capabilities associated to each smart machine should be distributed among the Edge, Fog and Cloud, taking into account the requirements imposed by the industrial scenario.

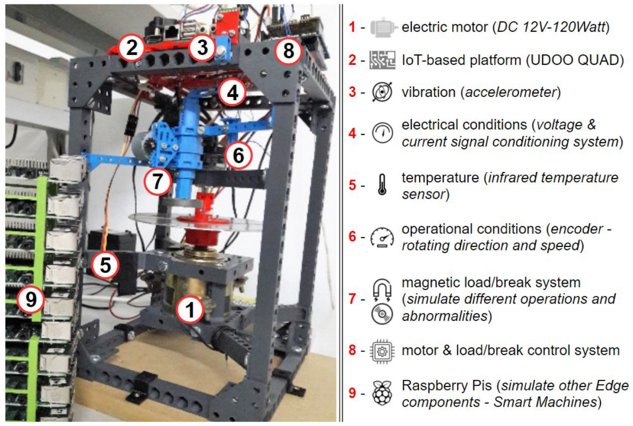

FIGURE 7. Case study setup: the smart machine testbed.

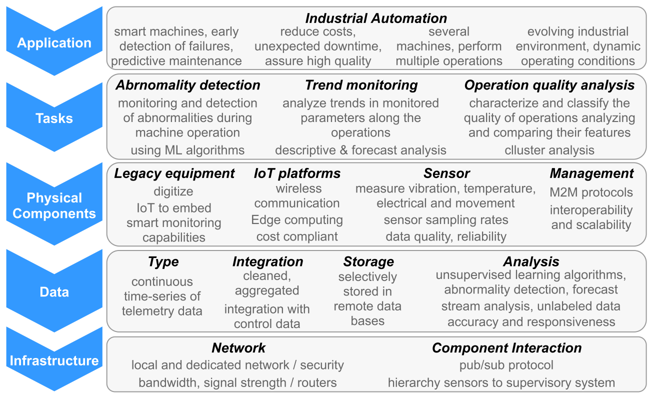

FIGURE 8. Summary of the identified concerns and requirements for the case study scenario.

A. IDENTIFYING THE DATA ANALYSIS CAPABILITIES CONCERNS AND REQUIREMENTS

The first part of the proposed framework encompasses the use of the conceptual guideline to identify and evaluate the concerns, requirements and constraints of the data analysis capabilities to be developed. Fig. 8 summarizes the outputs of this process for the case study considered in this work.

The case study comprises an application of smart machines in the industrial automation domain. For this purpose a testbed that comprise an electric motor that was instrumented with sensors and an IoT platform (UDOO QUAD) that enables the measurement of several parameters and their local processing [31]. The testbed also includes 10 Raspberry Pis to simulate other smart machines, and a network infrastructure that comprises computers and virtual machines (used as Fog/Cloud platforms) to support the testing of different scenarios.

The solution to be developed requires the condition monitoring and abnormality detection of the machine operation, aiming to reduce the unexpected downtime and ensure the high quality of products. The focus is on taking advantage of the state of the art ML algorithms to build data models capable to continuously analyze the machine operational data and produce alerts when unexpected events are detected. Other tasks planned for the smart machine include the analysis of some parameters behaviors in order to monitor KPIs, and identify and forecast trends. In this context, three data analysis tasks are considered in this case study:

1) *Abnormality detection* – continuously monitor the electric current parameter to detect abnormalities related to the load patterns of the machine operation.
2) *Trend monitoring* – analyze the machine operation to identify trends and forecast the energy consumption, temperature and vibration.
3) *Operation quality analysis* – characterize and classify the quality of the performed operations based on the analysis of the machine's parameters.

The physical components comprise legacy machines that need to be digitalized with IoT and ML technologies to embed smart data analysis capabilities. Such data analysis tasks require sensors that can measure mechanical (e.g., vibration and temperature), electrical (e.g., current and voltage) and operational (e.g., motor rotating direction and speed) parameters. The data sampling rate and quality can be limited by the sensor technology and cost, network bandwidth and local processing capabilities. However, they should be defined in conjunction with the data analysis algorithms to attend a desired level of accuracy and responsiveness.

The sensors provide a continuous time-series of telemetry data, leading to huge volumes of data. In order to reduce the data management and storage costs, the samples should be cleaned, aggregated and selectively stored in remote data bases, for data models improvement purposes. Additionally, the integration with the existing system should be considered, in order to retrieve the control information to support the data analysis models. Besides that, the data samples will not be labeled (i.e., annotated with normal or abnormal labels), and most of them will comprise normal behavior, since abnormal events should be rare. In these cases, unsupervised learning algorithms usually represent a suitable option. These types of algorithms are trained to learn the patterns of normal behaviors, thus any data sample that does not follow these patterns is identified as an abnormality. Besides the classification of current samples, such algorithms can also be used to predict events in advance. Based on the identified data analysis tasks, ML techniques like Autoencoders can be used to detect abnormalities in time-series data while clustering techniques can be used to group and classify the quality of operation based on the similarities of operating conditions.

Regarding the network infrastructure, given the environment characteristics and security concerns, a local wireless network infrastructure can be easily adopted without affecting the existing industrial control network. It should provide bandwidth and cover all the machines, ensuring the connectivity and responsiveness constraints, as well as the company data security policies. Regarding the components interaction,

TABLE 1 **Evaluation of the Data Analysis Tasks Using the Fuzzy System**

Fuzzy input criteria for data analysis tasks		*abnormality detection*	*trend monitoring*	*operation quality analysis*
processing (ms)		10	300	2500
data persistence (mins)		0	15	30
responsiveness (ms)		100	1000	2000
bandwidth (Mbps)		10	10	10
number of nodes (units)		50	50	50
message rate (msgs/sec)		30	30	30
message size (Kb)		1	1	1
Fuzzy output crisp values		*1.92*	*8.06*	*8.08*
layer suitability [0-1]	Edge	0.95	0	0
	Fog	0.05	0.06	0.05
	Cloud	0	0.94	0.95

a hierarchical approach should be adopted, since the solution does not require autonomous components capable to collaborate and perform local diagnosis. The outputs of the solution, i.e. the monitoring status and abnormality alerts, are intended to be provided as a service that should be integrated in an existing monitoring system, thus no human interfaces should be needed.

B. DEFINING WHERE TO DEPLOY THE DATA ANALYSIS CAPABILITIES

After identifying the required data analysis capabilities and the related concerns regarding the given industrial application scenario, the proposed Fuzzy Logic recommendation system can be used during the system design phase to suggest the most suitable layer to deploy them. Note that, during the design phase the values of the criteria may need to be estimated based on the system features and requirements, or, e.g., regarding some preliminary experiments that test the performance of the algorithms and computing platforms. This is not a concern, since the Fuzzy approach supports the decision making based on uncertain and estimated values.

In this sense, Table 1 presents the criteria of the three data analysis tasks defined in the case study, as well as the output of the Fuzzy system (defuzzyfied crisp value), and the respective layer suitability, computed according to the equations (2), (3), (4). The Fuzzy system was implemented in Python, using the library skfuzzy that implements the Mamdani Fuzzy type. The centroid method was used for the defuzzyfication process.

In this case study, the three identified data analysis tasks consider the same components and infrastructure. Therefore, some criteria are the same for all the tasks, like the available bandwidth that, in terms of the upload speed, is based on a regular WiFi network connection ($\sim$10 Mbps), as well as the number of nodes (50), the message rate ($\sim$30 msgs/sec) and the message size ($<$1 Kb). These criteria present low values, indicating that this case study is characterized by a low network infrastructure complexity. However, the required responsiveness and computing resources are dependent of the data analysis tasks, and still need to be evaluated before choosing the most suitable layer.

As example, the first data analysis task is related to the early detection of abnormality during the machine operation, which does not require any data persistence (0 mins) since it is based on data stream analysis, and the processing time of the data analysis algorithm is very low (∼10 ms in the worst case) since the processing of telemetry data is relatively simple and fast. However, the required responsiveness for these tasks is very low (∼100 ms), indicating that it is critical for the system to identify the occurrence of an abnormality as quickly as possible. Considering these values for the Fuzzy input criteria, the output of the Fuzzy inference system provides a crisp value of 1.92 that, according to the equations (2), (3) and (4), indicates that the Edge is the most suitable layer, presenting a suitability value of 0.94 (see Table 1).

The second data analysis task, that encompasses the analysis of trends in the operational data, requires some data persistence, since it must consider several operations over the last 15 minutes. This requires the processing of a greater volume of data that directly affects the algorithm processing time (∼300 ms). This task requires trends and forecasts of some KPIs to be presented after each operation to monitor their evolution. Since trends are observed in the long-term, the required responsiveness is not critical (1000 ms), i.e. at most 1 s after the end of the operation. Considering these characteristics, the Fuzzy inference system gave a crisp value of 8.06, that indicates the Cloud as the most suitable layer with a suitability value of 0.94 (Table 1). This output can be justified based on the increase of the computational resource complexity, caused by the high value of processing and data persistence, but also by the relaxed value for the responsiveness.

The third data analysis task is similar to the previous one, requiring the processing of batches of data from multiple operations. In this case, it has more computational complexity, in terms of the processing (∼2500 ms) and data persistence (30 min). Moreover, the required responsiveness is less critical (∼2000 ms), mainly given the descriptive analytics nature of this task, i.e., identify and characterize behaviors in the historical operational data. Although both criteria, processing and responsiveness, present values above the range defined in the Fuzzy system (Fig. 6), they are automatically adjusted to the maximum value, with the full membership in the very high Fuzzy set. Based on this analysis and considering the other criteria, specially the low network communication complexity, this task is recommended to be performed at the Cloud (crisp value of 8.08 and suitability value of 0.95) that attends the responsiveness requirements, and also contribute to not overload the local system.

C. EXPERIMENTS AND ANALYSIS OF RESULTS

The data analysis capabilities described in the case study were implemented and deployed in the computational layer as recommended by the Fuzzy Logic recommendation system (discussed in the previous sub-section).

In the experiments, a set of Raspberry Pi 3 B+ were used as the Edge platforms, while a local and a remote computer (4 cores/2.8 GHz/16 GB) were used as Fog and Cloud platforms.

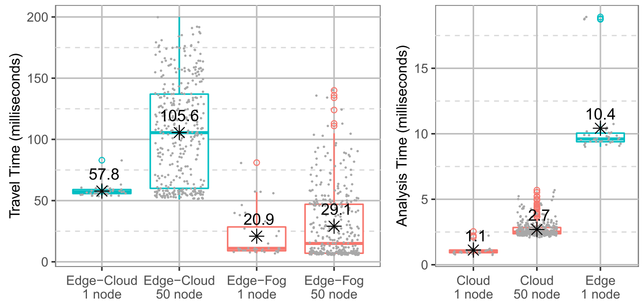

FIGURE 9. The distribution of travel and data analysis times for the early detection of abnormality task.

35 parameters are measured every 30 ms, being required ∼250 Bytes to be sent in a message. Although this represents low values for the *message rate* and *message size* criteria, depending on the number of nodes, they can easily overload the network bandwidth. In this context, the preliminary experiments showed that performing data analysis at the Edge resulted in ∼4.1 KB sent and ∼5.9 KB received per node (only status message sent/received), against ∼14 MB sent (403 msgs/operation) and ∼5.9 KB received, when the raw data are sent to be processed at Fog/Cloud. The execution of an experimental test takes ∼32 minutes, leading to a bandwidth consumption of ∼61 Kbps per node, while 50 nodes require ∼3 Mbps.

For the early detection of abnormality task, an Autoencoder, based on LSTM (Long-Short Term Memory) layers [6], was implemented using the DeepLearning4J Java library (deeplearning4j.org). The model was configured with 400 × 50 × 10 × 50 × 400 neurons per layer and was trained to detect abnormalities in the time-series data stream of the motor electric current parameter [31]. The experiments used a dataset with 120.904 data samples collected from the testbed. Half of it has no abnormalities and was used to train the model, while the other half, used for the tests, contains 12 abnormalities in 150 operations.

Fig. 9 illustrates the experiments' results for the early detection of abnormality task. The distribution and average values of the travel time between Edge and Fog/Cloud (left chart) illustrate how the network can affect the communication between these platforms, also considering the effects of data overload (e.g., higher magnitude and variability) when 50 Edge nodes share the same network. On the right, the chart shows the distribution and average of the times that the platforms of each layer takes to process a data sample using the Autoencoder model. The Edge presents the highest average value (∼10 ms), reflecting its lower computing resources when compared to the computers used in the other layers. In this case, the Fog and Cloud have similar computing resources, which are equally affected by the data overload (1 vs 50 Edge nodes).

This experiment indicates that this Edge platform fulfills the requirement of a processing time of 10 ms. On the other hand, the travel time presented average values of ∼29 ms

TABLE 2 Average Times of Data Transport and Analysis for Different Layers

layers	travel time (ms)	abnormality detection	trend monitoring	operation quality analysis
		average processing time (ms)		
Edge	0	10.4	306.4	2470.5
Fog	29.1	2.7	48.4	353.2
Cloud	105.6	2.7	48.4	353.2

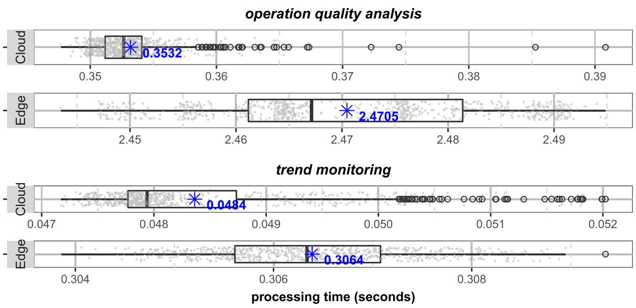

FIGURE 10. Distribution and average processing times of two data analysis tasks, when deployed on Edge and Cloud layers (the x-axis presents different scales to better illustrate the different processing times obtained in each computational layer).

and ~105 ms, for Fog and Cloud platforms, respectively, that should be accounted in the total responsiveness time, estimated at ~100 ms for this data analysis task. Thus, Cloud is not a suitable solution and, although Fog could be used, other criteria like the big number of nodes and messages that share the same wireless network can compromise the connectivity (as illustrated by the variability in Fig. 9 - left side and discussed in the previous section). Table 2 summarizes the travel and processing times for the data analysis tasks regarding each computing layer.

In conclusion, the achieved results support the Edge as a suitable solution to deploy the early abnormality detection task, as suggested by the Fuzzy decision system (discussed in the previous sub-section).

The other two data analysis capabilities were implemented and tested following the same ideas. Both were implemented in python and deployed in Edge and Cloud in order to test their processing times. The outputs regarding the data processing times are illustrated in Fig. 10 and Table 2. The trend monitoring task analyzes the raw data of each operation and extracts KPIs (e.g., the energy consumption, number of rotation, average temperature, vibration). Additionally, an Exponential Smoothing method based on the Holt-Winters algorithm was used to forecast the next operations. Different from the abnormality detection capability, this one considers the processing of batches of data that requires much more computing resources, as illustrated by the average processing times (~306 ms Edge vs. ~48 ms Fog/Cloud). If considered only the estimated responsiveness (1000 ms), this capability could be deployed in the Edge. However, it would consume significant computing resources, compromising the other tasks performed by the Edge platform, that explains the Cloud as a more suitable layer, as suggested by the Fuzzy decision approach that considers the combination of the seven criteria.

The operation quality analysis task follows a similar approach, where several features are extracted from the raw data of operations. Then, a Hierarchical Agglomerative Clustering algorithm is used to cluster the operations according to their behaviors. This process comprises the processing of batches of data from many operations, requiring considerable computing resources. This is illustrated by the high average processing time, ~353 ms in the Cloud and ~2470 ms in the Edge. This also confirms the Cloud as the suitable approach to perform this task, as recommended by the Fuzzy decision system. Although Fog also attends the requirements to support this task, the proposed Fuzzy system was modeled considering that the Fog is a solution when the requirements can not be attended by either Edge or Cloud.

D. ASSESSMENT OF THE FUZZY RECOMMENDATION SYSTEM AND LESSONS LEARNED

The described experimental tests clearly show that the use of the proposed Fuzzy decision-making system provides a lesser ad-hoc approach that simplifies the complexity required to decide of what computing layer is more suitable to deploy a given data analysis task. Although the case study evaluated in the previous sub-sections only explored three data analysis tasks in a specific application scenario, during the tuning process and the mapping of the expert knowledge in the Fuzzy rules, the feasibility of the Fuzzy approach was continuously evaluated in an empirical manner, by the experts and system developers that participated in the implementation of this approach. This evaluation was based on the analysis of the solution space that was computed by considering the different values of the input criteria (as illustrated by the 3D charts of Fig. 6). In this way, based on their feedback, the Fuzzy sets and rules were fine-tuned until the outputs attend their expectations.

This assessment allows to identify some of the main benefits and drawbacks of the Fuzzy approach for this kind of application, as briefly described in the Table 3.

In this system, the outputs are explainable based on the linguistic terms and the Fuzzy rules that map the experts' knowledge. The charts with the solution space for each Fuzzy sub-system, also support such explanation, enabling to visualize the relation between the input criteria and the outputs of the proposed Fuzzy system, and obtain the high-level information about how the final solution was produced. For instance, considering the solution space (Fig. 6), a higher level of Edge suitability can be explained by higher values of the *Comm. Infrastructure Complexity* (caused by high number of nodes, message size or message rate), lower level of Computing Resource Complexity (caused by low processing and data persistence), and high responsiveness, if the bandwidth is low, or low responsiveness, if the bandwidth is high.

As already mentioned, this approach was defined and tuned considering the characteristics of the case study. In this case, in order to support other application scenarios some important

TABLE 3 Identified Benefits and Drawbacks in the Proposed Fuzzy Approach

Benefits	*Drawbacks*
Enables and facilitates the representation of the expert knowledge, and uncertainties	May be biased on the experience and knowledge of those who designed the system
The range of input criteria and the rules can be adapted to incorporate new knowledge and fine-tune the approach or support new scenarios	Experiments are required to fine-tune the system, i.e., after a change in Fuzzy sets or rules, the solution space should be computed to check if the outputs are compliant
As a recommendation expert/rule-based system, its outputs are explainable	Experience and knowledge of multiple experts may require agreements to solve conflicts in system design
Different from ML-based methods, there is no need for datasets to create a model; additionally, if data is available, some Fuzzy approaches can extract the rules, being less dependent on the expert knowledge	Dependent on the know-how and experience of experts that can be challenging to be extracted; additionally, using automatic approaches to extract rules from datasets can be biased or not able to proper generalize the patterns

aspects should be considered. For instance, the ranges of the input criteria are highly dependent on the application scenario. Thus, it is not a good approach to define them for a general solution that can fit different scenarios, since it may increase the complexity to define and fine-tune the Fuzzy rules to achieve the desired outputs. In this case, the best approach is to fine-tune the Fuzzy system considering the technologies and related aspects of the application scenario to be assessed. Also related to the definition of the ranges of the input criteria, although the proposed approach only considers quantitative input criteria, it is also possible to use qualitative ones, e.g., by considering the values of the input range as a level or rating (e.g., 0-10), like in the output criteria of the three Fuzzy systems. This can support the use of subjective criteria that cannot be measured by tools or experiments, but instead provided based on the experience of users.

Another related aspect is the definition of the Fuzzy sets and rules that can be challenging. Besides directly affecting the number of Fuzzy rules and the complexity of the system, there may be conflicts if it is necessary to combine the knowledge of multiple domain experts, an agreements must be found between them.

As a final remark, the proposed Fuzzy decision support approach differs from other decision-making frameworks that could also be used in this solution, especially those based on pairwise comparison, like the multi-criteria decision-making methods, mentioned in Section III-C. While such methods would require the comparison of the Cloud, Fog and Edge alternatives for each data analysis task, the Fuzzy approach allows mapping the knowledge of experts, in terms of a set of rules and an inference mechanism, that can be used to assess different tasks. Besides that, it enables to use the knowledge of highly experienced experts by less experienced engineers, who would otherwise rely on their own knowledge to evaluate the alternatives.

V. CONCLUSION

Although data analysis is a key enabler of CPS, the traditional Cloud-centric solutions face some limitations that can be addressed by Edge computing as a complementary approach. However, balancing the data analysis capabilities along Cloud-Edge layers is not a straightforward task.

In this context, this paper presents a framework to support the design of Cloud-Edge data analysis capabilities that considers a guideline to identify aspects and concerns associated to each data analysis task. Additionally, a Fuzzy Logic approach, based on seven criteria, is defined to support a lesser ad-hoc decision-making regarding the most suitable computing layer to deploy a given data analysis task.

The framework was conceptually validated in an industrial CPS case study, based on a set of smart machines. The guideline was used to support the evaluation of the requirements/constraints of the three identified data analysis tasks, while the Fuzzy Logic system was used to recommend the most suitable layer to deploy them. The identified data analysis tasks were implemented in the Edge and Cloud layers, with the achieved performance allowing to validate the Fuzzy system recommendations. The validation of the proposed system was performed in an empirical manner, as traditionally done for recommendation systems, where the feedback from the users compare the initial requirements defined for the case study with the achieved results from the recommendations.

When compared with the current state-of-the-art, the proposed Fuzzy recommendation system constitutes an innovative approach to reduce the complexity required to determine the most suitable layer to deploy a given data analysis task by using a set of criteria that evaluates the computational and communication capabilities. It also proved to be robust to map the experts' knowledge and flexible to adapt for specific application scenarios, thus comprising a suitable tool to support a lesser ad-hoc design for distributed data analysis in CPS.

Future work will be devoted to further tuning the Fuzzy-based multi-criteria recommendation system, namely the fuzzy sets and rules, and particularly the adequacy to consider other qualitative and quantitative criteria.

REFERENCES

[1] H. Xu, W. Yu, D. Griffith, and N. Golmie, "A survey on industrial Internet of Things: A cyber-physical systems perspective," *IEEE Access*, vol. 6, pp. 78238–78259, 2018.

[2] Y. Lu, "Industry 4.0: A survey on technologies, applications and open research issues," *J. Ind. Inf. Integration*, vol. 6, pp. 1–10, 2017.

[3] Y. Liao, F. Deschamps, E. de Freitas Rocha Loures, and L. F. P. Ramos, "Past, present and future of Industry 4.0-a systematic literature review and research agenda proposal," *Int. J. Prod. Res.*, vol. 55, no. 12, pp. 3609–3629, 2017.

[4] T. Wuest, D. Weimer, C. Irgens, and K.-D. Thoben, "Machine learning in manufacturing: Advantages, challenges, and applications," *Prod. Manuf. Res.*, vol. 4, no. 1, pp. 23–45, 2016.

[5] J. Lee, H. Davari, J. Singh, and V. Pandhare, "Industrial artificial intelligence for industry 4.0-based manufacturing systems," *Manuf. Lett.*, vol. 18, pp. 20–23, 2018.

[6] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, vol. 48, pp. 144–156, 2018.

[7] H. Breivold and K. Sandström, "Internet of things for industrial automation - challenges and technical solutions," in *Proc. IEEE Int. Conf. Data Sci. Data Intensive Syst.*, 2015, pp. 532–539.

[8] M. Aazam, S. Zeadally, and K. A. Harras, “Deploying fog computing in industrial Internet of Things and industry 4.0,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 10, pp. 4674–4682, Oct. 2018.
[9] L. Li, K. Ota, and M. Dong, “Deep learning for smart industry: Efficient manufacture inspection system with fog computing,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 10, pp. 4665–4673, Oct. 2018.
[10] M. Chiang and T. Zhang, “Fog and IoT: An overview of research opportunities,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.
[11] R. Mahmud, R. Kotagiri, and R. Buyya, *Fog Computing: A Taxonomy, Survey and Future Directions*. Singapore: Springer, 2018, pp. 103–130.
[12] K. Dolui and S. K. Datta, “Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing,” in *Proc. Glob. Internet Things Summit*, 2017, pp. 1–6.
[13] M. Satyanarayanan, “The emergence of edge computing,” *IEEE Comput.*, vol. 50, no. 1, pp. 30–39, Jan. 2017.
[14] Y. Ai, M. Peng, and K. Zhang, “Edge computing technologies for Internet of Things: A primer,” *Digit. Commun. Netw.*, vol. 4, no. 2, pp. 77–86, 2018.
[15] S. Teerapittayanon, B. McDanel, and H. Kung, “Distributed deep neural networks over the cloud, the edge and end devices,” in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst.*, 2017, pp. 328–339.
[16] M. de Brito, S. Hoque, R. Steinke, A. Willner, and T. Magedanz, “Application of the fog computing paradigm to smart factories and cyber-physical systems,” *Trans. Emerg. Telecommun. Technol.*, vol. 29, no. 4, 2018, Art. no. e3184.
[17] P. Hu, S. Dhelim, H. Ning, and T. Qiu, “Survey on fog computing: Architecture, key technologies, applications and open issues,” *J. Netw. Comput. Appl.*, vol. 98, pp. 27–42, 2017.
[18] J. Pan and J. McElhannon, “Future edge cloud and edge computing for Internet of Things applications,” *IEEE Internet Things J.*, vol. 5, no. 1, pp. 439–449, Feb. 2018.
[19] Y. Sahni, J. Cao, S. Zhang, and L. Yang, “Edge mesh: A new paradigm to enable distributed intelligence in Internet of Things,” *IEEE Access*, vol. 5, pp. 16441–16458, 2017.
[20] H. El-Sayed *et al.*, “Edge of things: The big picture on the integration of edge, IoT and the cloud in a distributed computing environment,” *IEEE Access*, vol. 6, pp. 1706–1717, 2018.
[21] M. Aazam, S. Zeadally, and K. A. Harras, “Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities,” *Future Gener. Comput. Syst.*, vol. 87, pp. 278–289, 2018.
[22] L. Lin, X. Liao, H. Jin, and P. Li, “Computation offloading toward edge computing,” *Proc. IEEE*, vol. 107, no. 8, pp. 1584–1607, Aug. 2019.
[23] Q. D. La, M. V. Ngo, T. Q. Dinh, T. Q. Quek, and H. Shin, “Enabling intelligence in fog computing to achieve energy and latency reduction,” *Digit. Commun. Netw.*, vol. 5, no. 1, pp. 3–9, 2019.
[24] L. Yin, J. Luo, and H. Luo, “Tasks scheduling and resource allocation in fog computing based on containers for smart manufacturing,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 10, pp. 4712–4721, Oct. 2018.
[25] P. Bellavista and A. Zanni, “Feasibility of fog computing deployment based on docker containerization over raspberrypi,” in *Proc. 18th Int. Conf. Distrib. Comput. Netw.*, 2017, pp. 1–10.
[26] J. Lee, B. Bagheri, and H.-A. Kao, “A cyber-physical systems architecture for industry 4.0-based manufacturing systems,” *Manuf. Lett.*, vol. 3, pp. 18–23, 2015.
[27] C. C. Byers, “Architectural imperatives for fog computing: Use cases, requirements, and architectural techniques for fog-enabled IoT networks,” *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 14–20, Aug. 2017.
[28] M. Ashouri, P. Davidsson, and R. Spalazzese, “Cloud, edge, or both? towards decision support for designing iot applications,” in *Proc. 5th Int. Conf. Internet Things: Syst., Manage. Secur.*, 2018, pp. 155–162.
[29] M. Velasquez and P. T. Hester, “An analysis of multi-criteria decision making methods,” *Int. J. Operations Res.*, vol. 10, no. 2, pp. 56–66, 2013.
[30] A. Mardani, A. Jusoh, and E. K. Zavadskas, “Fuzzy multiple criteria decision-making techniques and applications–Two decades review from1994 to 2014,” *Expert Syst. With Appl.*, vol. 42, no. 8, pp. 4126–4148, 2015.
[31] J. Queiroz, P. Leitão, J. Barbosa, and E. Oliveira, “Agent-based approach for decentralized data analysis in industrial cyber-physical systems,” in *Industrial Applications of Holonic and Multi-Agent Systems*. Cham, Switzerland: Springer, 2019, pp. 130–144.

JONAS QUEIROZ received the B.S. and M.S. degrees in computer science from São Paulo State University, Brazil, in 2007 and 2012, respectively. He is currently working toward the Ph.D. degree in informatics engineer with the Faculty of Engineering, University of Porto, Porto, Portugal.

Since 2019, he has been a Researcher with Research Center in Digitalization and Intelligent Robotics, Polytechnic Institute of Bragança, Bragança, Portugal. His research interests include the application of multi-agent system, data analysis, IoT and edge computing for the development of autonomous and intelligent components in the realization of cyber-physical systems in the context of the 4th industrial revolution.

PAULO LEITÃO (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering, from the University of Porto, Porto, Portugal, in 2004. He joined the Polytechnic Institute of Bragança, in 1995, where he is currently a Full Professor with the Department of Electrical Engineering and Coordinator of CeDRI, Research Centre in Digitalization and Intelligent Robotics. He has authored or coauthored four books and more than 300 papers in high-ranked international scientific journals and conference proceedings (peer-review). His research interests include industrial informatics, intelligent and reconfigurable systems, cyber-physical systems, the Internet of Things, distributed data analysis, factory automation, multi-agent systems, holonic systems, and self-organized systems. He was the General Co-Chair of several international conferences, namely IEEE INDIN’18, SOHOMA’16, IEEE ICARSC’16, HoloMAS’11, and IFAC IMS’10. He is a Senior Member of the Systems, Man, and Cybernetics Society (SMCS), Chair of the IEEE IES Technical Committee on Industrial Agents, and a Member at-Large of the IEEE IES Administrative Committee (AdCom). He was the Chair of the established IEEE 2660.1-2020 standard.

Dr. Leitão was the recipient of the six Scientific Paper Awards and is the coauthor of three patents. He has coordinated/participated in several national and international research projects (EU FP7 and H2020) and networks of excellence.

EUGÉNIO OLIVEIRA received the Ph.D. degree in artificial intelligence from the New University of Lisbon, Lisbon, Portugal, in 1984.

He was a Guest Academic with IBM/IEC in Belgium (84–85). He has been an Emeritus and a Full Professor (Artificial Intelligence) with the University of Porto, Porto, Portugal. He is Co-Founder of LIACC, Artificial Intelligence and Computer Science Laboratory. He created the Ph.D. Program on informatics engineering with the University of Porto. His current research interests include software agents trust and reputation models, adaptive agents for E-Business, intelligent transportation systems, emotional-like agents, agent’s strategies for cooperation and negotiation, text mining, AI and ethics. He was the Area Chair for Agents with the European Conference on Artificial Intelligence - ECAI 2008, and the General Co-Chair for EPIA - AI conference in 2017.

Dr. Oliveira was awarded with the Gulbenkian Prize for Science and Technology. He successfully supervised 20 Ph.D. students and more than 50 Master students. He was involved in research projects, funded by either national or international funding agencies.