Evaluation Metrics for Collaborative Fault Detection and Diagnosis in Cyber-Physical Systems

Luis Piardi*‡¶, André Oliveira ¶, Pedro Costa ‡§, Paulo Leitão*†
* Research Center in Digitalization and Intelligent Robotics (CeDRI), Instituto Politécnico de Bragança, Campus de Santa Apolónia, 5300-253 Bragança, Portugal.
Email: {piardi, pleitao}@ipb.pt
† Laboratório Associado para a Sustentabilidade e Tecnologia em Regiões de Montanha (SusTEC), Instituto Politécnico de Bragança, Campus de Santa Apolónia, 5300-253 Bragança, Portugal
‡ Faculty of Engineering University of Porto (FEUP), Rua Dr. Roberto Frias, 4200-465 Porto, Portugal.
Email: pedrogc@fe.up.pt
§ Institute for Systems and Computer Engineering, Technology and Science (INESC TEC), 4200-465 Porto, Portugal
¶ Universidade Tecnológica Federal do Paraná (UTFPR), Av. Sete de Setembro, 80230-901, Curitiba-PR, Brazil.
Email: andreoliveira@utfpr.edu.br

2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA) | 979-8-3503-6123-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/ETFA61755.2024.10710719

***Abstract*—Cyber-physical systems (CPS) rapidly expand within industrial contexts in a new era of digitalization, processing power, and inter-device communication capabilities. These advancements integrate technologies such as the Internet of Things (IoT), artificial intelligence (AI), and cloud and edge computing, granting processes and operations a high degree of autonomy. In addition, these interconnections foster collective intelligence arising from information exchange and collaboration between components, often outperforming individual capabilities. This collective intelligence manifests in fault detection and diagnosis (FDD) tasks within CPS, as it significantly improves the flexibility, performance, and scalability. However, the inherent complexity of CPS poses challenges in determining the best configuration of the collaboration parameters, such as when and how to collaborate, wherein incorrect adjustments may lead to decision errors and compromise the system's performance. With this in mind, this paper proposes seven metrics to evaluate collaboration performance for fault detection and diagnosis in multi-agent systems (MAS)-based CPS, evaluating when the collaboration is beneficial or when the collaboration parameters need to be adjusted. The experiments focus on collaborative fault detection in temperature and humidity sensors within warehouse racks, where the proposed evaluation metrics point out the impact of collaboration on the detection task, as well as possible actions to be adopted to improve the agent's performance.**

***Index Terms*—Cyber-Physical System, Collaborative Fault Detection, Collaborative Fault Diagnosis, Multi-agent System.**

I. Introduction

Cyber-physical systems (CPS) are complex engineered systems that monitor, coordinate, control, and integrate physical devices or processes using embedded computing, sensing, and network communication. Connecting disparate systems and entities into a cohesive whole through the close coupling of cyber and physical domains, CPS offers a promising framework to meet the demands of significant scalability, resilience, adaptability, and reliability while providing sophisticated functions [1]. Furthermore, CPS plays a vital role in the collective intelligence of systems [2], which emerges from the interaction and collaboration between components that exhibit intelligence at the local level with the incorporation of new technologies (e.g., artificial intelligence (AI), Internet of Things (IoT), and cloud computing) [3]. In a distributed way, collective intelligence in CPS improves tasks such as control, planning, scheduling, sensing, monitoring, and fault tolerance, namely detection, diagnosis, and recovery, allowing flexibility, scalability, reliability, adaptability, and resilience in contemporary and future systems [4].

Specifically for fault detection and diagnosis (FDD) applications in industrial CPS, the autonomous execution of these tasks prematurely increases the success rate of fault recovery and prevents undesirable shutdowns. In this context, collective intelligence through the collaboration between different components can improve the individual capacity for detection and diagnosis by introducing the experience and particular skills of other components, enabling adaptability to different operating conditions and achieving higher detection and diagnostic accuracy by combining the strengths of different AI algorithms embedded in different components.

The principles of multi-agent systems (MAS) [5] fit collaborative FDD in the CPS context by distributing detection and diagnosis among a network of intelligent agents, each autonomous, equipped with local knowledge derived from AI for FDD. Simultaneously, they collaborate to assist other agents. Collective intelligence of FDD emerges from collaboration among agents, each one contributing with its knowledge and skills. This approach allows for a change in how fault tolerance operates, from local reactive and corrective to a more distributed, predictive, and precise manner, with successful detection and diagnostic at a premature stage increasing the success rate of fault recovery [6].

Despite the expected benefits mentioned above, engineering the MAS-based CPS exhibiting collaborative FDD is not trivial. It faces complexity in configuring the collaboration parameters, such as when to start a collaboration, which agents are selected to collaborate, and how to make decisions based on feedback from other agents. An incorrect configuration of these collaboration parameters can remain incorrect or even lead to errors in detecting and diagnosing faults, compromising the robustness and resilience of the individual component or the whole system.

This work focuses on study metrics that can evaluate collaboration efficiency in industrial CPS, particularly in the context of collaborative fault detection and diagnosis. The objective is to assess whether benefits are offered to the system through collaboration and, if not, to identify the required adjustments to improve the collaboration parameters. For this purpose, seven metrics are introduced, known as the Seven Collaboration Metrics (7CM), and a case study involving temperature and humidity sensors in warehouse racks is used to evaluate collaboration effectiveness in fault detection.

The rest of the paper is organized as follows: Section II addresses the related work for collaboration in fault detection and diagnosis and metrics for collaborative and distributed systems. Section III presents the concept of FDD based on MAS, and then Section IV introduces the metrics proposed to evaluate collaborative fault detection and diagnosis. Section V presents the case study, describes the experiments, and discusses the results obtained, and finally, Section VI presents the conclusions and points out the future work.

II. Related Work

Collaboration is crucial to achieving common goals and enhancing the CPS capabilities that components cannot accomplish individually. Collaboration represents the fundamental basis that enables intelligent collective behaviors that emerge from the interaction among a group of autonomous and cooperative agents endowed with simple or intelligent behaviors that seek to execute desired system behaviors [7]. This section briefly discusses the importance of collaboration for FDD in the CPS context and then discusses the evaluation metrics for FDD and collaborative or distributed systems.

A. Collaborative Fault Detection and Diagnosis

Fault detection and diagnosis have received significant attention in the literature as fundamental stages of fault tolerance and are used to maintain the operation of systems in the face of fault episodes. The detection objective is to determine whether there is a fault anywhere in the system. Simultaneously, diagnosis aims to provide specific information about the fault by classifying and distinguishing the nature of the fault [8]. Both the detection and diagnosis stages can significantly improve the resilience and performance of components by introducing collaboration capabilities into the FDD. Through communication, agents can share information and request data analysis, thereby improving the accuracy and effectiveness of FDD, enabling agents to leverage the collective intelligence of the system [4].

Several FDD methods, such as model-based, knowledge-based, and data-driven approaches, have been explored, as pointed out in the survey [9]. Each of these methods has its advantages and disadvantages. A method that works well under specific conditions or scenarios might not work well under another when different aspects or scenarios change [6]. Collaboration between agents with individual FDD capabilities creates an ecosystem that extracts the strengths of each method through collaboration and information exchange [10].

The engineering of collaborative fault detection and diagnosis in CPS poses several technological and scientific challenges and opens new avenues for research. For example, one of the major challenges is decentralizing the data analysis in each agent, allowing the self-capability of FDD [11], [12]. Another major challenge lies in supporting data uncertainty and inconsistency caused by temporal or spatial myopia, i.e., the lack of complete data information can result in wrong decisions [12]. The development of these issues will pave the way for collaborative FDD in distributed systems with collaborative decision support, facing data uncertainty, inconsistency, incompleteness, and AI model inaccuracy.

Although the challenges outlined serve as foundational elements for collaborative FDD, this paper concentrates on a crucial aspect of metrics to evaluate the impact of collaboration on agents. Misguided collaborative behavior not only misleads an agent, but also compromises the integrity of the entire system. Hence, metrics that pinpoint parameters for enhancing each agent's collaborative behaviors, such as the trigger to initiate collaboration, selecting suitable collaborative agents, or making decisions based on collaborative information, are imperative for understanding and predicting agent behaviors within a collaborative FDD.

B. Evaluation Metrics

FDD evaluation metrics in CPS are crucial to ensuring the robustness and reliability of the system. Studies propose temporal metrics such as the time to detect a fault, time for system performance degradation, and time to diagnose the fault, as indicated by the resilience curve in [13]. Additionally, statistical metrics are adopted to assess reliability, such as the failure rate (number of expected failures in a given period), mean time to failure (i.e., expected time until the first occurrence of a failure) and mean time between failures (i.e., average time between system failures) [14].

Additionally, it is usual to adopt metrics for each agent, highlighting the performance and effectiveness of FDD techniques deployed for specific applications. For example, precision, accuracy, recall, and F1-score are adopted to evaluate data-driven techniques to adjust selected features and the detection and diagnosis model. For instance, the random forest machine learning algorithm is adopted for fault detection and diagnosis in wind energy conversion, employing these performance metrics [15]. Moreover, a common language framework like DXF has been introduced to compare diagnostic algorithms' performance under identical conditions and compute relevant metrics [16]. These metrics help assess the effectiveness of FDD approaches in CPS, facilitating the design and development of more robust and resilient applications.

From another perspective, the academic community also dedicates its efforts to proposing evaluation metrics for distributed and collaborative systems. For instance, metrics are proposed to assess resilience, referencing a CPS design, structure, stability, and performance under cyber attacks [17]. In automated production, metrics are utilized to evaluate the flexibility and adaptability of cyber-physical production systems, as highlighted in the work [18]. To measure the dependability characteristics of a CPS, a digital dependability identity is adopted, resulting in the evaluation of CPS quality metrics [19]. However, during the research involving this study, no related works were found that evaluate the collaboration of a CPS, especially for FDD, indicating whether collaboration translates into benefits for the system or component.

III. Fault Detection and Diagnosis in MAS-based Industrial Cyber-Physical Systems

The emergence of collaborative intelligence in FDD requires an ecosystem composed of a network of cyber-physical components, where each component has local intelligence and compatible collaboration models. This collaboration and information exchange among distributed components enhances the FDD through a comprehensive understanding of condition changes and the proper detection and diagnosis. For instance, when a single mobile robot detects and diagnoses a fault in its motors, this information is shared to prevent other robot agents from taking a crowded path [10]. Similarly, after detecting problems on IoT sensors, the agent node sends to agents with higher processing capabilities for a thorough fault diagnosis [20].

MAS is a crucial facilitator technology that aids in the engineering of collaborative FDD, enabling the dissemination of intelligence capabilities among networked agents; each one is autonomous in terms of detection and diagnosis, possesses local intelligence and collaborative capabilities, sharing individual knowledge with other agents in the system, as shown in Fig. 1. In this system, each physical asset is associated with an agent, which, among the behaviors of control, order execution, and management, can detect and diagnose faults and collaborate to assist other agents in FDD tasks.

The intelligent overall behavior derives from the collaboration among individual agents, with decisions being taken in a decentralized manner, which overcomes the classic problems presented by traditional centralized and monolithic FDD architectures that are not anymore enough to address the current requirements imposed on industrial CPS [12]. The capability for fault detection and diagnosis appears on two levels. Firstly, autonomously, with self-FDD processes running locally in individual agents, which should respond fast and efficiently to disturbances, and secondly, collaboratively, with decision processes based on the interaction among these individuals through cooperation and data exchange, for example, adopting the collaboration protocol presented in Fig. 1. In this approach, the collaboration models that support the interaction among individual agents can contribute to detecting and diagnosing faults more effectively, highlighting the detection of non-observable faults at the individual level or supporting the event uncertainty when detecting and diagnosing a fault, and thus contributing to improving the accuracy of the FDD.

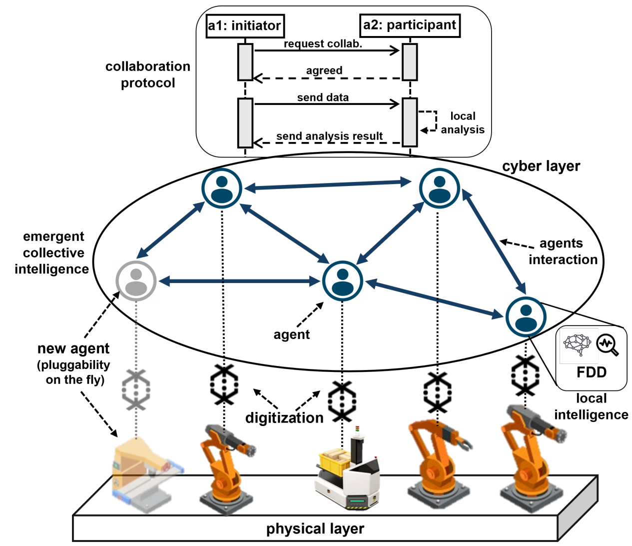

Fig. 1. Collaborative fault detection and diagnosis in cyber-physical systems implemented with MAS technology.

Consider a MAS with a set of j agents where each $agent_i$ has individual detection and diagnosis capabilities. For the fault detection stage on the input sample ϵ_i coming from digitized data related to $agent_i$, a detection method $fdt_i(\epsilon_i)$ is used. Suppose that there are two classes for detection, and $agent_i$ assigns an output $fdt_i(\epsilon_i)$ for each input sample ϵ_i according to Eq. 1:

$$fdt_i(\epsilon_i) = \begin{cases} 0, & \text{if fault free} \\ 1, & \text{if fault detected} \end{cases}, i \in [1, j] \quad (1)$$

The fault diagnosis stage is triggered upon a fault is detected, i.e., $fdt_i(\epsilon_i) = 1$. Each $agent_i$ has a fault diagnosis method tailored to handle k pre-defined fault classes. Consequently, the objective of $agent_i$ is to categorize each input sample ϵ_i with a fault into one of the k fault classes, yielding the diagnostic output $fdi_i(\epsilon_i)$ according to the Eq. 2:

$$fdi_i(\epsilon_i) = \begin{cases} 1, & \text{if fault type 1} \\ 2, & \text{if fault type 2} \\ ... & \\ k, & \text{if fault type k} \\ k+1, & \text{if fault unknow} \end{cases}, i \in [1, j] \quad (2)$$

For both stages, the functions $S(fdt_i(\epsilon_i))$ and $S(fdi_i(\epsilon_i))$ measure the degree of similarity $[0, 1]$ between the sample ϵ_i and the k classes according to Eq. 3 and Eq. 4.

$$S(fdi_i(\epsilon_i)) = \begin{cases} s_0, & \text{sim. for fault free} \\ s_1, & \text{sim. for fault detected} \end{cases}, \sum_{x=0}^{1} s_x = 1 \quad (3)$$

$$S(fdi_i(\epsilon_i)) = \begin{cases} s_1, & \text{sim. for fault type 1} \\ s_2, & \text{sim. for fault type 2} \\ \dots & \\ s_k, & \text{sim. for fault type k} \end{cases}, \sum_{x=1}^{k} s_x = 1 \quad (4)$$

The collaboration process starts when the similarity for the detection or diagnostic stage of an agent has similarity uncertainty, i.e., they do not have significant similarity values to determine a class for a given input. Fig. 2 presents the collaboration sequence, highlighting the essential collaboration parameters, namely collaboration trigger τ, agent selection σ, and decision-making ρ.

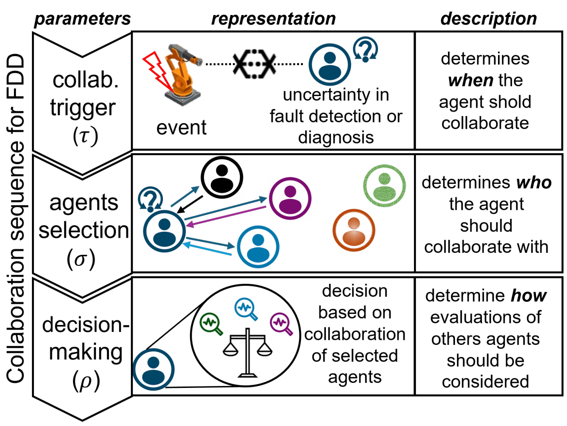

Fig. 2. Sequence for collaborative FDD and parameters description.

The collaboration trigger parameter, denoted as τ, alerts the agent when the detection or diagnosis model lacks certainty or similarity for classifying an event, prompting the need for collaboration to support decision-making. The agent selection parameter, represented by σ, plays a crucial role in selecting the most appropriate agents for collaboration, considering factors such as compatibility of detection and diagnosis models to provide analysis on uncertain data. Lastly, the decision-making parameter, denoted as ρ, determines how the agent will process the analyses from other agents' models to determine whether the event is a fault (in the case of detection) or the type of fault (in the case of diagnosis). For example, decision-making methods may include voting based on a majority or a weighted function according to the degree of similarity in analysis from each selected agent. These collaboration parameters must be instantiated according to each application scenario, where each agent can be configured independently.

IV. Proposed Metrics to Evaluate the Collaboration: The 7CM

The collaboration enables the collective intelligence to operate, allowing each agent, through optimal information exchange, to transcend its individual capacity in activities such as fault detection and diagnosis. Similarly, when established incorrectly, collaboration may not bring benefits or even have a negative impact; for example, excessive interactions can consume significant communication bandwidth and processing resources or even decrease task performance by inducing errors in detection or diagnosis. Therefore, it is essential to establish metrics to evaluate the behavior and outcomes of the collaboration among agents in an industrial CPS.

The proposed method to evaluate the collaboration, entitled Seven Collaboration Metrics ($7CM$), comprises seven metrics that allow assessing and adjusting the collaborative behavior in each agent's detection or diagnosis models of each agent. These metrics are based on collaboration outputs regarding the quantity of collaboration (qc), i.e., the number of times the agent's collaboration trigger (τ) is fired. As illustrated in Fig. 3 the metrics analyze whether the collaborations remain or modify the result initially classified by the agent detection or diagnosis local model to verify the benefit of the collaboration process for the agent.

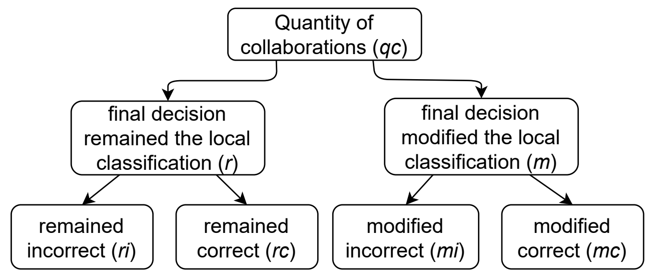

Fig. 3. Schematic tree presenting characteristics as collaboration for decision as collaboration output.

The first metric, CM_1, evaluates the agent's anxiety, checking if the collaboration triggers τ is activated excessively. The reason for the quantity of collaboration (qc) is analyzed, i.e., the inferences that activated the collaboration trigger with the quantity of inferences (qi), as presented by Eq. 5. If this ratio is high, it indicates that the agent presents excessive collaboration, requiring an adjustment to the τ parameter, or that the detection or diagnostic model needs to be updated to become more confident and not trigger the collaboration so often. On the other hand, if this ratio is low, the agent is not requesting collaboration and, therefore, contains only local behavior. $\mathcal{D}$ defines the domains of the variables.

$$CM_1 = \frac{qc}{qi} \quad : \quad \mathcal{D} = \begin{cases} \{qi \in \mathbb{N}^*\} \\ \{qc \in \mathbb{N}^* \mid 0 < qc \leq qi\} \end{cases} \quad (5)$$

The CM_2 metric represents the collective error and checks whether the agent selection and decision-making in collaboration could be more efficient. It is analyzed when the local model is incorrect, and the collaboration maintains the incorrect fault classification decision. Eq. 6 shows whether, after the collaboration, the decision remained incorrect ri; in this case, τ is performing as expected. However, the selection σ and decision-making parameter ρ are inefficient and must be updated.

$$CM_2 = \frac{ri}{qc} \quad : \quad \mathcal{D} = \begin{cases} \{ri \in \mathbb{N} \mid \ ri \leq qc\} \\ \{qc \in \mathbb{N}^* \mid 0 < qc \leq qi\} \end{cases} \quad (6)$$

Eq. 7 presents the CM_3 metrics that indicate when the collaborative behavior of the agent tends to confirm if the local classification decision is correct since it analyzes whether the decision after the collaboration remained correct rc. The high ratio may indicate an insecure agent, meaning the τ collaboration trigger must be adjusted.

$$CM_3 = \frac{rc}{qc} \quad : \quad \mathcal{D} = \begin{cases} \{rc \in \mathbb{N} \mid \ rc \leq qc\} \\ \{qc \in \mathbb{N}^* \mid 0 < qc \leq qi\} \end{cases} \tag{7}$$

The collaboration between agents can incorrectly modify the decision (mi), leading the agent that starts the collaboration into error. Eq. 8 presents the CM_4 metric and defines it as the worst collaboration scenario when this value is high. To correct this, updating the trigger τ or decision-making ρ is necessary since the model is already classifying correctly without collaboration.

$$CM_4 = \frac{mi}{qc} \quad : \quad \mathcal{D} = \begin{cases} \{mi \in \mathbb{N} \mid \ mi \leq qc\} \\ \{qc \in \mathbb{N}^* \mid 0 < qc \leq qi\} \end{cases} \tag{8}$$

The CM_5 metric presented in Eq. 9 points out the most desired behavior of the collaboration, as the decision of the collaboration modifies correctly (mc) the local classification. This metric highlights that the collaboration prevents a fault from going undetected or from false fault alerts or even incorrect diagnoses. In this sense, when this ratio is high, it indicates that τ, σ, and ρ are set efficiently. The result of the collaboration can be used to retrain and improve the detection model.

$$CM_5 = \frac{mc}{qc} \quad : \quad \mathcal{D} = \begin{cases} \{mc \in \mathbb{N} \mid \ mc \leq qc\} \\ \{qc \in \mathbb{N}^* \mid 0 < qc \leq qi\} \end{cases} \tag{9}$$

The CM_6 metric adds the undesired collaboration decision, that is, ri and mi. At the same time, the CM_7 metric adds the desired decision, that is, rc and mc. Therefore, an agent that can get a low CM_6 and a high CM_7 has good detection and diagnostic models and collaboration parameters τ, σ, and ρ well configured.

$$CM_6 = \frac{ri + mi}{qc} \ : \ \mathcal{D} = \begin{cases} \{ri + mi \in \mathbb{N} \mid \ ri + mi \leq qc\} \\ \{qc \in \mathbb{N}^* \mid 0 < qc \leq qi\} \end{cases} \tag{10}$$

$$CM_7 = \frac{rc + mc}{qc} \ : \ \mathcal{D} = \begin{cases} \{rc + mc \in \mathbb{N} \mid \ rc + mc \leq qc\} \\ \{qc \in \mathbb{N}^* \mid 0 < qc \leq qi\} \end{cases} \tag{11}$$

Table I summarizes the 7CM approach, presenting the behavior indicated by each metric, describing each one, and highlighting the parameters that have the most significant impact on their influence.

The proposed metrics support agents' continuous improvement during their collaborative operation in fault detection and diagnosis. The 7CM can assist each agent in adapting during its execution, identifying moments when more collaboration is needed or when this behavior needs to be reduced. Additionally, it can provide indications of when the FDD models must be updated.

TABLE I
SUMMARY OF THE 7CM TO EVALUATE COLLABORATIVE FAUT DETECTION AND DIAGNOSIS IN CPS.

Metrics	*Behavior represented*	*Description*	*Parameters with the most influence*
CM_1	anxiety	evaluate the collaboration rate	τ FDD model
CM_2	collective error	locally wrong and after the collab. the error remains	σ ρ
CM_3	insecurity	locally correct, and after the collab. confirm the correct	τ FDD model
CM_4	error induction	locally correct, but when collab. it is induced to error	τ σ ρ
CM_5	correction	locally wrong, and when collab. it is corrected	τ σ ρ
CM_6	collab. undesired	collab. that negatively impact the agent	all parameters
CM_7	collab. desired	collab. that positively impact the agent	all parameters

V. EXPERIMENTS AND DISCUSSION

An experimental case study was used to illustrate the applicability of the proposed evaluation metrics, the 7CM for updating the collaboration parameters during the fault detection stage, improving the performance of each agent.

A. Case Study

The proposed evaluation metrics of collaboration in CPS were applied in ARENA [21], a mock-up warehouse platform, illustrated in Fig. 4. The ARENA has three classes of assets, namely products, racks, and robots. Each asset has an associated agent corresponding to its class, with processing capacity for autonomous decision-making and collaboration capability aiming to achieve the warehouse goals and FDD. More information about the MAS-based CPS architecture deployed in the ARENA can be consulted in [22].

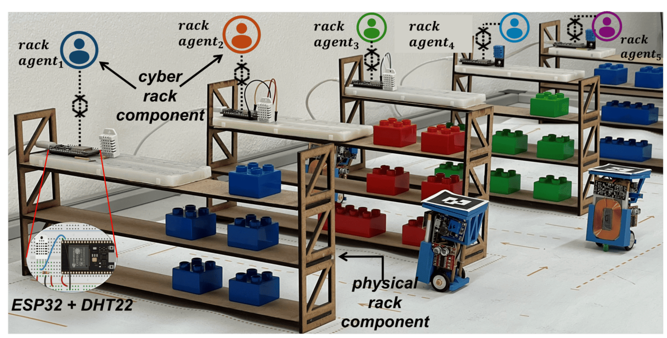

Fig. 4. ARENA highlighting the racks cyber physical components and agents.

The experiments are conducted on the five agent racks, implemented on the robot operating system (ROS) platform programmed in Python and running on dedicated Raspberry Pi 3b+ devices. Each rack is responsible for managing space and ensuring the storage of products under controlled thermal conditions, maintaining temperatures between 17°C and 20°C and air humidity between 35% and 40%. Furthermore, each agent hosts a fault detection algorithm locally based on the support vector machine (SVM) classifier model. These SVM models are trained with distinct and unique datasets, resulting in a diversity of SVM models tailored to detect fault samples in temperature and humidity data captured from DHT22 sensors installed in each rack.

The SVM algorithms are implemented using the scikit-learn Python library [23] with its standards training parameters, processing a window of 60 temperature and humidity samples as input. The output of the SVM algorithm adheres to Eq. 1 and Eq. 3: when a fault-free sample is detected, the output is "0" with the corresponding degree of similarity s_0, and if a fault is detected, the output is "1" with the corresponding degree of similarity s_1.

Fig. 5 presents the test data utilized in the experiment, comprising 1440 temperature and humidity observations collected at ten-second intervals, totaling four hours of data acquisition. Within this dataset, the initial 859 observations were deemed healthy, followed by the injection of faults into the remaining 581 samples using Gaussian noise in temperature and humidity. For consistency, this test data was injected into all agents during the three experiments, firstly to illustrate the impact of collaboration on fault detection and subsequently to highlight the effects of calibrating collaboration parameters after adopting the proposed 7CM evaluation metric.

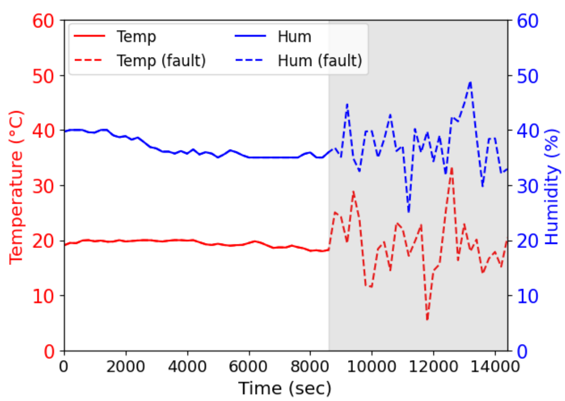

Fig. 5. Test data used in each agent for the three experiments with noise fault injected in temperature and humidity sensor.

B. Preliminary Experiments and Results

Three experiments were conducted to validate the proposed evaluation metric using the temperature and humidity test data. The first experiment examines the fault detection performance of local models embedded in each agent without collaboration. In the second experiment, agents collaborate using predefined collaboration parameters (τ, σ, ρ) set equally for all agents. The proposed evaluation metric, the 7CM, was applied in the second experiment to assess the collaborative behavior of each agent and propose individual improvements in the collaboration parameters to enhance each agent's performance in fault detection, as presented in the third experiment. The individual performance of the agents is compared in terms of precision, recall, and F1-score in detecting faults in the test data (see Fig. 5). The precision value measures the agent's ability to accurately classify fault-free samples as such without misclassifying them as fault samples. On the other hand, the recall value measures the agent's ability to detect all fault samples correctly. The F1-score is the weighted harmonic mean of precision and recall, incorporating the importance of both precision and recall. For precision, recall, and F1-score, the best value is 100%, and the worst score is 0%.

Fig. 6 displays the individual performance of the five agents in the non-collaborative experiment. Individually, reasonable performance is observed, with the F1-score of $agent_1$, $agent_3$, $agent_4$, and $agent_5$ around 90%, despite $agent_2$ having low recall values directly impacting its corresponding F1-score. The $agent_3$ successfully identified all faulty observations, with a recall of 100%, but misclassified 16% of fault-free observations as faulty, resulting in a precision of 84%. In contrast, $agent_1$, $agent_2$, $agent_4$, and $agent_5$ had precision values higher than recall. Therefore, the collaboration is expected to foster the collective intelligence, enhancing both recall and precision, surpassing the individual capacity of each agent.

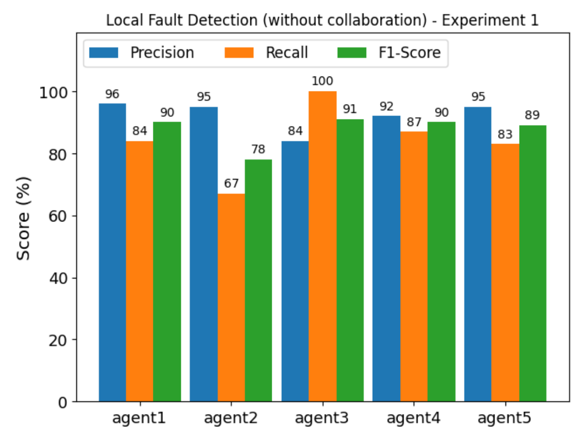

Fig. 6. Individual agents performance for fault detection based on test data.

In the second experiment, the collaboration among the agents for fault detection was introduced. For this purpose, exploratory and equal collaboration parameters were assigned to all agents, namely the collaboration trigger $\tau = 0.7$, meaning that a given agent will initiate collaboration if the detection result has a similarity (S) below 0.7. The selection parameter σ selects all agents in the case study, as there are only five agents, and all are compatible. Lastly, the decision-making parameter ρ is configured to classify the sample following the majority of analyses of all agents, i.e., the majority voting rule. Figure 7 presents the performance of each agent in classifying the test data, showing a noticeable improvement in some agent's performance. However, despite the collaboration, $agent_1$ and $agent_3$ maintained their fault detection performances almost stagnant. Nevertheless, they were fundamental in the improvements achieved by $agent_2$, $agent_4$, and $agent_5$, with their F1-score improving by 2%, 3%, and 4%, respectively. It is worth highlighting the considerable improvement in recall for $agent_4$ and $agent_5$, both by 6%.

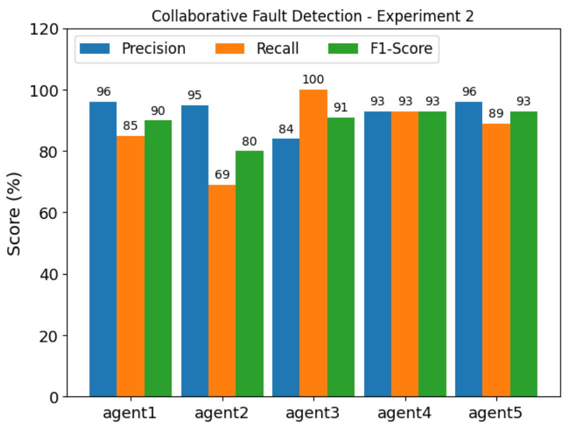

Fig. 7. Performance of agents with collaborative fault detection, with equal parameters for all agents: $\tau = 0.7$, $\sigma = all$ and $\rho = majority\ voting$.

Despite the observed improvement with the introduction of collaboration among the agents, this behavior needs to be assessed to identify aspects that each agent needs to improve regarding the collaboration parameters. Table II presents the results of applying the proposed 7CM, highlighting the collaboration outputs. CM_1 indicates that few collaboration events occurred, for example, $agent_3$ requested collaboration in approximately 1% of its observations to detect faults, while $agent_2$ requested collaboration around 12% of the test data sample. Therefore, it is necessary to properly increase the collaboration trigger so that agents can request more collaboration to enhance performance in fault detection.

TABLE II
COLLABORATION METRICS (7CM) FOR SECOND EXPERIMENT.

		$agent_1$	$agent_2$	$agent_3$	$agent_4$	$agent_5$
collab. output	qi	1440	1440	1440	1440	1440
	qc	91	179	12	110	78
	ri	26	3	0	2	0
	rc	26	150	7	22	19
	mi	17	6	0	2	1
	mc	22	20	5	84	58
collab. metrics[‰]	CM_1	0.06	0.12	0.01	0.08	0.05
	CM_2	0.29	0.02	0.00	0.02	0.00
	CM_3	0.29	0.84	0.58	0.20	0.24
	CM_4	0.19	0.03	0.00	0.02	0.01
	CM_5	0.24	0.11	0.42	0.76	0.74
	CM_6	0.47	0.05	0.00	0.04	0.01
	CM_7	0.53	0.95	1.00	0.96	0.99

Considering the Table II, through individual analysis, $agent_1$ requested collaboration in 6% of its observations (CM_1) but failed to leverage its performance with collaborations, as indicated by CM_6, which represents the undesirable collaboration index, being very close to the desirable collaboration index CM_7. This high CM_6 value is attributed to the collective error CM_2 (individually wrong, and when collaborating, remains wrong) and the collaboration inducing to error CM_4 (i.e., individually correct, but when collaborating, is induced to error wrongly). The most desirable behavior when collaborating is the correction of a wrong detection (CM_5), which accounts for 24% of collaborations performed for $agent_1$. To improve the collaborative initiative of $agent_1$, its collaboration trigger will be raised to $\tau = 0.85$. The selection parameter σ remains the same, selecting all agents. Lastly, the decision-making parameter ρ also remains unchanged, considering the majority of votes.

The $agent_2$ stood out as the most proactive in initiating collaborations, as shown by CM_1. However, 84% of these collaborations merely confirmed that its local decision was correct, as indicated by CM_3. Although not a negative indicator (as these data could be used to retrain its model), it does not directly impact the classification performance, remaining similar to that obtained in experiment 1. With this in mind, the collaboration parameters for $agent_2$ were adjusted, increasing the collaboration trigger value to $\tau = 0.9$. The selection parameter remains unchanged, but the decision-making parameter ρ is altered to adopt the decision of the voter with the highest similarity.

The $agent_3$ and $agent_5$ were the least proactive in initiating the collaboration, with CM_1 at 0.01 and 0.05, respectively. Due to this fact, the obtained corrections (CM_5) did not apparently affect their performance or could have been even better, given that $agent_5$ showed significant improvement when collaborating. Additionally, these two agents' desired collaborations CM_7 are the highest. Therefore, the only change for these agents is to considerably increase the collaboration trigger to $\tau = 0.9$ while keeping the other parameters unchanged.

The $agent_4$ exhibited good behavior, being the second agent to initiate collaboration most frequently, as indicated by CM_1, and showed a high degree of correction, with CM_5 at 76%. This suggests that the collaboration parameter of decision-making ρ is suitable for it. In order to further encourage its collaboration initiative, only the collaboration trigger will be adjusted to $\tau = 0.85$.

Figure 8 illustrates the performance of each agent with the changes in parameters τ, σ, and ρ obtained after analysis and reflection on Table II. The $agent_1$, $agent_4$, and $agent_5$ showed good performance, significantly improving compared to the previous experiments. For example, the recall of $agent_1$ improved by 14% compared to the first experiment and 13% compared to the second. The $agent_3$ showed a slight improvement of 1% in precision compared to previous experiments. The $agent_2$ improved the recall by 6% and the F1-score by 4% compared to the previous experiment but presented a precision decrease of 1%, resulting from incorrect modifications (CM_4) due to the change in the decision-making ρ. Nevertheless, this adopted decision-making parameter positively impacts the recall value and, consequently, on the F1-score, improving by 8% and 4% compared to the first and second experiments, respectively.

The case study presented in this section focused on collaboration in the fault detection stage. However, for collaborative fault diagnosis, the 7CM is also helpful as a metric for evaluating collaboration, simply by changing the classification between two classes (with a fault or without fault for detection)

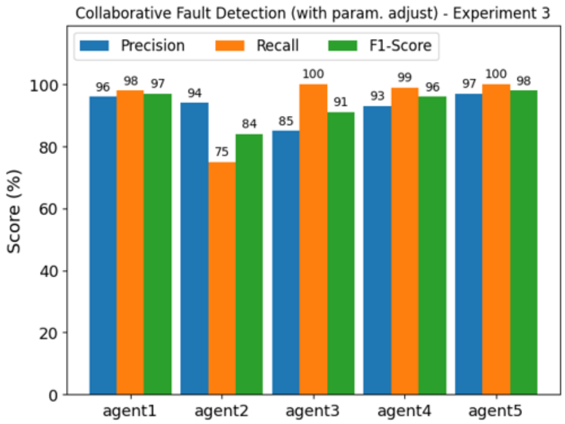

Fig. 8. Performance of agents with collaborative fault detection with collaborations parameters calibrated by 7CM evaluation metrics.

to a scenario with a list of faults of different types, as presented in the rationale of this work.

VI. Conclusions and Future Work

Collaboration in CPS results in an ecosystem endowed with collective intelligence abilities. This phenomenon is evident in operations such as collaborative fault detection and diagnosis, where agents transcend their individual capabilities through collaboration, enhancing their fault detection and diagnosis performance. However, collaboration can have the opposite effect when poorly established, leading to errors and unnecessarily consuming communication and processing resources.

With this in mind, this study introduces a novel approach to assess the collaboration among agents for fault detection and diagnosis. The Seven Collaboration Metrics, named 7CM, are designed to evaluate collaborative behaviors such as anxiety, uncertainty, collective error, error induction, and corrections. The experiments conducted in this work compare the performance of five agents in three scenarios (without collaboration, collaborating, and collaboration corrected according to the proposed metrics) in detecting precision faults in temperature and humidity sensors installed in a warehouse setup. The preliminary results are promising, indicating that collaboration enhances the individual performance of each agent, and fine-tuning the collaboration parameters based on the analysis of the results of applying the proposed metrics leads to significant improvements.

In future work, the aim is to explore developing digital twin models to enhance the collaboration in fault detection and diagnosis based on the proposed evaluation metrics, providing decision support to update the collaboration parameters to improve the agents' performance. Additionally, effort will be devoted to exploring experiments in fault diagnosis.

Acknowledgment

This work was supported by national funds through FCT/MCTES (PIDDAC): CeDRI, UIDB/05757/2020 (DOI: 10.54499/UIDB/05757/2020) and UIDP/05757/2020 (DOI: 10.54499/UIDP/05757/2020); and SusTEC, LA/P/0007/2020 (DOI: 10.54499/LA/P/0007/2020). Luis Piardi thank FCT for the PhD Grant UI/BD/151286/2021 (DOI: 10.54499/UI/BD/151286/2021).

References

[1] K. Zhang, *et al.*, "Advancements in industrial cyber-physical systems: an overview and perspectives," *IEEE Trans. on Ind. Informatics*, 2022.
[2] P. Leitão, J. Queiroz, and L. Sakurada, "Collective intelligence in self-organized industrial cyber-physical systems," *Electronics*, vol. 11, no. 19, p. 3213, 2022.
[3] Y. Lu, "Industry 4.0: A survey on technologies, applications and open research issues," *Journal of Ind. information integration*, vol. 6, pp. 1–10, 2017.
[4] L. Piardi, P. Leitão, and A. S. de Oliveira, "Fault-tolerance in cyber-physical systems: literature review and challenges," in *IEEE 18th Int. Conf. on Industrial Informatics (INDIN)*, 2020, pp. 29–34.
[5] M. Wooldridge, *An introduction to multiagent systems*. John wiley & sons, 2009.
[6] Y. S. Ng and R. Srinivasan, "Multi-agent based collaborative fault detection and identification in chemical processes," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 6, pp. 934–949, 2010.
[7] T. W. Malone and M. S. Bernstein, *Handbook of collective intelligence*. MIT press, 2022.
[8] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Trans. on dependable and secure computing*, vol. 1, no. 1, pp. 11–33, 2004.
[9] A. Abid, M. T. Khan, and J. Iqbal, "A review on fault detection and diagnosis techniques: basics and beyond," *Artificial Intelligence Review*, vol. 54, no. 5, pp. 3639–3664, 2021.
[10] L. Piardi, P. Costa, A. Oliveira, and P. Leitão, "Collaborative Fault Detection and Diagnosis Architecture for Industrial Cyber-Physical Systems," in *IEEE Int. Conf. on Ind. Technology (ICIT)*, 2022, pp. 1–6.
[11] A. Dorri, S. S. Kanhere, and R. Jurdak, "Multi-agent systems: A survey," *IEEE Access*, vol. 6, pp. 28 573–28 593, 2018.
[12] L. Piardi, P. Leitão, P. Costa, and A. Schneider de Oliveira, "Collaboration and self-organization to enable self-healing in industrial cyber-physical systems," in *Int. Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing*. Springer, 2023, pp. 532–543.
[13] S. Hosseini, K. Barker, and J. E. Ramirez-Marquez, "A review of definitions and measures of system resilience," *Reliability Engineering & System Safety*, vol. 145, pp. 47–61, 2016.
[14] D. J. Smith, *Reliability, maintainability and risk: practical methods for engineers*. Butterworth-Heinemann, 2021.
[15] R. Fezai, *et al.*, "Effective random forest-based fault detection and diagnosis for wind energy conversion systems," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6914–6921, 2020.
[16] A. Feldman, T. Kurtoglu, S. Narasimhan, S. Poll, and D. Garcia, "Empirical evaluation of diagnostic algorithm performance using a generic framework," *Int. Journal of Prognostics and Health Management Volume 1*, p. 24, 2013.
[17] M. Segovia, J. Rubio-Hernan, A. R. Cavalli, and J. Garcia-Alfaro, "Cyber-resilience evaluation of cyber-physical systems," in *Int. Symp. on Network Comp. and Applications (NCA)*. IEEE, 2020, pp. 1–8.
[18] B. Vogel-Heuser and J. Prieler, "Evaluation of selected metrics for flexibility of cyber physical production systems," in *IEEE Conf. on Automation Sci. and Engineering (CASE)*. IEEE, 2017, pp. 701–708.
[19] G. Regan, *et al.*, "Quality improvement mechanism for cyber physical systems—an evaluation," *Journal of Software: Evolution and Process*, vol. 32, no. 11, p. 2295, 2020.
[20] S. U. Jan, Y. D. Lee, and I. S. Koo, "A distributed sensor-fault detection and diagnosis framework using machine learning," *Information Sciences*, vol. 547, pp. 777–796, 2021.
[21] L. Piardi, V. C. Kalempa, M. Limeira, A. S. de Oliveira, and P. Leitão, "Arena—augmented reality to enhanced experimentation in smart warehouses," *Sensors*, vol. 19, no. 19, p. 4308, 2019.
[22] L. Piardi, P. Costa, A. Oliveira, and P. Leitão, "MAS-based Distributed Cyber-physical System in Smart Warehouse," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 6376–6381, 2023.
[23] F. Pedregosa, *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.