Robotics and Autonomous Systems 205 (2026) 105680

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Enhancing Robotic Mission Analysis via Process Mining and Visual Analytics

Sara Pettinari [a,*], Barbara Re [b], Lorenzo Rossi [b], Francesco Tiezzi [c]

[a] Gran Sasso Science Institute, L'Aquila, Italy
[b] School of Science and Technology, University of Camerino, Camerino, Italy
[c] Dipartimento di Statistica, Informatica, Applicazioni, University of Florence, Florence, Italy

ARTICLE INFO

Keywords:
Robotic systems
Process mining
Visual analytics
Multi-perspective event logs
Mission analysis

ABSTRACT

Robotic systems are increasingly used to automate complex tasks in dynamic environments, making mission analysis essential to improve reliability and performance. Among the existing techniques for analyzing robots' behavior, process mining is particularly suitable since it enables the extraction of high-level behavioral patterns from event logs, providing actionable insights into the overall system behavior. However, the large volume and the low-level of data generated by robotic systems pose challenges for the effective use of process mining, which mainly relies on high-level representations of system activities. To address this, robotic mission execution should be recorded at a higher level of abstraction to enable its discovery and reasoning in terms of control-flow. At the same time, low-level contextual data, as spatial occupancy, energy consumption, and interactions, must be integrated to provide a comprehensive understanding of the mission. This integration can be achieved by combining process mining with visual analytics, thus enhancing the automated mission discovery with interactive visualizations of contextual data and enabling mission analysis from multiple perspectives. This paper presents a methodology for preparing and analyzing robotic mission data via process mining and visual analytics. The methodology guides robotic experts in preparing event logs compliant with process mining standards and supports mission analysis through the RoboTrace tool, which enables process discovery and contextual visualizations. The methodology is assessed through a multi-robot case study and an evaluation with domain experts to show its feasibility, interpretability, usability, and utility.

1. Introduction

Robotic systems are becoming increasingly versatile, capable of performing different tasks and interacting with other systems, humans, and each other [1]. This evolution has broadened the scope and complexity of robotic missions, making it essential to analyze and interpret performed behavior in a way that is both explainable and insightful for developers, analysts, and end-users. In this regard, various methodologies can help to spot drawbacks in the robotic system by analyzing its behavior, i.e., when and how tasks have been done, to gather information useful for monitoring and optimizing it. Common approaches in the literature range from formal verification [2], which is often infeasible due to the complexity of the system [3], to quantitative analysis [4], which requires constant system observation. Other approaches rely on manually inspecting messaging logs across multiple terminal windows and correlating logged information, which highlight the limitations of current tools in providing a cohesive and efficient way for analyzing the overall robotic mission [5]. In this context, *process mining* [6] has emerged as a promising approach to address this need by discovering and analyzing process models that represent the executed behavior in an intuitive and process-driven way [7,8]. These process models offer a high-level abstraction of robotic operations, allowing analysts to understand and reason about behavioral patterns and decisions taken during mission execution. However, process mining requires event logs that capture system events at a high-level of abstraction, i.e., indicating the performed activity. In contrast, robotic systems typically generate low-level data, such as sensor readings or actuator commands. This mismatch limits the direct applicability of process mining in the robotics domain. Therefore, to enable process mining-driven analysis, robots should be equipped with the capability to record high-level activity information, explicitly describing the actions performed, which would facilitate the generation of event logs suitable for process mining.

At the same time, robots operate in highly dynamic environments, where their behavior is shaped by contextual factors such as spatial occupancy, energy consumption, and interactions. During mission execution, this results in the generation of vast amounts of low-level data, each capturing a distinct contextual perspective. These perspectives are essential for a comprehensive understanding of the robotic mission. Still, they are challenging to integrate with process mining techniques,

* Corresponding author.
E-mail addresses: sara.pettinari@gssi.it (S. Pettinari), barbara.re@unicam.it (B. Re), lorenzo.rossi@unicam.it (L. Rossi), francesco.tiezzi@unifi.it (F. Tiezzi).

https://doi.org/10.1016/j.robot.2026.105680
Received 23 June 2025; Received in revised form 27 May 2026; Accepted 28 July 2026
Available online 7 August 2026
0921-8890/© 2026 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

which struggle to handle the unstructured nature of low-level data [9]. In this respect, there is growing interest in process mining techniques complemented by visual representations that can explain additional aspects related to the underlying process model [10]. Specifically, *visual analytics* combines automated analysis with interactive visualizations, enabling a better understanding of large and complex datasets by leveraging the human ability to identify patterns in unstructured data. By integrating automated process mining techniques with interactive visualizations, it becomes possible to extract deeper insights from event logs [11,12]. In addition, in robotics, visual analytics approaches are well-established and widely used to provide robotic experts with intuitive and interactive visualizations. These visualizations facilitate the exploration of large datasets, representing contextual information such as spatial movements and resource usage (e.g., energy consumption).

Although visual analytics has the potential to analyze low-level robotic data, and process mining provides high-level mission abstraction, these approaches have rarely been exploited together to perform robotic mission analysis. Indeed, current practices for logging and analyzing robotic executions are typically ad hoc and rely mainly on low-level data from sensors, actuators, and interaction messages. Such data are useful for inspecting specific components, but they are usually not explicitly connected to the mission activities being performed. As a result, they do not directly support the reconstruction and analysis of robotic missions at a higher level of abstraction, where developers need to reason about executed activities, deviations, resource consumption, or coordination issues. To bridge this gap, we propose a novel methodology explicitly designed to integrate these two analytical approaches, allowing robotic experts to analyze and reason about executed missions in a comprehensive way. By combining high-level process models with the visualization of lower-level aspects, our solution enhances the analysis of robotic system behavior across both single- and multi-robot scenarios. In this perspective, the goal of mission analysis is not limited to reconstructing what happened during execution. Rather, it is to provide developers with evidence to understand why certain behaviors occurred, identify anomalies, bottlenecks, resource-related issues, or coordination failures, and use these insights to guide iterative improvements to the robotic mission design and implementation. In [8], we proposed a methodology to guide robotic experts in preparing event logs tailored for process mining applications, focusing on discovering control-flow and spatial perspectives. Building on this work, we extend the methodology by focusing on the phases that make process mining usable for robotic mission analysis. Specifically, (*i*) we discuss how mission executions can be represented through activity-centric event logs enriched with contextual information; (*ii*) we analyze key perspectives that influence robotic mission execution, namely performance, spatial occupancy, energy consumption, and communication; (*iii*) we show how discovered process models can be complemented with coordinated visual perspectives to support the interpretation of mission-level behavior; and (*iv*) we introduce RoboTrace, a supporting tool that automates process discovery and enhances it with visual representations of low-level data, enabling contextual exploration of robotic executions. Finally, (*v*) we assess the methodology in a smart agriculture multi-robot scenario and through an expert-based evaluation, focusing on feasibility, interpretability, usability, workload, and perceived utility.

The rest of the paper is organized as follows. Section 2 presents the core concepts of robotic mission analysis, process mining, and visual analytics. Section 3 outlines the motivation behind the proposed methodology, which is presented in Section 4. Section 5 describes the implementation supporting the proposed methodology, while Section 6 applies and evaluates the methodology in a multi-robot scenario. Section 7 reviews the state-of-the-art by presenting studies that apply process mining techniques to robotic system analysis, explore visual analytics approaches for robotic system analysis, and examine the integration of process mining and visual analytics across various domains. Section 8 discusses assumptions and limitations of the proposed methodology, while Section 9 accounts for the threats to validity. Finally, Section 10 presents the final remarks.

2. Context

This section first introduces the idea behind performing robotic mission analysis. After that, it presents the core concepts of process mining and visual analytics disciplines.

2.1. Robotic mission analysis

Robotic mission analysis focuses on examining the behaviors exhibited by robots while performing tasks, offering insights that are crucial for both runtime monitoring and postmortem evaluation [13]. This field has gained increasing attention in recent years due to the growing complexity of autonomous robots and the necessity to ensure accountability, transparency, and explainability in their operations [14,15].

Enabling mission analysis requires mechanisms for logging and inspection. A logging mechanism collects mission-related information, such as occurred events, utilized software and hardware components, and executed interactions. Differently, an inspection mechanism answers analytical questions, such as when and where an event occurred, which hardware or software components triggered it, why it happened, how the resulting behavior was composed, and who or what was affected [16]. Furthermore, such mechanisms facilitate the identification of bottlenecks and critical decision points, supporting performance optimization in terms of both time and resource utilization [17]. By reconstructing task sequences and environmental contexts, mission analysis enables a comprehensive assessment of system behavior from initial conditions to final outcomes.

The need for robot mission analysis becomes even more evident in multi-robot systems, where individual and cooperative tasks must be analyzed. The distributed nature of multiple robots introduces significant challenges, requiring the ability to track the behavior of single robots as well as the overall mission. In fact, changes in the behavior of a single robot can affect the whole system, impacting performances, other robots' activities, and the successful completion of the mission [5].

2.2. Process mining

Process mining is a well-established discipline for analyzing systems behavior, consisting of many techniques for discovering, monitoring, and improving the actual processes. Process mining relies on *event logs*, generated by the systems under investigation [6]. Event logs record data coming from systems' executions in an activity-centric manner, reporting at least two *attributes*, i.e., timestamp and activity name. Further attributes can be included in the events to provide context data that provides information about other perspectives. Events referring to the same system execution, i.e., *case*, are grouped into *traces* of events ordered by time. By analyzing such event logs, usually in the eXtensible Event Stream (XES) format, process mining reconstructs the process model describing the system's behavior, spots bottlenecks and unwanted deviations, and gets insights on the execution performances such as duration and costs [6]. Indeed, event logs can be firstly used to conduct three basic types of process mining analysis: process discovery, conformance checking, and process enhancement.

Process discovery techniques take an event log and create a process model that adequately describes the underlying process. Nevertheless, the correctness of the result depends both on the log completeness and on the used discovery algorithm. Usually, process discovery techniques resort to many notations to describe processes, ranging from Directly-Follows Graphs (DFGs) to Business Process Model and Notation (BPMN) and Petri nets [18]. Within this work, we adopt DFGs as the process model representation. Specifically, as exemplified in Fig. 1, a DFG is a directed graph where nodes represent activities and directed arcs represent the directly-follows relation between them. Two special nodes denote the start and the end of the process. An arc from activity *a* to

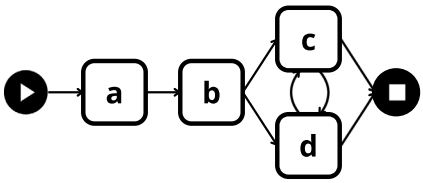


Fig. 1. Example of a DFG.

activity b indicates that b directly follows a in at least one execution trace. Nodes and arcs can be annotated with frequency or performance information. The construction of a DFG [19] is driven by the information recorded in the event log. First, the set of distinct activity labels observed in the log is extracted to define the nodes of the graph. Then, for each execution trace, consecutive pairs of activities are identified to determine directly-follows relations. These relations are aggregated across all traces to create the directed arcs of the graph, optionally enriched with frequency counts or timing statistics.

Conformance checking techniques relate events in the event log to activities in the process model and compares both. The goal is to enable the analysis of the quality of a process model discovered from event data, the identification of potential deviations, and the projection of real traces onto process models [20].

Finally, *process enhancement* aims to change or extend an existing process model. Specifically, process extension techniques aim to incorporate different perspectives to achieve a high-precision model. However, process improvement techniques are a supervised way to modify an existing model, by producing a model that properly reflects reality [21].

2.3. Visual analytics

Visual analytics is the science of analytical reasoning supported by interactive visual interfaces, aiming to maximize human capacity to perceive, understand, and reason about complex situations effectively [22]. As data grows increasingly massive, dynamic, and complex, traditional analytical methods struggle to provide clear, explainable, and actionable insights. Visual analytics addresses this challenge by integrating computational analysis with human cognition, enabling the extraction of meaningful insights from complex datasets through the use of visualizations as a bridge between human intelligence and computational power [23].

The scalability of visualizations is a key consideration, particularly when dealing with high-dimensional data. Techniques such as interactive filtering and hierarchical decomposition help mitigate information overload and improve usability [23]. Effective visual representations support the cognitive processes of pattern recognition, anomaly detection, and trend identification, ultimately improving decision-making.

The term visual analytics is also widely applied in domains such as business intelligence and operational decision support, where dashboards and interactive reports facilitate exploratory data analysis [24]. These tools empower users to engage with data through dynamic visualizations, enabling them to explore different perspectives, compare patterns, and generate insights, thus enhancing data-driven decision-making.

3. Motivations

This section first illustrates the running scenario used throughout the paper, then it presents the motivations behind this work by highlighting the challenges associated with data preparation and analysis when applying process mining techniques to robotic mission analysis.

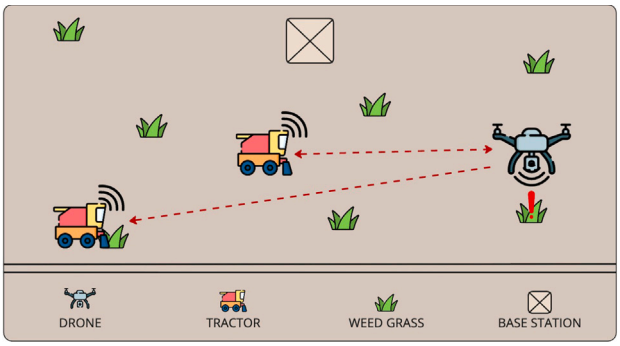


Fig. 2. Running scenario overview.

3.1. Running scenario

The proposed scenario (showcased in Fig. 2) consists of one drone and two tractors that cooperate to identify and remove weed grass in farmland, thus increasing productivity. All the robots are equipped with a controller, a battery, and several sensors and actuators. At the system start-up, the drone takes off and starts the exploration of the field. During the flight, it can recognize weed grass areas, and when found, it sends the weed position to the tractors. This enacts the tractors, which send back to the drone their current position. The drone can hence elect the closest tractor and notify it. At this point, the selected tractor starts moving towards the field to reach the weed grass area and cut it. The drone execution can be interrupted if the drone battery runs out or if the whole field has been explored and cleaned. At the same time, the tractors' execution can be interrupted by a low battery.

We built the smart agriculture scenario upon the ROS framework, and we deployed the robotic system on the Gazebo simulator[1]. Additionally, we leveraged the *ros2bag* library to record data published on topics during the system execution.

3.2. On the need to prepare robotic data for process mining

The idea of using process mining to extract and analyze the behavior of robotic systems goes hand in hand with the existence of several works exploiting process-driven approaches for the specification of robots' missions [25–27]. Indeed, process models are intuitive enough to be understood by humans and formal enough to be deployed in real robots. Nevertheless, to use process mining in this context, we need to map the event data generated by a robotic system with the concept of the event log. During its execution, a robot produces event data in a *peripheral-centric* way, keeping track of data exchanged by sensors and actuators [28]. Specifically, a robotic peripheral-centric event is typically characterized by the *time* in which the event is produced, by the name of the *peripheral* that triggered it, and by other attributes storing additional data. For instance, a robotic event triggered by a laser sensor may store the distance data read as an attribute. These event data represent that the robot has accessed a peripheral functionality, whereas process mining techniques need event logs with an *activity-centric* structure, meaning that each event must represent a clear robotic activity, i.e., a high-level activity that should be executed to fulfill an objective in the process model, like a movement action to reach a destination.[2]

[1] https://gazebosim.org/home

[2] Notably, we adopt the term *activity* to align with process mining terminology. In the robotic domain, this corresponds to the concept of a *task*, i.e., a repetitive and simple coordinated robotic behavior that is realized as a combination of skills [29,30].

Table 1
Event logs of the same robotic execution with different granularity.

Time	Peripheral	Other Data
09:33:20.147	rotor	...
09:33:20.147	camera	...
09:33:20.149	rotor	...
09:33:21.239	gps	...
09:33:30.941	camera	...
09:33:32.410	gps	...
09:33:32.495	laser	...
09:33:32.499	laser	...
09:33:32:655	camera	...
09:33:33.003	rotor	...
09:33:33.110	laser	...
09:33:33.118	laser	...
09:33:33.120	laser	...
09:33:33:240	camera	...
09:33:33.921	laser	...

(a) Peripheral-centric

Time	Activity	Lifecycle
09:33:20.147	Go To	start
09:33:20.147	Detect Weed	start
09:33:33.240	Detect Weed	complete
09:50:33.955	Go To	complete

(b) Activity-centric

To better clarify the difference between peripheral- and activity-centric event logs, we take as an example the drone from the running scenario equipped with rotors, GPS, camera, and laser sensors, that has to move around (activity *Go To*) while looking for a weed grass (activity *Detect Weed*). Table 1 shows an excerpt of a peripheral-centric event log during drone exploration execution, whereas Table 1 reports the same event log abstracted to an activity-centric point of view. The peripheral-centric log describes the working flow of the robot peripherals, e.g., the first two rows of Table 1 indicate that the robot has used a rotor and the camera. Instead, the activity-centric log provides a high-level view of the robotic execution, e.g., the first two rows of Table 1 show that the robot has concurrently started the Go To and Detect Weed activities. To graphically visualize a correlation between the peripheral- and the activity-centric events and properly present the robotic granularity problem, we colored the events belonging to the Go To activity in purple, while the events correlated to the Detect Weed activity are in yellow. In doing so, the obtained mixed occurrence of the colors in Table 1 shows that the peripheral events are produced in a continuous and unordered way, thus preventing the identification of a pattern for recognizing the related high-level activities. Moreover, the peripheral event log indicates that both activities depend on the laser sensor, which creates a problem in identifying whether an activity is finished.

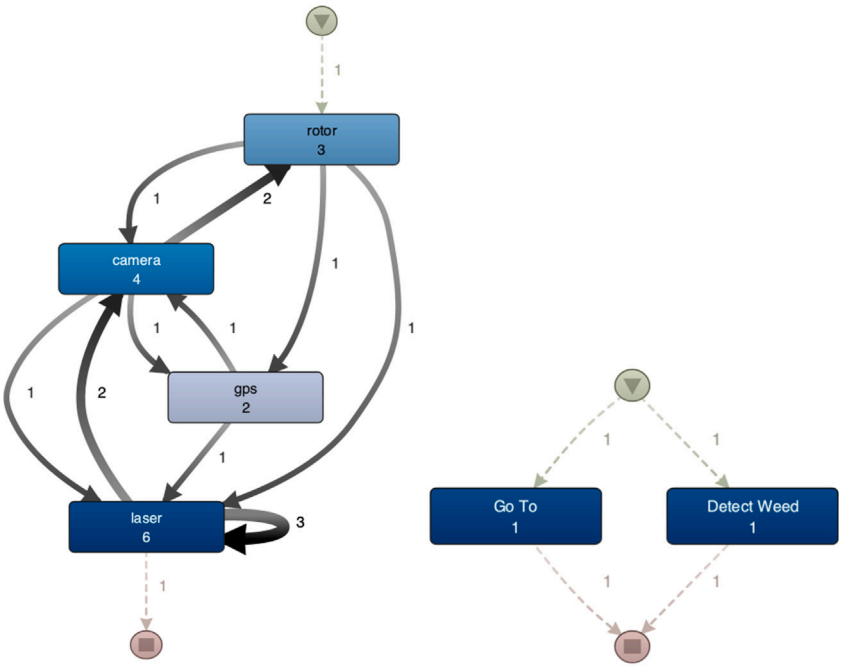
Additionally, to better illustrate the need for activity-centric data, we leverage the example logs in Table 1, which are associated with a single case, to discover the corresponding process models, in the form of DFGs. Specifically, Fig. 3(a) depicts the DFG generated from the event log in Table 1. The resulting process is unstructured, commonly referred to as a spaghetti process, where activities and their dependencies are challenging to discern. In the obtained process, each node represents the name of a peripheral used by the robot, reflecting its internal behavior, i.e., the operational flow of the peripherals. This representation shows the interactions between components and may be useful for stakeholders interested in analyzing low-level system behavior, such as sensor activation patterns or component usage. For example, in Fig. 3(a), the frequent directly-follows relations between the camera, laser, and gps nodes highlight recurring sensing loops, while the self-loop on the laser node indicates repeated activations of the same sensor. However, since the log records only peripheral-centric events, the resulting model does not directly express the structure of the robot's mission. Furthermore, the example peripheral log contains only a subset of the data generated during the execution of the robot's activities. If a complete mission execution were considered, the number of nodes and their connections would increase exponentially, further complicating the process analysis. In contrast, Fig. 3(b) illustrates the DFG derived from the event log in Table 1. This process model provides an intuitive and process-driven representation of the robot's activities, clearly depicting the parallel execution of the Go To and Detect Weed activities. For the purpose of high-level mission analysis, such as understanding how activities are executed over time, event logs capturing activity-level information enable the discovery of an intuitive and analyzable process model.

Therefore, to properly apply process mining techniques, robotic systems' mission data should be prepared in the form of an activity-centric event log. More specifically, to perform a comprehensive analysis, a robotic event log should contain a collection of robotic executions, possibly initiated in different settings, where a case comprises sequences of high-level robotic activities recorded during a complete run.

3.3. On the need to go beyond control-flow

Robotic mission analysis using process mining techniques relies on process models that represent control flow, outlining the sequence of activities and decision points within a mission [7,31,32]. While this representation provides an intuitive and process-driven way to understand performed activities, it is insufficient to capture the complexity of real-world robotic operations fully.

From the literature, four interrelated main perspectives emerge in robotic mission analysis. Specifically, activity or mission completion time, energy consumption, and spatial-related aspects are primarily examined to find an optimal tradeoff among these three [33,34], such as optimizing the time a robot takes to reach the target location and the energy consumption for the activity [35]. In this context, some approaches explore multi-robot cooperation as a strategy to improve performance [36,37], introducing the need to analyze a fourth perspective concerning communication. More specifically, the ***time*** perspective offers insights into the performance of individual activities and the overall mission. This analysis can be conducted using process mining techniques, leveraging time-related data from the event log. Indeed, a performance DFG can represent both the duration of each activity and the transition times between activities, enabling analysts to identify bottlenecks, delays, or inefficiencies in the execution flow. The ***spatial*** perspective analysis is required since robots operate in dynamic, physical environments where spatial constraints and navigation challenges directly influence mission execution. However, spatial data is continuous and unstructured [8], which makes it incompatible with standard process model representations. This requires a bridge to connect behavioral analysis with the corresponding spatial context, ensuring that spatial dynamics are not overlooked in mission evaluations. The ***energy*** consumption perspective is also critical in robotic missions [36,37]. Traditional control-flow models do not account for battery depletion, which can trigger mission deviations or adaptive behavior. By integrating energy consumption analysis, it becomes possible to evaluate battery-driven decisions and assess how efficiently activities are performed in terms of energy use, supporting more energy-aware mission planning. Finally, the ***communication*** perspective is essential in missions involving multiple robots or interactions with external systems. Communication issues can lead to failures (e.g., in a search and rescue mission, if one robot finds a survivor but loses communication, the survivor might not be rescued in time), or mission deviations (e.g., one robot substituting for another before resuming its original mission). While some process notations support modeling communication, few studies focus on discovering and analyzing process models



(a) DFG from data in Table 1a (b) DFG from data in Table 1b

Fig. 3. Discovery comparison.

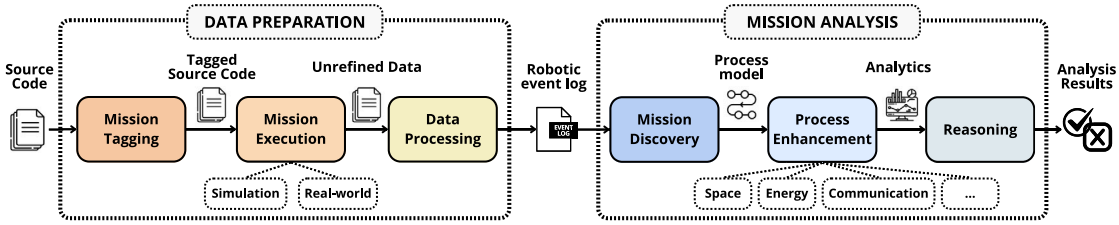


Fig. 4. Proposed methodology.

suitable for distributed systems [38]. Integrating this perspective helps capture the impact of communication on mission performance and robot coordination.

Therefore, a mission is not only defined by its executed activities but is also influenced by multiple perspectives shaped by the dynamic environment in which robots operate. To achieve a comprehensive understanding, it is necessary to go beyond control-flow and integrate additional contextual perspectives that impact mission outcomes. Integrating these perspectives allows robotic mission analysis to enrich a process-driven view with a more dynamic, context-aware one. This would better support decision-making by linking each mission and activity to the corresponding contextual perspective, offering deeper insights into performance, behaviors, and potential improvements.

4. Methodology

This section presents the methodology we designed to support a robotic expert (e.g., a robotic system developer or analyst) in the automatic extraction and processing of a robotic event log from the execution of a robotic system and its analysis through process mining and visual analytics.

The proposed methodology, depicted in Fig. 4, consists of a *data preparation* phase, composed of mission tagging, mission execution, and data processing, involving robotic event log preparation, and an *mission analysis* phase, composed of mission discovery, process enhancement, and reasoning, enabling robotic mission analysis via process mining and visual analytics. Notably, the data preparation phase is built upon the tag-based multi-perspective methodology presented in [8], but has been specialized to better consider multiple perspectives impacting robotic missions.

Mission tagging. The first step guides the robotic expert in integrating tags in the source code, making it possible to determine where a relevant activity begins and finishes. Notably, the user performing this sub-step is typically an expert of the robotic system, who has complete knowledge of the code controlling the robots' behavior; therefore, the tagging integration is a task that would not require much effort from the expert (as also supported by the qualitative evaluation presented in Section 6.2).

The expert indicates points, i.e., tags, in the code that trigger events during the robot's execution. Specifically, the expert specifies a list of attributes relevant to analyzing the robot mission with process mining. A tag has to report at least the name of the *activity*, and its execution status, i.e., the *lifecycle*, which either indicates the start or the completion status of the activity, i.e., *start* or *complete*. Notably, an essential characteristic is that the robotic activities must share the same and suitable level of granularity [39]. For example, in Fig. 5 we can see the composition of a robotic activity named *Explore*, performed by the drone from the running scenario to inspect a specific area. This activity can be seen as the combination of more granular activities, i.e., *Go To* and *Detect Weed*, which in turn are realized through a combination of various peripheral-centric robotic data produced by the robot. Notably, the concept of a tag directly related to the source code allows an

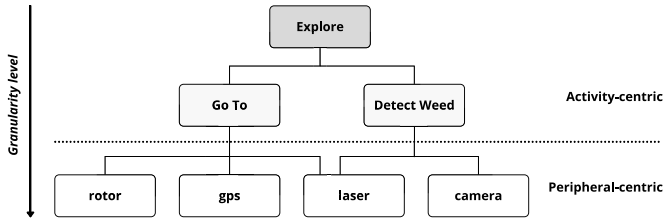


Fig. 5. Robotic activity composition.

expert to choose the desired level of granularity, hence passing from a peripheral-centric to an activity-centric point of view suitable for the analysis that will be applied to the event logs.

Mission execution. In the second step, the code is executed multiple times in a simulation environment or a real-world scenario. Notably, a simulation environment allows for checking the system's correctness while avoiding deploying incorrect behaviors into physical devices. Additionally, the simulation eases and speeds up the massive execution of the system while allowing testing in different environment settings. During the system execution, a listener is integrated to catch the event data. Specifically, since the robot's perspectives, e.g., spatial position or battery level, can change during the execution of an activity, a listener should record tagged data, and all the contextual perspectives that could be useful to get insights into the robotic mission. Moreover, in this step, the robotic expert must define how to segment the mission execution into meaningful cases for analysis. This is necessary because process mining requires a case identifier to group events that belong to the same execution instance. In the context of robotic systems, a case can correspond to a specific robot, a complete mission run, or a defined operational period. For example, one might consider a mission instance concluded when the robot returns to the docking station. The choice of how to segment missions, and thus define cases, depends on the robot's operational logic and the specific insights the expert aims to extract from the analysis.

Data processing. During the third step, the recorded event data needs to be processed to become an event log suitable for process mining analysis. First, to comply with process mining requirements, the produced data has to be segmented based on the case identified during the previous step. In addition, this step involves enriching activity-centric event logs, i.e., tagged data, with all relevant perspectives chosen by the expert. Since a robot's perspectives can change during the execution of an activity, this step should manipulate the events by associating each activity execution with the temporal evolution of its corresponding perspectives. This can be done by introducing additional events with the same activity name, a lifecycle status labeled as *inprogress* [40], and a set of values reflecting the multi-perspective information relevant to the system. As a result, the output is a robotic event log, translated in XES format, where each recorded event represents a meaningful robotic activity and is enriched with the temporal evolution of all perspectives of interest. An example of a XES-based robotic log with multiple perspectives is shown in Listing 1.

```
...
<event>
    <date key="time:timestamp" value="2025-04-01T12:11:18" />
    <string key="concept:name" value="explore" />
    <string key="lifecycle:transition" value="start" />
    <string key="lifecycle:state" value="none" />
    <float key="data:x" value="0.0" />
    <float key="data:y" value="5.0" />
    <float key="data:z" value="1.4" />
    <string key="org:resource" value="drone" />
    <string key="message:id" value=null />
</event>
<event>
    <date key="time:timestamp" value="2025-04-01T12:11:20" />
    <string key="concept:name" value="explore" />
    <string key="lifecycle:transition" value="none" />
    <string key="lifecycle:state" value="inprogress" />
    <float key="data:x" value="0.5" />
    <float key="data:y" value="4.8" />
    <float key="data:z" value="1.4" />
    <string key="org:resource" value="drone" />
    <string key="message:id" value=null />
</event>
...
```

Listing 1: XES-based example of robotic event log.

Mission discovery. In the fourth step, the methodology adopts process discovery to provide a process model that describes the robotic mission, in terms of executed activities, by observing the event log. Since the generated log follows the XES standard, various process discovery techniques can be applied. The selection of a discovery algorithm depends on the specific analysis objectives, as different representations offer unique insights into the structure and flow of activities.

In this paper, we chose to represent the robotic mission as a DFG because it provides an intuitive visualization, thus making it easier to interpret robotic missions. Moreover, a DFG allows analysts to avoid direct manipulation of event logs, such as filtering operations, which may be computationally expensive. Instead, these operations can be performed more efficiently by working directly on the DFG [19]. The discovery algorithm, presented in [19], extracts from the log the events produced during the mission execution and collects the directly-follows relations, i.e., binary relations indicating whether an activity occurs immediately after another. The relations between the performed activities are graphically represented as a DFG, i.e., a directed graph whose nodes represent the process activities and the arcs indicate a directly-follows relation. In this form, the DFG summarizes the different executions of the robotic mission seen in the event log. Each path from the start to the end node represents a sequence of activities performed during the mission execution. In the case of a frequency DFG, the labels on the nodes indicate how often each activity has been performed, while the labels on the arcs show how frequently the connected activities have been executed in sequence. In contrast, for a performance DFG, the labels on the nodes represent the duration of each activity, and the labels on the arcs indicate the time taken for transitions between activities.

Process enhancement. Since the discovery of the process focuses only on the control-flow, this step leverages multiple visualizations to extend the process view with additional information, resulting in a more extensive analytics representation. Indeed, the process enhancement step builds on the robotic event log by exploiting the explicit correlation between activities and contextual data recorded during mission execution. Rather than embedding contextual information directly into the process model, each perspective is analyzed through dedicated visualizations that are semantically aligned with the discovered control-flow. This design allows analysts to relate observed mission behavior to contextual perspectives without overloading the process model or losing information, while keeping the enhancement step independent of the specific process representation adopted.

Based on the motivation presented in Section 3, we choose to incorporate robot spatial occupancy, energy consumption, and communication as key perspectives for enhancing the discovered process model, thus enabling a more context-aware representation of robotic missions. These perspectives complement the time perspective, which is inherently captured during mission discovery.

- *Spatial Perspective*. The spatial perspective strongly impacts the control-flow, as robots operate in dynamic and often unknown environments [41]. To capture this, each activity in the discovered process model is linked to a set of positional data points representing the robot's location during execution. This perspective enables analysts to relate activity execution to environmental

conditions, such as identifying areas where specific behaviors repeatedly occur or where navigation issues arise, thus fostering the reasoning on how this may impact robotic behavior.

- *Energy Perspective*. Battery availability is critical for task execution, determining whether a robot can complete a task or needs to recharge before proceeding [42]. To support this analysis, the process model is enhanced with energy consumption data for each activity, alongside an overall mission discharge profile. By correlating energy consumption with activity execution, this perspective supports the identification of energy-intensive activities, possibly energy-driven deviations, and potential causes of mission termination.
- *Communication Perspective*. The distributed and cooperative nature of robotic systems requires a timely and accurate exchange of information among devices [43]. Indeed, by exploiting communication mechanisms, the robots can coordinate with each other and manage task assignments. Therefore, the process model is extended to incorporate inter-robot communication, capturing message exchange in terms of duration and loss rates. This perspective enables reasoning about coordination effectiveness by relating communication failures or delays to specific activities and performed activities within the mission. Notably, communication may also involve interactions with humans or external systems.

Together, these perspectives enable analysts to enhance control-flow inspection and reason about why specific mission behaviors occurred under particular contextual conditions.

Reasoning. The final step of the methodology involves reasoning on the produced analytics to derive mission analysis results. In this step, insights gathered from the process model and visualizations are examined to identify hidden patterns, correlations, and insights from performed activities, mission, and contextual perspectives. This reasoning can be driven by automated systems, where specialized algorithms can identify anomalies and relationships, or guided by human expertise, with robotic experts leveraging intuition and experience to interpret the interplay between the process model and visualizations. This analysis helps to understand how different perspectives influence the execution of robotic activities. Moreover, it can highlight potential inefficiencies or areas for improvement in mission performance, providing opportunities for optimization. Whether automated or human-driven, this step aims to provide analysis results capable of fostering a deeper understanding of the system and supporting informed decision-making.

5. Implementation

The proposed methodology can be automated both for data processing and for mission analysis. For data processing, a collection of scripts facilitates the automatic translation of unrefined data, generated during the robot's execution, into robotic event logs suitable for the application of process mining techniques. This implementation is tailored to ROS-based systems, as ROS is recognized as the de facto standard for robot application development [44]. For mission analysis, the RoboTrace tool enables both the process discovery and its enhancement with multiple perspectives. Notably, these implementations are loosely-coupled and flexible enough to be used independently of each other. The source code to reproduce these steps is available online [45].

Data processing. Data generated during robots' execution can be acquired using the *ros2bag*[3] library. This library serves as a listener for a custom set of ROS topics transmitted within the robot's network. However, the recorded data are stored in a format specific to ROS, which needs to be processed to create event logs. To facilitate this processing, a set of scripts has been developed, supporting the data processing workflow illustrated in Fig. 6. These scripts enable the transformation of the unrefined ROS data into robotic event logs.

After the execution of the system, the unrefined ROS data is given as input to the *Data Conversion* script. This script decodes the ROS-based data into several files, representing both the events generated by the system's tags and other perspectives identified by the developer. However, the retrieved event data requires further refinement to create process mining-compliant event logs. This refinement is carried out by the *Data Enrichment* script, which enriches the tag-based data by first defining each mission execution as a case identifier, then incorporating the perspectives of interest both in the tag-based data and by introducing additional events with the same activity name, an *inprogress* lifecycle status, and a set of values reflecting the evolution of the perspectives. The resulting *robotic event log* is in the form of Listing 1.

Process discovery and enhancement. The tool enabling mission analysis is RoboTrace, a web application, that automates the discovery and enhancement steps. It is composed of two modules. One is responsible for generating the process model from the event log, achieving this through DFG discovery. The second module utilizes the information about the generated process model to graphically visualize the received process model and enhance it with multiple perspectives.

The first module enables the discovery of a DFG based on the event log given as input. It exploits the *pm4py*[4] library both for importing and manipulating the event log and for discovering the DFG. The interface of this module offers functionalities for both uploading event logs and applying filters to them. Indeed, since the DFG discovery correlates the events based on temporal occurrence, analysts can exploit the interface to apply filters to the event log before initiating the process discovery. This allows them to refine the event log and focus the analysis on specific aspects of interest. At present, the module incorporates two primary filters: one for the case identifier and another for the robot identifier. The discovery output consists of a collection of nodes, edges, and labels that represent the DFG structure. Additionally, to make the graph more user-friendly, and thus ease the analysis, the module appends a *start node* to the events that serve as a start event, and an *end node* to the events that serve as end nodes, ultimately providing a clear and easily navigable process model for analysis.

The second module receives the collection of nodes, edges, and labels produced by the first module. It enables the generation of an enhanced process model by integrating the perspectives' view with the activities. This module is intentionally designed as a unique analytical interface that brings together control-flow and contextual information, allowing analysts to reason about mission execution without switching between separate tools or views. Rather than merging all the information into a single visualization, which could hinder interpretability, each perspective is rendered through a dedicated plot, generated from the information retrieved in the event log. Additionally, each plot style has been selected to ensure clear and efficient visualization and analysis of the corresponding information. Specifically, *spatial occupancy* visualization has been represented via a 3D scatter plot. Indeed, the scatter plot enables the visualization of the relationship between three quantitative variables [24], in this case, the x, y, and z dimensions. To enhance human reasoning, points have been associated with different colors and shapes to distinguish between activities performed by each robot. *Energy consumption* visualization has been represented via a line chart. Specifically, a line chart shows how quantitative values have changed over time for different categories [24]. For each robot, we plotted the time progression on the x-axis and the battery levels on the y-axis. Additionally, to enhance readability and facilitate analysis, we segmented the battery discharge plots by separating them according to different cases. Finally, *communication* visualization has been represented as a stacked bar chart. Specifically, a stacked bar

[3] https://github.com/ros2/rosbag2

[4] https://processintelligence.solutions/pm4py

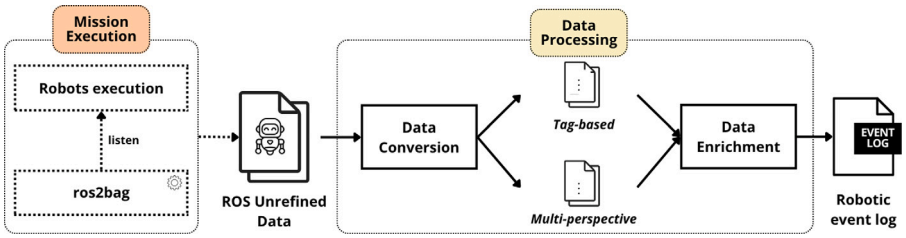


Fig. 6. Data processing workflow.

chart shows how different categories combine to form a whole across major items. Each part's size reflects its quantitative proportion, represented as stacked segments [24]. For the robotic scenario, we selected to represent *received* and *lost messages*. Together, these visualizations provide an interactive and comprehensive view of mission execution, combining control-flow, spatial occupancy, energy usage, and communication dynamics. Additionally, the analyst can perform more accurate analysis over single activities by selecting a node in the graph and the perspectives of interest.

Notably, to improve visualization scalability, users can exploit interactivity features provided by the plots. Specifically, *filtering* allows users to include or exclude data from the display, such as visualizing only the spatial occupancy during a specific activity. Meanwhile, *highlighting* enables users to zoom in on areas of interest, such as focusing on the final moments of a case execution to observe battery depletion in detail.

6. Proof of concept

This section shows the feasibility and effectiveness of the methodology. For this purpose, we apply to the running scenario presented in Section 3, the steps prescribed by the methodology to produce event logs and analyze them via process mining and visual analytics techniques. Additionally, we conduct a qualitative user evaluation with robotic domain experts to assess the perceived feasibility, interpretability, added-value, utility of the proposed methodology.

6.1. Methodology at work

Following, we present the methodology steps applied to the smart agriculture scenario.

Mission tagging. To be compliant with the ROS architecture of the running scenario, a custom ROS message capable of containing the tag data is defined. Therefore, the message is defined to contain the *activity* field that identifies the action that the robot is performing, and the *lifecycle* that determines the status of the related activity. Once the message has been defined, the activities performed by the robots are identified, selecting a level of abstraction appropriate for behavioral analysis. Specifically, the drone activities are: *take off*, *explore*, *weed found*, *weed position*, *tractor position*, *closest tractor*, *low battery*, *field cleaned*, *return to base*, and *land*. For the tractors *weed position*, *tractor position*, *closest tractor*, *move*, *cut grass*, *low battery*, *return to base*, and *stop*. Consequently, tags are placed in the points of the code where activities begin or finish. Notably, during tag integration, the activities related to messages end with a *!*, if they are of the sender type and with a *?* if they are of the receiver type [46]. To better clarify the tag concept, Listing 2 shows an excerpt of the tag integration in the ROS code. Here, the *Cut grass* function call is preceded and succeeded by publishing a topic containing the aforementioned fields.

```
tag = TagTopic()
tag.activity = "cut_grass"
tag.lifecycle = "start"
ros_publisher.publish(tag)
CutGrass(ros_node)
tag.lifecycle = "complete"
ros_publisher.publish(tag)
```

Listing 2: Excerpt of tag integration

Mission execution. Once we completed the tag integration, we ran the system in the Gazebo simulator. Fig. 7 shows two instances of the mission execution with related activities graphically tagged. Despite current simulation solutions are not able to emulate the real-world phenomena in a sufficiently realistic manner [13], our choice of using the simulation environment was mainly driven by the possibility of executing simulations with different initial settings. It allowed us to set random locations of weeds and variable battery discharge for each robot, as well as to produce a significant amount of data by running 70 simulations of the system execution. Specifically, we defined that each simulation run corresponds to a case identifier. Notably, the number of executed simulations was chosen as a trade-off between generating diverse mission instances, with variable weed locations and battery discharge levels, and the computational resources available for running Gazebo simulations.

Data processing. The execution of the simulations generates robotic event data that is processed to extract activity-centric event logs enriched with the selected perspectives. Table 2 shows an excerpt of the robotic event log related to a simulation run. Notably, for the sake of presentation, the log is shown in a tabular format. However, it is also generated in compliance with the XES standard, similar to Listing 1. The *Case* value uniquely identifies one run of the system execution, and the *Time* value stores a timestamp, i.e., the date and time in which the event occurred. *Activity* and *Lifecycle* are related to the high-level data generated via tags, and respectively represent the event name and its execution status. Considering the spatial perspective, the x, y, and z values store the position of the robot when an activity starts, is in progress, or completes. To distinguish the robot that generates a given event, its identifier is saved in the *Resource* field. *Msg* stores the message identifier shared between robots. Finally, *Battery* stores robots' battery levels at the time the event is generated.

Mission discovery. The event log produced by the data processing step can be easily uploaded into its interface. After uploading the event log, the analyst can discover the DFG depicting the control flow of the multi-robot system. Fig. 8 represents the frequency DFG of the overall mission of the robotic scenario. The resulting DFG provides an overall view of the mission execution by presenting the frequency, across all cases, during which an activity occurred, or the performance of an activity, as well as the frequency/time duration of the relation between the connected activities. Since the DFG is directly derived from the event log, the discovered model is fully compliant with the observed execution, ensuring maximal fitness with respect to the input data. Notably, exploiting the RoboTrace tool, the DFG can also be easily navigated, explored, and filtered.



Fig. 7. Case study in Gazebo.

Table 2
Excerpt of robotic event log.

Case	Time	Activity	Transition	State	Attributes					
					x	y	z	Resource	Msg	Battery
...	...	...	...	...	...	...	...	...	...	...
38	10:08:53.12	explore	null	inprogress	0.59	3.33	1.48	drone	null	97
38	10:08:55.01	explore	null	inprogress	0.99	4.01	1.48	drone	null	97
38	10:08:56.55	explore	null	inprogress	1.12	4.52	1.48	drone	null	96
38	10:08:57.61	explore	complete	null	1.66	5.03	1.48	drone	null	96
38	10:08:57.88	weed_found	start	null	1.66	5.03	1.48	drone	null	96
...	...	...	...	...	...	...	...	...	...	
38	10:08:58.00	weed_position!	complete	null	1.66	5.03	1.48	drone	m52	94
38	10:08:58.30	weed_position?	start	null	0.25	1.12	0.0	tractor_1	m52	85
38	10:08:58.30	weed_position?	complete	null	0.25	1.12	0.25	tractor_1	m52	85
38	10:08:58.30	tractor_position!	start	null	0.25	1.12	0.0	tractor_1	m53	84
38	10:08:58.32	tractor_position!	complete	null	0.25	1.12	0.0	tractor_1	m53	84
38	10:08:58.34	weed_position?	start	null	0.17	1.12	0.0	tractor_2	m52	71
38	10:08:58.35	tractor_position?	start	null	1.66	5.03	1.48	drone	m53	93
38	10:08:58.35	tractor_position?	complete	null	1.66	5.03	1.48	drone	m53	93
38	10:08:58.36	weed_position?	complete	null	0.17	1.12	0.0	tractor_2	m52	71
38	10:08:58.37	tractor_position!	start	null	0.17	2.04	0.0	tractor_2	m54	71
38	10:08:58.39	tractor_position!	complete	null	0.17	2.04	0.0	tractor_2	m54	71
38	10:09:00.91	tractor_position?	start	null	1.66	5.03	1.48	drone	m54	93
...	...	...	...	...	...	...	...	...	...	...

Process enhancement. After mission discovery, the enhancement step takes the discovered process model as input and supports the joint exploration of control-flow and contextual perspectives selected by the expert. Rather than modifying the process model itself, this step provides an integrated analytical interface that enables correlating mission activities with spatial, energy, and communication information. The resulting interface, depicted in Fig. 9, provides the expert with a dashboard to visualize the interleaving of the mission process model, the spatial occupancy of the robots when performing the mission, robots' energy consumption for each case, and the communication metrics regarding received and lost messages. This allows experts to inspect how specific activities relate to contextual conditions, supporting exploratory-driven reasoning. Moreover, if the energy perspective inspection is active, DFG nodes are enhanced with a visualization showing the average energy consumption of each activity.

Therefore, the visualizations presented for the running scenario aim to demonstrate how the enhancement step can enrich mission analysis through coordinated visual analytics without constraining the underlying process representation, thereby prioritizing interpretability and exploratory analysis in support of human-driven mission understanding.

Reasoning. To reason on the robot mission, we start with an exploration of the mission's DFG to provide an initial understanding of the sequence and consistency of executed actions. As illustrated in Fig. 8, the overall behavior of the system showed that the robots followed the expected workflow. Indeed, in this representation, event connections are established only based on temporal occurrences. Consequently, events executed by different robots cannot be differentiated. Nevertheless, this DFG offers a comprehensive overview of the multi-robot mission. It reveals a consistent pattern where the drone always concludes (via the *land* activity) its execution before the tractors.

The DFG is filtered to dig deeper by isolating the drone's process, generating a new DFG for the mission of this robot. Fig. 10 depicts the process of the drone. The process shows that the drone properly performed the desired behavior, by starting with a *take off* activity and then moving to the *explore* activity. During this phase, it can navigate and look for weed grasses. When a weed is identified, it triggers a *weed found* and notifies the tractors through the *weed position* message. Afterwards, when it has received tractor responses, it moves to the *tractor position* status. Then, it can elect the tractor closest to the weed, i.e., *closest tractor*, and restart its exploration. Finally, the drone's execution ends when it spots a *low battery* state. Therefore, it can perform the *return to base* and *land* activities. Notably, the extracted process shows that the drone has never performed the *field cleaned* activity, meaning that it has been unable to explore the whole field. Indeed, the analysis of the mission process model shows that the running robot worked

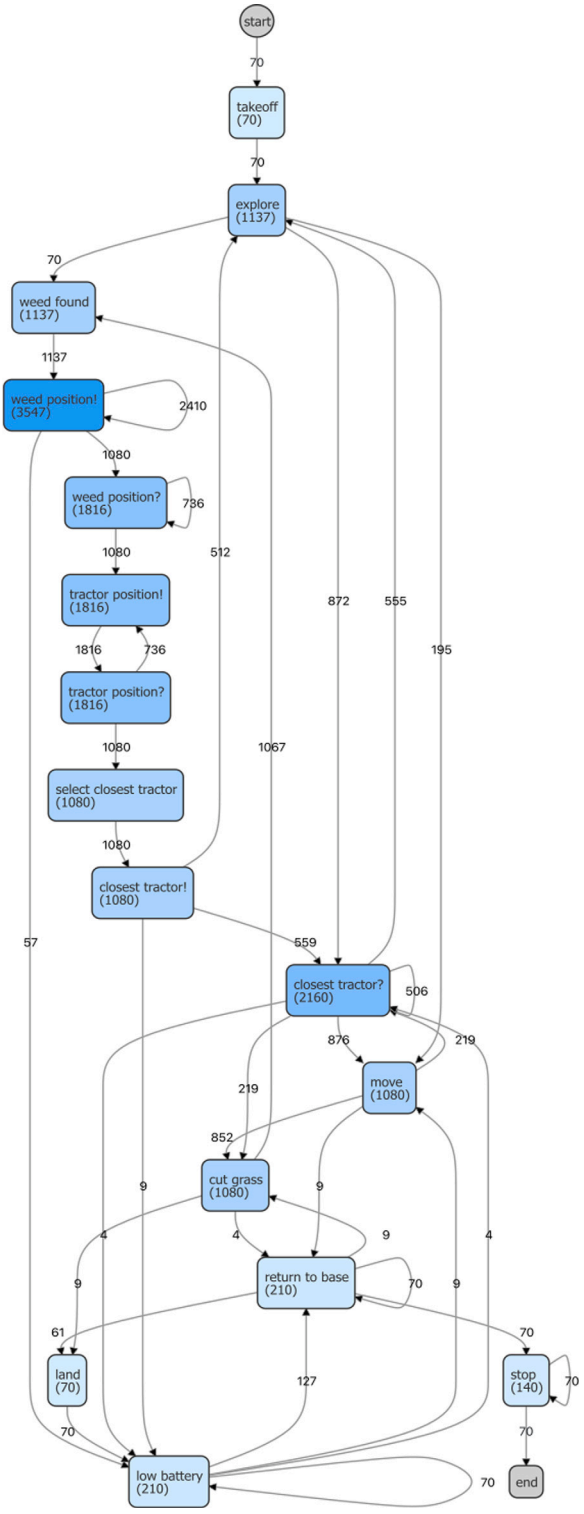


Fig. 8. DFG of the smart agriculture scenario.

as expected without performing any incorrect behavior. However, to better understand why there was no expected activity in the process, we inspect other system perspectives to investigate possible causes, such as a programming error or a lack of resources.

We initially focus on the spatial perspective, mapping the drone's movements across the operational area. This visualization, depicted in Fig. 11, reveals an incomplete exploration pattern, with unexplored sections of the field. Specifically, the drone always takes off from the base station and starts exploring the field. During the exploration, it spots some weeds and reports them to the tractors. If, on the one hand, this sequence of actions has already been verified with the DFG, the visualization of the space helps to identify further potential errors in the system. Specifically, although the drone has properly performed the desired activities, it could not explore most of the field. This might be due to a programming error preventing the robot from navigating correctly. Additionally, a robotic developer may have overestimated the capabilities of the robots. For instance, a single drone may not be sufficient for exploring the intended field, or the two tractors are too slow to satisfy the demands of the drone.

Looking for further clarity, we move to a finer DFG granularity, focusing on a single execution run, specifically, we discuss case 1. By inspecting the corresponding spatial plot, we observe that the drone remains stationary in a specific area while executing the *weed position!* activity. This continues until a *low battery* event occurs (see Fig. 12). This observation raises new questions about the underlying cause of the blockage. To investigate further, we integrate energy consumption data into the analysis, combining it with the performance DFG to incorporate a time perspective as well. As depicted in Fig. 13, a considerable amount of energy usage is associated with the *weed position!* activity, which, on average, takes 10 s to complete and recurs every 10 s. This suggests that this activity significantly impacted the drone's resources, thus affecting the successful completion of the mission.

Since a potential issue is linked to an activity representing inter-robot communication, we take the initial log and filter on case 1, keeping also the tractors, so that we can retrieve received and lost messages. Indeed, the communication analysis reveals 8 *weed position* messages were lost, suggesting that the drone might have awaited tractor responses. This communication breakdown could explain both the standstill in Fig. 12 and the resulting battery drain, as the drone may have continued consuming energy while attempting to receive a response (see Fig. 14).

Summing up, the iterative nature of the human-driven reasoning step, i.e., moving progressively from high-level mission insights to detailed investigations of space, energy, and communication data, highlights the strength of combining process mining with visual analytics. Each layer of investigation enhances the understanding of the mission, transforming an initial anomaly into an intuitive diagnosis of the possible issues affecting mission execution.

6.2. User evaluation

To investigate the practical applicability of the proposed methodology, we conducted an expert-based evaluation with robotics researchers from academia. The evaluation combined semi-structured interviews and standardized questionnaires to examine the perceived feasibility of activity tagging, the interpretability of the discovered process model, the added value of the visual perspectives, and the usefulness of the proposed mission analysis approach with respect to current tools and practices.

Methodology and domain experts selection. We followed the *qualitative survey empirical standard* and its essential attributes [47] to perform semi-structured interviews. The interview study was complemented with post-session standardized *questionnaires*, namely System Usability Scale (SUS) and NASA Task Load Index (NASA-TLX), to collect descriptive indicators of perceived usability and workload. Participants were selected through *convenience sampling* [48], with the goal of collecting in-depth expert feedback. The recruitment criteria focused on participants' knowledge and expertise in robotic software engineering, mission specification, and reasoning. This choice is consistent with

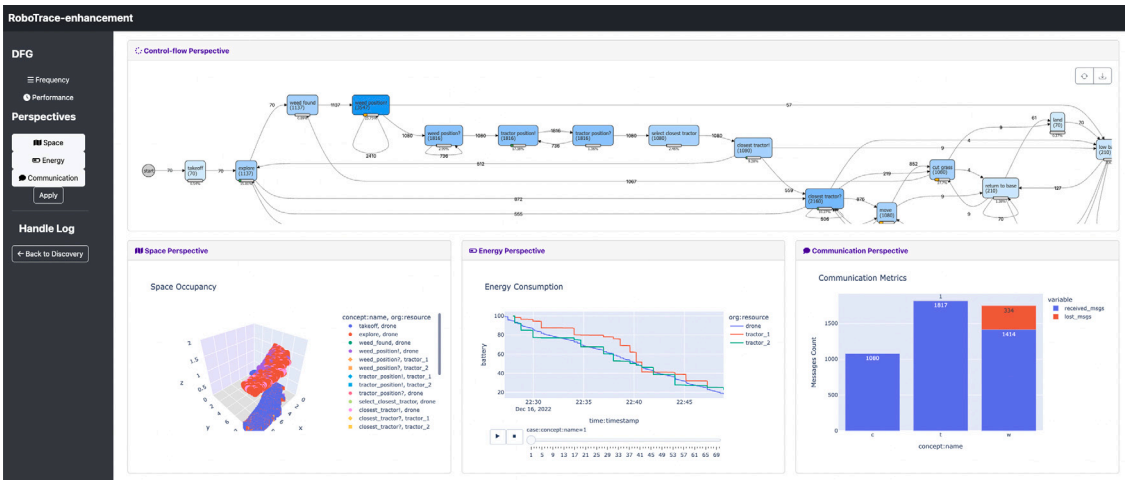


Fig. 9. RoboTrace-enhancement interface.

established empirical practices in software engineering research, where expert-based qualitative studies are commonly employed to gain qualitative insight into novel approaches [49–51]. While the choice of using SUS [52,53] and NASA-TLX [54] was adopted to assess the subjective usability and workload induced by the interaction with the tool. These two instruments capture complementary aspects of the user experience: SUS is a compact and widely used standardized questionnaire for perceived usability, whereas NASA-TLX provides a multidimensional assessment of workload, including mental demand, temporal demand, effort, perceived performance, and frustration. This is particularly relevant in our setting, where participants were asked to inspect and reason about robotic mission analysis through a novel visualization dashboard. For anonymity purposes, the experts are referred to throughout the text as *[ex:1]–[ex:5]*. All participants have contributed research publications and are actively involved in teaching courses in this domain. Specifically, they include researchers and faculty members working in computer science-related areas. Their combined expertise spans runtime verification, software engineering, business process models, privacy-enhancing technologies, and service-oriented computing, while also addressing the design, verification, and implementation of robotic systems. Overall, we interviewed five domain experts.

Experimental protocol. To clarify the evaluation procedure, we summarize the protocol followed during the user evaluation. The study combined two complementary components: (i) a semi-structured interview aimed at collecting expert feedback on the proposed approach, and (ii) a questionnaire-based assessment aimed at collecting descriptive indicators of perceived usability and workload. Each session was conducted synchronously, involved one participant at a time, and lasted approximately one hour.

At the beginning of each session, the interviewers introduced the study objectives and provided the background needed to understand the approach. The introduction covered the problem addressed by RoboTrace, the role of process models in representing robotic missions, the tagging mechanism used to generate activity-centric event logs, and the selected visual perspectives. The same explanatory material was used for all participants, namely the explanatory slides available in the replication package, followed by a demonstration of the RoboTrace prototype.

The interview part was grounded in the case study presented in Section 6.1. This case was used as a representative example to support expert reasoning about the approach, as it illustrates the main steps of the methodology, from tagging to process discovery and visualization. The discussion followed a semi-structured guide organized around four evaluation dimensions: (i) feasibility of the tagging activity, (ii) interpretability of the resulting process model, (iii) added value of the generated visual perspectives, and (iv) usefulness of the approach with respect to baseline representations. For each dimension, experts were asked open-ended questions and encouraged to think aloud, commenting on strengths, weaknesses, ambiguities, limitations, and possible improvements.

After the interview, participants proceeded to the questionnaire-based part of the evaluation. This part consisted of two steps. First, participants used RoboTrace to analyze three synthetic event logs generated for the study, each representing a different robotic mission scenario, and performed five analysis tasks reflecting the main perspectives supported by the tool. Second, after completing the tasks, they filled in the SUS and NASA-TLX questionnaires to report their perceived usability of RoboTrace and the perceived workload associated with the analysis. The use of synthetic logs avoided biases related to participants' familiarity with the robotic system discussed during the interview, as well as biases caused by recognizable or context-dependent mission details.

Specifically, participants were asked to: identify the typical sequence of activities and the main phases of the mission (control-flow); detect delays, bottlenecks, or unusually long-running activities (performance); use the recorded spatial coordinates to estimate the area covered by each activity and, for multi-robot scenarios, determine whether robots operated in the same or distinct regions (space); assess whether battery-related conditions affected the mission execution (energy); and identify delayed or missing messages between robots, for multi-robot scenarios (communication). For each synthetic log, we defined a reference ground truth describing the expected findings for these tasks, including the main control-flow sequence, potential performance issues, spatial patterns, battery-related issues, and communication anomalies. Participants compared their observations with this ground truth as a qualitative correctness check. At the end of the tasks, participants completed the SUS and NASA-TLX questionnaire to collect descriptive evidence on perceived usability and workload. The results were analyzed descriptively and used to complement the qualitative findings obtained from the interviews.

Two of the authors conducted all sessions following the same protocol. After the fifth interview, no significant new themes emerged regarding the evaluated dimensions, suggesting thematic saturation within the scope of the evaluation. To mitigate participants' bias in the qualitative analysis, the interviewers followed the same semi-structured guide, avoided leading questions, and encouraged participants to discuss both positive and negative aspects of the approach. Moreover, the experts reflected on their own assumptions and expectations before and during the analysis, and discussed emerging interpretations to reduce the influence of individual bias. The experts' answers were then analyzed qualitatively to identify recurring themes and viewpoints,

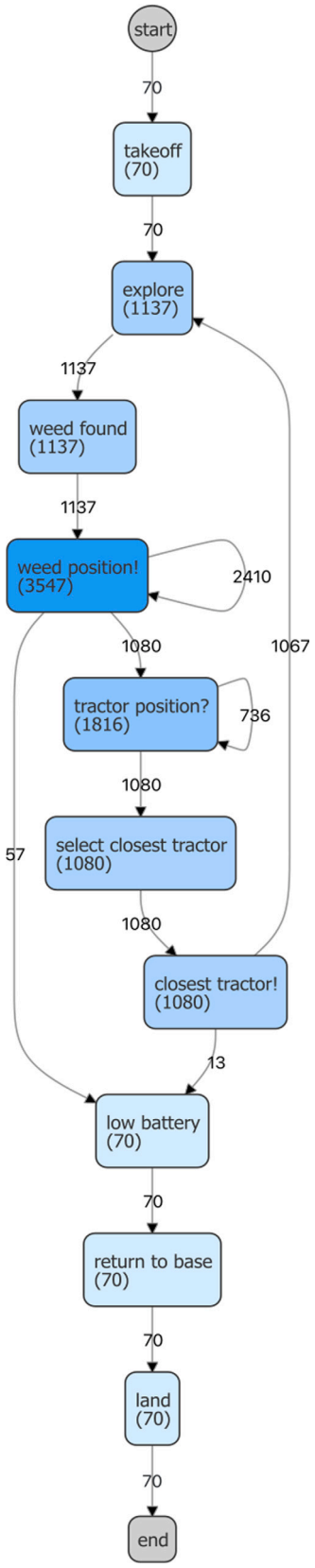


Fig. 10. DFG filtered on the *drone* resource.

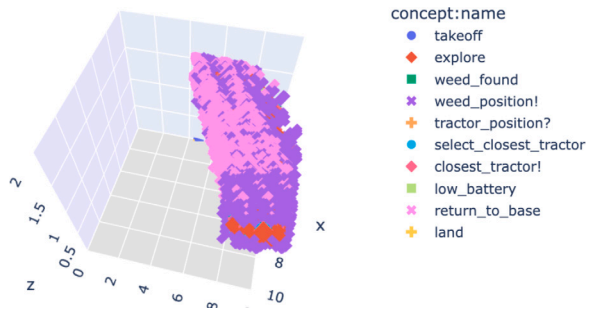


Fig. 11. Drone *spatial occupancy*.

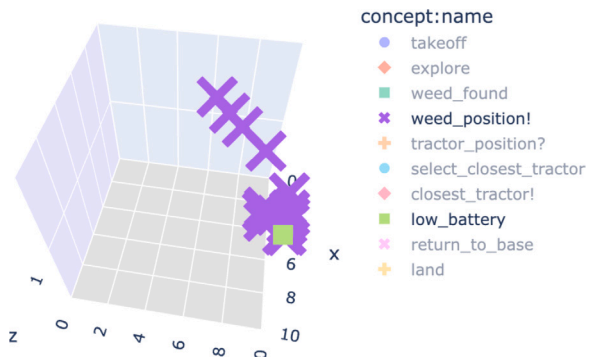


Fig. 12. Drone filtered *spatial occupancy* for case 1.

focusing on their relevance and coherence with respect to the evaluation goals. The overall material, including the explanatory slides, the interview guide, the RoboTrace prototype, the questionnaires, the synthetic event logs, the reference ground truth, and the reconstructed interview transcripts, is available in the online repository [45].

Results of the interview-based validation. We submitted to experts' evaluation the proposed methodology and its application to the running scenario (cf. Section 6.1). The questions in the interview were designed to gather their opinions on the feasibility of the tagging integration, the interpretability and added-value of process models and visualizations, and the utility of the proposed mission analysis with respect to baseline approaches, beyond the scenario. Below is the set of open-ended questions the domain experts were asked.

- **Q1** How would you assess the effort required to define and integrate activity tags into a robotic mission? In which scenarios do you think tagging would be feasible, and in which might it become impractical?
- **Q2** Based on the process model, can you describe what happened during the mission?
- **Q3** Did the additional perspectives influence or refine your understanding of the mission? How did the combination of the process view and these visualizations affect your reasoning about the mission behavior?
- **Q4** How would you normally analyze a robotic mission like this using the tools, data, or practices you currently rely on?
- **Q5** Do you think you could have reached similar insights or conclusions using your usual tools or approaches?

Feasibility of activity tagging (Q1). Experts generally perceived the integration of activity tags into a ROS-based system as technically feasible, especially when developers have control over the codebase, and described the required implementation changes as localized. However, several participants emphasized that the main challenges are conceptual rather than technical. For instance, *[ex:3]* pinpointed that

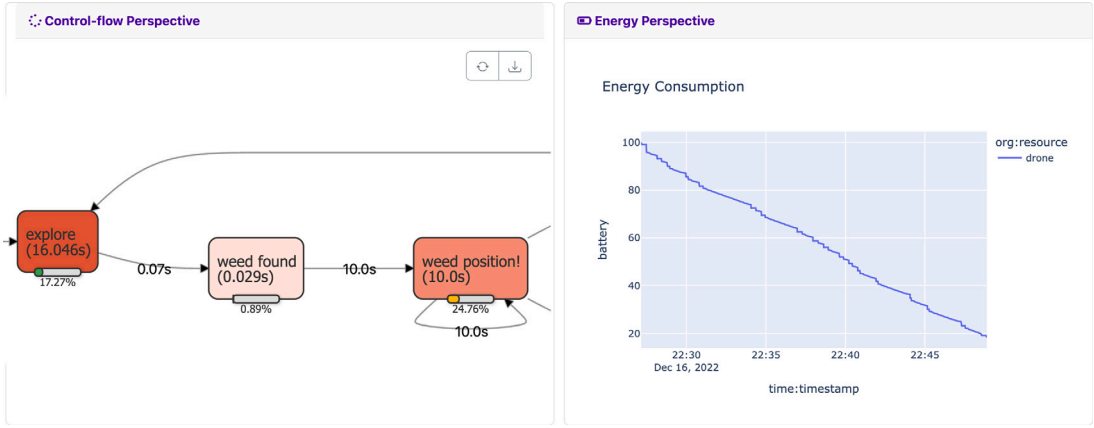


Fig. 13. Drone *energy consumption* for case 1.

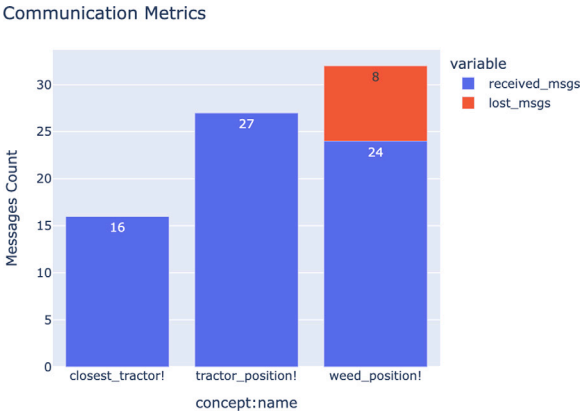


Fig. 14. *Communication* perspective for case 1.

selecting an appropriate abstraction level and identifying meaningful activities may be non-trivial design decisions. *[ex:1]* and *[ex:2]* noted that tagging becomes more challenging when behaviors overlap, are nested, or are reactive. Overall, tagging was considered feasible when developers have control over the codebase and sufficient understanding of the mission structure, while guidelines or best-practice patterns were suggested to reduce the risk of inconsistent tagging (*[ex:3]*, *[ex:5]*).

Interpretability of the process model (Q2). Experts reported that the discovered process model helped them obtain a structured overview of the mission execution and reconstruct the main sequence of activities and interactions. Several participants explicitly stated that the process view supports understanding the overall mission flow and distinguishing frequent from infrequent behaviors (*[ex:2]*, *[ex:4]*). Additionally, *[ex:1]* highlighted its usefulness for understanding the sequential nature of the mission, specifically in cases where sequencing is not explicitly represented by the developer, such as when using behavior trees or similar control structures. At the same time, *[ex:3]* observed that interpretability depends on the chosen abstraction level, as too fine-grained tagging may lead to complex models that are harder to read, while coarse-grained tags may lead to a loss of information. In multi-robot settings, *[ex:4]* and *[ex:5]* experts highlighted the importance of distinguishing activities done by different robots, either through filtering or additional annotations.

Added-value of visual perspectives (Q3). All experts agreed that the contextual perspectives refined their understanding of the mission beyond what could be inferred from the process model alone. In particular, *[ex:1]* and *[ex:4]* emphasized the usefulness of spatial visualizations for identifying navigation patterns and abnormal behaviors, while *[ex:1]*, *[ex:2]*, and *[ex:5]* highlighted the relevance of energy-related views for pinpointing energy-intensive tasks and potential battery issues. Communication-related perspectives were perceived as particularly useful in multi-robot scenarios, as they helped reveal coordination issues and message losses (*[ex:1]*, *[ex:2]*). Overall, *[ex:2]*, *[ex:3]*, and *[ex:4]* experts appreciated that these perspectives have a direct connection to robots' activities and are presented as separate but coordinated views, avoiding visual overload.

Utility with respect to baseline approaches (Q4–Q5). When describing their usual analysis practices, experts reported to rely mainly on manual log inspection (*[ex:1]*, *[ex:5]*), ad-hoc plotting (*[ex:1]*, *[ex:2]*, *[ex:4]*), and, in some cases, requirement- or monitor-based analysis (*[ex:2]*, *[ex:3]*, *[ex:4]*). Experts *[ex:1]* and *[ex:5]* noted that similar insights could technically be achieved by leveraging, or customizing, existing tools or libraries to extract each required aspect separately (e.g., the process model and individual plots). However, this would require manual effort to connect heterogeneous data sources. In contrast, some experts perceived the proposed approach as potentially reducing the manual effort needed to connect different sources of evidence, by presenting process-oriented and contextual information within a unified analysis environment (*[ex:2]*, *[ex:3]*).

Beyond the evaluation dimensions captured by Q1–Q5, experts also provided additional suggestions during the final discussion. The interviews revealed a shared interest in the contextual visualizations we introduced. In addition, some experts suggested extending the current visualizations with additional ones. For instance, a Gantt-like chart to capture temporal aspects of task execution and complement the performance DFG (*[ex:1]*), or a social-network graph to provide a different view on multi-robot communication (*[ex:5]*). In this direction, the methodology and tool have been designed to naturally support the integration of additional visual perspectives. Another direction emerging from the discussion concerns the integration of runtime verification and monitoring techniques to complement the proposed visual analysis. Experts *[ex:2]*, *[ex:3]*, and *[ex:4]* noted that some behaviors may be challenging to assess only via visual inspection and suggested combining visualizations with property-based or requirement-driven analysis to strengthen the analysis of mission executions.

> ***Summary and emerging insights.*** The interviews provided preliminary support for the expected usefulness of visualization-based analysis for obtaining a structured overview of mission execution, while also revealing non-trivial methodological design implications. In particular, experts highlighted that the main challenge of activity tagging lies in choosing meaningful abstraction levels, especially for overlapping, nested, or reactive behaviors. They also emphasized that model interpretability depends on tag granularity, since fine-grained tags increase analysis detail but may reduce readability. Finally, the feedback identified tagging guidelines, additional temporal and communication-oriented views, and integration with requirement-based monitoring as relevant directions for extending the approach. Thus, the evaluation provides exploratory evidence that helps clarify where the methodology may be useful, what design choices affect its applicability, and how it can be further developed.

Overall, the expert feedback indicates that the proposed methodology is perceived as practically feasible in practice, effective in supporting mission understanding, and useful in complementing existing robotic mission analysis approaches, while also revealing potential for extension and identifying several promising directions for future research.

Results of the questionnaire-based validation. To complement the qualitative interviews, we collected questionnaire-based feedback on the perceived usability and workload associated with using RoboTrace for robotic mission analysis. The questionnaire was administered after participants had interacted with the tool and completed the assigned analysis tasks on synthetic robotic event logs. Specifically, the evaluation material consisted of synthetic logs covering three representative robotic mission scenarios: a single-drone mission, a multi-robot warehouse scenario, and a multi-robot search-and-rescue scenario. For each log, participants were asked to complete five analysis tasks reflecting the main perspectives supported by the tool: control-flow, performance, spatial workspace, battery, and communication analysis. After task responses, participants were able to compare the extracted insights with the reference ground truth defined for each synthetic log, to ground their subsequent assessment of perceived usability and workload in the analysis outcomes. Overall, the study involved 5 participants, each completing 5 analysis tasks on 3 event logs.

After completing the tasks, participants filled in the SUS and NASA-TLX questionnaires. The SUS questionnaire consists of ten items rated on a five-point Likert scale, ranging from 1 to 5 (with 1 = strongly disagree, to 5 = strongly agree). The items cover different aspects of perceived usability, including willingness to use the system frequently, perceived complexity, ease of use, need for technical support, integration of functions, consistency, learnability, cumbersomeness, confidence, and the amount of prior learning required before using the system. The NASA-TLX questionnaire was used to collect participants' perceived workload while performing the analysis tasks with the RoboTrace tool. It considers six workload dimensions: mental demand, physical demand, temporal demand, performance, effort, and frustration. We used the raw NASA-TLX variant, considering the six workload dimensions without applying pairwise weighting.

Fig. 15 shows the distribution of the SUS Likert-scale responses. Overall, the responses suggest a positive perception of RoboTrace usability among the involved participants. For the positively worded items, participants mostly agreed or strongly agreed that they would use the system frequently (SUS1), that it was easy to use (SUS3), that its functions were well integrated (SUS5), that most users would learn it quickly (SUS7), and that they felt confident using it (SUS9). Conversely, for the negatively worded items, participants consistently disagreed or strongly disagreed that the system was unnecessarily complex (SUS2), required technical support (SUS4), was inconsistent (SUS6), or cumbersome to use (SUS8). Responses to SUS10 were also mostly negative, suggesting that participants did not perceive a substantial amount of prior learning as necessary before using the tool. These results suggest that RoboTrace was perceived as usable, coherent, and learnable by the involved experts.

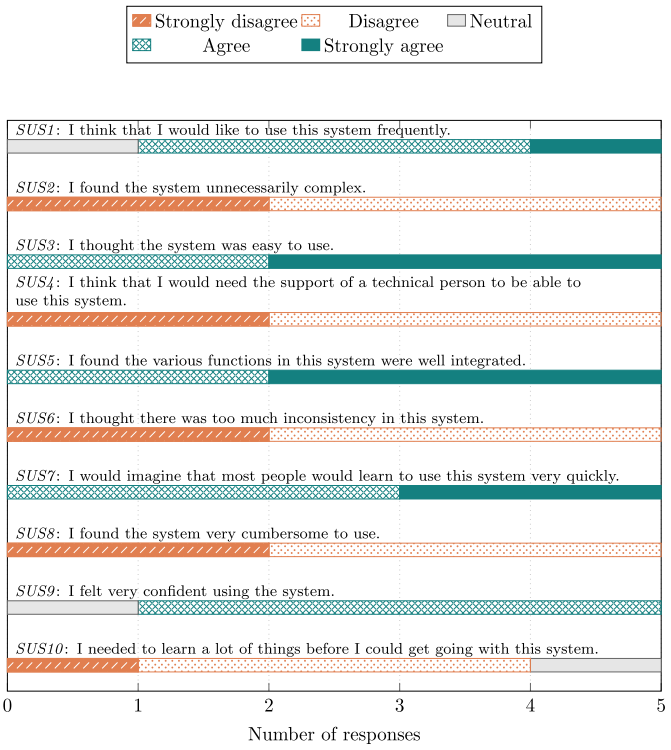


Fig. 15. Distribution of SUS Likert-scale responses.

Concerning perceived workload, overall, participants reported a moderate level of mental demand, with an average score of 3.0 out of 10 (with 1 = very low, to 10 = very high). Effort was also perceived as moderate, with an average score of 2.4, while physical demand and temporal demand were low, both averaging 1.6. Frustration was also low, with an average score of 1.8. Finally, the performance-related item (with 1 = perfect/success, to 10 = failure/unsuccessful) obtained an average score of 2.6, suggesting that participants generally perceived themselves as able to complete the assigned analysis tasks, also with respect to the provided ground truth.

These results suggest that using RoboTrace required some cognitive effort, mainly due to the reasoning involved in interpreting robotic event logs and their visual representations without knowing the reference scenarios, but did not impose substantial physical, temporal, or frustration-related workload. As for the SUS results, these values are interpreted descriptively and used to complement the qualitative findings rather than to support statistically generalizable claims.

Overall, the questionnaire-based evaluation suggests that the proposed approach can support the robotic system development lifecycle by supporting developers to analyze mission executions at a higher level, identify anomalies and execution inefficiencies, and consequently guide subsequent refinements of the robotic system implementation. Rather than replacing existing analysis practices, the goal of the proposed methodology and tool is to complement traditional low-level analysis/debugging approaches by providing mission-oriented abstractions and visual perspectives that facilitate the interpretation of complex robotic behaviors.

7. State of the art

This section examines the state of the art by considering works applying process mining techniques to robotic systems, visual analytics

in robotic systems, and integrating process mining and visual analytics in various domains. Since no existing methodology combines process mining and visual analytics specifically for robotic mission analysis, we structure our review around these three dimensions to present the current state of the art and related limitations.

Process mining & robotic systems. Recently, a few works have experimented with applying process mining techniques to analyze and diagnose a robotic system. In [55], the author proposes a process conformance verification using a process mining tool on the event log of a single-robot system. The approach aims to check the correctness of the robot's execution by supporting the correctness and quality assurance of a robot from design to execution. Considering the application of a robotic system in the manufacturing domain, in [31] the authors exploit process mining to get insight from the robot's executions. Indeed, the aim is to assess the use of process mining to discover robots' emergent behavior, and the decision-making process, find repetitive patterns, and identify bottlenecks. Focusing on multi-robot scenarios, in [7] the authors assess the application of process mining to extract robots' missions and perform analysis related to the execution time, resource occupancy, and area usage. In [32], the authors apply process mining techniques to event logs produced in a robotic soccer scenario. Process discovery has been applied to construct a process model of both the team formations and the behavior of each robot. Additionally, decision-mining techniques have been exploited to discover the conditions leading to team formation. In [56], the authors apply process mining to continuously discover process models that serve as a basis for robot learning. The discovered models provide insights into scheduling challenges and inefficiencies, highlighting deviations in average throughput times across different scenarios. This analysis reveals how variations in decision rules influence emergent behaviors, facilitating the identification and understanding of such behaviors within multi-robot systems.

Overall, the presented works identified a common drawback in the application of process mining in robotic systems: the huge amount of generated low-level data. This may lead to obtaining processes that reflect low-level behaviors but do not support effective reasoning about the overall mission execution. Moreover, the difficulty in extracting and manipulating robotic data leads to a shortage of process mining approaches that could be exploited or designed to address the dynamics of robotic scenarios.

Visual analytics & robotic systems. Visual analytics has been exploited in the literature to improve the interpretation and analysis of robotic systems' complex behaviors. In [17], the authors propose an architecture to support stakeholders during the robotic mission design, runtime, and postmortem analysis phases. Specifically, they propose different visualizations that support the reasoning and analysis of the robot's tasks at mission runtime and postmortem. These visualizations primarily target the representation of performed actions, their duration, the used resources, and different metrics and statistics of actions grouped by action type or resource usage. Considering multi-robot scenarios, in [5], the authors address the need to understand activities and decisions made by a multi-robot system where each robot has its own worldview. They propose a visual analytics system designed to aid robotic experts in identifying and analyzing robotic decisions, detecting desynchronization between different worldviews, and performing root-cause analysis to uncover underlying issues. In [57], the authors investigate the performance of robotic swarms in surveillance tasks, focusing on spatial distribution and the impact of communication failures. By leveraging graphical and interactive visualizations, the study provides insights into individual and collective swarm behaviors. Specifically, it highlights the importance of robots' communication patterns, task duration, and spatial visitation rates. In an industrial context, in [58], the authors present an approach that leverages visual analytics to trace anomalies when automatic decision evaluation becomes infeasible due to state space explosion. This method relies on interactively deducing concrete dependencies through expert-based analysis.

Overall, the presented works highlight visual analytics as a promising solution for providing robotic experts with intuitive and easily interpretable visualizations of robotic system behavior, facilitating both real-time monitoring and postmortem analysis. However, the proposed techniques focus on specific aspects of robot behavior, thus overlooking a comprehensive analysis of the overall mission execution.

Process mining & visual analytics. Despite the research fields of process mining and visual analytics having evolved independently for years [12], some studies have explored their integration to enhance systems analysis and interpretation. In [59], the authors propose a multi-perspective approach that combines process models and visualizations to assist users in analyzing processes from different perspectives. The core objective is to compare the execution of a process by considering perspectives related to time, performance, and resource usage. This is achieved by overlaying visualizations of control-flow, time, performance, and resource consumption of various process variants, providing a comprehensive view of the process. In [60], the authors introduce an approach to identify differences between two executions of the same process, addressing the challenges associated with comparing different logs. The proposed solution offers visualization techniques specifically designed to analyze variations in resource reallocation and activity rescheduling between the two logs. These visualizations enable users to easily gain insights into how resources and activities have been managed differently across the two event logs. In [61], authors propose an approach for identifying bottlenecks in business processes by analyzing event logs and visualizing the results. The objective is to integrate visual analytics within process mining to deliver explainable and interpretable insights without exposing users to complex and difficult-to-interpret process models. The primary focus is on visualizing the time dimension, representing activity and trace durations to enhance human cognition and facilitate intuitive analysis. Recently, new frameworks have been proposed to support the integration of process mining techniques with visual analytics. In [62], the authors develop a framework for industrial applications that utilizes historical and real-time production data to predict optimal scheduling for fulfilling new requests while meeting customer time requirements across different simulated scenarios. The proposed solution integrates visual analytics to generate interactive representations of process simulation results, enabling decision-makers to identify key patterns and trends for more effective production planning. Specifically, the framework includes a what-if analysis step, offering visual insights into the potential impact of adjustments to the production plan based on simulated scenarios. Differently, in [63], the authors propose a domain-independent framework that leverages multiple visual representations to capture different informational dimensions of the same processes, tailoring them to the specific needs and background knowledge of process workers and stakeholders. The framework focuses on prioritizing visualization, considering the type of data to be visualized, the analytical tasks the visualization aims to support, and the intended users, and adopting a process-oriented representation.

Overall, integrating interactive visual data analysis with process mining algorithms enhances the comprehensibility of complex information structures and facilitates the discovery of new insights [12]. Many existing approaches focus on creating visualizations based on typical data stored within event logs, such as resource usage or time. Whereas, a few approaches go further by considering additional information and shaping it in the form of a process model, during a preprocessing phase. Therefore, this information is constrained to conform to a process-oriented view, rather than being leveraged alongside the discovered process model representing system behavior to derive additional insights.

8. Discussion

The strength of the proposed methodology lies in the separation between the data preparation and mission analysis phases, with the event log serving as a common point. This structure enhances flexibility, enabling each phase to be extended or adapted without affecting the others. Although the methodology is tailored for robotic experts, it aims to foster the integration of robotics, process mining, and visual analytics approaches. Nonetheless, its application is based on assumptions and is subject to limitations, which are discussed as follows.

Tagging performed activities. The first step of the proposed methodology involves tagging the performed activities, which directly impacts the quality of the event log produced. This step requires decisions on the activity granularity and the type of contextual information stored, which depend on the system under analysis, the analysis goals, and the correctness of the applied tags. Although manual tagging offers flexibility, it is prone to errors and inconsistencies. Therefore, iterative refinement based on the quality of the extracted event log helps to ensure proper logs. This assumption of manually recording robots' activities is aligned with practices also adopted with commercial robots.[5] As an alternative, model-driven approaches, such as the use of behavior trees, can support structured and automated tagging. These approaches facilitate runtime logging of mission execution and provide standardized status information (e.g., success, running, failure), thereby reducing the need for manual intervention and enhancing consistency across logs.

Abstraction of low-level data. Robots typically generate low-level data (e.g., sensor readings, actuator commands), which poses challenges for applying process mining techniques that require high-level activity information. To address this, our methodology assumes that these low-level data are recorded together with high-level event logs, thanks to the tagging step. In cases where this is not feasible, such as with AI-driven systems that do not provide transparent behavior, we suggest that additional techniques such as activity recognition or classification should be employed [64]. However, such cases fall outside the scope of the current methodology.

Event logs of robotic systems. Robotic systems typically generate heterogeneous execution data rather than structured, case-oriented event logs. These data are often continuous, multimodal, and not explicitly associated with a notion of process instance [64]. As a consequence, identifying cases and their starting points can be challenging, similarly to what has been observed in other domains such as with software robots [65]. Our methodology does not assume the availability of a predefined case notion, acknowledging that segmentation depends on both the analysis goal and the specific robotic system under study. When a clear case notion is not available, existing approaches for (semi-)automated segmentation, such as [66–68], can be integrated within the preprocessing step to identify process instances. This issue may arise when missions are executed continuously without explicit resets between runs, or when a single long-running execution spans multiple mission cycles. It may also occur in highly variable environments, where context-driven adaptations blur the identification of mission instances, making it difficult to determine where a case logically starts and ends.

Log timestamp synchronization. Timestamp synchronization is crucial when dealing with distributed data sources like multi-robot systems [69]. In scenarios where clocks are unsynchronized, misalignment of event sequences may occur, leading to inaccurate process discovery results. This challenge could be faced during log processing by detecting and repairing timestamp imperfections [70,71]. In the proposed methodology, we assume that event logs are properly synchronized through hardware or software mechanisms, ensuring data consistency across different robots or system components. Moreover, in our experiments, we avoided synchronization challenges by utilizing a simulator where the clocks of all robots were inherently synchronized.

Process discovery algorithms. The proposed methodology is designed to be flexible, enabling its application across different robotic systems and missions. While the current implementation used DFGs for process discovery, the methodology is not limited to this specific representation. By preparing robotic event logs in the XES standard, the methodology supports extensibility in the discovery phase. This allows exploiting alternative process discovery algorithms and notations [19], depending on the specific requirements of the robotic expert and the objectives of the analysis.

Process mining with contextual perspectives. To enrich the mission analysis with relevant contextual data, we extend the event log format to capture continuous perspectives, such as spatial occupancy and energy consumption. This extension allows for the inclusion of contextual factors that traditional process mining techniques do not account for. While this extension enables the analysis of robotic missions, a major challenge lies in developing general-purpose tools capable of interpreting continuous perspectives in a standardized way. Indeed, without a standard for representing such continuous data within event logs, tool development remains ad hoc and tailored to specific use cases. Differently, having standardized event log representations capable of supporting continuous perspectives would facilitate the incorporation of new analytical dimensions within existing process mining tools and libraries.

Robotic systems variability. From a conceptual perspective, the utility of process mining for analyzing robotic missions benefits both domains. Although robotic missions may be highly variable or even unique, due to autonomy, interaction, and context-dependent decision-making, process discovery techniques can be used to reconstruct and make explicit the actual sequence of actions performed by the robot. Even in a single execution, the observed behavior may differ from the intended or expected one, as actions may be selected at runtime based on contextual conditions. At the same time, from a process mining perspective, the high variability characterizing robotic systems introduces a new dimension of complexity for existing techniques. Aspects like autonomy, interaction, and adaptability lead to executions with strong contextual dependencies, infrequent behaviors, and non-repetitive execution paths, which may challenge traditional process mining assumptions [72]. Rather than limiting the applicability, this variability motivates the extension and adaptation of process mining techniques, positioning robotic systems not only as an application domain but also as a driver for the evolution of process mining towards increasingly adaptive and robotized scenarios.

9. Threats to validity

This section discusses potential threats to the validity of the proposed methodology and its evaluation, and outlines the measures adopted to mitigate them according to [73].

Threats to internal validity. Threats to internal validity relate to the study design and the extent to which the reported findings are supported by the collected data. A potential threat arises from manual activity tagging, which may introduce inconsistencies and affect the quality of the generated event logs and the resulting process models. The qualitative evaluation indicated that, while the technical integration of tags requires limited technical effort, tagging the activities can be challenging in complex, reactive, or overlapping behaviors. This risk is mitigated by encouraging structured and model-driven mission specifications (e.g., behavior trees), as discussed in Section 8.

[5] https://docs.pal-robotics.com/ari/sdk/23.1.13/management/data-logging.html

Threats to construct validity. Threats to construct validity concern whether the methodology accurately captures the intended concepts of robotic mission analysis. The proposed methodology relies on tagging robotic activities, where inappropriate granularity may either hide relevant behavior or lead to too complex models. This threat is mitigated by allowing robotic experts to tailor activity definitions to analysis objectives, thereby supporting the selection of the desired level of granularity. Another threat concerns the representation of continuous contextual data associated with discrete activities. While this association enables compatibility with process mining techniques, it may oversimplify the evolution of contextual perspectives. The use of visual analytics mitigates this risk by enabling these perspectives to be analyzed alongside, rather than embedded within, the process model.

Threats to external validity. Threats to external validity concern the generalizability of the results beyond the evaluated scenario. The experimental evaluation focuses on a single robotic mission scenario, which may limit direct generalization to other missions or domains. This choice was made to demonstrate the feasibility and applicability of the methodology in depth, rather than to provide a broad empirical comparison. We partially mitigate this limitation through a qualitative evaluation with robotic experts, assessing feasibility, interpretability, added value, and practical relevance beyond the specific scenario. Nevertheless, extending the evaluation to multiple robotic missions and real-world deployments is a key direction for future work.

Threats to conclusion validity. Threats to conclusion validity relate to the reliability of the conclusions drawn from the analysis. The methodology relies on human-driven reasoning supported by visual analytics, which introduces a degree of subjectivity. However, the qualitative evaluation showed consistent patterns across participants, i.e., the interpretability and usefulness of the analysis interface, as well as recurring limitations of baseline approaches. Experts also suggested that automated verification or property-based analyses could complement visualizations, indicating a promising direction for strengthening conclusions without replacing human-centered analysis.

10. Conclusions

This paper explores the application of process mining and visual analytics techniques to enhance the analysis of robotic missions. To this end, we proposed a methodology that guides robotic experts in preparing event logs from mission execution. These logs serve as the input for discovering mission execution in the form of process models, enabling the analysis of executed activities from a control-flow perspective and their enhancement with visualizations describing contextual factors that may have influenced mission outcomes. The methodology has been implemented to facilitate both event log preparation and mission analysis. The latter has led to the development of the RoboTrace tool, which supports robotic experts in mission reasoning by automatically discovering process models and enhancing them with spatial occupancy, energy consumption, and communication perspectives. The proposed approach has been assessed in a smart agriculture multi-robot scenario, through an experiment conducted in a simulated environment, showing its applicability and potential through a step-by-step analysis of robotic mission execution.

As future work, we plan to validate the effectiveness of process mining and visual analytics in multiple real-world scenarios by applying the proposed methodology to commercial robotic systems. This will provide empirical evidence of its utility in multiple settings and support the assessment of its generalizability beyond a single scenario, while also enabling the creation of reusable robotic datasets. Additionally, as a need highlighted by the qualitative evaluation, we aim to provide conceptual guidelines to support robotic experts in selecting suitable activity abstraction levels during the tagging phase. Finally, we aim to complement the methodology with automated verification mechanisms, such as checking formal properties associated with activities and contextual perspectives, to strengthen the analysis by supporting both explainability and correctness assessment.

CRediT authorship contribution statement

Sara Pettinari: Writing – review & editing, Writing – original draft, Validation, Software, Resources, Methodology, Investigation, Conceptualization. **Barbara Re:** Writing – review & editing, Methodology, Investigation, Funding acquisition, Conceptualization. **Lorenzo Rossi:** Writing – review & editing, Methodology, Investigation, Conceptualization. **Francesco Tiezzi:** Writing – review & editing, Methodology, Investigation, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been partially funded by (a) the MUR (Italy) Department of Excellence 2023–2027 for GSSI; (b) the Italian Ministry of University and Research (MUR), Department of Excellence projects 2023–2027 ReDS ‘Rethinking Data Science’ - Department of Statistics, Computer Science, Applications - University of Florence.

Data availability

Data will be made available on request.

References

[1] G.-Z. Yang, J. Bellingham, P.E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, R. Merrifield, et al., The grand challenges of science robotics, Sci. Robot. 3 (14) (2018) eaar7650.
[2] M. Luckcuck, M. Farrell, L.A. Dennis, C. Dixon, M. Fisher, Formal specification and verification of autonomous robotic systems: A survey, ACM Comput. Surv. 52 (5) (2019) 100:1–100:41.
[3] F. Vicentini, M. Askarpour, M.G. Rossi, D. Mandrioli, Safety assessment of collaborative robotics through automated formal verification, IEEE Trans. Robot. 36 (1) (2019) 42–61.
[4] A. Afzal, C.L. Goues, M. Hilton, C.S. Timperley, A study on challenges of testing robotic systems, in: International Conference on Software Testing, Validation and Verification, IEEE, 2020, pp. 96–107.
[5] S.S. Bae, F. Rossi, J.V. Hook, S. Davidoff, K. Ma, A visual analytics approach to debugging cooperative, autonomous multi-robot systems’ worldviews, in: Conference on Visual Analytics Science and Technology, IEEE, 2020, pp. 24–35.
[6] W. van der Aalst, Process Mining - Data Science in Action, Second Edition, Springer, 2016.
[7] J.J. Roldán, M.A. Olivares-Méndez, J. del Cerro, A. Barrientos, Analyzing and improving multi-robot missions by using process mining, Auton. Robots 42 (6) (2018) 1187–1205.
[8] F. Corradini, S. Pettinari, B. Re, L. Rossi, F. Tiezzi, A methodology for the analysis of robotic systems via process mining, in: Enterprise Design, Operations, and Computing, in: LNCS, vol. 14367, Springer, 2023, pp. 117–133.
[9] A. Koschmider, M. Aleknonyte-Resch, F. Fonger, C. Imenkamp, A. Lepsien, K. Apaydin, D. Janssen, D. Langhammer, T. Ziolkowski, Y. Zisgen, Process mining for unstructured data: Challenges and research directions, in: Modellierung, in: LNI, vol. P-348, Gesellschaft für Informatik e.V., 2024, pp. 119–136.
[10] C. Klinkmüller, R. Müller, I. Weber, Mining process mining practices: An exploratory characterization of information needs in process analytics, in: Business Process Management, in: LNCS, vol. 11675, Springer, 2019, pp. 322–337.
[11] W. van der Aalst, A. Adriansyah, A.K.A. de Medeiros, et al., Process mining manifesto, in: Business Process Management Workshops, in: LNBIP, vol. 99, Springer, 2011, pp. 169–194.
[12] S. Miksch, C. Di Ciccio, P. Soffer, B. Weber, Visual analytics meets process mining: Challenges and opportunities, IEEE Comput. Graph. Appl. 44 (6) (2024) 132–141.
[13] R. Caldas, J.A.P. García, M. Schiopu, P. Pelliccione, G. Rodrigues, T. Berger, Runtime verification and field-based testing for ROS-based robotic systems, IEEE Trans. Softw. Eng. 50 (10) (2024) 2544–2567.
[14] S. Wachter, B. Mittelstadt, L. Floridi, Transparent, explainable, and accountable AI for robotics, Sci. Robot. 2 (6) (2017) eaan6080.
[15] M. Autili, M. De Sanctis, P. Inverardi, P. Pelliccione, Engineering digital systems for humanity: a research roadmap, ACM Trans. Softw. Eng. Methodol. (2025).

[16] F.J. Rodríguez-Lera, Á.M. Guerrero-Higueras, F.M. Rico, J. Gines, J.F.G. Sierra, V.M. Olivera, Adapting ROS logs to facilitate transparency and accountability in service robotics, in: Robot 2019: Fourth Iberian Robotics Conference, in: Advances in Intelligent Systems and Computing, vol. 1093, Springer, 2019, pp. 587–598.

[17] S.G. Brunner, P. Lehner, M.J. Schuster, S. Riedel, R. Belder, D. Leidner, A. Wedler, M. Beetz, F. Stulp, Design, execution, and postmortem analysis of prolonged autonomous robot operations, Robot. Autom. Lett. 3 (2) (2018) 1056–1063.

[18] W. van der Aalst, Process mining: a 360 degree overview, in: Process Mining Handbook, in: LNBIP, vol. 448, 2022, pp. 3–34.

[19] W. van der Aalst, Foundations of process discovery, in: Process Mining Handbook, in: LNBIP, vol. 448, Springer, 2022, pp. 37–75.

[20] J. Carmona, B. van Dongen, M. Weidlich, Conformance checking: foundations, milestones and challenges, in: Process Mining Handbook, in: LNBIP, vol. 448, Springer, 2022, pp. 155–190.

[21] M. de Leoni, Foundations of process enhancement, in: Process Mining Handbook, in: LNBIP, Springer, 2022, pp. 243–273.

[22] K.A. Cook, J.J. Thomas, Illuminating the Path: The Research and Development Agenda for Visual Analytics, IEEE Computer Society, 2005.

[23] W. Cui, Visual analytics: A comprehensive overview, IEEE Access 7 (2019) 81555–81573.

[24] A. Kirk, Data Visualisation: A Handbook for Data Driven Design, SAGE Publications Ltd, 2019.

[25] K. Bourr, F. Corradini, S. Pettinari, B. Re, L. Rossi, F. Tiezzi, Disciplined use of BPMN for mission modeling of multi-robot systems, in: Forum At Practice of Enterprise Modeling, vol. 3045, CEUR-WS.org, 2021, pp. 1–10.

[26] F. Corradini, S. Pettinari, B. Re, L. Rossi, F. Tiezzi, A BPMN-driven framework for multi-robot system development, Robot. Auton. Syst. 160 (2023) 104322.

[27] K. Otsu, S. Tepsuporn, R. Thakker, T.S. Vaquero, J.A. Edlund, W. Walsh, G. Miles, T. Heywood, M.T. Wolf, A.-A. Agha-Mohammadi, Supervised autonomy for communication-degraded subterranean exploration by a robot team, in: Aerospace Conference, IEEE, 2020, pp. 1–9.

[28] Y. Bertrand, J.D. Weerdt, E. Serral, A bridging model for process mining and IoT, in: Process Mining Workshops, in: LNBIP, vol. 433, Springer, 2021, pp. 98–110.

[29] RobMoSys Project, Wiki, 2019, URL https://www.robmosys.eu/wiki/.

[30] S. García, D. Strüber, D. Brugali, A. Di Fava, P. Pelliccione, T. Berger, Software variability in service robotics, Empir. Softw. Eng. 28 (1) (2023) 24.

[31] J.-F. Jimenez, G. Zambrano-Rey, S. Aguirre, D. Trentesaux, Using process-mining for understating the emergence of self-organizing manufacturing systems, IFAC-PapersOnLine 51 (11) (2018) 1618–1623.

[32] A. Rozinat, S. Zickler, M. Veloso, W. van der Aalst, C. McMillen, Analyzing multi-agent activity logs using process mining techniques, in: DARS, Springer, 2008, pp. 251–260.

[33] M. Quann, L.V. Ojeda, W. Smith, D.M. Rizzo, M.P. Castanier, K. Barton, An energy-efficient method for multi-robot reconnaissance in an unknown environment, in: American Control Conference, IEEE, 2017, pp. 2279–2284.

[34] Í.R. da Costa Barros, T.P. do Nascimento, Robotic mobile fulfillment systems: A survey on recent developments and research opportunities, Robot. Auton. Syst. 137 (2021) 103729.

[35] M.A. Hossen, S. Kharade, B.R. Schmerl, J. Cámara, J.M. O'Kane, E.C. Czaplinski, K.A. Dzurilla, D. Garlan, P. Jamshidi, CaRE: Finding root causes of configuration issues in highly-configurable robots, IEEE Robot. Autom. Lett. 8 (7) (2023) 4115–4122.

[36] J.K. Verma, V. Ranga, Multi-robot coordination analysis, taxonomy, challenges and future scope, J. Intell. Robot. Syst. 102 (1) (2021) 10.

[37] A. Benkrid, A. Benallegue, N. Achour, Multi-robot coordination for energy-efficient exploration, J. Control. Autom. Electr. Syst. 30 (6) (2019) 911–920.

[38] F. Corradini, S. Pettinari, B. Re, L. Rossi, F. Tiezzi, A technique for discovering BPMN collaboration diagrams, Softw. Syst. Model. (2024) 1–21.

[39] S.J. van Zelst, F. Mannhardt, M. de Leoni, A. Koschmider, Event abstraction in process mining: literature review and taxonomy, Granul. Comput. 6 (3) (2021) 719–736.

[40] M. zur Muehlen, K.D. Swenson, BPAF: A standard for the interchange of process analytics data, in: Business Process Management Workshops, in: LNBIP, vol. 66, Springer, 2010, pp. 170–181.

[41] I. Afyouni, C. Ray, C. Claramunt, Spatial models for context-aware indoor navigation systems: A survey, J. Spat. Inf. Sci. 4 (2012) 85–123.

[42] D. Lee, Resource-based task allocation for multi-robot systems, Robot. Auton. Syst. 103 (2018) 151–161.

[43] P. Mazdin, B. Rinner, Distributed and communication-aware coalition formation and task assignment in multi-robot systems, IEEE Access 9 (2021) 35088–35100.

[44] A. Koubaa, et al., Robot Operating System (ROS), vol. 1, Springer, 2017.

[45] S. Pettinari, B. Re, L. Rossi, F. Tiezzi, Enhancing robotic mission analysis via process mining and visual analytics (replication package), 2025, https://github.com/SaraPettinari/RoboPM.

[46] R.M. Van Eijk, F.S. De Boer, W. Van Der Hoek, J.-J.C. Meyer, Process algebra for agent communication: A general semantic approach, Commun. Multiagent Syst.: Agent Commun. Lang. Conversat. Policies (2003) 113–128.

[47] ACM SIGSOFT, Empirical standards for software engineering, 2020, https://github.com/acmsigsoft/EmpiricalStandards/blob/master/docs/standards/QualitativeSurveys.md.

[48] S.J. Stratton, Population research: convenience sampling strategies, Prehospital Disaster Med. 36 (4) (2021) 373–374.

[49] D.I.K. Sjøberg, J.E. Hannay, O. Hansen, V.B. Kampenes, A. Karahasanovic, N. Liborg, A.C. Rekdal, A survey of controlled experiments in software engineering, IEEE Trans. Softw. Eng. 31 (9) (2005) 733–753.

[50] B. Amir, P. Ralph, There is no random sampling in software engineering research, in: Proceedings of the 40th International Conference on Software Engineering Companion Proceeedings, ACM, 2018, pp. 344–345.

[51] S. Baltes, P. Ralph, Sampling in software engineering research: a critical review and guidelines, Empir. Softw. Eng. 27 (4) (2022) 94.

[52] A. Bangor, P.T. Kortum, J.T. Miller, An empirical evaluation of the system usability scale, Intl. J. Human–Computer Interact. 24 (6) (2008) 574–594.

[53] P. Vlachogianni, N. Tselios, Perceived usability evaluation of educational technology using the system usability scale (SUS): A systematic review, J. Res. Technol. Educ. 54 (3) (2022) 392–409.

[54] S.G. Hart, NASA-task load index (NASA-TLX); 20 years later, in: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 50, Sage publications, 2006, pp. 904–908.

[55] T.C. Nicoleta, Process mining on a robotic mechanism, in: International Conference on Software Testing, Validation and Verification Workshops, IEEE, 2021, pp. 205–212.

[56] R.H. Bemthuis, M. Koot, M.R.K. Mes, F.A. Bukhsh, M. Iacob, N. Meratnia, An agent-based process mining architecture for emergent behavior analysis, in: International Enterprise Distributed Object Computing, IEEE, 2019, pp. 54–64.

[57] C.D.G. Linhares, C.R. Tinoco, J.R. Ponciano, G.M.B. Oliveira, B.A.N. Travençolo, A network-based visual analytics approach for performance evaluation of swarms of robots in the surveillance task, in: Intelligent Systems, in: LNCS, vol. 13653, Springer, 2022, pp. 61–76.

[58] M. Kiermeier, S. Feld, Visual analytics for root cause analysis in self-organizing industrial systems, in: International Conference on Industrial Informatics, IEEE, 2018, pp. 315–320.

[59] A. Pini, R. Brown, M.T. Wynn, Process visualization techniques for multi-perspective process comparisons, in: J. Bae, S. Suriadi, L. Wen (Eds.), Asia Pacific Business Process Management, in: LNBIP, vol. 219, Springer, 2015, pp. 183–197.

[60] W.Z. Low, W.M.P. van der Aalst, A.H.M. ter Hofstede, M.T. Wynn, J.D. Weerdt, Change visualisation: Analysing the resource and timing differences between two event logs, Inf. Syst. 65 (2017) 106–123.

[61] A. Kaouni, G. Theodoropoulou, A. Bousdekis, A. Voulodimos, G. Miaoulis, Visual analytics in process mining for supporting business process improvement, in: Novelties in Intelligent Digital Systems, in: Frontiers in Artificial Intelligence and Applications, vol. 338, IOS Press, 2021, pp. 166–175.

[62] S. Agostinelli, D. Benvenuti, A. Casciani, F.D. Luzi, M. Marinacci, A. Marrella, J. Rossi, A context-aware framework to support decision-making in production planning, in: Advanced Information Systems Engineering, in: LNCS, vol. 14663, Springer, 2024, pp. 248–264.

[63] A. Alman, A. Arleo, I. Beerepoot, A. Burattin, C. Di Ciccio, M. Resinas, Tiramisù: making sense of multi-faceted process information through time and space, J. Intell. Inf. Syst. (2024) 1–23.

[64] F. Corradini, S. Pettinari, B. Re, L. Rossi, M. Sampaolo, Enabling process mining on multimodal robotic data, in: Advanced Information Systems Engineering Workshops, in: LNBIP, Springer, 2025, pp. 257–269.

[65] S. Agostinelli, A. Marrella, M. Mecella, Exploring the challenge of automated segmentation in robotic process automation, in: Research Challenges in Information Science, in: LNBIP, vol. 415, Springer, 2021, pp. 38–54.

[66] C.W. Günther, A. Rozinat, W.M.P. van der Aalst, Activity mining by global trace segmentation, in: Business Process Management Workshops, in: LNBIP, vol. 43, Springer, 2009, pp. 128–139.

[67] L. Esposito, F. Leotta, M. Mecella, S. Veneruso, Unsupervised segmentation of smart home logs for human habit discovery, in: International Conference on Intelligent Environments, IEEE, 2022, pp. 1–8.

[68] S. Agostinelli, F. Leotta, A. Marrella, Interactive segmentation of user interface logs, in: Service-Oriented Computing, in: LNCS, vol. 13121, Springer, 2021, pp. 65–80.

[69] M. Alsayegh, A. Dutta, P. Vanegas, L. Bobadilla, Lightweight multi-robot communication protocols for information synchronization, in: International Conference on Intelligent Robots and Systems, IEEE, 2020, pp. 11831–11837.

[70] P.M. Dixit, S. Suriadi, R. Andrews, M.T. Wynn, A.H.M. ter Hofstede, J.C.A.M. Buijs, W.M.P. van der Aalst, Detection and interactive repair of event ordering imperfection in process logs, in: Advanced Information Systems Engineering, in: LNCS, vol. 10816, Springer, 2018, pp. 274–290.

[71] D.A. Fischer, K. Goel, R. Andrews, C.G.J. van Dun, M.T. Wynn, M. Röglinger, Enhancing event log quality: Detecting and quantifying timestamp imperfections, in: Business Process Management, in: LNCS, vol. 12168, Springer, 2020, pp. 309–326.

[72] F. Corradini, S. Pettinari, B. Re, L. Rossi, F. Tiezzi, Process mining meets robotic systems: Challenges and opportunities, in: Internet of Things Meets Business Process Management: A Synergistic Integration, Springer Nature, 2026, pp. 203–217.

[73] C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, A. Wesslén, Experimentation in Software Engineering, Springer, 2012.