Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 192 (2021) 3966–3978

Procedia
Computer Science

www.elsevier.com/locate/procedia

International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2021, 8-10 September 2021, Szczecin, Poland

Robust Reasoning for Autonomous Cyber-Physical Systems in Dynamic Environments

Anne Håkansson[a,b], Aya Saad[b,*], Akhil Anand[b], Vilde Gjærum[b], Haakon Robinson[b], Katrine Seel[b]

[a] *Department of Computer Science, The Arctic University of Norway (UiT), Tromsø Norway*
[b] *Department of Engineering Cybernetics, The Norwegian University of Science and Technology (NTNU), Trondheim, Norway*

Abstract

Autonomous cyber-physical systems, CPS, in dynamic environments must work impeccably. The cyber-physical systems must handle tasks consistently and trustworthily, i.e., with a robust behavior. Robust systems, in general, require making valid and solid decisions using one or a combination of robust reasoning strategies, algorithms, and robustness analysis. However, in dynamic environments, data can be incomplete, skewed, contradictory, and redundant impacting the reasoning. Basing decisions on these data can lead to inconsistent, irrational, and unreasonable cyber-physical systems' movements, adversely impacting the system's reliability and integrity.

This paper presents the assessment of robust reasoning for autonomous cyber-physical systems in dynamic environments. In this work, robust reasoning is considered as 1) the capability of drawing conclusions with available data by applying classical and non-classical reasoning strategies and algorithms and 2) act and react robustly and safely in dynamic environments by employing robustness analysis to provide options on possible actions and evaluate alternative decisions. The result of the research shows that different common existing strategies, algorithms and analyses can be provided together with a comparison of their applicabilities, benefits, and drawbacks in the context of cyber-physical systems operating in dynamically changing environments. The conclusion is that robust reasoning in cyber-physical systems can handle dynamic environments. Moreover, combining these strategies and algorithms with robustness analysis can support achieving robust behavior in autonomous cyber-physical systems while operating in dynamically changing environments.

Keywords: Robust reasoning; robustness analysis; autonomous cyber-physical systems (CPS); dynamic environment

[*] Corresponding author.
E-mail address: aya.saad@ntnu.no

10.1016/j.procs.2021.09.171

1. Introduction

Autonomous cyber-physical systems (CPS) are systems capable of making decisions and operating independently. CPS intertwine computations, physical processes, and networking, acting on environment data and managing actuators [12]. The computational resources and physical systems interact over networks, and computation devices and physical processes affecting each other via feedback loops. Hence, the physical actions of the CPS affect the computations and vice versa autonomously. CPS are individual agents, each characterized by being goal-oriented and self-directed. In the case of autonomous agents, these CPS can be individual movable systems that interact with other objects and systems in the environment, carrying out tasks that might be less informative. For example, fully autonomous vehicles, like water vessels navigating and operating near-shore areas or inland waterways, self-driving cars operating in heavy traffic, as well as self-flying drones must at least partly handle static and dynamic environments.

In dynamic environments, an autonomous system is said to be robust when it has predictable behavior while making decisions and stable operations when performing complex tasks [16]. Still, the robustness of the system's behavior remains a distant achievement due to unexpected situations. A robust system should withstand a certain amount of disruptions without interrupting the execution of the tasks or negatively affecting the performance. In a dynamically changing environment, the collected data typically suffer from being incomplete, uncertain, imperfect, erroneous, or contain noise, as well as being contradictory and redundant. These data can lead to inconsistent or irrational decisions directing CPS towards unexpected movements, causing them to be unreliable and behave unreasonably, leading in some cases to loss of integrity and dangerous situations in the environment. For autonomous CPS to behave robustly in dynamic environments, they must draw valid conclusions by applying decision-making on the data at hand and acting accordingly. Thus, CPS require robust reasoning by utilizing one or a combination of the following: 1) reasoning strategies or algorithms 2) robustness analysis for evaluating the system's behavior.

This paper presents classical and non-classical reasoning strategies and algorithms and robustness analysis and their applicability to achieve robust autonomous CPS operating in dynamic environments. Here we use the term *classical reasoning strategies (classical RS)* for resolution-based systems applied in CPS [15]. Conversely, the term *non-classical reasoning strategies and algorithms (non-classical RSA)* refer to strategies utilizing machine learning (ML) algorithms and stream reasoning. ML algorithms, such as reinforcement learning, model and reason with temporal logic and spatial data [21]. These ML algorithms often handle reasoning by applying classical RS, like inductive reasoning and case-based reasoning with rewards, in the model. Stream reasoning continuously handles data streams observed by CPS. Stream reasoning works with temporal and spatial logic to analyze performances of CPS and find deviations, to recover from failures and other problems [10]. *Robustness analysis (RA)* has a two-fold application: 1) ML algorithms use RA on images and videos for pattern-matching and decision-making; 2) RA uses ML and fuzzy logic and model-based reasoning to analyze decisions and measure the proposed actions in a step-wise manner. In this case, RA measures the distinctions between the initial decision, i.e., commitment and acceptable options, and evaluates the implications of those decisions on the system's stability without destroying its flexibility. The result of the research, in this paper, concludes that it is feasible to apply existing strategies and analyses to CPS. The study of robust reasoning shows that the autonomous CPS can pursue robust decision-making (RDM) in dynamic environments by proposing a sequence of action options that are stable and flexible to the changing nature of the environment. The aim is to identify and provide a combination of classical RS and non-classical RSA and robustness analysis to achieve robust behavior in autonomous CPS operating in dynamically changing environments. New derived options, which represent adaptive decision strategies, are evolving, over time, in response to new information. By continuously drawing conclusions, a robust CPS can consider the overall system behavior while tolerating errors in input data.

2. Related Work

An early attempt to perform robust reasoning where computer systems incorporated classical reasoning strategies, such as rule-based reasoning, similarity-based reasoning and commonsense reasoning [32]. A two-level architecture is used to find common patterns in commonsense reasoning to provide basic patterns and show characteristics of these patterns. These patterns shall handle: 1) partial information, i.e., conclusions must be drawn even without knowing all relevant information, 2) uncertain and fuzzy information without absolute certainty in the information, 3) lack of matching rules, and 4) lack of consistency and completeness due to fragmented rules and inheritances among rules. A more recent attempt is robust stream reasoning with constant data streams like video streams, social media data streams, and industrial data streams with autonomous robotic systems. For example, robust stream reasoning can combine temporal reasoning with qualitative spatial reasoning to draw conclusions on data that is provided with uncertainty on streams [10]. For example, De Leng [10] explores an adaptive reconfiguration procedure that robustly manages the data in streams. This adaptive procedure performs spatial-temporal stream reasoning to reason with inter-temporal and spatial relations.

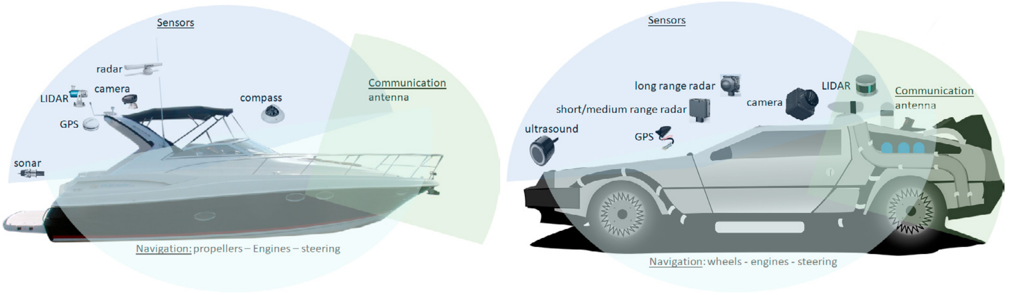
Another important branch of robust systems is robustness analysis. The robustness analysis for CPS focuses on engineering design approaches with methods that guarantee robust performance and trustworthiness even when the underlying assumptions are falsified. Depending on the application under consideration, the methods can outline appropriate metrics that verify the systems' input-output stability against uncertainty sources [34, 26]. Previous work on robustness analysis for ML models varies between 1) adding robustness verification approaches to adversarial training [36, 37] and 2) constructing a robust ensemble classifier through classifier-diversification [41]. One paper introduces a notion of prediction consistency where neural networks jointly learn the predictions and their reasoning labels [24]. The neural network predictions are accepted when the predicted class is mapped with the predicted reasoning labels while following a pre-specified mapping function. Another paper explores the possibility of using joint inference mechanisms as an extension to the traditional machine learning classifier model [39]. In this work, a full robustness analysis framework is implemented. The framework is carried out in two steps: the first applies ensemble classifiers, and the second provides probabilistic guarantees robustness for two types of reasoning components: 1) Markov logic networks and 2) Bayesian networks. In addition, a highly interesting research to handle robust reasoning in CPS has been combining the different classical reasoning strategies to provide reasoning [32]. Another research is using stream reasoning with ML algorithms that can handle episodes in sequential events [10]. Still, robust reasoning for robust behavior in the environment using reasoning strategies has been given little attention in the literature.

To the best of our knowledge, there is no other paper presenting robust reasoning with a spectrum of reasoning strategies and algorithms and robustness analysis to achieve robust autonomous CPS in dynamic environments. The work presented in this paper focuses on robust reasoning with the most common reasoning strategies and algorithms that can be applied in CPS but also the most appropriate strategy or algorithm on the available data. The robustness analysis in CPS commonly focuses on the analysis and design of the systems, as well as testing and evaluating to verify robust behavior. However, robustness analysis also includes measuring the initial option expectation (action) with possible options inferred by the system evaluating the compatibility between the expectation and options and the acceptability of these options, which is shown in this paper. An interesting solution for autonomous CPS in dynamic environments is a combination of robust reasoning strategies and algorithms with robustness analysis since it can provide two-step uncertainty verification (authentication) and provide robust behavior in the environment.

3. Autonomous Cyber-Physical Systems and reasoning in dynamic environments

Autonomous Cyber-Physical Systems operate on a combination of continuous computations and physical performances independently of interventions of human beings. These CPS have various levels of complexity and can be anything from a small self-management device to a fully autonomous vehicle operating in surrounding environments, communicating with other CPS. Viewing the CPS as a fully autonomous mobile vehicle implies several tightly and seamlessly integrated components (hardware and software) and networks for moving around and performing tasks. The CPS are *cyber* in every *physical component*. The cyber includes computation, communication, and control, whereas physical is manufactured and continuous operating sensors and devices. These must work synchronously to manage the mobile CPS in static and dynamic environments. These CPS are closed-loop systems that take the sensing data into account and adapt the control laws offering justifiable actuation and control. Autonomous mobile vehicles include sensors, devices, and other actuators for different purposes. For example, there are sensors and devices, systems and cameras for handling measurements concerning the vehicle, including electrical power and capacity, and surrounding environment with motions and movement directions. There are also actuators for navigation systems and maneuvering systems, such as steering and speed controls. With the computing data perceived in the environment, the CPS can maneuver the entire vehicle autonomously and communicate with other CPS.

Many of the autonomous CPS have several commonly utilized hardware components, which besides cameras and sensors, often are radars, LIDAR systems, GPS, sonar and ultrasound devices, and antennas. See Fig. 1 for typical



(a) (b)

Fig. 1. List of typical hardware components for (a) autonomous ship (b) autonomous car.

hardware components for (a) autonomous ship and (b) autonomous car. With these components, the autonomous CPS can maneuver the autonomous vehicles by interacting, extracting data, and deriving conclusions. These conclusions become action options. The current CPS are goal-oriented and self-directed software that deals with the available data and other software systems' processes. The CPS are multi-tasking and handle, for example, distance keeping, path tracing, and object tracking, in parallel. However, the robust CPS need to handle every possible, sometimes unexpected, situation in the dynamic environment. These situations require robust behavior with completely reliable decision-making of fully autonomous CPS. These decisions must be valid and sound in the sense that the CPS can act on the decision and behave in a robust and trustworthy manner. The mobile autonomous CPS must dynamically reorganize and reconfigure components in real-time. This reconfiguration requires synchronizing the input data from the different integrated systems and orchestrating the computations of interrelated parts to autonomously making decisions according to the given task. These decisions are communicated to all computational and physical components, i.e., actuators, and other sensors and devices, to act upon. From the actions, the physical components pick up new input data on which the CPS need to make decision and send back to the components. The process is a continuous loop of actions and reactions.

Depending on the data from the hardware components, different kinds of reasoning strategies can be applied. In the case of structured, straightforward raw sensor data, robust reasoning strategies or algorithms can support decision-making but, when handling images and videos, robustness analysis is required due to the nature of the data. This paper refers to robust reasoning as reasoning strategies or algorithms to make robust decisions and analysis to measure the validity of these decisions and evaluate their implications on the system's overall performance. The reasoner may benefit from a combination of robust reasoning strategies, algorithms, and robustness analysis evaluations to derive valid conclusions from the available data in dynamically changing environments. Reasoning strategies and algorithms for robust system behavior and robustness analysis and their applicability, benefits, and drawbacks in CPS are the key points in this paper.

4. Robust Reasoning Strategies for Robust Cyber-Physical Systems

Reasoning strategies refer to a wide range of implemented classical and non-classical reasoning techniques that allow intelligent systems to make decisions. It started as outlining a systematic decision framework [2] to then dealing with the dynamic nature of the environment. An overview of classical (classical RS) and non-classical reasoning strategies and algorithms (non-classical RSA), how they apply robustness in CPS, their benefits and drawbacks are presented in this section and summarized in Table 1 and Table 2 in the attached appendix.

4.1. Classical reasoning strategies for Robust Cyber-Physical systems

For robust cyber-physical systems, the classical RS are used in different kinds of decision-support systems. This paper focuses on deductive reasoning, inductive reasoning, abductive reasoning, case-based reasoning, monotonic and non-monotonic reasoning, and probabilistic reasoning, as classical strategies for making robust decisions in CPS.

Deductive reasoning is a top-down approach that preserves the truth while asserting rules or logical relations on data and decisions in CPS [15]. It is an intuitive representation of the CPS system's behavior [33]. It has been used to outline goal-directed and highly reliable "closed-loop commanding" autonomous agents that operate over long periods [23]. Temporal constraint-based planning and scheduling are the tools utilized to represent and reason about time [30]. With deductive reasoning, conclusions are deduced through formal specifications that exert correct and sound inference. This allows the CPS to make robust and more trustworthy decisions and draw valid conclusions. For instance, an autonomous vessel or an autonomous car, depicted in Fig. 1, uses deductive reasoning to decide the shortest path between the current position and the destination, which is provided a global picture of the current system and its destination GPS positions. The main drawback of deductive reasoning is that detailed logical relations between the different concepts stored in the knowledge base must be hardcoded to be reliable and predefined thoroughly within the system. Applying deductive reasoning on dynamic knowledge bases has been done [15], but this is a complicated task since it can lead to problems within the knowledge base. Sometimes is impossible to achieve soundness, especially within a complex, distributed, and dynamic environment [18]. In addition, deductive reasoning strategies are susceptible to noise, which might yield inconsistent system states [11].

Inductive reasoning is a bottom-up approach that, from a limited set of observations or premises, deduces general conclusions [15]. Machine learning approaches that learn to exploit statistical patterns in datasets are considered as a type of inductive reasoning strategy[42]. CPS can use inductive reasoning to predict the uncertain behavior or changes in their surrounding [33]. Hence, CPS can deduce general conclusions with a level of certainty from available knowledge about the environment. For instance, an autonomous vessel or car can use inductive reasoning to conclude the system's position and its state as a consequence of accelerating the speed after a certain amount of time. This happens when the CPS are aware of the current systems' state observed by the sensors that confirm that no obstacles are along the way and that the system (vessel or car) is moving forward. The main drawback of using inductive reasoning in CPS is when many facts, critical to the inference are missing, causing high uncertainty and, thereby, wrong conclusions. A unified framework for agents that deploy both deductive and inductive reasoning has been employed by several different researchers [7, 42, 33] to ensure system soundness. Soundness implies that a conclusion, provable in the deductive system, is true for all interpretations or structures on which the conclusion is based.

Abductive reasoning is utilized when knowledge about the problem is incomplete. The CPS generate reasonable hypotheses based on assumptions and infer information that is generally plausible [9, 15]. Abductive learning can be regarded as a special kind of weakly supervised learning where the supervision information comes from knowledge reasoning [42]. In a distributed agent-based environment, each agent arranges its local knowledge base and uses abductive logic to reason about distributed and incomplete knowledge among the other agents in the environment. The agent's global knowledge base, in turn, is built upon agents' agreed hypotheses [9, 31]. For instance, the LIDAR observes a moving hull shape in front of the surface vessel or a moving object with wheels in front of the car and reports that this object is most likely another moving vessel or car. This information can further be passed to the system in order to decelerate its current speed such that the distance for collision avoidance is kept safe during navigation. The main drawback of abductive reasoning is when a high degree of inconsistency arises from deduced hypotheses. This inconsistency may lead to contradictory results and wrong conclusions, especially when different system agents provide these hypotheses.

Case-based reasoning provides solutions to problems by analyzing similarities with other problems. They use analogical reasoning to infer solutions based on case histories [3]. There are four common steps in case-based reasoning for CPS: retrieve, reuse, revise and retain. Retrieve is about retrieving cases from the knowledge base that can solve the problem. Reusing is mapping the solutions of these previous cases to the target problem. Revise is about testing the solution in the real world and, if necessary, revising the case. Retain is about storing the successfully applied solutions as a new case in the memory. For instance, an autonomous vessel or car can apply case-based reasoning on a delicate situation like maneuver from collision by using solutions to the situation that has occurred earlier. A primary benefit of case-based reasoning is that it builds up evolving experiences that can be exchanged between different CPS systems. However, the main drawback of case-based reasoning is due to the fact that it depends on previous experiences that might lack similar cases or fail to match them.

In monotonic reasoning, the entailment and facts are freely derived and extended with additional assumptions. When a conclusion is drawn, it will remain the same even when other information is added to the existing knowledge in the knowledge base. On the other hand, non-monotonic reasoning is based on revealing dynamic truth values. In non-monotonic reasoning the conclusions are tentative, meaning that the inference can be revised [15]. The main drawback of monotonic reasoning is that inferences are likely to change and cannot be handled by the reasoner. The main drawback of non-monotonic reasoning is that there are cases when the inference should remain unchanged in the knowledge base to be able to build new knowledge, like hard facts that will not or shall not change. Also, the system can reach conflicting inferences, i.e., conclusions, and steps shall be taken to preserve or restore consistency. Monotonic reasoning in CPS has static truth values during the inference. Such inferred values are easy to follow and formally prove. However, this fact contradicts the environment's dynamic property. Dynamically revealing new values for the truth through non-monotonic reasoning opens up for incorrect conclusions drawn by the CPS.

Probabilistic reasoning is based on dynamic approaches for reasoning. To handle the uncertainty, the representation of the information in these approaches is qualitative rather than having precise and discrete numerical formulations [23]. CPS, like autonomous vessel or car, use probabilistic reasoning to draw conclusions when the data presented is coupled with uncertainty or when the environment is dynamically changing. The qualitative information representation, in this case, is described through probabilistic models or possibilistic models. A tool for estimating probability distributions of potential outcomes is stochastic models. Stochastic models allow random variation in one or more inputs over time and, hence, have the highest computation complexity as they reason about the data in a pointwise manner [1]. Probabilistic models build the nearest probability distribution representation of the data [4] whereas possibilistic models build on possibility measures by treating fuzzy numbers as possibility distribution. Fuzzy and convex models belonging to the possibilistic world have less computational complexity. The former [8] builds the nearest fuzzy membership function [40] that fits the data whereabouts, while in convex models [28, 27], the data is represented as intervals. Reasoning, in all of the cases mentioned above, applies formal logic to the model representations in order to draw inferences on the acquired knowledge. Inferred conclusions are coupled with weights indicating the degree of uncertainty or showing how likely the solution is close to the truth value. The benefit is that the system ensures that the qualitative constructs of information are reliable when covering all possible cases. However, the coupling of conclusions and weights can sometimes be disadvantageous when it is unclear for the end-user to define the solution's true value.

4.2. Non-classical reasoning strategies and algorithms for Robust Cyber-Physical systems

The non-classical RSA for robust decisions in CPS, in this paper, are machine learning algorithms and stream reasoning. They are defined as non-classical because they are based on a combination of classical reasoning strategies. However, they do not carry out resolution-based reasoning, thus inheriting the properties from the foundational classical strategies.

Machine learning approaches can be excellent tools for adopting inductive reasoning strategy for data-mining agent applications. ML schemes that predict the environment and future system states adopt regression models or hidden Markov models. CPS interpret the dynamics of the physical environment at a set of sequential states. Temporal semantics need to be introduced to explain the physical behavior over time [16]. The main drawback of using ML for reasoning is that the environment is partially known to the system, and since this approach is intrinsically inductive, the lack of facts, significant to the inference, sometimes may lead to wrong conclusions and decisions. Moreover, ML cannot train in real-time but can be used as a model for reasoning and prediction. Reinforcement learning is an ML approach adopted by intelligent systems to take actions in their environment. While doing so, the reinforcement learning algorithms tend to maximize a cumulative reward while accomplishing a specific task. A system formed by multiple CPS can adopt a centralized reinforcement learning model to decide cooperatively on actions forming a plan and execute them to accomplish a specific task in a decentralized manner. In a dynamically changing environment, where data may be non- or partially observed, each CPS learns an optimal behavior policy and interacts with other agents. A cost/reward function is used to measure the effects of a decision to enhance the learning under incomplete or erroneous sensor data [20, 25]. A drawback with reinforcement learning is that the reward function can be very hard to determine.

Stream reasoning is an emerging field to create intuitively complex reasoning strategies that operate on continuously changing data over time [13]. In systems adopting stream reasoning, input data is generally produced by virtual or physical sensors, or by social networks, which can provide additional end-user information to the system. Streams should be findable, accessible, interoperable, and reusable (FAIR) to achieve robustness. The FAIR characteristics are ensured through building models, ontologies, and vocabularies [35]. Functional aspects of heterogeneous data streams, such as publication, sharing, discovering, and validating while maintaining privacy, federation, and security, are essential in robust stream reasoning systems. Reasoners and agents' collaboration and negotiation must be defined through vocabularies and communication protocols that deal with knowledge, goals, and behavioral patterns dynamicity [35]. The system can adopt stream scheduling strategies under uncertain conditions to overcome limitations due to different levels of uncertainties and agents' beliefs [6]. A stream descriptor provides quantitative and qualitative information about the captured streams. The stream reasoner takes decisions on how to process the data stream to find, report and predict failures or malfunctions. An important aspect to consider is the discretization abstraction of streams. The stream design should deal with heterogeneous data while implementing a synchronization methodology to orchestrate their granularity and interoperability. Data Stream Management Systems are used to transfer data streams into timestamped windows on which logical operators are applied. To identify complex relations over the transformed time-windows (or events), Complex Event Processors are adopted [11]. These complex event processor systems provide inductive and/or deductive reasoning that handles dynamic knowledge representation having streaming flexibility to respond with strict and timely deadlines and provide reliable feedback [6]. The stream reasoning temporal feature involves optimization and planning under uncertainty associated with data and the reasoning operations [11, 22, 14, 13]. The types of noise a distributed stream reasoner may encounter [11], can arise in the content of the stream if it contains erroneous data. Another type of noise can be found in the order of the stream timestamps due to some delay caused by the network or communicating media. The introduced noise may lead to wrong reasoning and decisions. Stream reasoning commonly reasons on parts of a data stream that are significant to CPS. With this reasoning, an autonomous vessel or car can speed up the reasoning and find the cause and effect of an occurrence and predict failures, such as malfunctioning parts that might break down in an expected period of time.

5. Robustness Analysis

Robustness analysis (RA) is an old term that has been used in many domains with various interpretations and hence definitions. In this paper, RA is defined as a reasoning method for structuring problem situations and evaluating decisions that can be staged sequentially, and allowing correct functioning in the presence of data noise and environmental fluctuations. Moreover, the RA can represent the functionality of a system under perturbations and quantify their impacts. RA measures the degree to which the system can function correctly with unforeseen or erroneous inputs when executing the set of actions proposed by the reasoner. To analyze, reasoning with fuzzy logic, model-based reasoning, and reasoning using ML algorithms, such as genetic algorithms and deep learning, can be applied. In the literature, the term RA is coupled with uncertainty and flexibility. It can be found in many approaches depending on the application at hand. One way to accomplish RA is, for example, to study complex phenomena or theories by generating heterogeneous models with varying assumptions about the phenomenon [38]. The fundamental structure of each model consists of three main parts: a structure, a robust property, and a set of conditions. The robustness of the phenomenon, or theory, is ensured if the diversified models reach common results. Inferred common results from the distinct models, despite their differences, should be similar to trust the reliability of the predictions and explanations. This theory can be represented as: for every i, $M_i \vDash R, i \in 1, ..., n$ where $M_1, ..., M_n$ are the distinct models and R is the common result or explanation of the model to the given phenomenon. This diversification concept has been used for constructing and confirming hypotheses [29]. The reliability of RA depends heavily on diversified detection means, which is coupled with a high degree of heterogeneity. A hypothesis is incrementally confirmed by ruling out evidences that could support it or eliminating the possibilities that can falsify it with respect to its competing hypotheses. Another form of RA is to gauge the robustness of the operating processes by perturbation on the measured data [17]. With the model predictive control method, robust systems can satisfy constraints on control signals and state variables to minimize errors between the system's output and the adaptive filtering output with unknown parameters. A tool to improve the robustness efficiency of the system design is to incorporate random variability in the problem formulation [5]. By adding the random variability, the system takes into consideration the probabilistic representation of unexpected failures and unforeseen situations. The probabilistic representations are introduced as modifications in the objective function or additional conditions. RA for designing CPS can be achieved by imposing certain assumptions on the dynamicity of the environment, which is partly known at design time [26].

Robustness Analysis for Robust Cyber-Physical Systems can handle task management robustness by measuring the robustness of real-time scheduling algorithms. The measuring can, for example, be evaluating the computation time perturbations that interrupt the scheduling during execution. As stated earlier, a CPS system is considered robust when it operates with predictable and highly stable behavior, irrespective of the growing complexity of the required task and the dynamicity of the surrounding environment. To guarantee the system robust behavior in a dynamically changing environment, e.g., a RA framework that focuses on robust reasoning exerted by CPS, can be used. This framework can include step-wise methods for managing uncertainty in decisions and actions of CPS, carried out by evaluating and verifying the robustness of the optimal solutions with variations in CPS intentions. Robust system operations shall tolerate erroneous input or partial faults in its computing capabilities, internal and external communications, and data processing [16]. Robust decisions computationally enable such operations via concepts, tools, and processes, i.e., robust decision making (RDM). RDMs derive the CPS robust actions and reactions under deep uncertainty. RDM outlines specific frameworks that suggest robust strategies, evaluate issues for their vulnerabilities, and propose approaches detailing the sequence of actions that yield robust solutions rather than optimal ones [19]. RA for autonomous CPS is typically employed on three levels: the input data, the proposed set of actions, and the results after exerting the actions. *The input data* observed or captured by the sensors highly affect RDMs for CPS. Having a diversified set of sensors in the system and confirming the observations through their measurement agreements ensures the accuracy of the measurements and the robustness of the CPS decisions made on such measurements. *The proposed set of actions* are applicable when decisions under uncertainty are staged. This set of actions are used to evaluate the flexibility of the initial decisions' commitments of actions under conditions of uncertainty by leaving acceptable options of actions at the planning horizon for future decision choices [2]. In this situation, the CPS should incorporate uncertainty in the problem formulation. This can be performed by introducing assumptions to unforeseen disturbances or faults. An essential component for a robust CPS is a feedback loop added *after executing the set actions*. This feedback loop will assist a close observation of the system behavior on the proposed set of actions, and accordingly, the reasoner can adjust the plan for better interaction with the environment's dynamicity. The system behavior can be evaluated via how the results are affected by the past means of detection [29]. The results are, e.g., observations, measurements, predictions, theorems' and the means of detection are, e.g., experiments, laboratory instruments, sensory modalities, derivations from axioms, models, theories, axiomatic systems, computer simulations, and formal models. A successful increment of RA can indeed confirm a target hypothesis. RA for Robust CPS has two-fold aspects: 1) RA in ML, implying that ML algorithms use RA for training and providing valid alternative decisions from collected data in CPS; 2) RA with ML, which means that RA incorporates ML algorithms for reasoning and evaluation of the different system's alternative decisions according to initial commitment, i.e., expected decision of the system.

Robustness analysis in Machine Learning: RA is utilized by ML algorithms to handle incomplete, incorrect, conflicting, and skewed data in images and videos. The ambition of RA is to obtain proper and valid outcomes despite the quality of the input, which might be problematic. As mentioned above, ML algorithms adopt inductive reasoning strategies for data-mining agent applications. With RA, the algorithms can increase the probability of a certain action option by evaluating decisions that can be staged sequentially. Hence, RA in ML allows correct operations functionalities despite data noise and environmental fluctuations. A problem with RA in ML is that ML algorithms may not be considered robust since they are highly dependent on the quality of the input data provided to the system. For instance, data can lack significant input data. For example, a feature may have too many missing data points, i.e., hardware component, due to loss of particular time frames. In addition, the lack of examining the content of the system and the possible actions, i.e., operation explainability, makes it difficult to interpret the output of the algorithm. When erroneous data is introduced, the system can fail to maintain its stability.

Robustness analysis with Machine Learning: RA uses ML to analyze decisions and measure the actions in a stepwise manner. RA measures the distinctions between the initial option, i.e., action commitment and acceptable action options, and evaluates the implications of those decisions on the system's stability without destroying flexibility. As mentioned above, RA using ML can represent the functionality of a system with perturbations and quantify the impacts. Robustness analysis with ML can use probabilistic reasoning, in particular fuzzy logic, and model-based reasoning to measure the degree to which the system can function correctly with unforeseen or erroneous inputs when executing the set of actions proposed by the reasoner. Model-based reasoning is an inference method based on a model of the physical world. In CPS, rule-based approaches are used, and during execution, the CPS combine the model knowledge with observed data to derive conclusions using inductive reasoning. Combining ML with rule-based models can be a promising research direction and can yield more powerful problem solvers that incorporate the advantages of both worlds. ML and rule-based algorithms form a constructed hybrid system that must comply with specific measurements based on robustness analysis to guarantee an RDM output.

6. Conclusions and further work

This paper presents the assessment of robust reasoning for autonomous CPS in dynamic environments. The dynamic characteristics complicate the decision to take a set of actions since the data can be inconsistent and irrational, leading to unreasonable movements of the CPS. Several existing classical and non-classical reasoning strategies and algorithms applied on autonomous CPS are described together, with the benefits and drawbacks of the different strategies and algorithms. Moreover, robustness analysis is described as a means to reason with the possible action options. When executing the set of actions proposed by the reasoner, the analysis evaluates the quality of the conducted decisions by measuring the degree to which the system can function correctly with unforeseen or erroneous inputs.

This paper also sheds light on combinations of reasoning strategies, algorithms, and robustness analysis to achieve robust reasoning, thus maintaining the CPS overall stability and ensuring its robust behavior in a dynamically changing environment. Tables 1, 2, and 3 in the attached appendix summarize the application of the different robust reasoning strategies for autonomous CPS, discussed in this paper, and present a brief review of their benefits and drawbacks. The conclusion is that several different common classical RS, non-classical RSA, and RA can support the robustness of CPS in dynamically changing environments. Moreover, a combination of these classical RS and non-classical RSA and RA can achieve robust behavior in autonomous CPS in dynamic environments.

As future work, different combinations of robust reasoning strategies and robustness analysis will be explored in detail to establish the possibility of providing an improved robust behavior for the CPS. This is especially important in cases where either robust reasoning strategies or robustness analysis cannot reach valid conclusions from the available data. The belief is that a combination of robust reasoning strategies, algorithms, and robustness analysis evaluations can be a means to create robust autonomous CPS in dynamic environments.

Acknowledgements

This work has been supported by the IDUN-project - NFR 295920 - the Research Council of Norway (RCN).

References

[1] Terje Aven and Uwe Jensen. *Stochastic models in reliability*. Springer, 1999.

[2] PG Bennett, SA Cropper, CS Huxham, and J Rosenhead. Rational analysis for a problematic world. 1989.

[3] Ralph Bergmann, Ralf Schenkel, Lorik Dumani, and Stefan Ollinger. Recap-information retrieval and case-based reasoning for robust deliberation and synthesis of arguments in the political discourse. In *LWDA*, pages 49–60, 2018.

[4] Pierre Bessière, Christian Laugier, and Roland Siegwart. *Probabilistic reasoning and decision making in sensory-motor systems*, volume 46. Springer, 2008.

[5] Christian Bucher. Robustness analysis in structural optimization. *Structure and Infrastructure Engineering*, 5(4):287–293, 2009.

[6] Davide Calvaresi and Jean-Paul Calbimonte. Real-time compliant stream processing agents for physical rehabilitation. *Sensors*, 20(3):746, 2020.

[7] Longbing Cao, Vladimir Gorodetsky, and Pericles A Mitkas. Agent mining: The synergy of agents and data mining. *IEEE Intelligent Systems*, 24(3):64–72, 2009.

[8] Christer Carlsson and Robert Fullér. *Fuzzy reasoning in decision making and optimization*, volume 82. Physica, 2012.

[9] Anna Ciampolini, Evelina Lamma, Paola Mello, Cesare Stefanelli, and Paolo Torroni. An implementation for abductive logic agents. In *Congress of the Italian Association for Artificial Intelligence*, pages 61–71. Springer, 1999.

[10] Daniel De Leng. *Robust Stream Reasoning Under Uncertainty*, volume 2006. Linköping University Electronic Press, 2019.

[11] Daniele Dell'Aglio, Emanuele Della Valle, Frank van Harmelen, and Abraham Bernstein. Stream reasoning: A survey and outlook. *Data Science*, 1(1-2):59–83, 2017.

[12] Patricia Derler, Edward A Lee, Stavros Tripakis, and Martin Törngren. Cyber-physical system design contracts. In *Proceedings of the ACM/IEEE 4th International Conference on Cyber-Physical Systems*, pages 109–118, 2013.

[13] Carmine Dodaro, Thomas Eiter, Paul Ogris, and Konstantin Schekotihin. Managing caching strategies for stream reasoning with reinforcement learning. *Theory and Practice of Logic Programming*, 20(5):625–640, 2020.

[14] Thomas Eiter, Paul Ogris, and Konstantin Schekotihin. A distributed approach to lars stream reasoning (system paper). *Theory and Practice of Logic Programming*, 19(5-6):974–989, 2019.

[15] Anne Håkansson, Ronald Hartung, and Esmiralda Moradian. Reasoning strategies in smart cyber-physical systems. *Procedia Computer Science*, 60:1575–1584, 2015.

[16] Fei Hu, Yu Lu, Athanasios V Vasilakos, Qi Hao, Rui Ma, Yogendra Patil, Ting Zhang, Jiang Lu, Xin Li, and Neal N Xiong. Robust cyber–physical systems: Concept, models, and implementation. *Future generation computer systems*, 56:449–475, 2016.

[17] Wiktor Jakowluk and Karol Godlewski. Robustness analysis of the estimators for the nonlinear system identification. *Entropy*, 22(8):834, 2020.

[18] Philip N Johnson-Laird. Deductive reasoning. *Annual review of psychology*, 50(1):109–135, 1999.

[19] Robert J Lempert. Robust decision making (rdm). In *Decision Making under Deep Uncertainty*, pages 23–51. Springer, Cham, 2019.

[20] Matteo Leonetti, Luca Iocchi, and Peter Stone. A synthesis of automated planning and reinforcement learning for efficient, robust decision-making. *Artificial Intelligence*, 241:103–130, 2016.

[21] Alex S Leong, Arunselvan Ramaswamy, Daniel E Quevedo, Holger Karl, and Ling Shi. Deep reinforcement learning for wireless sensor scheduling in cyber–physical systems. *Automatica*, 113:108759, 2020.

[22] Alessandra Mileo, Minh Dao-Tran, Thomas Eiter, and Michael Fink. Stream reasoning. 2017.

[23] Nicola Muscettola, P Pandurang Nayak, Barney Pell, and Brian C Williams. Remote agent: To boldly go where no ai system has gone before. *Artificial intelligence*, 103(1-2):5–47, 1998.

[24] Vedant Nanda, Junaid Ali, Krishna P Gummadi, and Muhammad Bilal Zafar. Unifying model explainability and robustness via reasoning labels.

[25] Praveen Palanisamy. Multi-agent connected autonomous driving using deep reinforcement learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2020.

[26] Matthias Rungger and Paulo Tabuada. A notion of robustness for cyber-physical systems. *IEEE Transactions on Automatic Control*, 61(8):2108–2123, 2015.

[27] Aya Saad, Thom Frühwirth, Carmen Gervet, Michael Leuschel, and Tom Schrijvers. The p-box cdf-intervals: A reliable constraint reasoning with quantifiable information. *Theory and Practice of Logic Programming*, 14(4-5):461, 2014.

[28] Aya Saad, Carmen Gervet, and Slim Abdennadher. Constraint reasoning with uncertain data using cdf-intervals. In *International Conference on Integration of Artificial Intelligence and Operations Research Techniques in Constraint Programming*, pages 292–306. Springer, 2010.

[29] Jonah N Schupbach. Robustness analysis as explanatory reasoning. *The British Journal for the Philosophy of Science*, 69(1):275–300, 2018.

[30] Lokendra Shastri, Venkat Ajjanagadde, et al. From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings using temporal synchrony. *Behavioral and brain sciences*, 16:417–417, 1993.

[31] Fernando Soler Toscano and Fernando R Velázquez Quesada. Abduction for (non-ominiscient) agents. 2010.

[32] Ron Sun. An efficient feature-based connectionist inheritance scheme. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(2):512–522, 1993.

[33] Andreas L Symeonidis, Kyriakos C Chatzidimitriou, Ioannis N Athanasiadis, and Pericles A Mitkas. Data mining for agent reasoning: A synergy for training intelligent agents. *Engineering Applications of Artificial Intelligence*, 20(8):1097–1111, 2007.

[34] Paulo Tabuada, Sina Yamac Caliskan, Matthias Rungger, and Rupak Majumdar. Towards robustness for cyber-physical systems. *IEEE Transactions on Automatic Control*, 59(12):3151–3163, 2014.

[35] Riccardo Tommasini, Davide Calvaresi, and Jean-Paul Calbimonte. Stream reasoning agents: Blue sky ideas track. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1664–1680, 2019.

[36] Florian Tramer and Dan Boneh. Adversarial training and robustness for multiple perturbations. *arXiv preprint arXiv:1904.13000*, 2019.

[37] Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. On the convergence and robustness of adversarial training. In *ICML*, volume 1, page 2, 2019.

[38] Michael Weisberg. Robustness analysis. *Philosophy of science*, 73(5):730–742, 2006.

[39] Zhuolin Yang, Zhikuan Zhao, Hengzhi Pei, Boxin Wang, Bojan Karlas, Ji Liu, Heng Guo, Bo Li, and Ce Zhang. End-to-end robustness for sensing-reasoning machine learning pipelines. *arXiv preprint arXiv:2003.00120*, 2020.

[40] Lotfi Asker Zadeh, George J Klir, and Bo Yuan. *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers*, volume 6. World Scientific, 1996.

[41] Junfei Zhang, Dong Li, and Yuhang Wang. Predicting tunnel squeezing using a hybrid classifier ensemble with incomplete data. *Bulletin of Engineering Geology and the Environment*, 79(6):3245–3256, 2020.

[42] Zhi-Hua Zhou. Abductive learning: Towards bridging machine learning and logical reasoning. *Science China Information Sciences*, 62(7):76101, 2019.

	Robust reasoning in CPS	Benefits	Drawbacks
Deductive	The reasoning process by deducing information about the environment and the system state from all available facts is robust because the conclusion can be formally proved. To ensure robust deductive reasoning, the input data collection obtained from several observatory methods (sensors) must agree on the measurement and the values of these facts.	The correct reasoning is formally proved through formal specifications.	Knowledge-base must be hardcoded with correct and complete data in hand. Susceptibility to noise and changes can lead to inconsistency. Lack of flexibility when data is incomplete.
Inductive	The reasoning process is exerted by deducing general information with a level of certainty about the environment from a limited set of observations or facts. To ensure robustness, vital facts that significantly decrease the uncertainty level of the drawn conclusions must exist.	The correct reasoning is formally proved to make the drawn robust conclusion of the CPS more trustworthy.	The lack of facts that are significant to the inference might lead to a high level of uncertainty and result in wrong conclusions.
Abductive	When knowledge about the problem is incomplete, a plausible hypothesis about what is commonly true in the environment is inferred.	Ensure plausible inferred hypothesis among agents is true to reason about the incomplete knowledge in the environment.	The high degree of inconsistency in the inferred hypotheses provided by different alternatives may lead to contradictory results and confusion. The hypothesis cannot be formally proved valid to the environment.
Case-based reasoning	Use analogical reasoning to infer solutions based on similar problems through retrieve, reuse, revise and retain.	Evaluate the degree of similarities between the cases to select the best matching experience that applies to the case under consideration.	Lack of similar cases or insufficient evidence matching similar cases.
Monotonic and non-monotonic reasoning	Monotonic reasoning is based on static truth value. Non-monotonic reasoning is based on revealing dynamic truth values.	Ensure a suitable representation of the true value.	Having a static truth value contradicts the dynamic property of the environment. Dynamically revealing new values for the truth opens up for incorrectness in conclusions.
Probabilistic reasoning	Intuitive qualitative representation of information that draws conclusions when the data is uncertain or when the environment is dynamically changing.	Ensure the qualitative constructs of information is reliable, covering all possible cases.	When the solution is coupled with weights, it is sometimes hard to quantify the resulting true value.

Table .1. Classical reasoning strategies for Robust CPS

	Robust reasoning in CPS	Benefits	Drawbacks
Machine Learning / Reinforcement Learning	Adopt inductive reasoning strategy for data-mining. Learn the set of actions that form a plan. Interpret the dynamics of the physical environment at a set of sequential states. Each CPS learns an optimal behavior policy to interact with other agents.	The cost/reward function is used to measure the effects of a decision to enhance the learning under incomplete or erroneous sensor data.	The environment is partially observed. Machine learning models cannot be trained on new information in real-time.
Stream Reasoning	Create complex reasoning strategies that operate on continuously changing data over time. Functional aspects of heterogeneous data stream such as publication, sharing, discovering, and validating while maintaining privacy, federation, and security are essential in robust stream reasoning systems.	The system adopts stream scheduling strategies under uncertain conditions to overcome limitations due to different levels of uncertainties.	Types of noise that exist in a stream reasoning distributed system: • The content of the stream can be affected and contain erroneous data- • Timestamps can be out-of-order.

Table .2. Non-classical reasoning strategies for Robust CPS

	Robustness analysis in CPS	**Benefits**	**Drawbacks**
Robustness analysis in ML	Provide valid decisions from data. Increase the probability of a certain hypothesis, i.e. perceived knowledge on the input level, the proposed set of actions, and the results after executing the actions.	Seek to obtain proper and valid outcomes despite the quality of the input which might be problematic. A successful increment of robustness analysis can confirm a target hypothesis.	When multiple systems agree on wrong interpretations of, for example, the input, the consensus can confirm wrong assumptions and can lead to erroneous knowledge and wrong decisions.
Robustness analysis with ML	Combine ML with rule-based models to create RDM systems that proposed a robust set of actions. For autonomous CPS, robustness analysis evaluates the system behavior by measuring how the results are affected by the proposed set of actions outlined by the RDM by attaching a feedback loop that monitors the system status after executing the proposed set of actions.	Evaluate issues of vulnerability in the set of actions outlined by the RDM, and propose approaches detailing the sequence of actions that yield robust solutions rather than optimal ones.	Hybrid systems combining ML and rule-based models must comply with specific measurements to guarantee an RDM output.

Table .3. Robustness analysis for robust CPS