Risk and Reliability Evaluation of Future Industrial Automation Systems: A Systematic Literature Review and Research Agenda

Andrey Morozov [1], **Tagir Fabarisov** [2], **Silvia Vock** [3], **Georg Siedel** [3], **Victor Bolbot** [4], **Stefan Voß**[3]
[1] University of Stuttgart, Stuttgart, Germany, andrey.morozov@ias.uni-stuttgart.de
[2] University of Luxembourg, Luxembourg, tagir.fabarisov@uni.lu
[3] Bundesanstalt für Arbeitsschutz und Arbeitsmedizin, Dresden, Germany, {last, first}@baua.bund.de
[4] Aalto University, Helsinki, Finland, victor.bolbot@aalto.fi

Today, terms such as Sustainable Production, Industrial Cyber-Physical Systems, Cyber-Physical Production Systems (CPPS), Software-Defined Manufacturing, Smart Manufacturing, Industry 4.0, Industry 5.0, System of Systems, Internet of Things, Human-in-the-Loop, and Digital Twins are widely used. These concepts emphasize key characteristics of modern and future production systems, including heterogeneity, structural and behavioral complexity, intelligence, autonomy, reconfigurability, and human centrism. They also highlight the growing importance of reliable and up-to-date risk assessment, safety, and reliability measures, given the significant environmental, economic, and social demands. However, current industrial risk analysis methods lag behind the rising technical sophistication of such systems. It remains unclear whether existing methods can capture complex failure scenarios of dynamic, AI-driven systems with advanced software architectures.

This paper discusses the main challenges facing safety engineers in industrial automation. We provide a classification and overview of available risk and reliability analysis methods and metrics, supported by a systematic review of 92 papers. The review addresses questions such as: which CPPS aspects must be considered, which methods are applicable, what are their advantages and limitations, and how can methods be combined? The findings reveal the need to extend classical approaches toward dynamic risk assessment, probabilistic model checking, AI-based techniques, digital twins, and intelligent fault injection. The study provides both a comprehensive overview of current risk and reliability assessment methods for CPPS and a roadmap for advancing their future development.

Acronym List

AI	Artificial Intelligence
BDMP	Boolean logic-Driven Markov Processes
BN	Bayesian Networks
BTA	Bow-Tie Analysis
CCD	Cause-Consequence Diagram
CCA	Cause-Consequence Analysis
CFT	Component Fault Trees
CPS	Cyber-Physical Systems
CPPS	Cyber-Physical Production Systems
CTMC	Continuous-Time Markov Chain
DEPM	Dual-graph Error Propagation Model
DET	Dynamic Event Trees
DL	Deep Learning
DRBD	Dynamic Reliability Block Diagrams
DRL	Deep Reinforcement Learning
DT	Digital Twin
DTMC	Discrete-Time Markov Chain
DFT	Dynamic Fault Trees
ETA	Event Tree Analysis
ET	Event Trees
ESD	Event Sequence Diagrams
FV	Formal Verification
FI	Fault Injection
FT	Fault Trees
FTA	Fault Tree Analysis
FMEA	Failure Mode and Effects Analysis
FMECA	Failure Mode, Effects, and Criticality Analysis
FRAM	Functional Resonance Analysis Method
GAN	Generative Adversarial Networks
GSPN	Generalized Stochastic Petri Nets
HCL	Hybrid Causal Logic
HAZOP	Hazard and Operability Analysis
HSPN	Hybrid Stochastic Petri Net
IMC	Interpreted Markov Chain
MC	Model Checking
MDP	Markov Decision Process
MDT	Mean Downtime
ML	Machine Learning
MTBF	Mean Time Between Failures

MTTF	Mean Time to Failure
PHR	Process Hazard Review
PMC	Probabilistic Model Checking
PCTL	Probabilistic Computation Tree Logic
PRM	Probabilistic Relational Models
QA	Quality Assessment
RL	Reinforcement Learning
RBD	Reliability Block Diagrams
RFT	Repairable Fault Trees
RM	Risk Method
SA	Static Analysis
SDM	Software-Defined Manufacturing
SEFT	State Event Fault Trees
SLR	Systematic Literature Review
SPN	Stochastic Petri Nets
STPA	System-Theoretic Process Analysis
TP	Theorem Proving

1 Introduction

Risk is commonly defined as the combination of the likelihood of a hazardous event and the potential consequences of that event on the system's functionality, performance, and overall safety [1]. New concepts such as Cyber-Physical Production Systems (CPPS), Software-Defined Manufacturing, and Smart Factories have introduced innovative technologies in industrial automation. These concepts emphasize aspects like high autonomy, intelligence, heterogeneity, spatial and logical distribution, high structural and behavioral complexity, reconfigurability, and human-in-the-loop integration. They pose serious new challenges for risk assessment. Currently available risk assessment and hazard identification methods, such as FMEA, FTA, and STPA, vary in their applicability and effectiveness when addressing the unique challenges posed by modern CPPS. It is important to evaluate the applicability of these methods to ensure they can adequately assess and manage the risks associated with advanced production systems. In this paper, we conduct a comprehensive Systematic Literature Review (SLR) to identify the limitations and gaps in existing risk assessment methods for CPPS. We evaluate how these methods cope with the challenges of CPPS.

Key contributions of this paper:

Identification of thirteen CPPS aspects that challenge risk assessment.
Evaluation of existing risk methods and their applicability to CPPS.
Highlighting gaps in current risk assessment methodologies.
Suggesting promising directions for future research and development.

The remainder of this paper is organized as follows: Section 2 provides the necessary background information about risk, risk assessment, and how the evolution of industrial automation systems challenges risk assessment. Section 3 provides an overview of modern risk assessment methods. Section 4 discusses the structure and procedure of the conducted SLR. Section 5 presents the results and their interpretation, identifies gaps, and suggests future research directions. Section 6 compares the SLR with other similar meta-reviews, explaining the necessity of the new SLR and distinguishing it from other works. Finally, Section 7 concludes the paper.

2 Risk assessment and its challenges

2.1 What is risk?

We define risk by answering three questions [1]: "What can go wrong?", "How likely is it that that will happen?", "What are the consequences?". Similar concepts appear in numerous other papers, including seminal works of Terje Aven [4, 3]. The answer to the first question helps to define possible system failure scenarios. The answer to the second question provides the qualitative or quantitative estimation of likelihoods of these scenarios to happen. The answer to the third question gives the understanding of the potential hazard level of each scenario. Therefore, formally, risk can be defined as the set of triplets: $R = \{\langle s_i, p_i, x_i \rangle\}$. Where s_i is a scenario identification or description, p_i is the likelihood, e.g., frequency or probability, of that scenario, and x_i is the consequence of that scenario, i.e., the measure of damage. Figure 1 sketches eight system execution scenarios. Scenarios s_5, s_6, and s_7 are failure scenarios. Some experts said that "risk is probability times consequence", reporting the risk as a single value, the product of the probability and measure of hazard [3]. However, in most cases, it is required to communicate the likelihood and potential consequence for each scenario separately to distinguish between high-probably low-damage and low-probability high-damage scenarios [1].

The risk analysis is closely related to the dependability theory [5] that defines crucial terminology and taxonomy for risk analysis and risk reduction. Dependability addresses three groups of definitions: (i) attributes (availability, reliability, safety, confidentiality, integrity, and maintainability), (ii) means (fault prevention, fault tolerance, fault removal, and fault forecasting), and (iii) threats (faults, errors, and failures). The last group, threats, is the most relevant for us. A fault is a defect in the system that can be activated and cause an error. An error is an incorrect internal state of the system, or a discrepancy between the intended behavior

[1] The definition of risk used in this paper follows the work of Kaplan and Garrick [1]. However, other definitions exist, e.g., highlighting the importance of uncertainty quantification. For more details, refer to the works of Terje Aven [2, 3].

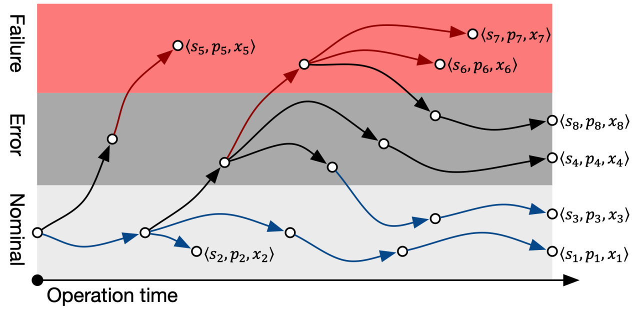
Failure
Error
Nominal
Operation time
$\langle s_5, p_5, x_5 \rangle$
$\langle s_7, p_7, x_7 \rangle$
$\langle s_6, p_6, x_6 \rangle$
$\langle s_8, p_8, x_8 \rangle$
$\langle s_4, p_4, x_4 \rangle$
$\langle s_3, p_3, x_3 \rangle$
$\langle s_2, p_2, x_2 \rangle$
$\langle s_1, p_1, x_1 \rangle$

Fig. 1: System execution scenarios represented as sequences of system states. The states are classified into three categories: nominal states, erroneous states, and failure states. Each scenario is represented as a triplet $\langle s_i, p_i, x_i \rangle$. Where s_i is a scenario description, p_i is the likelihood, and x_i is the measure of hazard.

of a system and its actual behavior. A failure is an incorrect external state of the system, when the system displays behavior that is contrary to its specification. The circles in Figure 1 describe system states that are classified into three categories: (i) Normal state, in which every single component of the system operates in the specified nominal mode; (ii) Erroneous State, in which there are errors in a part of the system and some of the system components might be failed, but the system operates correctly at a potentially lower performance level; (iii) Failure State, in which the system cannot perform its function (e.g., because of a failure of a critical set of components).

Each scenario starts in the initial system state, which is nominal. System components might have faults. However, most of them stay dormant and do not manifest as errors or failures during the system execution. Correct system execution scenario s_1 models this nominal case. Scenario s_2 is also a "correct" scenario, but the system was stopped earlier. The other scenarios contain faults activations and errors occurrences. An error is an incorrect internal state of the system. However, the system can operate according to its specification. Errors tend to propagate through the system, causing other errors or failures (i.e. the fault-error-failure chain [5]). Scenario s_4 represents the situation when the system accomplishes the task being in an erroneous state. In scenario s_3, an error is detected and mitigated, bringing the system back to nominal operation. In contrast, in scenarios s_5, s_6, and s_7 errors cause system failures.

The tasks of various risk analysis methods discussed in this paper are to identify scenarios (s_i), estimate the likelihoods (p_i), and potential damages (x_i). The failure prevention can be implemented either by fault-tolerance mechanisms that prevent the system from jumping to an erroneous state after the fault activation or by error detection and recovery mechanism like in s_3. Some systems can be even restored after a failure. This situation is shown in scenario s_8. Obviously, the identification of all failure scenarios is impossible [1]. The goal is to concentrate on the most hazardous and the most probable ones. The challenges to the risk analysis are discussed in Section 2.4.

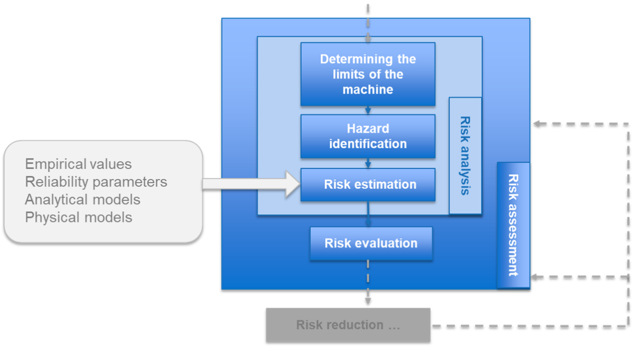

Fig. 2: Iterative process for risk assessment and risk reduction according to the international standard ISO 12100:2010.

2.2 Risk assessment

Risk assessment is often mandatory and the concepts discussed above are integrated into industry standards. In Europe, the Machinery Directive plays an important role in the field of industrial automation. The Machinery Directive 2006/42/EC was replaced in 2023 by a new Machinery Regulation 2023/1230. This new regulation will apply from 20 January 2027. The currently valid Machinery Directive and the new Machinery Regulation both define safety objectives for various types of machines. It explicitly demands a risk assessment to be performed by the manufacturer before the machine is placed on the European market. The risk assessment aims at providing decision support on whether there is a need for additional safety measures and how the safety system needs to be designed.

The standard ISO 12100:2010 [6] elaborates requirements of the Machinery Directive with respect to the general risk assessment procedure and the risk reduction. It specifies risk assessment and risk reduction procedures for machinery as an iterative process (see Figure 2). The risk assessment begins by determining the machine's limits, which include the intended use and reasonably foreseeable misapplications, the time boundaries such as lifespan and maintenance intervals, the spatial limits like interactions between humans and the machine, and other limits such as environmental constraints. In a next step the hazards are identified. For this step several techniques exist, ranging from checklists to fault and event trees. In the risk estimation step the severity and the probability of the hazardous event are combined and estimated. Quantitative and qualitative methods can be applied for this step. Fi-

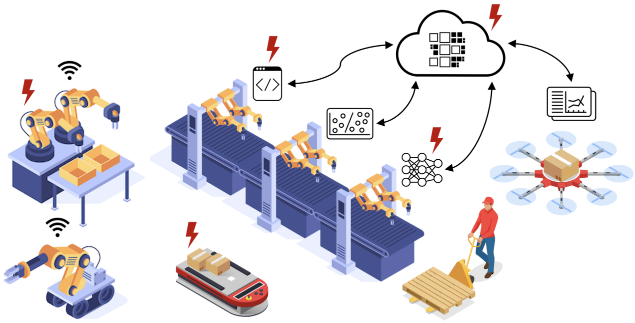

Fig. 3: A production system consisting of multiple heterogeneous physical and virtual (cyber) assets.

nally, the risk evaluation step compares the estimated risks with risk acceptance criteria which are detailed, e.g., in standards or can be derived from the current state of technology (see Figure 4). Similar procedures are defined in the other risk standards, e.g. in the more generic standard ISO 31000 [7].

2.3 Trends in productions systems

In recent decades, numerous terms describing new as well as established concepts and trends have emerged. These concepts partially overlap, have many controversial definitions in different sources and communities, and sometimes primarily serve marketing and political purposes. However, they do highlight particular technical aspects of modern and future production systems and are essential to be mentioned in this paper.

Cyber-Physical Systems: The widely adopted Cyber-Physical Systems (CPS) [8] concept has been reflected in production as Industrial CPS, Cyber Manufacturing, and Cyber-Physical Production Systems (CPPS). These concepts describe the heterogeneous system components, such as software, hardware, and mechanical parts, emphasizing the intensive interaction of networked computational nodes (i.e., cyber) with the surrounding processes (i.e., physical). Also, CPS usually refers to the System of Systems concept, a collection of dedicated systems that pool their resources and capabilities together to offer more functionality and performance than simply the sum of the constituent systems.

Industry 4.0: Industry 4.0 [9] is another popular concept focused on creating Smart Factories exploiting CPPS, the Internet of Things (IoT), and several particular IT technologies such as Edge Computing, which brings computation closer to data sources; Fog Computing, which extends cloud capabilities to the edge of the network; Cloud Computing, offering scalable resources and services; and Cognitive Computing, which enhances decision-making through AI and machine learning.

Sustainable and human-centric production: The relatively recent Industry 5.0 concept provides a fresh vision of an industry that aims beyond efficiency and productivity as the sole goals and reinforces the role and the contribution of industry to society. The Industry 5.0 concept is created around Sustainable and resource-efficient production, the Human-in-the-Loop [10], and even the Human-centric approach to digital technologies, including regulating the application of AI.

Digital Twin: The widely-used Digital Twin (DT) [11] concept is closely connected to CPPS and Industry 4.0. A DT in production refers to an advanced virtual representation that serves as the real-time digital counterpart of physical assets. Like most of the above-mentioned concepts, the DT doesn't have a precise definition. However, it is common to distinguish between a digital model, a digital shadow, and a digital twin depending on how tight the digital and physical processes are coupled.

Flexible production: Smart and Advanced Manufacturing highlight high adaptability, rapid design changes of production lines, and digitalization. Similarly, the Software-Defined Manufacturing (SDM) [12] concept stresses that the software is solely decisive for system configuration, and frequent software updates enable high flexibility and adjustable production.

Artificial Intelligence: Separately, we must mention the enormous growth of Machine and Deep Learning as well as other Artificial Intelligence (AI) [13] technologies and their influence on production systems evolution. AI is primarily used as an assisting technology for non-safety critical tasks. However, there is a clear trend to exploit AI technologies in safety critical application. These applications are regulated since 2024 in the European AI Act [14]. This regulation is a risk based approach and demands a life-cycle based risk management process. However, the regulation defines safety and health objectives, but how they are to be operationalized, e.g. by standards, is still an open question.

2.4 New risk assessment challenges

The concepts mentioned in Section 2.3 emphasize the necessity of an accurate and up-to-date risk assessment process including the implementation of respective safety measures. Leaving aside the marketing purposes of the aforementioned trends, we identify several technical aspects and the challenges each aspect poses for the risk assessment of a future production system like the one shown in Figure 3. The aspects were identified based on background knowledge described in studies such as [15, 16]. For the remainder of this paper, we will refer to systems with these technical aspects as CPPSs.

Heterogeneity: CPPS and Industry 4.0 imply that production systems consist of multiple interconnected types of sub-systems and components, including various mechanical parts (joints, grippers, wheels, belts, transmissions, etc.), hardware parts (smart sensors, sensor arrays, actuators, controllers, network devices, computing hardware, etc.), and software parts including the classical deterministic algorithms as well as AI components (control software, network protocols, HMIs, digital twins, edge and cloud services, MES and ERP systems, etc.). From the risk perspective, heterogeneity means that the system can be affected by very different types of faults: mechanical failures of actuators and other physical parts, heat, vibration, sensor noise, drift, or freeze, network delays and package drops, bit-flips in CPU, RAM, or network, logical faults, software bugs, as well as flaws in the data for utilized AI algorithms. It is essential to consider all these fault classes and, what is more challenging, their combinations and interference.

Complexity: CPPSs are becoming increasingly complex both from structural and behavioral points of view. Structural complexity increases as more components and sub-systems are integrated using multiple and usually non-unified interfaces. Behavioral complexity implies multiple use cases, modes of operation, system states, and operational profiles. Complexity inherently challenges safety; as system complexity increases, reliability tends to decrease. This is reflected in the never-aging K.I.S.S. principle: Keep It Simple, Stupid [17]. A growing structural complexity usually leads to a disproportionate increase in behavioral complexity: The number of system execution scenarios can grow exponentially with the number and complexity of system components. Thus, it becomes challenging to evaluate this vast number of system execution scenarios and identify the safety-critical ones. More details how complexity affects risk assessment of sociotechnical systems can be found in [18].

Intelligence: The concepts discussed above emphasize that software becomes the central part of the system. This affects the risk assessment in three ways. First, the software makes the system more intelligent and autonomous, which increases the behavioral complexity discussed above. A system consisting of distributed agents controlled by complex algorithms will have a huge number of hardly predictable execution scenarios that should be identified and analyzed. Second, the analysis of software reliability, which directly influences system safety, is a complex task by itself. Thorough reliability analysis requires the application of cutting-edge methods, usually based on a combination of efficient fault injection techniques and formal methods. Practically, the engineers apply strict design, implementation, testing, and deployment guidelines during the development of the safety-critical software. These software components are then assumed to be fault-free during risk assessment. This assumption may not hold for the complex software of modern and future production systems. In addition to errors that originate in the software, the software also acts as a medium for the propagation of data and timing errors from other system components. Third, the reliability analysis of AI is an entirely open question at the moment [19]. Despite several methodologies published in recent years [20,21,22,23] and the first attempts to define AI safety standards, this area is still largely uncharted territory from a risk assessment point of view. It is very challenging to identify safety-related requirements for different machine learning tasks, model architectures and their hyper parameters, and most importantly, the testing and training data sets. Moreover, AI systems need to be evaluated in terms of various aspects [24], such as their robustness to random [25,26] or adversarial [27,28] errors in the input data and to faults in the hardware on which they are deployed [29]. The importance of safe AI is highlighted in [30] and the a holistic view of the literature related to ML and DL applications in the context of safety, risk, and reliability is given in [31] and [32].

Reconfigurability: Reconfigurability, re-purposeability, and flexible adaptation of the production systems to changing environments, goals, and requirements are emphasised by Smart, Advanced, and Software-Defined Manufacturing (SDM) concepts. These concepts imply that the system consists of universal and easily configurable robotic components that could accomplish diverse tasks. Thus, the software defines the production process by combining and controlling these generic robotic stations. From a risk assessment point of view, this brings us back to the need for detailed software reliability analysis, as the software now controls not only particular hardware but the entire production process. However, the main challenge lies precisely in the fact that the system can be changed during its operation. This shifts the paradigm of risk assessment towards automated and continuous system analysis. Traditional production systems have rare software updates. Each machine is designed and programmed to perform a specific task. Therefore, the risk assessment is conducted once before the operation, mostly manually or in a semi-automated fashion. With SDM, each update can significantly change the system behavior. Therefore, the risk analysis must be performed continuously before each software update. Advanced hybrid and highly flexible risk models and automated risk model generation algorithms are the key enabling technologies.

Human-centrism: Industry 5.0, human-centric, sustainable, and resource-efficient production bring additional challenges to the risk assessment. Humans are an inevitable part of the production process. Current trends show that complete autonomy is not possible, not only because of imperfect technology, but also for various social and ethical reasons. The Human-in-the-loop concept treats the human as an integral part of the system. Industry 5.0 puts humans at the center, shifting the focus from production efficiency to human needs. From a risk analysis point of view, a human is very complex and, at the same time, a very fragile part of the system. As stated above, complexity is the main enemy of safety, and indeed humans are the source of most system failures. At the same time, humans are the most valuable part of the system, and they have to be protected. Thus, the risk assessment should also be human-centric, taking into account not only physical hazards but also various ethical, social, and cognitive aspects. The sustainability also implies the reduction of environmental damage, such as global carbon dioxide emissions. Therefore, the risk assessment should take into account not only the classical direct risk to human operators but also long-term environmental safety goals.

3 Risk assessment methods

Multiple risk assessment methods help engineers to identify failure scenarios, quantify their likelihoods, and estimate hazards. There are many methods, each with its own unique flexibility, and even more extensions and combinations of them. In this chapter, we provide only a brief overview of these methods. For more details, please refer to the papers cited in this section as well as the result tables in Section 5. Figure 4 shows our proposed classification based on the background knowledge on risk methods. Other classifications are of course possible, e.g., division on linear and non-linear models as suggested in the recent paper "The future of safety science" [78]. In our classification, first, we distinguish between qualitative and quantitative methods. The quantitative methods are further classified into static and dynamic methods. A separate group consists of quantitative metrics that help to describe failure likelihoods for systems and their components. These methods, metrics, and their application vary a lot from industry to industry and heavily depend on guidelines, standards, and tools applied.

3.1 Qualitative methods

Qualitative methods usually provide guidelines and define steps that need to be taken to identify and possibly rank potential failure scenarios in a systematic way. They typically include classical Failure Mode and Effect Analysis (FMEA) [40] and HAZard and OPerability (HAZOP) study [41], as well as more modern System-Theoretic Process Analysis (STPA) [43] and Functional Resonance Analysis Method (FRAM) [42]. The other qualitative techniques are various What-if methods, Checklists, and multiple Probabilistic Hazard Analysis methods [45] such as Preliminary Hazard Analysis (PrHA) [79], Major and Direct Hazard Analysis (MHA/DHA) [80], Process Hazard Review (PHR) [81], Cause-Consequence Analysis (CCA) [82], Bow-Tie Analysis (BTA) [83, 84], to name a few.

The hazard levels and the likelihoods of the scenarios are estimated using qualitative scales, e.g., severity rank from 1 (no damage) to 5 (serious damage) or likelihood that takes values such as negligible, low, medium, high, and very high. These methods give no numerical estimations of risk but rather provide means for identification, classification, and sometimes prioritization of the failure scenarios, e.g., according to the Risk Priority Number in FMEA. The qualitative methods are based on expert opinion and qualification.

The completeness of the failure scenario identification strongly depends on how thoroughly and deeply the analysis is performed. FMEA and HAZOP are straightforward step-by-step procedures that are highly dependent on expert knowledge, STPA and FRAM are more complex and take into account system architecture and behavior aspects. FMEA is still the most commonly used qualitative method with many extensions and specifications such as Failure mode, Effects, and Criticality Analysis (FMECA) [85], Design FMEA [86], Process FMEA [87], and Software FMEA [88]. A brief search on the IEEE Xplore yields around 1000 papers with tittle containing FMEA, whereas HAZOP yildes around 300 papers. However, STPA is becoming more and more popular in technical systems domains, including manufacturing and industrial automation. Different combinations of the methods are possible, e.g., HAZOP can be used as a part of FMEA, or FMEA as a part of STPA. Table 1 summarizes the features of FMEA, HAZOP, and STPA methods.

Formal Verification (FV) is a subset of formal methods focused on proving or disproving the correctness of systems with respect to certain formal specifications or properties, using mathematical and logical reasoning to ensure that a system behaves as intended. Classically [89] , FV includes: (i) Theorem Proving (TP) - logical inference to prove that a system satisfies its specifications, (ii) Model Checking (MC) - automatically verifying whether a finite-state model of a system meets a given specification, typically expressed in temporal logic, and (iii) Static Analysis (SA) - analyzing software code to verify properties without executing the program. MC [44] is the most commonly used FV method for risk and reliability analysis since it helps to automatically identify potential failure scenarios given a formal system model, e.g. a sate machine, and a model checking tool, e.g. SPIN [90], NuSMV [91], or UPPAAL [92]. FV methods are added to the qualitative group. Note that Probabilistic Model Checking (PMC), which not only checks system correctness but also provides a probabilis-

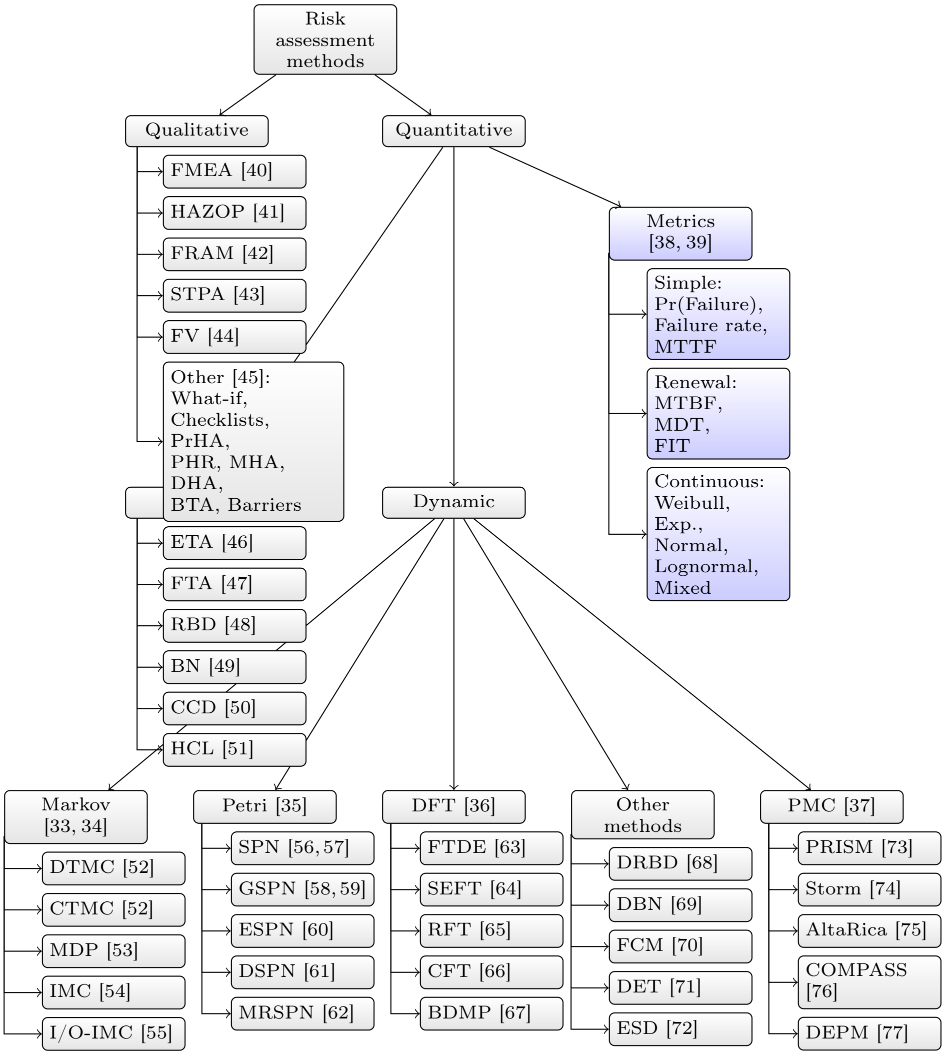

Fig. 4: Classes of risk assessment methods.

tic assessment, is part of quantitative dynamic methods.

3.2 Likelihood metrics

Whereas the primary goal of the qualitative methods is to identify failure scenarios and estimate their hazard levels, the quantitative methods are focused on the numerical evaluation of the likelihood for each scenario. The quantification is usually done using one of the metrics shown in Figure 4. These metrics are also called reliability metrics since they are commonly used for the estimation of system or component reliability, the probability that the system or component will continue performing its functions during the defined period and under the specified operational conditions [5]. Reliability metrics are common for risk assessment since the particular failure of the system or its components usually plays a keyrole in the likelihood of a failure scenario.

In this study, we focus on likelihood metrics, which help evaluate the probability of various failure scenarios. Additionally, several well-established risk metrics incorporate the hazard or consequence aspect of failure scenarios, such as Individual Risk Per Annum (IRPA), Localized Individual Risk (LIRA), and Potential Loss of Life (PLL), among others. For more information on these metrics, refer to the seminal work by Johansen and Rausand [93]. The application of these metrics varies across industries, and even within CPPS, they differ significantly depending on the specific type of plant or system. For instance, a review of consequence-based metrics in the chemical process industry is presented in [94].

The metrics are classified into simple, renewal, and continuous. Simple metrics are single numbers, also called point estimates. Pr(Failure) is the probability of a particular failure scenario. The failure rate is the estimated number of times a component or a system fails in a specified period. Mean Time to Failure (MTTF) is the average time until a specific failure. Renewal metrics are used under the assumption that a system or component can be restored after a failure. The most common renewal metrics are Mean Time Between Failures

Table 1: Comparison of qualitative risk methods

	FMEA	HAZOP	STPA
Purpose	Identify and prioritize failure modes and effects of system components	Identify and assess (process) hazards and deviations from nominal scenarios	Identify and analyze consequences of hazardous control actions
Features	Analytical, proactive, focus on failure modes, prioritization is based on likelihood, detectability, and severity	Systematic, focus on deviations of system parameters using guide words such as less, more, too early, etc	Systematic, causal, focus on unsafe control actions, takes into account system control structure
Application complexity	Easy to moderate	Moderate	Moderate
Expert knowledge required	Moderate to high in failure modes of system components	High in process industry, hazards, and safeguards	High in system theory and control engineering
Ranking	Risk Priority Number	Risk matrix	Primarily qualitative analysis
Inputs	System description, system decomposition, failure mode information, failure effects	Process description, system decomposition, guide words, deviation scenarios	System control structure, unsafe control actions, hazardous states

(MTBF), number of Failures in Time (FIT), and Mean Downtime (MDT). Renewal metrics are less suitable for risk assessment because they do not directly define the likelihood of a particular failure scenario. However, they are commonly required to estimate the hazards of a specific event. Finally, continuous metrics help to represent the likelihood of an unwanted event in the form of a probability distribution. They help to model such aspects as component wear out or so-called "infant mortality" or any other time-related changes in the failure probabilities. Weibull, Exponential, Normal, Lognormal, and mixed distributions are commonly used. The exponential distribution is used because of its simplicity, and Weibull is used because of flexibility, e.g., it can model increasing, decreasing, and constant failure rates.

The metrics are used for two reasons: (1), as discussed above, to estimate the likelihood of a particular failure scenario, and (2) as reliability figures of system components that are commonly used as inputs for static and dynamic quantitative risk models. This information is either provided by the component producers or can be found in commonly used component reliability prediction guidelines like FIDES [95], NRPD [96], or MIL-HDBK-217F [97]. MTBFs and FITs are the most common metrics. Reliability data depends on multiple factors, and finding a trustworthy number for a particular system and its operational conditions is challenging. Also, these metrics are usually generic. They define the probability of a component failure but do not specify the type of failure. It is almost impossible to find metrics for specific failure modes of a component which commonly restricts the application of advanced quantitative methods.

3.3 Static quantitative methods

Quantitative risk assessment methods help not only to discover but also to numerically estimate the likelihood of failure scenarios. The component reliability data, discussed in the previous section, and the known likelihoods of other undesired events are inputs for the quantitative system-level methods. The methods such as Event Tree Analysis (ETA) [46], static Fault Tree Analysis (FTA) [47], Reliability Block Diagrams (RBD) [48], Bayesian Networks (BN) [49], and their extensions and combinations such as Cause-Consequence Diagram (CCD) method [50] or Hybrid Causal Logic (HCL) [51] are static quantitative risk methods. These methods are called static in contrast to the dynamic techniques discussed in the next section. Table 2 summarizes the features of these methods.

Event Trees (ET) explore system success and failure scenarios as a tree starting from an initiation event through other events to arrive at the success or failure completion of system operation. Usually, each event has two stochastically defined outcomes: success and failure.

Table 2: Comparison of static quantitative risk methods

	ETA	FTA	BN
Purpose	Analyze the sequence of events leading to a particular hazardous outcome	Analyze system components failures, other undesired events, and their logical combinations leading to a hazardous outcome	Analyze probabilistic relationships between undesired events
Features	Intuitive representation of failure scenarios, modeling of sequential failures	Hierarchical representation of failure causes, logical dependencies, modeling of redundancy	Modeling of probabilistic dependencies and common cause failures.
Application complexity	Easy to moderate	Moderate	Moderate to difficult
Expert knowledge required	Moderate in event modeling and probability assessment	Moderate in fault modeling and logical analysis	High in probabilistic modeling and inference
Quantification	Calculation of the probabilities of failure scenarios as a product of event outcome probabilities, prioritization of failure scenarios	Calculation of top event probabilities (e.g. system failure) as logical combination of basic events (e.g. single component faults), identification of minimal cut-sets	Calculation of posterior probabilities, calculation of interested probabilities given evidences
Inputs	Initial event, event sequences, branching probabilities	Failure modes logic (gate structure), probabilities of basic events	Events, prior and conditional probabilities

Event trees provide a transparent and intuitive representation of failure scenarios. The probability of each failure scenario is computed as the product of the probabilities of the events over the path. Rarely, the events are defined with continuous metrics. In this case, a bit more complex quantification is required. A Fault Trees (FT) describes how failures of system components and other undesired events (basic events) can combine to cause a particular failure scenario (top event). FTs utilize logical gates such as AND, OR, and K-out-of-N. For example, the system will fail if both primary and spare components fail. FT analysis is traditionally focused on the identification of so-called cut sets, unique combinations of basic events leading to the top event. The quantitative analysis yields various likelihood metrics such as the probability of system failure, the mean number of failures for discrete-time, the mean downtime (unavailability), MTTF, as well as MTBF for continuous-time FTs that can model failure rates. These metrics can be computed either directly using top-down or bottom-up methods or using underlying Binary Decision Diagrams (BDDs) models [98]. BDD-based methods are used in most cases since they allow the computation of the system unreliability and the probabilities of the cut sets at the same time. RBDs and FTs share the same underlying concept. An RBD models a system as a series of blocks connected in parallel or series configuration. Each block represents a component of the system with a failure rate. Parallel blocks indicate redundant subsystems or components, like the AND-gate in FTs. By contrast, any failure along a series path causes the entire path to fail, like the OR-gate in FTs. The underlying RBD computation methods are similar to the FTs. However, an RBD models a system success, while an FT models a system failure. RBDs and FTs are usually used interchangeably, but FTs are more popular for risk assessment because they help focus on failure scenarios rather than on system successful execution. BN are employed in the presence of statistical dependencies between the events when the fact of the occurrence of one undesired event changes the probability of another event, e.g., common cause failures. A BN represents a set of random variables, e.g., component failures and their probabilistic relationships, allowing analysts to quantify the uncertainty associated with each variable and evaluate the overall risk. BN can estimate the probability of specific risk events and their potential consequences.

The risk models mentioned above can be combined. For example, the CCD method models the failure logic (like an FT) but has the extra capability to analyze systems subject to sequential failures (like an ET). Similarly, the logic of Bow-Tie Analysis (BTA) [83] can be represented as a combination of fault and event trees, enabling its use for quantitative analysis. HCL integrates hierarchical ETs, FTs, and BNs models. HCL is proven to be an effective combination that leverages the advantages of all static risk methods, such as sequential failures (ETs), structural failure logic (FTs), and dependent events (BNs). HCL models are quantified via the transformation into a BDD.

3.4 Dynamic quantitative methods

Static risk assessment methods cannot model dynamic aspects such as component repairs, transient faults, complex control flow structures, multi-rate components, built-in fault tolerance mechanisms, and the propagation of data and timing errors, which are common in distributed systems with complex software. Essentially, static methods cannot adapt to critical system changes. Dynamic risk assessment methods, on the other hand, can address these aspects but come with increased modeling and computational complexities. Table 3 summarizes the features of three methods from this category. The following section briefly presents the most widely used dynamic risk assessment methods. Each method is designed to tackle specific challenges inherent to CPPS. For a more thorough and detailed exploration, refer to the sources in Figure 4.

Dynamic methods as a rule employ underlying Markov chain models. These models utilize probabilistic transitions to describe the evolution of a system over time from one system state to another, taking into account reliability factors such as fault activation, error propagation, error detection and correction, component and system failures, and repairs. Discrete and Continuous Time Markov Chains (DTMC, CTMC) [52] are the most commonly used models. Markov Decision Process (MDP) [53] extends DTMCs by incorporating both stochastic and deterministic transitions. Interactive [54] and Input/Output Markov chains [55] enhance the models by considering reactions to external events. Quantification of Markov models is achieved through various methods, including exact linear algebra methods, Monte Carlo simulations, and heuristic methods. Discrete-time models use transition probabilities, while continuous-time models employ continuous metrics, often limited to exponential distributions.

Stochastic Petri Nets (SPNs) [56, 57] are employed for modeling parallel processes in complex systems or systems of distributed agents. Generalized Stochastic Petri Nets (GSPNs) [58, 59] is an extension of SPNs, comprising immediate transitions that fire instantaneously and timed transitions with exponentially distributed random firing delays. SPNs and GSPNs follow the assumption of exponential firing delays. Extended Stochastic Petri Nets (ESPNs) [60] allow for generally distributed firing times. Deterministic and Stochastic Petri Nets (DSPNs) [61] enable deterministic and exponentially distributed firing delays with transitions. Markov Regenerative Stochastic Petri Nets (MRSPNs) [62] encompass immediate transitions, exponentially distributed transitions, and generally distributed timed transitions.

Dynamic Fault Trees (DFTs) [36] and their various extensions, including Fault Trees with Dependent Events (FTDE) [63], Repairable Fault Trees (RFT) [65], State-Event Fault Trees (SEFT) [64], Component Fault Trees (CFT) [66], and Boolean logic-Driven Markov Processes (BDMP) [67] form the broad class of dynamic risk assessment methods. DFTs extend intuitive, logical failure representation of static fault trees with additional dynamic gates such as PAND, FDEP, and SPARE, allowing for the modeling of complex behaviors, interactions, and failure sequences of system components [99]. For example, a SPARE gate activates spare units when primary units fail, continuing until no more spares are available. Basic events in a DFT can be dormant, active, or failed, with the failure probability of dormant components reduced by the dormancy factor. Quantitative analysis of DFTs typically involves applying standard methods to their operational semantics in terms of CTMCs, Interactive and Input-Output Markov Chains, or dynamic BNs. Other dynamic methods include the Dynamic extension for Reliability Block Diagrams (DRBD) [68], which shares similarities with DFTs, Dynamic Bayesian Networks (DBNs) [69], a dynamic extension of BNs, and Dynamic Event Trees (DET) [71] and Event Sequence Diagrams (ESD) [72], which are dynamic extensions of event trees. Dynamic methods are computationally more complex than static methods and often suffer from the problem of state space explosion, where the state space of an underlying system model grows exponentially with system complexity.

Probabilistic Model Checking (PMC) [37] is a distinct group of methods that offers a generalized approach to other dynamic risk assessment techniques. PMC methods are designed to analyze system behavior and verify properties using formal verification techniques. These methods rely on formal modeling, probabilistic specifications, and model-checking algorithms to assess probabilities of failures. Typically, PMC methods utilize various Markov or Petri models, including DTMC, CTMC, MDP, or GSPN. However, the system is often described using high-level languages such as PRISM [73], Altarica [75], or Dual-graph Error Propagation Model (DEPM) [77] formalism. DFTs can also serve as input for PMC tools like STORM. The key aspect of PMC lies in the analysis process, which is based on temporal logic queries, such as Probabilistic Computation Tree Logic (PCTL). These techniques enable the analysis of highly customizable risk metrics, such as "the probability of system recovery after a specified system failure during the defined time interval," in addition to traditional reliability metrics. Nevertheless, applying PMC techniques can be challenging and requires a high level of expertise in probability theory, formal methods,

Table 3: Comparison of dynamic quantitative risk methods

	Markov	Petri	PMC
Purpose	Quantify system failure scenarios using state models with probabilistic transitions	Efficiently analyze concurrent and multi-agent systems	Quantify the probability of user-defined temporal logic properties in stochastic systems
Features	Stochastic state transient analysis, Markov property limitation, state space explosion	Parallel processes, synchronizations, concurrency analysis	Flexible analysis with probabilistic temporal logic properties and formal methods
Application complexity	Moderate to difficult	Moderate to difficult	Difficult
Expert knowledge required	High in stochastic processes	High in concurrent processes	High in probability theory and formal methods
Quantification	Probabilistic estimation of state reachability, steady-states, safety and liveness properties	Probabilistic estimation of reachability, boundedness, liveness, reversibility	Numerical evaluation of PCTL properties
Inputs	States, transition probabilities, initial state distribution, states of interest	Places and transitions, transition activation probabilities, synchronisation, places of interest	Stochastic model (DTMC, CTMC, MDP, SPN, etc), probabilistic temporal logic property specifications

and model checking. Two widely used PMC tools are PRISM [100] and STORM [74], although there are other available tools that employ various algorithms and numerical methods.

4 Systematic Literature Review: Procedure

The risk assessment challenges discussed in Section 2.4 and the methods outlined in Section 3 establish the foundation for our Systematic Literature Review (SLR). The primary question we aim to address is, "Can the existing methods adequately address the identified challenges?" If they cannot, we must consider whether it is necessary to combine, adapt, and extend these methods or if entirely new approaches are required. With this consideration in mind, we conducted the SLR. The SLR procedures, including the search strategy, research questions, sources, and statistical methods, are detailed in this section. The procedure follows the guidelines presented in [101]. The study selection was partially carried out in the online tool Parsif.al [102].

4.1 Search strategy

1. Planning:
   - (a) Elaboration and formalization of SLR objectives and definition of Research Questions (RQs), see Table 4.
   - (b) Identification of literature sources, see Section 4.2.
   - (c) Development of search strings based on the defined RQs, see Section 4.2.
   - (d) Quality Assessment (QA) criteria identification, involving the formulation of inclusion and exclusion criteria presented as a checklist, tailored to the RQs.
2. Conducting:
   - (a) Initial study Selection: Evaluation of how well the acquired studies address the defined RQs, based on abstracts, paper contribution paragraphs, results overviews, and conclusions.
   - (b) Kappa cross-validation: A critical step to verify reviewer agreement and ensure the absence of bias, further detailed in Section 4.3.
   - (c) Detailed paper screening: Detailed filtering of studies prior quality assessment, assessing the credibility of the publication source, and the extent to which the research addressed the specific research question.
   - (d) Quality Assessment: Refinement and filtering of studies prior to data extraction, assessing the quality of writing and soundness of results across relevant papers.
   - (e) Snowballing: After the quality assessment

Table 4: Eight research questions in the SLR and corresponding literature sources addressing each question.

RQ 1[1]	Which aspects of CPPS must be covered by the risk methods? We aim to identify relevant technical aspects, such as intelligence, heterogeneity, and complexity, that must be encompassed by risk methods. We explore existing literature to discuss these critical aspects and their incorporation into risk assessment. Addressed in [103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 77, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157].
RQ 2	Which risk methods can be applied for the CPPS analysis? After identifying the pivotal aspects of CPPS, we employ the SLR to assess the suitability of existing risk methods for addressing CPPS-related challenges. Addressed in [103, 104, 105, 107, 108, 158, 159, 160, 109, 161, 110, 162, 41, 163, 164, 165, 166, 111, 112, 167, 113, 168, 114, 115, 116, 117, 169, 170, 119, 171, 120, 122, 172, 123, 124, 125, 126, 127, 129, 173, 174, 175, 176, 177, 130, 178, 179, 180, 131, 132, 181, 182, 183, 184, 185, 186, 133, 187, 134, 135, 136, 77, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 149, 150, 151, 152, 153, 154, 155, 156, 157].
RQ 3	What are the advantages and drawbacks of the existing risk methods? Our analysis focuses on the advantages and drawbacks of various risk methods, considering factors such as industrial maturity, availability of tools and benchmarks, data requirements, uncertainty sensitivity, and computational complexity. This evaluation helps us ascertain the practical applicability of each method. Addressed in [104, 110, 162, 163, 111, 167, 117, 169, 170, 123, 125, 127, 129, 173, 174, 188, 130, 179, 132, 181, 182, 183, 133, 134, 135, 136, 77, 138, 141, 147, 149, 151, 154, 156].
RQ 4	How can different risk methods be combined for a more effective analysis? Beyond evaluating individual methods, we explore the integration of multiple risk methods tailored to specific tasks, which could yield more reliable results. Addressed in [110, 163, 167, 123, 127, 173, 174, 188, 130, 132, 183, 133, 134, 136, 141, 149, 152, 155].
RQ 5	Where can the input data for the risk methods be found? Although various sources and guidelines are available, the input data for risk methods often remains generic and incomplete. We explore the challenges of obtaining suitable input data and strategies to enhance the quality and relevance of data used in CPPS risk assessments. Addressed in [108, 159, 160, 166, 113, 189, 190, 122, 123, 126, 191, 77, 154, 155].
RQ 6	In which phase of CPPS development can risk methods be applied? Different risk assessment methods are suitable for different life cycle phases of CPPS development. Stimulative methodologies like model-based Fault Injection or Software/Hardware-in-the-Loop may address specific failure modes but are applicable only after some prototypical system implementation. Conversely, analytical methods such as fault trees or Markov chains could be implemented during the earlier design phase. Addressed in [170, 124, 127, 129, 187, 136].
RQ 7	Are there any major gaps in the available risk assessment methods that should be covered? We investigate whether the defined CPPS aspects might not be fully covered by existing methods and require an extension. Our goal is to determine if there are significant gaps in risk methods that need future coverage. Addressed in [192, 132, 182, 187].
RQ 8	What are the challenges for risk assessment of CPPS that contain AI components? This dedicated RQ addresses the unique challenges of assessing risks in CPPS with AI components. We aim to understand the key differences between AI and non-AI components in risk assessment and what risk assessment solutions are currently considered in industrial standards and research communities. Addressed in [192, 123, 191, 138, 139].

[1] Please note that Sections 2.4 and 3 also describe key CPPS aspects that challenge risk analysis and available risk assessment methods. These sections are essential for understanding the issues addressed in this SLR. They were initially written based on our background knowledge and were partially updated during the course of the SLR.

stage we circulated the preliminary results and added 19 papers based on expert recommendations.

3. Analysis:
   (a) Data Extraction: Gathering necessary information from selected papers using a data extraction form. This information is crucial for addressing the RQs in the subsequent data analysis phase.
   (b) Data Analysis: Analysis of raw data from extraction forms, structured and presented in the format of a report paper.
   (c) Systematic Mapping: Mapping the SLR findings to the originally defined RQs and formulating the conclusions of the paper.

4.2 Search sources and criteria

We selected the ACM Digital Library, IEEE Xplore, Web of Science, and Scopus as our primary literature sources. The search string was formulated based on the research questions and the background information regarding risk methods and challenges discussed earlier. We considered only papers published from 1995 to 2023. The selection of papers was conducted in two stages. In the first stage, we assessed papers identified by the initial search based on their titles, abstracts, contributions, and conclusions. Through this preliminary review, we quickly discarded papers that were clearly irrelevant. In the second stage, we thoroughly read and evaluated the selected papers against our inclusion and exclusion criteria. The primary criteria were the applicability of the method presented for risk assessment of CPPS and the overall quality of the paper. To qualify for inclusion, a paper had to address at least one of the first four research questions. The detailed list of criteria is presented in the following section. The complete search string used for this process is detailed in Listing 1.

```
("Document Title": Reliability OR "Document
    Title": Risk OR "Document Title": Safety OR
    "Document Title": Dependability OR "
    Document Title": Resilience OR "Document
    Title": robustness) AND ("Document Title":
    Method* OR "Document Title": Model* OR "
    Document Title": Evaluation OR "Document
    Title": Estimation OR "Document Title":
    Prediction OR "Document Title": Assessment
    OR "Document Title": Forecasting OR "
    Document Title": Analysis)
AND ("Abstract": Method* OR "Abstract": Model*
    OR "Abstract": Evaluation OR "Abstract":
    Estimation OR "Abstract": Prediction OR "
    Abstract": Assessment OR "Abstract":
    Forecasting OR "Abstract": Analysis) AND ("
    Abstract": Cyber-Physical Production System
    OR "Abstract": Cyber-Physical Production
    Systems OR "Abstract": CPPS OR "Abstract":
    cpps OR "Abstract": Cyber-Physical System
    OR "Abstract": Cyber-Physical Systems OR "
    Abstract": CPS OR "Abstract": cps OR "
    Abstract": Industrial OR "Abstract":
    Automation OR "Abstract": Production OR "
    Abstract": Manufacturing OR "Abstract":
    Industrial Robotics OR Robot OR "Abstract":
    Smart Factories OR "Abstract": Smart
    Factory OR "Abstract": Industry 4.0 OR "
    Abstract": Industry 5.0 OR "Abstract":
    Industrie 4.0 OR "Abstract": Industrie 5.0
    OR "Abstract": Networked Automation Systems
    OR "Abstract": Digital Twin OR "Abstract":
    Advanced Engineering OR "Abstract":
    Advanced Systems OR "Abstract": Advanced
    Systems Engineering OR "Abstract":
    Intelligent OR "Abstract": Intelligence OR
    "Abstract": Artificial OR "Abstract":
    Autonomous OR "Abstract": adaptive)
NOT("Abstract": Cyber-security OR "Abstract":
    Attack OR "Abstract": Security OR "Abstract
    ": Threat OR "Abstract": Optimization OR "
    Abstract": Economic OR "Abstract":
    deployment OR "Abstract": Energy OR "
    Abstract": Grid OR "Abstract": Electric OR
    "Abstract": Medi* OR "Abstract": Health OR
    "Abstract": Rule OR "Abstract": Semiconduct
    OR "Abstract": Market OR "Abstract":
    Business OR "Abstract": Supply OR "Abstract
    ": Chain OR "Abstract": Electronic OR "
    Abstract": Logistics OR "Abstract": Coal OR
    "Abstract": Mine OR "Abstract":
    Development OR "Abstract": Real Estate OR "
    Abstract": Investment OR "Abstract":
    Financing OR "Abstract": Structure OR "
    Abstract": Dam OR "Abstract": wear OR "
    Abstract": Fatigue OR "Abstract": Power OR
    "Abstract": plants OR "Abstract": Hydraulic
    OR "Abstract": agricultural OR "Abstract":
    Oil OR "Abstract": Gas OR "Abstract":
    Pipeline OR "Abstract": Bank OR "Abstract":
    Ecological OR "Abstract": Soci*)
```

Listing 1: Search string used for the literature search.

Note that by using the "NOT" operator in the search string, we excluded other application areas to focus specifically on industrial CPPS. Although there is considerable overlap, we treat cybersecurity as a separate domain. Our risk method analysis primarily focuses on risks stemming from accidental faults rather than intentional attacks. We cover many probabilistic methods, which are less commonly applied to cyber attacks. However, numerous studies connect security and safety, such as [193].

4.3 Statistics

The initial literature search identified 2,199 unique papers out of 2,366 papers found in total. After an initial screening (step 2.a), 730 articles were selected for further detailed analysis before the quality assessment. In total, 1,634 studies were rejected during the screening and selection process. The most common reason for exclusion was that the papers were outside the project's scope, leading to the removal of 352 papers. Additionally, 140 papers were excluded due to their focus on unrelated topics such as attack trees, security, or sectors like finance and insurance, which did not

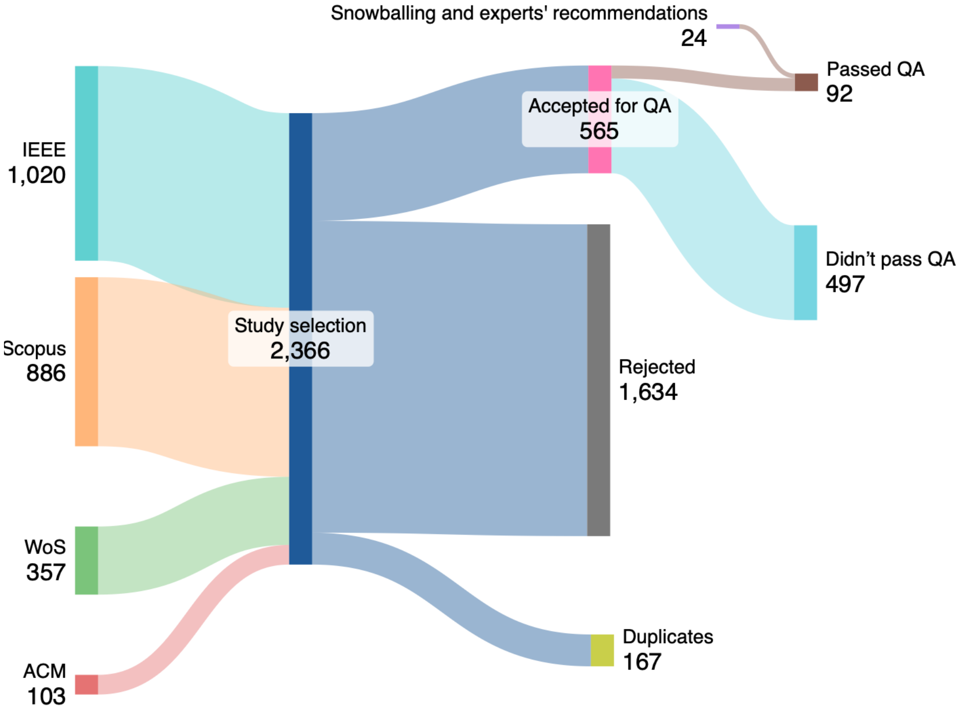

Fig. 5: Number of papers at each literature review stage.

align with the project's primary focus areas of computer science or electrical engineering. A further 161 papers were excluded because they addressed specific aspects of reliability rather than the broader, system-level term. Examples of these papers discussed frameworks for risk evaluation in maintenance, analyses of reliability and degradation mechanisms in battery performance, circuit modeling for ISO 10605 electrostatic discharge tests, and methodologies for risk indices in humanoid robots navigating on uneven terrain. In a subsequent selection round (step 2.c), an additional 165 papers were excluded based on factors such as language, source credibility, and relevance to the research questions. A Sankey diagram was constructed to visualize these processes (see Figure 5). All criteria for inclusion and exclusion were combined for clarity.

To ensure the trustworthiness of paper inclusions, we conducted a cross-validation study (step 2.b). The experts involved had complete autonomy in deciding which papers to include or exclude, without any pre-set quotas or targets. This approach follows the free-marginal distribution principle [194], which necessitates the use of an appropriate measure for inter-rater agreement to counteract the effects of prevalence and bias [195]. Consequently, we calculated the inter-rater agreement using Randolph's free-marginal multirater kappa statistic, which is influenced solely by the number of rating categories and is unaffected by the symmetry of marginal distributions [196]. The calculated $\kappa = 0.55$ indicates a moderate level of agreement among raters. Based on this, the inclusion of 565 papers for further analysis is justified.

After a detailed screening of the papers, we conducted a quality assessment of the accepted papers (step 2.d) using a weighted scoring system to evaluate each study, see Table 5. The primary research question was assigned 10 points, underscoring its critical importance to our research objectives. Research questions 2 to 4, important for refining the study's relevance and quality, were allocated up to 1.5 points each. Lesser but still significant questions, 5 to 7, were capped at 1 point each. Papers needed a minimum score of 15 points to advance to the data extraction phase, a threshold set to ensure that only high-quality and highly relevant papers were included. Following this criterion, 68 papers met the threshold and were selected for detailed data extraction, while 479 papers did not meet the score requirement and were excluded. Subsequent to the initial quality assessment, we engaged in a snowballing step to ensure comprehensive coverage of relevant literature (step 2.e). This involved sharing initial findings with a panel of subject matter experts, who identified an additional 24 papers that were initially overlooked for various reasons. Following screening and quality assessment, 92 papers were included for data extraction.

5 Systematic Literature Review: Results

Each of the 92 papers was evaluated based on how comprehensively it addressed the defined RQs. Not all papers were fully effective in covering each RQ. The coverage of each RQ by the papers is shown in Table 4, with

Table 5: Quality assessment checklist for final paper acceptance decisions.

No.	Description	Points
1	Coverage of at least one of the first four RQs: Evaluation of the depth and comprehensiveness of the answers provided.	10
2	Maturity of the method: Presence of an implemented tool, description of an algorithm or mathematical model, or presentation of the idea as a concept.	1.5
3	Rigor of method evaluation: Representativeness of the case study, quality of experiments, and benchmarks used.	1.5
4	Comparison with other methods: Analysis based on state-of-the-art and experimental outcomes.	1.5
5	Clarity of writing: Quality of explanation regarding the problem of study and research context.	1
6	Clarity of contribution: Transparency in presenting the contributions and limitations of the study.	1
7	Validity of the research: Assessed by the number of citations received.	1

RQs 1 to 4 receiving the best coverage. It was not possible to completely address RQs 5 to 8 using only the selected papers; thus, we supplemented these questions with additional background knowledge and conclusions drawn from the information in the papers. The results of the SLR are presented in this section, organized by the corresponding RQs.

5.1 Which aspects of CPPS must be covered by the risk methods? (RQ1)

Table 7 summarizes the answer to the first research question. It identifies specific aspects of Cyber-Physical Production Systems (CPPS) that should be considered by risk assessment methods. We outline thirteen distinct characteristics that influence the risk profile of CPPS, ranging from AI components and autonomy to the complexity of system behaviors and structures. These thirteen aspects align with and provide detailed context for the challenges discussed in Section 2.4. The right column of the table highlights how each aspect presents unique challenges to risk assessment.

5.2 Which risk methods can be applied for the CPPS analysis? (RQ2)

The answer to RQ2 is structured in connection with the identified thirteen aspects. Our objective was to determine which aspects could be addressed by each group of risk methods. Each paper evaluated discusses the application of one or several risk methods to a CPPS-related system, as outlined in Section 3. We categorized the risk methods into the same categories as shown in the tree diagram in Figure 4. We put Machine Learning (ML) and Deep Learning (DL) methods that play the role of risk estimators into a separate category (ML/DL). We also separately treated Fault Injection (FI) methods, which utilize executable system models, prototypes, or the CPPS themselves to inject faults and analyze the consequences. FI methods are mostly based on the Monte Carlo method; more advanced methods enhance FI with Reinforcement Learning (RL) for guided FI that help find either the most common or the most dangerous failure scenarios. In general, we can distinguish two classes of risk methods described in the papers. The first class includes methods targeting specialized aspects. For instance, BinFI [191] is a FI tool specifically for ML applications. Another example is the research by [113] that presents a methodology for modeling the reliability of intelligent mechatronic systems that naturally covers the autonomy aspect of CPPS. To the second class belong the risk methods that can evaluate several CPPS aspects. Notably, ML-based approaches are versatile [108] [189], enabling the evaluation of dynamic and complex CPS based on time series data from different system levels and components. The three tables below provide an overview of the risk models covering the identified CPPS aspects: (1) qualitative methods in Table 8, (2) quantitative static methods in Table 9, and (3) quantitative dynamic methods in Table 10. These tables follow the structure of the tree in Figure 4. FI and ML/DL methods are included under dynamic methods since they consider the system changes over time. Additionally, several papers propose specific types of metrics that measure risk based on physical or other system properties, rather than suggesting system-level risk methods. These papers were classified under the Metrics category, added to the static methods in Table 9.

Based on the results of our SLR, we estimated how effectively each risk method (RM) addresses each CPPS aspect. Our findings are summarized in Table 6. Each cell of the table indicates whether a particular method provides any advantages for a specific aspect. We considered not only the number of papers suggesting the use of each method but also the context and manner in which they are recommended. For example, although many papers suggest using FTs, their application to future CPPS is questionable. Conversely, while only one paper suggests using FRAM [152], the method itself appears very promising for considering human and system adjustments. Additionally, we incorporated background knowledge in our analysis. For instance, Petri nets are known to be very helpful for distributed systems as they model parallel processes, and timed Petri nets focus specifically on timing analysis. Similarly, formal verification methods, such as model checking and PMC, are effective in optimization to combat state space explosion, thereby aiding in the analysis of systems with complex software components.

Table 6 reveals that among the classical qualitative methods, STPA is very promising. However, STPA focuses on the identification of failure scenarios rather than quantitative analysis. Additionally, STPA is not particularly helpful for software or AI components, as it primarily targets control architecture. Other classical qualitative methods such as FMEA and HAZOP are still useful for preliminary analysis but are generic and usually conducted by experts. For complex CPPS systems, automated approaches are needed. Classical quantitative static methods such as FTA and ETA struggle to address CPPS aspects effectively. Dynamic methods are clearly necessary. BNs are promising within this group, as they provide advantages in modeling probabilistic dependencies between undesired events. From the dynamic methods, PNs are helpful for parallel systems and timing analysis, especially when combined with PMC using tools like STORM. While DFTs are useful, it may be more effective to directly analyze underlying Markov models for complex systems. In some cases, Markov- and Petri Nets-based methods can leverage reconfigurability and repurposeability. But this requires extension of the methods with model-to-model transformation [197]. PMC is promising, provided that sufficiently detailed system models can be created for analysis. FI is very promising because it does not require building a formal model. However, FI can only reveal some failure scenarios and cannot guarantee comprehensive coverage. Nonetheless, smart FI is potentially the only option to address AI components, which are often black boxes and difficult to analyze analytically. It also important to mention that research groups are actively working on the formal verification of AI [198]. Finally, as expected, ML/DL methods are very promising. This group is diverse, with ML/DL being used directly as risk estimators or predictors. Additionally, ML/DL methods can be effectively combined with other methods, such as RL to guide FI or DL for solving and optimizing other risk models.

5.3 What are the advantages and drawbacks of the existing risk methods? (RQ3)

It is important to note that each of the mentioned methods is actually a group of methods with numerous extensions and combinations. Table 11 addresses the advantages and drawbacks of the canonical versions of these methods. Some of the facts in Table 11 are well-known and were confirmed by the SLR. Also, note that some advantages and disadvantages were already addressed in Section 3 and summarized in Tables 1, 2, and 3, and the SLR yielded similar results.

5.4 How can different risk methods be combined for a more effective analysis? (RQ4)

As shown above, each risk method has its strengths and weaknesses. They address certain aspects of a system but not all. Intuitively, it is logical to combine methods to enhance their modeling power. In the SLR, we explored how authors combine different methods. Table 12 provides an overview of how various RMs can be combined. The table was also extended with three well known combinations that were not a part of SLR: (i) HCL, which combines ESDs with FTs and BNs; (ii) CCD, which combine ETs and FTs; (iii) State Event Fault Trees (SEFT), which incorporate features of FTs and Markov Chains. Also, because the scope of our study is constrained, see Listing 1, we have omitted several combinations. For example, we are aware of various combinations of STPA with other methods such as STRIDE, FTA, FMEA, BBN, ESD, and many more.

Although combined methods offer broader coverage of CPPS aspects, we did not identify a single universally effective approach. This partially answers RQ7, being one of the major gap of the risk models landscape. We identified several noteworthy approaches that could be candidates for further expansion or combination to bridge the gap in risk assessment. For example, the author in [123] proposed an approach that combines FI with STPA. Another paper, [133], presents a FI approach combined with Formal Verification. For automated risk modeling for system-wide safety analysis, the authors presented the tool SafeATAC [173], which combines different risk models for evaluating various system levels. These methods are good examples of close, but not complete, coverage of the identified risk-challenging CPPS aspects.

Table 6: CPPS aspects and risk methods.

	Qualitative methods					Quantitative static methods				Quantitative dynamic methods					
CPPS aspect	FMEA	HAZOP	FRAM	STPA	FV	ETA	FTA	RBD	BN	Markov	Petri	DFT	PMC	FI	ML/DL
AI components	-	-	-	-	+	-	-	-	-	-	-	-	+	+++	++
Autonomy	-	-	-	++	++	-	-	-	+	+	-	-	++	+++	+++
Intelligence	-	-	-	-	++	-	-	-	+	-	-	-	++	+++	++
Heterogeneity	++	+	+	++	+	++	++	+	++	++	++	++	++	+	+++
Distributed and networked	-	-	-	-	-	-	+	+	++	-	+++	+	++	-	+
High structural complexity	+	-	-	++	++	-	++	+	++	-	-	-	+	+	+
High behavioral complexity	+	-	-	+	++	-	-	-	-	+	++	++	++	+++	++
Reconfigurability	-	-	+	-	++	-	-	-	+	-	-	-	++	++	++
Repurposeability	-	-	-	-	+	-	-	-	-	-	-	-	+	++	++
Software intensity	-	-	-	-	++	-	-	-	++	++	++	+	+++	+++	-
Time dependencies	-	+	-	++	++	++	-	-	-	+	+++	++	+++	-	-
Human-in-the-Loop	+	++	++	++	+	+	+	-	+	+	+	+	-	-	+
Sophisticated Failure Scenarios	+	-	-	++	++	-	-	-	+	++	++	++	+++	++	+

-	Not advantageous
+	Applicable
++	Usefull
+++	Advantageous

Table 7: CPPS aspects that should be covered by risk methods and respective challenges in risk assessment.

No.	CPPS Aspect	Risk assessment challenge
1	AI components: The system is controlled by learning algorithms (ML or DL) or contains AI-driven components.	Challenges include the reliability analysis of AI components, their "black box" nature, and susceptibility to errors in training data and adversarial attacks.
2	Autonomy: The quality of being self-governing, capable of making autonomous decisions.	Increases behavioral complexity with unpredictable execution scenarios and uncertainties.
3	Intelligence: Ability to maximize goal satisfaction through adaptive behavior in dynamic, uncertain environments.	Also increases behavioral complexity and the difficulty of thorough risk analysis.
4	Heterogeneity: The system comprises diverse components such as software, hardware, and physical elements.	Diverse fault types and their interactions make increases the number and complexity of failure scenarios.
5	Distributed and networked: The system includes spatially dispersed, networked, and task-dedicated components.	Network faults and synchronisation mechanisms increase the number and complexity of failure scenarios.
6	High structural complexity: A high number of components and subsystems, with complex compositions and interfaces.	The increased number of components, their hierarchy and interactions of subsystems leads to a potential exponential growth in failure modes.
7	High behavioral complexity: Characterized by higher-dimensional state spaces (e.g., multiple use cases, operating modes, system states) resulting in numerous potential execution scenarios.	Identifying potential failure scenarios becomes challenging due to huge number of execution possibilities.
8	Reconfigurability: Capability to add new components or modules easily, either during operation or between operations.	Continuous risk assessment is needed due to potential significant changes in behavior after each update.
9	Repurposeability: Ability to adapt the system for different purposes, such as reconfiguring for new products or tasks.	Each new configuration introduces new risks, requiring adaptive and dynamic risk models.
10	Software intensity: Marked by extensive and complex software components, interfaces, and cloud services utilization.	Complex software layers and interactions complicate the detection and mitigation of software failures. Advanced method for SW reliability and error propagation are required.
11	Time dependencies: Characterized by time-critical processes, potentially requiring real-time responses.	Failures can be originated because of the timing errors. Risk methods that consider time and temporal dependencies are required.
12	Human-in-the-Loop: Integrates human interactions within its operational framework, treating humans as essential components of the system.	Humans introduce variability and unpredictability, complicating risk assessments. Higher safety requirements because of the human in close proximity to potentially dangerous mechanical parts.
13	Sophisticated failure scenarios: Subject to failures inadequately addressed by standard risk analysis methods like FTA or Markov models.	Non-standard failures require innovative and possibly customized risk assessment techniques.

Table 8: CPPS aspects addressed by qualitative methods.

No.	CPPS Aspect	FMEA	HAZOP	FRAM	STPA	FV
1	AI components				[123]	[139,145]
2	Autonomy				[123]	[115,117]
3	Intelligence					
4	Heterogeneity	[104,134]	[104]	[152]	[149]	[135,136,137,139,146]
5	Distributed and networked	[104,134]	[104]	[153]		[135,136,147]
6	High structural complexity	[167,183, 141]		[152]	[123,141]	[119,133,137,146,147]
7	High behavioral complexity	[183,141]	[140]	[152,153]	[123,141, 149,155]	[133, 135, 136, 137, 139, 144,146,147,155]
8	Reconfigurability	[183,134]		[152]		[117,119]
9	Repurposeability	[183]				[117]
10	Software intensity	[134]				
11	Time dependencies	[161]		[153]		[135,137,136]
12	Human-in-the-Loop	[156]	[165,112, 140,154]	[152,153]	[155]	[115,137,139,155]
13	Sophisticated failurescenarios	[161, 167, 174, 134,156]	[165,112, 154]	[153]	[123, 149, 150, 151,155]	[115, 117, 119, 121, 133, 136,137,146,155]

Table 9: CPPS aspects addressed by static methods.

No.	CPPS Aspect	ETA	FTA	RBD	BN	Metrics
1	AI components	[142]				
2	Autonomy	[162]	[175]			
3	Intelligence				[113]	[192]
4	Heterogeneity		[106,107,163,111, 170,176,134,137]		[163,131,132]	[189,118]
5	Distributed and networked		[106,188,134]	[188]		
6	High structural complexity	[162]	[170,122,173,186, 137,127]	[173,177, 184]	[167,127,182]	
7	High behavioral complexity	[142]	[122,186,137,127]		[127,132,182]	[192,128,148]
8	Reconfigurability		[170,186,134]		[131]	[128]
9	Repurposeability	[183]				
10	Software intensity		[134]			[190]
11	Time dependencies		[111,137]		[132,157]	[189,171,192]
12	Human-in-the-Loop		[137]		[157]	
13	Sophisticated failurescenarios		[107,169,170,134, 137,127]		[167, 127, 132, 157]	[189,128]

Table 10: CPPS aspects addressed by dynamic methods.

No.	CPPS Aspect	Markov	Petri	DFT	PMC	FI	ML/DL	Other
1	AI components	[138]			[138]	[123,191]	[192]	
2	Autonomy	[164]			[117,120]	[123]		
3	Intelligence						[192]	
4	Heterogeneity	[103,159, 114, 116, 137,138]	[105, 158,111, 130,149]	[130,132]	[159,77,138, 146]	[143]	[189,180]	
5	Distributed and networked	[114]			[77]		[172,126, 108]	
6	High structural complexity	[183,137, 173]	[109, 168,130]	[125,130, 181,187]	[120,185,77, 146]	[123,133]	[172,126]	DET [173], FCM [178], DBN [183]
7	High behavioral complexity	[110,116, 179, 183, 137,138]	[109, 110,168, 144,149]	[132,181]	[185,77,138, 145,146]	[123, 124, 132, 133,143]	[192]	DBN [183]
8	Reconfigurability	[183]		[125,129, 187]	[117,120]		[126,180]	DBN [183]
9	Repurposeability			[125]	[117]			DBN [183]
10	Software intensity	[166]		[187]		[191]	[108]	
11	Time dependencies	[110, 41, 179,137]	[110, 111,130]	[130,132]	[77]		[189,192]	
12	Human-in-the-Loop	[137]	[105, 158]					
13	Sophisticated failure scenarios	[166,116, 137]	[168, 149]	[125,132, 181]	[117, 120, 174,77,146]	[123,124, 133,143]	[160,189]	FCM [178]

Table 11: Advantages and drawbacks of the risk methods.

Method	Advantages	Drawbacks
FMEA	Well-known; rich tool and guidelines support; helps analyze failure modes of different nature; can be combined with other methods	Too generic; usually manual, however can be also semi-automated with MBSE tools; relies on expert knowledge; not scalable
HAZOP	Systematic and thorough; identifies potential hazards of different types	Not automated; relies on expert knowledge; usually manual, however can be also semi-automated with MBSE tools; qualitative only
FRAM	Considers variability and imperfection of systems; designed for socio-technical systems	Relatively limited tool support; qualitative only; manual; relies on expert knowledge
STPA	Effective for identifying failure scenarios of control systems; systematic and well-structured; large community; rich tools and guidelines support	Partially automated; focuses on qualitative analysis[2]; requires relatively high expert knowledge
FV	Ensures correctness through formal methods such as model checking; guarantees all failure scenarios are considered within the provided system model; rich tool support; automated; many optimization algorithms to combat issues like state space explosion	Requires detailed system models; results depend on the quality of the system model provided; subject to state space explosion for complex systems; high expert knowledge required
ETA	Effective for event sequence analysis; simple; well-known and widely accepted; rich tool and guideline support; useful for high-level analysis of safeguarding systems; visual representation of failure sequences	Not scalable; the same knowledge can be represented with other methods more compactly; suitable only for high-level analysis; static, does not consider dynamic changes in the system over time
FTA	Logical and visual representation of system failure as combination of component failures or other undesired events; well-known and widely accepted; rich tool and guideline support; methods for automated FT generation from various system models are available; efficient BDD-based solvers	Static, limited in handling dynamic aspects; cannot handle dependencies between basic events; cannot model complex failure scenarios beyond the boolean logic of component failures and undesired events
RBD	Visual representation of systems with redundant components; good for modeling networks; shares the same underlying concept with FT; same computation methods can be used	Same as for FTA; more suitable for reliability analysis rather than risk assessment where the probability of failure is in focus rather than system successful execution; limited tool support compared to FTA
BN	Can model probabilistic dependencies of undesired events and dependent component failures; useful for uncertainty analysis; allows providing observations and evidence for more precise analysis; relatively rich tool support; methods for automated BN generation from various system models are available	Computationally intensive; static, limited in handling dynamic aspects; requires lots of quantitative input data (conditional probabilities) for each node; high expert knowledge required
Markov	Can model dynamic changes of system and components as a state transition system; can be used for PMC; relatively rich tool support; methods for automated generation from various behavioral system models are available; can be used as a "back end" for other dynamic risk methods such as DFT; can model more complex failure scenarios than static methods	Computationally intensive; state space explosion issues for complex systems; reaches its full potential only in combination with PMC and advanced solvers; only simple Markov chains are human readable; manual application is complicated; memoryless (Markov) property, transition probabilities/rates are independent; high expert knowledge required

[2] The STPA community tends to criticize quantitative risk assessment, arguing that the resulting numbers depend on multiple factors, can be highly erroneous, and may be misinterpreted by safety engineers, potentially leading to high risks. Instead, they suggest using STPA to identify and handle all potential risk scenarios without relying on probabilistic quantification. Therefore, the absence of quantification is not interpreted as a drawback by the STPA community.

Petri	Same as for Markov; effective for modeling parallel and concurrent processes; powerful method for modeling timing requirements (timed-PNs); rich tool support; many optimization and analysis methods; can analyze specific properties such as boundedness, liveness, and reversibility; can model more complex failure scenarios	Complex to construct manually; computationally intensive; supports only exponentially distributed events (in case of classical SPNs); high expert knowledge required
DFT	Same as for FTA; set of useful dynamic gates that allow modeling dynamic changes of system and components; intuitive FTA interface; easier to manually model compared to Markov or Petri; relatively good tool support; methods for automated generation of DFTs from various system models are available; PMC is possible but limited compared to Markov or Petri	Supports only limited (but useful) dynamic features; can model a limited set of dynamic failure scenarios; not suitable for modeling parallel processes; requires transformation to a Markov model for quantification
PMC	Same as for Markov, Petri, or DFT, since PMC can be done on these types of formal stochastic models; same as for deterministic model checking but extended with quantification; relatively rich tool support; can model and quantify any failure scenario allowed by the provided model using PCTL temporal logic requests	Requires detailed and probabilistic system models; high expert knowledge required in stochastic models and PCTL; complex and computationally intensive underlying methods; despite many optimization methods and advanced algorithms, still prone to state space explosion
FI	Does not require transformation/abstraction to any formal model; reveals realistic heterogeneous failure scenarios; usually easy and fast to implement; promising combination with reinforcement learning for guided and efficient FI; promising for analysis of AI components and complex intelligent systems when the adequate abstraction to formal models is impossible or impractical	Cannot guarantee comprehensive coverage of failure scenarios; requires a vast amount of system runs to cover as many fault types, parameters, times, and places of injections; usually used for verification purposes, not for risk assessment
ML/DL	Promising for analysis of AI components; promising for systems with high autonomy, heterogeneity, and reconfigurability/repurposeability because of the learning capabilities; very diverse application potential from generation and quantification of risk models to online system monitoring and anomaly detection; many advanced neural network architectures for different tasks; effective application as risk estimators/predictors; vast potential of combination with other methods such as FI or PMC	Requires huge datasets; complex to get large failure data sets since failures are rare events; limited access to the accidents and incidents data; black box methods (require methods for explainability); the correctness of the analysis cannot be guaranteed (research groups work on verification of neural networks); training and tuning are computationally intensive; limited interpretability of the analysis results

Table 12: Combinations of risk methods.

Risk Methods	How the combination was achieved
FMEA + STPA [141]	FMEA is complemented by insights from STPA.
SafeSysE = FMEA + FTA [134]	MBSA approach for automated generation of complementary FMEA and FTA from SysML models.
FMEA + BN [167]	FMEA is enhanced with a fuzzy Bayesian Network and the fuzzy best-worst method.
pFMEA = FMEA + PMC [174]	Probabilistic FMEA (pFMEA) integrates stochastic model checking (PRISM) techniques into the FMEA process.
FMEA + Markov + DBN [183]	FMEA and Markov chains of independent components are unified by Dynamic Object-Oriented BNs.

STPA + Petri [149]	Scenarios are identified with STPA and probabilities are quantified with SPNs.
STPA + FI [123]	Fault injection is guided by STPA.
FRAM + FV [152]	Model checking is used to aid FRAM.
FTA + BN [127]	Extended FTA for failure propagation is transformed into Bayesian Networks.
PRM = BN + FTA [163]	Probabilistic Relational Models (PRM) combine the probabilistic reasoning capabilities of BNs with the logical structuring of FTAs.
Petri + Markov [110]	A Hybrid Stochastic Petri Net (HSPN) models logical relationships between failures, which are transformed into an isomorphic Markov chain to solve system reliability.
SafeATAC = DET + FTA + RBD + Markov [173]	Combines four functional layers exploiting dynamic event trees, fault trees, RBDs, and Markov models.
DFT + Petri [130]	DFTs are translated to GSPNs, and fuzzy set theory combined with expert judgments is used to estimate failure data for basic events.
FI + FV [133]	Invariants generated with property-directed reachability model checking are used to speed up Monte Carlo fault injection.
DFT + BN [132]	DFTs are transformed into a Dynamic Evidential Network, which is a dynamic version of the combination of BNs and evidence theory.
COMPASS = FV + FI + Markov + FTA + FMEA [136]	An integrated MBSE approach based on FV techniques that includes fault injection, Markov analysis, and generates fault trees and FMEA tables.
HCL = ESD + FTA + BN [51]	A well-known risk method that combines: (i) ESDs at the top level to model scenarios, (ii) FTA for computing branching probabilities, and (iii) BN at the lower level to model common cause failures.
CCD = ETA + FTA [50]	A graphical and analytical model that combines event trees and fault tree logical gates.
SEFT = FTA + Markov [64]	Fault trees extended with probabilistic state models.
STPA + FV [155]	STPA as first step for FV to identify the safety constraints.

Table 13: Risk method and corresponding phases of system development life cycle.

Life cycle phase	Risk methods
Requirements	Formal Verification [115], PMC [120]
Design	FMEA [104, 161, 167, 174, 134], HAZOP [104, 165, 112, 154, 156], STPA [155, 141, 149, 150, 151, 123], FRAM [152, 153], Markov chains [103, 159, 110, 164, 114, 116, 173, 137, 138], Petri nets [105, 158, 109, 110, 168, 130, 144], ETA [162, 111], FTA [106, 107, 163, 111, 169, 170, 122, 173, 175, 176, 186, 134, 137, 127], DFT [125, 129, 130, 132, 181, 187], DET [173], RBD [173, 184], BN [163, 167, 113, 127, 132, 131, 182, 157], PMC [159, 117, 174, 185, 138, 145, 146], Formal verification [155, 115, 117, 121, 133, 135, 136, 137, 139, 144, 146, 147], DL [160, 192, 172, 126], FI [123, 124, 133, 143], FCM [178], Metrics [190, 170, 171, 192, 128]
Implementation	ML/DL [108, 143], Markov chains [41, 183], FMEA [183], DBN [183], FI [143]
Testing	DL [160, 143], Metrics [118, 171, 128], Fromal verification [133], PMC [120], FTA [175], DFT [130], Markov chains [183], Petri nets [130], FCM [178], FMEA [183], DBN [183], FI [133, 191, 143]
Operation	ML/DL [108, 160, 189, 192, 180, 143], FI [143], Markov chains [41, 166, 179, 183, 138], Petri nets [130, 149], Metrics [189, 118, 192, 148], Formal verification [119, 147], PMC [77, 138], ETA [142], DFT [129, 130, 187], FTA [188, 175], RBD [188, 177], FCM [178], DBN [183], FMEA [183, 141], HAZOP [140]

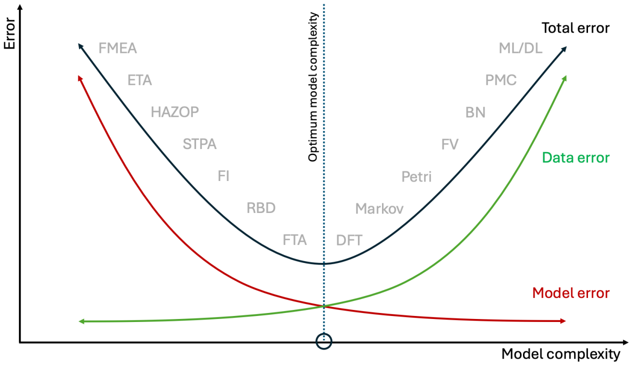

Fig. 6: Risk assessment error combines model error, which reflects how accurately the model represents the system, and input data error, which indicates the accuracy of model parameters, such as probabilities of failure of system components.

5.5 Where can the input data for the risk methods be found? (RQ5)

While addressing the research question concerning the search for input data for risk assessment methods in CPPS, we encountered significant gaps in the available literature. Although numerous sources and guidelines exist, the input data for these methods frequently remains generic and incomplete. This issue is compounded by the fact that not all methods have been rigorously tested on real systems, and comprehensive benchmarks that could provide a more detailed evaluation are scarce. Our investigation highlights the pressing challenges in acquiring suitable input data and delineates potential strategies to improve the quality and relevance of data used in CPPS risk assessments. The scarcity of empirical testing and the lack of detailed benchmarks limit our ability to definitively answer this research question, emphasising the need for more targeted research in this area.

In Section 3.2, we mentioned that the probabilities of system component failures, which are essential input data for system-level reliability and risk models, can either be provided by the component producers or found in commonly used component reliability prediction guidelines such as FIDES [95], NRPD [96], or MIL-HDBK-217F [97]. This data has two main issues. First, the provided data is averaged, meaning that specific failure probabilities for a particular component under particular operational conditions and environmental factors are not available. Second, the data is typically given as FIT or MTBF for generic component failures. For example, the data might indicate a sensor failure without specifying the nature of the failure, such as what is the probability of a delayed message, or what is the probability of a stuck-at-value conditions. This limitation restricts the application of complex methods needed to model specific failure conditions or nontrivial error propagation scenarios.

Figure 6 illustrates these issues using the bias–variance problem analogy. The error in risk assessment can be influenced by two factors: model error and data error. Model error reflects how accurately the selected model represents the system. For instance, FMEA does not consider system structure at all and relies solely on expert opinion. In contrast, methods like STPA and FTA are based on system architecture, while Markov models represent more complex system dynamics. In Figure 6, the risk models are roughly sorted according to their complexity. However, this is a very broad assumption that heavily depends on the level of detail in which the analysis is conducted with each model. Generally, more detailed models yield more precise results. Data error, on the other hand, reflects the accuracy of the input data fed into the model. For example, FTA requires the probabilities of system component failures, or Markov models require the probabilities of state transitions. The more accurate these estimations, the better the results. In general, as models become more complex and advanced, they require more input data and become increasingly sensitive to the quality of this data. Consequently, as model complexity grows, model error decreases, but data error increases. Thus, the choice of models depends on the available data. With the progression towards future CPPS, which present the aforementioned aspects, more advanced models are required. However, these models demand a large amount of high-quality input data, which is difficult to obtain. This poses a significant open question in the field - how to shift the Optimal Model Complexity point in Figure 6 to the right.

5.6 In which phase of CPPS development can risk methods be applied? (RQ6)

To identify gaps in risk assessment for CPPS, one of the main questions to address is the limitations of method applicability. Specifically, these limitations can pertain to different phases of a system's life cycle. Some methods are applicable only during the design phase, while others are suitable for the operational phase. Table 13 provides an overview of which phases the risk analysis methods from the SLR can be applied. Specific papers are cited as references within the corresponding life cycle phases. The results are rather straightforward and align with our expectations. Formal methods can be applied as early as the requirements definition phase. Methods like FTA, RBD, or BN require a preliminary high-level system design, such as component diagrams, while Markov chains require state machines. More advanced methods necessitate system prototypes or already implemented systems to gain sufficient insight into system behavior. FI methods, obviously, require at least a detailed executable model of the system. Similarly, ML/DL methods are applied in later phases because they require data for training and tuning. Another observation is that the later the phase, the more information is available about the system, leading to better risk assessment results.

5.7 Are there any major gaps in the available risk assessment methods that should be covered? (RQ7)

Answering RQ1, we have identified thirteen CPPS aspects that challenge the risk assessment of future CPPS. In addressing RQs 2-4, we determined how available risk methods can address these challenges. Based on this information, we can define the following six gaps that must be addressed. We do not aim to close these gaps but will try to suggest some ideas.

1. How to combine risk methods? Effective combinations of methods and hybrid risk models are crucial, as no single risk model can cover all aspects, as shown in Table 6. It is important to define interfaces between different risk models for easier integration. Developing a single model to cover all aspects is inefficient; instead, a theory that unites these models may be helpful. Practically, it is essential to develop hybrid risk models, solvers, and exchange formats. Several attempts include OpenPSA [199,200], OpenPRA [201, 202], SCRAM, etc. Additionally, following RQ6, it is important to combine methods with the system development process, ensuring that risk models are improved and synchronized throughout system development.
2. Where to get failure data to feed risk models? This question, raised in RQ5, highlights the need for data for complex risk models. This topic is closely related to the accident investigation which is a separate research field. Potential solutions include using AI methods for system monitoring, anomaly detection, and storing anomalous data to create larger datasets. Methods for adapting failure data from one system to another should also be developed.
3. How to assess risk facing continuous changes? Table 6 identifies the gap for reconfigurability/repurposeability. Software-defined systems will require continuous adaptation of risk models and assessments before each software update. Therefore, synchronization mechanisms between the system and the hybrid risk model are essential. This could be based on model-to-model transformation methods, or the digital twin paradigm, which can serve as a playground for FI and risk model adaptation, integrating risk assessment into the system lifecycle.
4. How to model complex failure scenarios? Systems are becoming prone to complex failure scenarios that only advanced risk models based on PMC can describe. However, PMC methods in risk assessment need further development, as they require high expertise in formal methods. Interfaces for risk and safety engineers must be simplified, and specific PMC tools for risk assessment should be developed, offering advantages like automated generation and synchronisation with system models.
5. How to identify critical failure scenarios? Even with detailed digital twins, and topnotch risk models, identifying failure scenarios among billions of execution possibilities remains challenging. Intelligent FI methods, extended with AI such as RL, might be helpful. However, injected faults must be realistic, and finding fault samples is difficult. One idea could be using generative adversarial networks (GAN) to create realistic faults. Additionally, large language models (LLM) could help guide and propose potential failure models at a high-level supporting experts.
6. How to assess the risk of AI components? Lastly, assessing the risk of safety-critical AI components will be discussed in the next subsection as a dedicated RQ8.

5.8 What are the challenges for risk assessment of CPPS that contain AI components? (RQ8)

Non-AI, traditional software is deterministic, predictable, and explainable. This allows the application of formal analytical methods from FV and PMC. In contrast, AI software operates as a black box, where weights and hyper-parameters of a neural network lack explicit meaning. Traditional formal methods and conventional software testing techniques are in general ineffective for AI software. Simply identifying corner cases and extrapolating them to the entire state space as it usually done with software testing is also insufficient since AI software cannot guarantee continuity. Therefore, FI methods provide a promising solution. This is highlighted in the first row of Table 6. FI tools like the already mentions BinFI [191] or several other methods such as TensorFI [203], InjectTF [204], or Ares [205] automate this task. Too achieve better performance FI can be extended with RL techniques that could help to identify fault parameters for the most critical failures. From the SLR papers we could highlight three papers that address risk assessment of AI components.

However, we should mention the several research groups who are actively working on the formal verification of AI [198]. From the SLR papers we could highlight the following three works. The authors in [192] propose a safe learning framework for complex dynamical systems. It defines a safe region and uses a supervisory control strategy to switch actions between the learning-based controller and a predefined corrective controller. The method presented in [138] addresses the challenges of dependability in Deep Reinforcement Learning (DRL) for Robotics and Autonomous Systems (RAS) by defining dependability properties in temporal logic and modeling them using a DTMC. It employs PMC to provide a holistic risk assessment, uncovering the need for customized optimization objectives and offering sensitivity analysis to environmental disturbances. The method presented in [139] introduces a risk modeling approach tailored to Collaborative AI systems, which aim to work with humans in a shared space to achieve common goals. This risk model includes goals, risk events, and domain-specific indicators that potentially expose humans to hazards, and it drives assurance methods through insights extracted from runtime evidence, demonstrated via an Industry 4.0 example involving a ML-equipped robotic arm collaborating with a human operator.

The field of AI is vast, and different methods are necessary for the risk assessment of various safety-critical AI components. Examples provided indicate that no single methodology is universally applicable. Additionally, the authors address AI explainability and trustworthiness in relation to system safety. Explaining how a component works is crucial for trust. Safety engineers struggle to formally verify large, complex systems due to the state space explosion problem, yet they trust deterministic and explainable software, allowing for effective safeguards. Thus, AI explainability is essential for risk assessment and safety assurance

6 Comparison with other meta-reviews

This section answers the question why this SLR was required and provides a comparative analysis with six identified meta-reviews that align with our research questions. We highlight difference in contributions, identify common challenges, and propose areas for further investigation.

Bolbot et al. [15] presents a paper review that closely aligns with the scope of our work. This paper is the most close and also provided the biggest influence into our SLR. While Bolbot et al. provide a comprehensive overview of safety assurance methods for generic CPS from 2006 to 2019, our review is extending to the end of 2023, and focuses exclusively on CPPS in the industrial and automation context. Out of the 68 papers reviewed by Bolbot et al., approximately 22 specifically address Industrial CPS, whereas our review encompasses 92 papers in this sector alone. Notably, the reviewed paper does not include the application of AI methods for risk assessment, which is a significant component of our analysis. We cover the integration and combination of the risk methods and specifically investigate the application of ML/DL techniques and their implications for CPPS.

Leimeister et al. [206], present a similar SLR encompassing 67 papers and employing comparable methods. This review excludes Petri nets, MML/DL, and model checking techniques that are important method groups with distinct advatages. The reviewed risk methods in [206] are considered without possibility for combinations. Also, their focus is on the offshore wind industry, our research is centered on industrial automation.

Hairing et al. [207] addresses various risk models extensively discussed in our SLR. This paper categorizes risk models into three groups: risk identification primarily aligned with the qualitative methods in our SLR, risk quantification and risk mitigation, which falls outside the scope of our SLR. Additionally, this study evaluates the applicability of these models across different system types, including physical, technical hardware, cyber, operational, organizational, and socio-economic systems. Our findings for CPPS are consistent with those reported in this paper.

Kabir et al. [208] presented a meta-review focusing on BNs, PNs, and DFTs, including their model-to-model transformations. The authors provided an in-depth analysis of the applicability of BN and PN for technical systems, which aligns well with the focus of our SLR. They conclude that both BN and PN are suitable for such applications, though they have certain limitations. Our SLR also examines BN, PN, and other models, and the conclusions of this study align with our finding.

Another notable meta-review is the paper by Villani et al. [209]. This paper offers an in-depth examination of safety standards and intuitive interfaces specifically for collaborative robotics within industrial contexts, with a focus on safety issues, user interfaces, and applications from 2017 to 2018. While Villani et al. provide a detailed overview of safety standards and human-robot collaboration in industrial settings, it does not include dynamic methods, focusing instead on traditional safety and interface issues.

Huck et al. [210] conducted a literature review on risk assessment methods for industrial human-robot collaboration (HRC) and evaluated interviews with HRC professionals to understand industry needs. They found that awareness of available methods is limited within the industry, with familiarity mainly restricted to HAZOP, while methods like STPA, FTA, formal verification (SAFER-HRC), expert systems, and simulation-based approaches are more commonly discussed in academic literature. This review examined 183 papers, selecting 53 for detailed analysis. In contrast, our review provides a broader perspective on risk assessment for industrial cyber-physical production systems, rather than focusing solely on human-robot systems. Another review [211] also addresses HRC safety but from an occupational health perspective, focusing on open challenges rather than specific risk methods, although it highlights hazard analysis and risk assessment as central concerns. Similarly, a survey on safety boundaries in human-robot interaction [212] aims to categorize recent literature by describing safety requirements during human-robot collaboration, emphasizing collaborative robot functions in specific processes. This survey explores methods for psychological safety in HRC and its influence on robot behavior, addressing the psychological factors of robot integration in industrial and social contexts. A search of safety analysis methods in [212] shows that most studies employ cognitive and task-analytic models along with traditional hazard analysis methods such as the Human Factors Analysis and Classification System (HFACS), FMEA, FTA, STPA, HAZOP, and SAFER-HRC. The findings are consistent with those of Huck et al. [210], though the number of papers evaluated is smaller due to the different focus of the study.

In conclusion, the need for our SLR is justified by its focus on CPPS within the industrial automation context, extending the review period to the end of 2023 and incorporating the latest advancements in AI for risk assessment. Our review surpasses previous meta-reviews by encompassing a broader range of methods and highlighting the integration and combination of risk methods, particularly emphasizing the application of ML/DL techniques and their implications for CPPS.

7 Conclusion

In this paper, we have conducted a Systematic Literature Review to evaluate the applicability and limitations of existing risk assessment methods for Cyber-Physical Production Systems. Through this review, we identified several aspects of CPPS that challenge traditional risk assessment methodologies, including high autonomy, intelligence, heterogeneity, spatial and logical distribution, high structural and behavioral complexity, reconfigurability, and human-in-the-loop integration. Our findings indicate that while there are numerous risk assessment methods available, none can fully address all the challenges posed by modern CPPS on their own. We highlighted key gaps in current methodologies, including the need for hybrid risk models, effective integration of various risk assessment methods, and the necessity of continuous risk assessment in reconfigurable systems. Additionally, there is a critical need for high-quality failure data to feed into these models and for innovative approaches to model complex failure scenarios, particularly those involving AI components. In conclusion, addressing the risk assessment challenges of CPPS requires a multifaceted approach, combining traditional and advanced methods, continuous data collection, and the integration of AI technologies.

Acknowledgment

The authors would like to acknowledge the following colleagues (alphabetical order) for participation as experts in the study selection process: Sheng Ding, Philipp Grimmeisen, Joachim Grimstad, Yuliang Ma, Thomas Mössner, Arne Sonnenburg, Martin Westhoven, Andreas Richter. This research is supported by the Bundesanstalt für Arbeitsschutz und Arbeitsmedizin (BAuA, Germany) under project number F2497.

References

[1] Kaplan, S., and Garrick, B. J., 1981. “On the quantitative definition of risk”. Risk analysis, 1(1), pp. 11–27.

[2] Aven, T., and Renn, O., 2009. “On risk defined as an event where the outcome is uncertain”. Journal of risk research, 12(1), pp. 1–11.

[3] Aven, T., 2012. “The risk concept—historical and recent development trends”. Reliability Engineering & System Safety, 99, pp. 33–44.

[4] Aven, T., 2014. “What is safety science?”. Safety Science, 67, pp. 15–20. The Foundations of Safety Science.

[5] Avizienis, A., Laprie, J.-C., Randell, B., and Landwehr, C., 2004. “Basic concepts and taxonomy of dependable and secure computing”. IEEE Transactions on Dependable and Secure Computing, 1(1), pp. 11–33.

[6] ISO ISO 12100:2010, 2010. Safety of machinery — General principles for design — Risk assessment and risk reduction. Standard, International Organization for Standardization, Geneva, CH, Nov.

[7] für Normung, I. O., 2018. ISO 31000: 2018, Risk Management - Guidelines. International Organization for Standardization. International Organization for Standardization.

[8] Xu, H., Yu, W., Griffith, D., and Golmie, N., 2018. “A survey on industrial internet of things: A cyber-physical systems perspective”. IEEE Access, 6, pp. 78238–78259.

[9] Rüßmann, M., Lorenz, M., Gerbert, P., Waldner, M., Justus, J., Engel, P., and Harnisch, M., 2015. “Industry 4.0: The future of productivity and growth in manufacturing industries”. Boston consulting group, 9(1), pp. 54–89.

[10] Nunes, D. S., Zhang, P., and Silva, J. S., 2015. “A survey on human-in-the-loop applications towards an internet of all”. IEEE Communications Surveys & Tutorials, 17(2), pp. 944–965.

[11] Barricelli, B. R., Casiraghi, E., and Fogli, D., 2019. “A survey on digital twin: Definitions, characteristics, applications, and design implications”. IEEE Access, 7, pp. 167653–167671.

[12] Thames, L., and Schaefer, D., 2016. “Software-defined cloud manufacturing for industry 4.0”. Procedia cirp, 52, pp. 12–17.

[13] Peres, R. S., Jia, X., Lee, J., Sun, K., Colombo, A. W., and Barata, J., 2020. “Industrial artificial intelligence in industry 4.0 - systematic review, challenges and outlook”. IEEE Access, 8, pp. 220121–220139.

[14] Council of European Union, 2024. Regulation of the european parliament and of the council laying down harmonised rules on artificial intelligence and amending regulations (ec) no 300/2008, (eu) no 167/2013, (eu) no 168/2013, (eu) 2018/858, (eu) 2018/1139 and (eu) 2019/2144 and directives 2014/90/eu, (eu) 2016/797 and (eu) 2020/1828. https://www.europarl.europa.eu/doceo/document/TA-9-2024-0138-FNL-COR01_EN.pdf.

[15] Bolbot, V., Theotokatos, G., Bujorianu, L. M., Boulougouris, E., and Vassalos, D., 2019. “Vulner-

abilities and safety assurance methods in cyber-physical systems: A comprehensive review". Reliability Engineering & System Safety, 182, pp. 179–193.

[16] Fabarisov, T., Siedel, G., Vock, S., and Morozov, A., 2021. "Aspects of industrial cps critical for risk assessment methods". Системная инженерия и информационные технологии, 3(3 (7)), pp. 23–29.

[17] Grossmann, I. E., and Harjunkoski, I., 2019. "Process systems engineering: academic and industrial perspectives". Computers & Chemical Engineering, 126, pp. 474–484.

[18] Johansen, I. L., and Rausand, M., 2014. "Defining complexity for risk assessment of sociotechnical systems: A conceptual framework". Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 228(3), pp. 272–290.

[19] Siedel, G., Voß, S., and Vock, S., 2021. "An overview of the research landscape in the field of safe machine learning". In Proceedings of IMECE 2021, The American Society of Mechanical Engineers, ed., p. Art.69390.

[20] Poretschkin, M., Hauben, S., Schmitz, A., and et al. Leitfaden zur gestaltung vertrauenswürdiger künstlicher intelligenz: Ki-prüfkatalog.

[21] Hong, Y., Lian, J., Xu, L., Min, J., Wang, Y., Freeman, L. J., and Deng, X., 2023. "Statistical perspectives on reliability of artificial intelligence systems". Quality Engineering, 35(1), pp. 56–78.

[22] Koopman, P., Kane, A., and Black, J., 2019. "Credible autonomy safety argumentation". In 27th Safety-Critical Systems Symposium, pp. 34–50.

[23] Zhao, X., Banks, A., Sharp, J., Robu, V., Flynn, D., Fisher, M., and Huang, X., 2020. "A safety framework for critical systems utilising deep neural networks". In Computer Safety, Reliability, and Security: 39th International Conference, SAFECOMP 2020, Lisbon, Portugal, September 16–18, 2020, Proceedings 39, Springer, pp. 244–259.

[24] Li, B., Qi, P., Liu, B., Di, S., Liu, J., Pei, J., Yi, J., and Zhou, B., 2023. "Trustworthy ai: From principles to practices". ACM Computing Surveys, 55(9), pp. 1–46.

[25] Hendrycks, D., and Dietterich, T., 2018. "Benchmarking neural network robustness to common corruptions and perturbations". In International Conference on Learning Representations.

[26] Siedel, G., Vock, S., Morozov, A., and Voß, S., 2022. "Utilizing class separation distance for the evaluation of corruption robustness of machine learning classifiers". arXiv preprint arXiv:2206.13405.

[27] Carlini, N., and Wagner, D., 2017. "Towards evaluating the robustness of neural networks". In 2017 ieee symposium on security and privacy (sp), Ieee, pp. 39–57.

[28] Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., Mittal, P., and Hein, M., 2021. "Robustbench: a standardized adversarial robustness benchmark". In Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2).

[29] Hanif, M. A., Khalid, F., Putra, R. V. W., Rehman, S., and Shafique, M., 2018. "Robust machine learning systems: Reliability and security for deep neural networks". In 2018 IEEE 24th international symposium on on-line testing and robust system design (IOLTS), IEEE, pp. 257–260.

[30] Reader, T. W., Katz-Navon, T., and Grote, G., 2023. "Safety science in the new age of work". Safety Science, 158, p. 105970.

[31] Leoni, L., BahooToroody, A., Abaei, M. M., Cantini, A., BahooToroody, F., and De Carlo, F., 2024. "Machine learning and deep learning for safety applications: Investigating the intellectual structure and the temporal evolution". Safety Science, 170, p. 106363.

[32] Hegde, J., and Rokseth, B., 2020. "Applications of machine learning methods for engineering risk assessment – a review". Safety Science, 122, p. 104492.

[33] Fuqua, N. B., 2003. "The applicability of markov analysis methods to reliability, maintainability, and safety". Selected Topic in Assurance Related Technologies (START), 2(10), pp. 1–8.

[34] Markov, A. A., 1906. "Rasprostranenie zakona bol'shih chisel na velichiny, zavisyaschie drug ot druga". Izvestiya Fiziko-matematicheskogo obschestva pri Kazanskom universitete, 15(135-156), p. 18.

[35] Petri, C. A., 1962. "Kommunikation mit automaten".

[36] Dugan, J. B., Bavuso, S. J., and Boyd, M. A., 1990. "Fault trees and sequence dependencies". In Reliability and Maintainability Symposium, 1990. Proceedings., Annual, IEEE, pp. 286–293.

[37] Baier, C., and Katoen, J.-P., 2008. Principles of model checking. MIT press.

[38] Rausand, M., and Høyland, A., 2004. System Reliability Theory: Models, Statistical Methods and Applications. Wiley-Interscience, Hoboken, NJ.

[39] Leemis, L., 1995. Reliability: Probabilistic Models and Statistical Methods. Prentice-Hall international series in industrial and systems engineering. Prentice Hall.

[40] Stamatis, D. H., 2003. Failure mode and effect analysis: FMEA from theory to execution. ASQ Quality Press.

[41] Althoff, M., Stursberg, O., and Buss, M., 2007. "Safety assessment of autonomous cars using verification techniques". In 2007 American Control Conference, IEEE, pp. 4154–4159.

[42] Hollnagel, E., 2012. FRAM, the functional resonance analysis method: modelling complex socio-technical systems. Ashgate Publishing, Ltd.

[43] Levenson, N., and Thomas, J., 2018. Stpa handbook. https://psas.scripts.mit.edu/home/get_file.php?name=STPA_handbook.pdf, March. Online; accessed 23 April 2023.

[44] Clarke, E., Grumberg, O., Kroening, D., Peled, D., and Veith, H., 2018. Model Checking, second edition. Cyber Physical Systems Series. MIT Press.

[45] Inc., P., 2017. Comparison of process hazard analysis (pha) methods. https://www.primatech.com/images/docs/comparison-of-pha-methods.pdf. Online; accessed 23 April 2023.

[46] Clemens, P., 2002. "Event tree analysis". JE Jacobs Sverdrup,.

[47] Vesely, W. E., Goldberg, F. F., Roberts, N. H., and Haasl, D. F., 1981. Fault tree handbook. Tech. rep., Nuclear Regulatory Commission Washington dc.

[48] Čepin, M., 2011. Reliability Block Diagram. Springer London, London, pp. 119–123.

[49] Bause, F., and Kritzinger, P. S., 2002. Stochastic petri nets, Vol. 1. Citeseer.

[50] Andrews, J., and Ridley, L., 2001. "Reliability of sequential systems using the cause—consequence diagram method". Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering, 215(3), pp. 207–220.

[51] Groen, F., and Mosleh, A., 2006. "An algorithm for the quantification of hybrid causal models". In Proceedings of the eighth international conference on probabil-istic safety assessment and management (PSAM8).

[52] Norris, J. R., 1998. Markov chains. No. 2. Cambridge university press.

[53] Howard, R. A., 1960. "Dynamic programming and markov processes.".

[54] Hermanns, H., and Hermanns, H., 2002. "Interactive markov chains". Interactive Markov Chains: And the Quest for Quantified Quality, pp. 57–88.

[55] Boudali, H., Crouzen, P., and Stoelinga, M., 2007. "Dynamic fault tree analysis using input/output interactive markov chains". In 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'07), IEEE, pp. 708–717.

[56] Marsan, M. A., 1990. "Stochastic petri nets: An elementary introduction". In Advances in Petri Nets 1989, G. Rozenberg, ed., Springer Berlin Heidelberg, pp. 1–29.

[57] Molloy, 1982. "Performance analysis using stochastic petri nets". IEEE Transactions on computers, 100(9), pp. 913–917.

[58] Marsan, M., Balbo, G., Chiola, G., Conte, G., Donatelli, S., and Franceschinis, G., 1991. "An introduction to generalized stochastic petri nets". Microelectronics Reliability, 31(4), pp. 699 – 725.

[59] Ajmone Marsan, M., Conte, G., and Balbo, G., 1984. "A class of generalized stochastic petri nets for the performance evaluation of multiprocessor systems". ACM Transactions on Computer Systems (TOCS), 2(2), pp. 93–122.

[60] Dugan, J. B., 1984. EXTENDED STOCHASTIC PETRI NETS: APPLICATIONS AND ANALYSIS (MODELING, RELIABILITY, PERFORMANCE). Duke University.

[61] Marsan, M. A., and Chiola, G., 1987. "On petri nets with deterministic and exponentially distributed firing times". In Advances in Petri Nets 1987 7, Springer, pp. 132–145.

[62] Choi, H., Kulkarni, V. G., and Trivedi, K. S., 1994. "Markov regenerative stochastic petri nets". Performance evaluation, 20(1-3), pp. 337–357.

[63] Buchacker, K., et al., 1999. "Combining fault trees and petri nets to model safety-critical systems". In High performance computing, The Society for Computer Simulation International, pp. 439–444.

[64] Kaiser, B., Gramlich, C., and Förster, M., 2007. "State/event fault trees—a safety analysis model for software-controlled systems". Reliability Engineering & System Safety, 92(11), pp. 1521–1537.

[65] Raiteri, D. C., Franceschinis, G., Iacono, M., and Vittorini, V., 2004. "Repairable fault tree for the automatic evaluation of repair policies". In International Conference on Dependable Systems and Networks, 2004, IEEE, pp. 659–668.

[66] Kaiser, B., Liggesmeyer, P., and Mäckel, O., 2003. "A new component concept for fault trees". In Proceedings of the 8th Australian workshop on Safety critical systems and software-Volume 33, pp. 37–46.

[67] Bouissou, M., 2002. "Boolean logic driven markov processes: A powerful new formalism for specifying and solving very large markov models". PSAM6, Puerto Rico.

[68] Distefano, S., and Xing, L., 2006. "A new approach to modeling the system reliability: dynamic reliability block diagrams". In Reliability and Maintainability Symposium, 2006. RAMS'06. Annual, IEEE, pp. 189–195.

[69] Murphy, K. P., 2002. Dynamic bayesian networks: representation, inference and learning. University of California, Berkeley.

[70] Kosko, B., 1986. "Fuzzy cognitive maps". International journal of man-machine studies, 24(1), pp. 65–75.

[71] Acosta, C., and Siu, N., 1993. "Dynamic event trees in accident sequence analysis: application to steam generator tube rupture". Reliability Engineering & System Safety, 41(2), pp. 135–154.

[72] Commission, U. N. R., et al., 1983. "Pra procedures guide (nureg/cr 2300), section 3.4.4.2 system event trees developed from event-sequence diagrams". Washington, DC.

[73] Kwiatkowska, M., Norman, G., and Parker, D., 2002. "Prism: Probabilistic symbolic model checker". In International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Springer, pp. 200–204.

[74] Dehnert, C., Junges, S., Katoen, J.-P., and Volk, M., 2017. "A storm is coming: A modern probabilistic model checker". In Computer Aided Verification: 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part II 30, Springer, pp. 592–600.

[75] Batteux, M., Prosvirnova, T., Rauzy, A., and Kloul, L., 2013. "The altarica 3.0 project for model-based safety assessment". In 2013 11th IEEE International Conference on Industrial Informatics (INDIN), pp. 741–746.

[76] Bozzano, M., Bruintjes, H., Cimatti, A., Katoen, J.-P., Noll, T., and Tonetta, S., 2017. "Formal methods for aerospace systems". In Cyber-Physical System Design from an Architecture Analysis Viewpoint. Springer, pp. 133–159.

[77] Morozov, A., Ding, K., Steurer, M., and Janschek, K., 2019. "Openerrorpro: A new tool for stochastic model-based reliability and resilience analysis". In 2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE), pp. 303–312.

[78] Swuste, P., Groeneweg, J., van Gulijk, C., Zwaard, W., Lemkowitz, S., and Oostendorp, Y., 2020. "The future of safety science". Safety Science, 125, p. 104593.

[79] Rausand, M., 2005. "Preliminary hazard analysis". Norwegian University of Science and Technology.

[80] Baybutt, P., 2003. "Major hazards analysis: An improved method for process hazard analysis". Process Safety Progress, 22(1), pp. 21–26.

[81] McKelvey, T., Rothschild, M., Gideon, J., Beasley, A., and Gressel, M., 1992. "Process hazards review applied to the use of anhydrous ammonia in agriculture: an example of chemical process safety for small business". Journal of loss prevention in the process industries, 5(5), pp. 297–303.

[82] Vyzaite, G., Dunnett, S., and Andrews, J., 2006. "Cause–consequence analysis of non-repairable phased missions". Reliability Engineering & System Safety, 91(4), pp. 398–406.

[83] de Ruijter, A., and Guldenmund, F., 2016. "The bowtie method: A review". Safety Science, 88, pp. 211–218.

[84] Mokhtari, K., Ren, J., Roberts, C., and Wang, J., 2011. "Application of a generic bow-tie based risk analysis framework on risk management of sea ports and offshore terminals". Journal of hazardous materials, 192(2), pp. 465–475.

[85] Gargama, H., and Chaturvedi, S. K., 2011. "Criticality assessment models for failure mode effects and criticality analysis using fuzzy logic". IEEE transactions on Reliability, 60(1), pp. 102–110.

[86] Narayanagounder, S., and Gurusami, K., 2009. "A new approach for prioritization of failure modes in design fmea using anova". World Academy of Science, Engineering and Technology, 49(524-31).

[87] Parsana, T. S., and Patel, M. T., 2014. "A case study: A process fmea tool to enhance quality and efficiency of manufacturing industry". Bonfring International Journal of Industrial Engineering and Management Science, 4(3), pp. 145–152.

[88] Goddard, P. L., 2000. "Software fmea techniques". In Annual Reliability and Maintainability Symposium. 2000 Proceedings. International Symposium on Product Quality and Integrity (Cat. No. 00CH37055), IEEE, pp. 118–123.

[89] Ouimet, M., and Lundqvist, K., 2007. "Formal software verification: Model checking and theorem proving". Embedded Systems Laboratory Technical Report ESL-TIK-00214, Cambridge USA, p. 24.

[90] Kazhamiakin, R., Pistore, M., and Roveri, M., 2004. "Formal verification of requirements using spin: A case study on web services". In Proceedings of the Second International Conference on Software Engineering and Formal Methods, 2004. SEFM 2004., IEEE, pp. 406–415.

[91] Cimatti, A., Clarke, E., Giunchiglia, F., and Roveri, M., 1999. "Nusmv: A new symbolic model verifier". In Computer Aided Verification: 11th International Conference, CAV'99 Trento, Italy, July 6–10, 1999 Proceedings 11, Springer, pp. 495–499.

[92] Behrmann, G., David, A., and Larsen, K. G., 2004. "A tutorial on uppaal". Formal methods for the design of real-time systems, pp. 200–236.

[93] Johansen, I. L., and Rausand, M., 2012. "Risk metrics: Interpretation and choice". In 2012 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 1914–1918.

[94] Villa, V., Paltrinieri, N., Khan, F., and Cozzani, V., 2016. "Towards dynamic risk analysis: A review of the risk assessment approach and its limitations in the chemical process industry". Safety Science, 89, pp. 77–93.

[95] Group, F., et al., 2009. "Reliability methodology for electronic systems". FIDES guide.

[96] Denson, W., Chandler, G., Crowell, W., Clark, A., and Jaworski, P., 1994. "Nonelectronic parts reliability data 1995". DTIC document.

[97] Handbook, M., 1991. "Reliability prediction of electronic equipment (mil-hdbk-217f)". Department of Defense.

[98] Rauzy, A., 1993. "New algorithms for fault trees analysis". Reliability Engineering & System Safety, 40(3), pp. 203–211.

[99] Ruijters, E., and Stoelinga, M., 2015. "Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools". Computer science review, 15, pp. 29–62.

[100] Kwiatkowska, M., Norman, G., and Parker, D., 2011. "Prism 4.0: Verification of probabilistic real-time systems". In International conference on computer aided verification, Springer, pp. 585–591.

[101] Keele, S., et al., 2007. Guidelines for performing systematic literature reviews in software engineering.

[102] Freitas, V., et al. Perform systematic literature reviews.

[103] Sun, X., Huang, N., Wang, B., and Zhou, J., 2014. "Reliability of cyber physical systems assessment of the aircraft fuel management system". In The 4th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent, IEEE, pp. 424–428.

[104] Schmittner, C., Ma, Z., Schoitsch, E., and Gruber, T., 2015. "A case study of fmvea and chassis as safety and security co-analysis method for automotive cyber-physical systems". In Proceedings of the 1st ACM Workshop on Cyber-Physical System Security, pp. 69–80.

[105] Xiao, M.-r., Dong, Y.-w., Gou, Q.-w., Xue, F., and Chen, Y.-h., 2020. "Architecture-level particular risk modeling and analysis for a cyber-physical system with aadl". Frontiers of Information Technology & Electronic Engineering, 21(11), pp. 1607–1625.

[106] Kühn, J., Schoonbrood, P., Stollenwerk, A., Brendle, C., Wardeh, N., Walter, M., Rossaint, R., Leonhardt, S., Kowalewski, S., and Kopp, R., 2015. "Safety conflict analysis in medical cyber-physical systems using an smt-solver.". In Software Engineering (Workshops), pp. 19–23.

[107] Di Maio, F., Mascherona, R., and Zio, E., 2019. "Risk analysis of cyber-physical systems by gtst-mld". IEEE Systems Journal, 14(1), pp. 1333–1340.

[108] Wang, H., 2020. "Research on real-time reliability evaluation of cps system based on machine learning". Computer communications, 157, pp. 336–342.

[109] Pinna, B., Babykina, G., Brinzei, N., and Pétin, J.-F., 2013. "Deterministic and stochastic dependability analysis of industrial systems using coloured petri nets approach". In Annual conference of the European safety and reliability association, ESREL 2013, Taylor & Francis Group, ISBN 978-1-138-00123-7, pp. 2969–2977.

[110] Wang, Q., Gao, J., Chen, K., and Yang, P., 2011. "Reliability assessment of manufacturing system based on hspn models and non-homogeneous isomorphism markov". In 2011 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, IEEE, pp. 182–186.

[111] Guerin, F., Todoskoff, A., Barreau, M., Morel, J.-Y., Mihalache, A., and Dumon, B., 2002. "Reliability analysis for complex industrial real-time systems: application on an antilock brake system". In IEEE International Conference on Systems, Man and Cybernetics, Vol. 7, IEEE, pp. 6–pp.

[112] Martin-Guillerez, D., Guiochet, J., Powell, D., and Zanon, C., 2010. "A uml-based method for risk analysis of human-robot interactions". In Proceedings of the 2nd International Workshop on Software Engineering for Resilient Systems, pp. 32–41.

[113] Kaul, T., Meyer, T., and Sextro, W., 2015. "Integrated model for dynamics and reliability of intelligent mechatronic systems". In European safety and reliability conference (esrel2015). Taylor and Francis.

[114] Fiondella, L., Lin, Y.-K., and Chang, P.-C., 2015. "System performance and reliability modeling of a stochastic-flow production network: A confidence-based approach". IEEE Transactions on Systems, Man, and Cybernetics: Systems, 45(11), pp. 1437–1447.

[115] Vicentini, F., Askarpour, M., Rossi, M. G., and Mandrioli, D., 2019. "Safety assessment of collaborative robotics through automated formal verification". IEEE Transactions on Robotics, 36(1), pp. 42–61.

[116] Bode, E., Herbstritt, M., Hermanns, H., Johr, S., Peikenkamp, T., Pulungan, R., Rakow, J., Wimmer, R., and Becker, B., 2008. "Compositional dependability evaluation for statemate". IEEE Transactions on Software Engineering, 35(2), pp. 274–292.

[117] Güdemann, M., Ortmeier, F., and Reif, W., 2006. "Safety and dependability analysis of self-adaptive systems". In second international symposium on leveraging applications of formal methods, verification and validation (isola 2006), IEEE, pp. 177–184.

[118] Sunilkumar, K., Sreejith, P., and Jayadas, N., 2011. "Service reliability analysis using competing risk models". In 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing, IEEE, pp. 1–5.

[119] Habermaier, A., Eberhardinger, B., Seebach, H., Leupolz, J., and Reif, W., 2015. "Runtime model-based safety analysis of self-organizing systems with s". In 2015 IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, IEEE, pp. 128–133.

[120] Braman, J. M., and Murray, R. M., 2008. "Safety verification of fault tolerant goal-based control programs with estimation uncertainty". In 2008 American Control Conference, IEEE, pp. 27–32.

[121] Alwi, S., and Fujimoto, Y., 2014. "Safety property comparison between gröbner bases and bdd-based model checking method". In 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), IEEE, pp. 511–516.

[122] Yevkin, O., 2009. "Truncation approach with the decomposition method for system reliability analysis". In 2009 Annual Reliability and Maintainability Symposium, IEEE, pp. 430–435.

[123] Rubaiyat, A. H. M., Qin, Y., and Alemzadeh, H., 2018. "Experimental resilience assessment of an open-source driving agent". In 2018 IEEE 23rd Pacific rim international symposium on dependable computing (PRDC), IEEE, pp. 54–63.

[124] Goswami, K. K., 1997. "Depend: A simulation-based environment for system level dependability analysis". IEEE Transactions on Computers, 46(1), pp. 60–74.

[125] Mohrle, F., Zeller, M., Hofig, K., Rothfelder, M., and Liggesmeyer, P., 2015. "Automated compositional safety analysis using component fault trees". In 2015 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), IEEE, pp. 152–159.

[126] Araújo, D. R., de Barros, G. H., Bastos-Filho, C. J., and Martins-Filho, J. F., 2017. "Surrogate models assisted by neural networks to assess the resilience of networks". In 2017 ieee latin american conference on computational intelligence (la-cci), IEEE, pp. 1–6.

[127] Dorociak, R., 2012. "Early probabilistic reliability analysis of mechatronic systems". In 2012 Proceedings Annual Reliability and Maintainability Symposium, IEEE, pp. 1–6.

[128] Balchanos, M., Li, Y., and Mavris, D., 2012. "Towards a method for assessing resilience of complex dynamical systems". In 2012 5th International Symposium on Resilient Control Systems, IEEE, pp. 155–160.

[129] Schneider, D., and Trapp, M., 2009. "Runtime safety models in open systems of systems". In 2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, IEEE, pp. 455–460.

[130] Kabir, S., Yazdi, M., Aizpurua, J. I., and Papadopoulos, Y., 2018. "Uncertainty-aware dynamic reliability analysis framework for complex systems". IEEE Access, 6, pp. 29499–29515.

[131] Zhong, X., Ichchou, M., and Saidi, A., 2010. "Reliability assessment of complex mechatronic systems using a modified nonparametric belief propagation algorithm". Reliability engineering & system safety, 95(11), pp. 1174–1185.

[132] Duan, R., Lin, Y., and Zeng, Y., 2018. "Fault diagnosis for complex systems based on reliability analysis and sensors data considering epistemic uncertainty". Eksploatacja i Niezawodność, 20(4).

[133] Bernardini, A., Ecker, W., and Schlichtmann, U., 2016. "Where formal verification can help in functional safety analysis". In Proceedings of the 35th International Conference on Computer-Aided Design, pp. 1–8.

[134] Choley, J.-Y., Mhenni, F., Nguyen, N., and Baklouti, A., 2016. "Topology-based safety analysis for safety critical cps". Procedia computer science, 95, pp. 32–39.

[135] Bittner, B., Bozzano, M., and Cimatti, A., 2017. "Timed failure propagation analysis for spacecraft engineering: the esa solar orbiter case study". In Model-Based Safety and Assessment: 5th International Symposium, IMBSA 2017, Trento, Italy, September 11–13, 2017, Proceedings 5, Springer, pp. 255–271.

[136] Bozzano, M., Bruintjes, H., Cimatti, A., Katoen, J.-P., Noll, T., and Tonetta, S., 2019. "Compass 3.0". In Tools and Algorithms for the Construction and Analysis of Systems: 25th International Conference, TACAS 2019, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2019, Prague, Czech Republic, April 6–11, 2019, Proceedings, Part I 25, Springer, pp. 379–385.

[137] Prosvirnova, T., Batteux, M., Brameret, P.-A., Cherfi, A., Friedlhuber, T., Roussel, J.-M., and Rauzy, A., 2013. "The altarica 3.0 project for model-based safety assessment". IFAC proceedings volumes, 46(22), pp. 127–132.

[138] Dong, Y., Zhao, X., and Huang, X., 2022. "Dependability analysis of deep reinforcement learning based robotics and autonomous systems through probabilistic model checking". In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, pp. 5171–5178.

[139] Camilli, M., Felderer, M., Giusti, A., Matt, D. T., Perini, A., Russo, B., and Susi, A., 2021. "Towards risk modeling for collaborative ai". In 2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN), IEEE, pp. 51–54.

[140] Hanna, A., Bengtsson, K., Götvall, P.-L., and Ekström, M., 2020. "Towards safe human robot collaboration-risk assessment of intelligent automation". In 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vol. 1, IEEE, pp. 424–431.

[141] Hanna, A., Bengtsson, K., Larsson, S., and Götvall, P.-L. "Risk assessment for intelligent and collaborative automation system by combining fmea and stpa". Available at SSRN 4530824.

[142] Müller, M., Jazdi, N., and Weyrich, M., 2022. "Towards situative risk assessment for industrial mobile robots". In 2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, pp. 1–8.

[143] Fabarisov, T., Morozov, A., Mamaev, I., and Grimmeisen, P., 2022. "Fidget: Deep learning-based fault injection framework for safety analysis and intelligent generation of labeled training data". In 2022 IEEE 27th International Conference on Emerging Technologies and Factory Au-

tomation (ETFA), IEEE, pp. 1–6.

[144] Yang, B., and Hu, H., 2021. "Robustness analysis of automated manufacturing systems with unreliable resources using petri nets". IEEE Transactions on Automation Science and Engineering, 19(4), pp. 3686–3699.

[145] Kwon, R., Kwon, G., Park, S., Chang, J., and Jo, S., 2024. "Applying quantitative model checking to analyze safety in reinforcement learning". IEEE Access.

[146] Morozov, A., Diaconeasa, M. A., and Steurer, M., 2020. "A hybrid methodology for model-based probabilistic resilience evaluation of dynamic systems". In ASME International Mechanical Engineering Congress and Exposition, Vol. 84669, American Society of Mechanical Engineers, p. V014T14A024.

[147] Gleirscher, M., Calinescu, R., and Woodcock, J., 2021. "Riskstructures: A design algebra for risk-aware machines". Formal Aspects of Computing, 33(4), pp. 763–802.

[148] FOURLAS, P. H. T. G. K., 2015. "Reliability and maintainability analysis of a robotic system for industrial applications: a case study". International Journal of Performability Engineering, 11(5), p. 453.

[149] Bensaci, C., Zennir, Y., Pomorski, D., Innal, F., and Lundteigen, M. A., 2023. "Collision hazard modeling and analysis in a multi-mobile robots system transportation task with stpa and spn". Reliability Engineering & System Safety, 234, p. 109138.

[150] Adriaensen, A., Pintelon, L., Costantino, F., Di Gravio, G., and Patriarca, R., 2021. "An stpa safety analysis case study of a collaborative robot application". IFAC-PapersOnLine, 54(1), pp. 534–539.

[151] Buysse, L., Whiteley, S., Vankeirsbilck, J., Vanoost, D., Boydens, J., and Pissoort, D., 2023. "Case study analysis of stpa on an industrial cooperative robot and an autonomous mobile robot". In The Future of Safe Systems-Proceedings of the Safety-Critical Systems Symposium, SCSC on Amazon, pp. 83–104.

[152] Zheng, Z., Tian, J., and Zhao, T., 2016. "Refining operation guidelines with model-checking-aided fram to improve manufacturing processes: a case study for aeroengine blade forging". Cognition, Technology & Work, 18, pp. 777–791.

[153] Adriaensen, A., Pintelon, L., Costantino, F., Di Gravio, G., and Patriarca, R., 2023. "Systems-theoretic interdependence analysis in robot-assisted warehouse management". Safety Science, 168, p. 106294.

[154] Guiochet, J., 2016. "Hazard analysis of human–robot interactions with hazop–uml". Safety science, 84, pp. 225–237.

[155] Dakwat, A. L., and Villani, E., 2018. "System safety assessment based on stpa and model checking". Safety science, 109, pp. 130–143.

[156] Ceylan, B. O., Karatuğ, Ç., Akyuz, E., Arslanoğlu, Y., and Boustras, G., 2023. "A system theory (stamp) based quantitative accident analysis model for complex engineering systems". Safety science, 166, p. 106232.

[157] Ale, B., Van Gulijk, C., Hanea, A., Hanea, D., Hudson, P., Lin, P.-H., and Sillem, S., 2014. "Towards bbn based risk modelling of process plants". Safety Science, 69, pp. 48–56.

[158] Adamyan, A., and He, D., 2002. "Analysis of sequential failures for assessment of reliability and safety of manufacturing systems". Reliability Engineering & System Safety, 76(3), pp. 227–236.

[159] Bhatti, Z. E., Roop, P. S., and Sinha, R., 2016. "Unified functional safety assessment of industrial automation systems". IEEE Transactions on Industrial Informatics, 13(1), pp. 17–26.

[160] Ung, S., Williams, V., Bonsall, S., and Wang, J., 2006. "Test case based risk predictions using artificial neural network". Journal of Safety Research, 37(3), pp. 245–260.

[161] Soltanali, H., Garmabaki, A., Thaduri, A., Parida, A., Kumar, U., and Rohani, A., 2019. "Sustainable production process: An application of reliability, availability, and maintainability methodologies in automotive manufacturing". Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 233(4), pp. 682–697.

[162] Hejase, M., Kurta, A., Aldemira, T., and Ozgunera, U., 2018. "The backtracking process algorithm: a dynamic probabilistic risk assessment method for autonomous vehicle control systems". In Proceedings of PSAM International Conference on Probabilistic Safety Assessment and Management (PSAM14), Los Angeles, California, USA.

[163] Konig, J., Nordstrom, L., and Osterlind, M., 2013. "Reliability analysis of substation automation system functions using prms". IEEE Transactions on smart grid, 4(1), pp. 206–213.

[164] Nyberg, M., 2018. "Safety analysis of autonomous driving using semi-markov processes". In Safety and Reliability–Safe Societies in a Changing World. CRC Press, pp. 781–788.

[165] Inam, R., Raizer, K., Hata, A., Souza, R., Forsman, E., Cao, E., and Wang, S., 2018. "Risk assessment for human-robot collaboration in an automated warehouse scenario". In 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), Vol. 1, IEEE, pp. 743–751.

[166] Koziolek, H., Schlich, B., and Bilich, C., 2010. "A large-scale industrial case study on architecture-based software reliability analysis". In 2010 IEEE 21st international symposium on software reliability engineering, IEEE, pp. 279–288.

[167] Gul, M., Yucesan, M., and Celik, E., 2020. "A manufacturing failure mode and effect analysis based on fuzzy and probabilistic risk analysis". Applied Soft Computing, 96, p. 106689.

[168] Jiang, Z., Zuo, M. J., and Fung, R. Y., 1999. "Stochastic object-oriented petri nets (sopns) for reliability modeling of manufacturing systems". In Engineering Solutions for the Next Millennium. 1999 IEEE Canadian Conference on Electrical and Computer Engineering (Cat. No. 99TH8411), Vol. 3, IEEE, pp. 1471–1476.

[169] Shu, L., Li, J., and Qiu, M., 2008. "Study on applying fault tree analysis based on fuzzy reasoning in risk analysis of construction quality". In 2008 International Conference on Risk Management & Engineering Management, IEEE, pp. 393–397.

[170] Kloos, J., Hussain, T., and Eschbach, R., 2011. "Risk-based testing of safety-critical embedded systems driven by fault tree analysis". In 2011 IEEE Fourth International Conference on Software Testing, Verification and Validation Workshops, IEEE, pp. 26–33.

[171] Vilardell, S., Serra, I., Abella, J., Del Castillo, J., and Cazorla, F. J., 2019. "Software timing analysis for complex hardware with survivability and risk analysis". In 2019 IEEE 37th International Conference on Computer Design (ICCD), IEEE, pp. 227–236.

[172] Altiparmak, F., Dengiz, B., and Smith, A. E., 2003. "Reliability estimation of computer communication networks: Ann models". In Proceedings of the Eighth IEEE Symposium on Computers and Communications. ISCC 2003, IEEE, pp. 1353–1358.

[173] Yousefi, A., Xie, R., Krishna, S., Shortle, J., and Zhang, Y., 2012. "Safety analysis tool for automated airspace concepts (safeatac)". In 2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC), IEEE, pp. 4C2–1.

[174] Aljazzar, H., Fischer, M., Grunske, L., Kuntz, M., Leitner-Fischer, F., and Leue, S., 2009. "Safety analysis of an airbag system using probabilistic fmea and probabilistic counterexamples". In 2009 Sixth International Conference on the Quantitative Evaluation of Systems, IEEE, pp. 299–308.

[175] Åslund, J., Biteus, J., Frisk, E., Krysander, M., and Nielsen, L., 2007. "Safety analysis of autonomous systems by extended fault tree analysis". International Journal of Adaptive Control and Signal Processing, 21(2-3), pp. 287–298.

[176] Vemuri, K. K., and Dugan, J. B., 1999. "Reliability analysis of complex hardware-software systems". In Annual Reliability and Maintainability. Symposium. 1999 Proceedings (Cat. No. 99CH36283), IEEE, pp. 178–182.

[177] Yang, Q., and Chen, Y., 2009. "Sensor system reliability modeling and analysis for fault diagnosis in multistage manufacturing processes". IIE transactions, 41(9), pp. 819–830.

[178] Jamshidi, A., Ait-Kadi, D., Ruiz, A., and Rebaiaia, M. L., 2018. "Dynamic risk assessment of complex systems using fcm". International Journal of Production Research, 56(3), pp. 1070–1088.

[179] Kołowrocki, K., and Soszyńska, J., 2006. "Reliability and availability analysis of complex port transportation systems". Quality and Reliability Engineering International, 22(1), pp. 79–99.

[180] De Silva, N., Ranasinghe, M., and De Silva, C., 2013. "Use of anns in complex risk analysis applications". Built Environment Project and Asset Management.

[181] Ge, D., Lin, M., Yang, Y., Zhang, R., and Chou, Q., 2015. "Reliability analysis of complex dynamic fault trees based on an adapted kd heidtmann algorithm". Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 229(6), pp. 576–586.

[182] Mi, J., Li, Y.-F., Peng, W., and Huang, H.-Z., 2018. "Reliability analysis of complex multi-state system with common cause failure based on evidential networks". Reliability Engineering & System Safety, 174, pp. 71–81.

[183] Weber, P., and Jouffe, L., 2006. "Complex system reliability modelling with dynamic object oriented bayesian networks (doobn)". Reliability Engineering & System Safety, 91(2), pp. 149–162.

[184] Helle, P., 2012. "Automatic sysml-based safety analysis". In Proceedings of the 5th International Workshop on Model Based Architecting and Construction of Embedded Systems, pp. 19–24.

[185] Glaß, M., Lukasiewycz, M., Haubelt, C., and Teich, J., 2010. "Towards scalable system-level reliability analysis". In Design Automation Conference, IEEE, pp. 234–239.

[186] Aliee, H., Glaß, M., Reimann, F., and Teich, J., 2013. "Automatic success tree-based reliability analysis for the consideration of transient and permanent faults". In 2013 Design, Automation & Test in Europe Conference & Exhibition (DATE), IEEE, pp. 1621–1626.

[187] Seidl, C., Schaefer, I., and Aßmann, U., 2013. "Variability-aware safety analysis using delta component fault diagrams". In Proceedings of the 17th International Software Product Line Conference co-located workshops, pp. 2–9.

[188] Fazlollahtabar, H., and Niaki, S. T. A., 2018. "Fault tree analysis for reliability evaluation of an advanced complex manufacturing system". Journal of Advanced Manufacturing Systems, 17(01), pp. 107–118.

[189] Chen, B., Liu, Y., Zhang, C., and Wang, Z., 2020. "Time series data for equipment reliability analysis with deep learning". IEEE Access, 8, pp. 105484–105493.

[190] Febrero, F., Moraga, M. A., and Calero, C., 2017. "Software reliability as user perception: Applica-

tion of the fuzzy analytic hierarchy process to software reliability analysis". In 2017 IEEE International Conference on Software Quality, Reliability and Security (QRS), IEEE, pp. 224–231.

[191] Chen, Z., Li, G., Pattabiraman, K., and DeBardeleben, N., 2019. "Binfi: An efficient fault injector for safety-critical machine learning systems". In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, pp. 1–23.

[192] Zhou, Z., Oguz, O. S., Leibold, M., and Buss, M., 2020. "A general framework to increase safety of learning algorithms for dynamical systems based on region of attraction estimation". IEEE Transactions on Robotics, 36(5), pp. 1472–1490.

[193] Carreras Guzman, N. H., Kozine, I., and Lundteigen, M. A., 2021. "An integrated safety and security analysis for cyber-physical harm scenarios". Safety Science, 144, p. 105458.

[194] Brennan, R. L., and Prediger, D. J., 1981. "Coefficient kappa: Some uses, misuses, and alternatives". Educational and psychological measurement, 41(3), pp. 687–699.

[195] Banerjee, M., Capozzoli, M., McSweeney, L., and Sinha, D., 1999. "Beyond kappa: A review of interrater agreement measures". Canadian journal of statistics, 27(1), pp. 3–23.

[196] Randolph, J. J., 2005. "Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa.". Online submission.

[197] Grimmeisen, P., Golwalkar, R., Ma, Y., and Morozov, A., 2023. "Automated and continuous risk assessment for ros-based software-defined robotic systems". In 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE), IEEE, pp. 1–7.

[198] Huang, X., Kwiatkowska, M., Wang, S., and Wu, M., 2017. "Safety verification of deep neural networks". In Computer Aided Verification, R. Majumdar and V. Kunčak, eds., Springer International Publishing, pp. 3–29.

[199] Batteux, M., Prosvirnova, T., and Rauzy, A., 2020. "The new open-psa format: a model-based approach". In Congrès Lambda Mu 22 «Les risques au cœur des transitions»(e-congrès)-22e Congrès de Maîtrise des Risques et de Sûreté de Fonctionnement, Institut pour la Maîtrise des Risques.

[200] Steven, E., and Antoine, R., 2007. "Open-psa model exchange format". The Open-PSA Initiative.

[201] Earthperson, A., Aras, E. M., Farag, A. S., and Diaconeasa, M. A., 2023. "Introducing openpra: A web-based framework for collaborative probabilistic risk assessment". In ASME International Mechanical Engineering Congress and Exposition, Vol. 87707, American Society of Mechanical Engineers, p. V013T15A014.

[202] Grimmeisen, P., Karimov, A., Diaconeasa, M. A., and Morozov, A., 2021. "Demonstration of a limited scope probabilistic risk assessment for autonomous warehouse robots with openpra". In ASME International Mechanical Engineering Congress and Exposition, Vol. 85697, American Society of Mechanical Engineers, p. V013T14A030.

[203] Laskar, S., Rahman, M. H., and Li, G., 2022. "Tensorfi+: a scalable fault injection framework for modern deep learning neural networks". In 2022 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), IEEE, pp. 246–251.

[204] Beyer, M., Morozov, A., Valiev, E., Schorn, C., Gauerhof, L., Ding, K., and Janschek, K., 2020. "Fault injectors for tensorflow: evaluation of the impact of random hardware faults on deep cnns". arXiv preprint arXiv:2012.07037.

[205] Reagen, B., Gupta, U., Pentecost, L., Whatmough, P., Lee, S. K., Mulholland, N., Brooks, D., and Wei, G.-Y., 2018. "Ares: A framework for quantifying the resilience of deep neural networks". In Proceedings of the 55th Annual Design Automation Conference, pp. 1–6.

[206] Leimeister, M., and Kolios, A., 2018. "A review of reliability-based methods for risk analysis and their application in the offshore wind industry". Renewable and Sustainable Energy Reviews, 91, pp. 1065–1076.

[207] Häring, I., and Häring, I., 2021. "Technical safety and reliability methods for resilience engineering". Technical Safety, Reliability and Resilience: Methods and Processes, pp. 9–26.

[208] Kabir, S., and Papadopoulos, Y., 2019. "Applications of bayesian networks and petri nets in safety, reliability, and risk assessments: A review". Safety science, 115, pp. 154–175.

[209] Villani, V., Pini, F., Leali, F., and Secchi, C., 2018. "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications". Mechatronics, 55, pp. 248–266.

[210] Huck, T. P., Münch, N., Hornung, L., Ledermann, C., and Wurll, C., 2021. "Risk assessment tools for industrial human-robot collaboration: Novel approaches and practical needs". Safety Science, 141, p. 105288.

[211] Giallanza, A., La Scalia, G., Micale, R., and La Fata, C. M., 2024. "Occupational health and safety issues in human-robot collaboration: State of the art and open challenges". Safety Science, 169, p. 106313.

[212] Zacharaki, A., Kostavelis, I., Gasteratos, A., and Dokas, I., 2020. "Safety bounds in human robot interaction: A survey". Safety Science, 127, p. 104667.