Closing the Gap Between IEC 61511 and the Use of Artificial Intelligence in Plant Safety

Ehsan Ghahremani – Critical Systems Labs Inc.
Jonathan Groves – Critical Systems Labs Inc.
Jeff Joyce – Critical Systems Labs Inc.
Laure Millet – Critical Systems Labs Inc.

WWW.CRITICALSYSTEMSLABS.COM

Closing the Gap Between IEC 61511 and the Use of Artificial Intelligence in Plant Safety

Ehsan Ghahremani
Critical Systems Labs Inc.
Vancouver, Canada
ehsan.ghahremani@cslabs.com

Jonathan Groves
Critical Systems Labs Inc.
Vancouver, Canada
jonathan.groves@cslabs.com

Jeffrey Joyce
Critical Systems Labs Inc.
Vancouver, Canada
jeff.joyce@cslabs.com

Laure Millet
Critical Systems Labs Inc.
Vancouver, Canad
laure.millet@cslabs.com

***Abstract*— There is an urgent need to reconcile the established best practices of functional safety and requirements of applicable safety standards, such as IEC 61511, with the increasing use of Machine Learning (ML) and other forms of Artificial Intelligence (AI). Possible uses of AI/ML in plant automation are motivated by a variety of objectives such as improving safety monitoring, optimizing production and reducing human supervision. Most applications of AI in this context unavoidably involve some measure of uncertainty. Traditional methods and measures for managing safety risk, such as those embodied in IEC 61511, are not necessarily helpful for addressing this inherent uncertainty. Other industries such as advanced automotive have taken steps to close the gap between functional safety and AI/ML safety – for example, the recent publication of ISO/PAS 8800. A key part of this strategy is recognizing (in the words of ISO/PAS 8800) that with the use of AI/ML "it is not possible to provide detailed requirements on the process or product characteristics required to achieve an acceptably low level of residual risk associated with the use of AI systems". Instead, closing the gap between functional safety and AI/ML safety requires assurance argumentation, ideally, in the form of a structured argument that captures the critical thinking that links safety claims to supporting evidence. This paper describes strategies for adapting conventional hazard and risk assessment methods to take account of AI/ML in SIL determination. This paper also explains how safety assurance argumentation can be combined with conventional functional safety to close the gap between IEC 61511 and the use of AI/ML in an industrial setting.**

Keywords—Safety Instrumented Systems (SIS), Artificial Intelligence (AI), Machine Learning (ML), IEC 61511, Functional Safety, Safety Assurance Case, Structured Argumentation.

I. Introduction

Artificial Intelligence and Machine Learning (AI/ML) are increasingly being considered for tasks that touch safety, such as pattern detection, anomaly recognition, diagnostics, and decision support. Yet organizations responsible for Safety Instrumented Systems (SIS) rightly conform their practices to IEC 61511, a standard built around deterministic behavior, static configurations, and evidence gathered through conventional verification and validation (V&V) efforts. The tension is obvious: AI/ML promises capability where deterministic methods might struggle, but it also introduces uncertainty and new failure modes that traditional process-sector guidance does not address specifically. In this paper, "AI/ML" refers to models whose behavior can be probabilistic and whose outputs are shaped by training data rather than fixed logic.

This paper is written from the perspective of practitioners who live at the intersection of AI/ML-enabled functionality and the traditional principles that constrain the design and implementation of an IEC 61511-compliant SIS. Our purpose in this paper is not to promote AI everywhere, nor to downplay regulatory obligations. Instead, we aim to help safety leaders decide when AI/ML is warranted, where it can be utilized without undermining confidence in the safety of a system, and how to assemble objective evidence that will stand up to independent assessment.

In this paper, we take a deliberately pragmatic stance. First, we outline decision criteria for selecting AI/ML only when conventional approaches are insufficient and when the organization can realistically develop, verify, validate, and maintain the model over time (Section II). Second, we characterize the specific mismatches between IEC 61511's expectations and the realities of AI/ML; mismatches that, if ignored, derail compliance efforts and erode confidence (Section III). Third, we propose mitigation strategies that preserve the spirit and objectives of IEC 61511 while bringing in additional lifecycle activities and evidence patterns that have emerged from adjacent safety work on AI systems (Section IV).

Two integration approaches are considered. The bottom-up approach confines AI/ML to well-bounded components placed alongside deterministic SIS components and develops component-level assurance cases as part of the overall system safety case. The top-down approach treats the system holistically and uses structured assurance arguments to demonstrate that residual risk is acceptably low. Neither approach simply "copies-pastes" practices from other domains. Both emphasize clear requirements, defined operating conditions, architectural separation and monitoring, disciplined data management, and a repeatable method for arguing safety claims with supporting evidence.

A recurring theme is transparency about what the AI/ML model can and cannot do, how it was trained and tested, how it is monitored in operation, and how changes are controlled. When system behavior is probabilistic rather than strictly deterministic, assurance shifts from demonstrating via conventional V&V that all software requirements are satisfied to bounding risk to an acceptable level and demonstrating that controls are effective. That does not relax expectations – it changes the evidence you must produce and the way you organize that evidence.

The intended takeaway is practical: readers should come away with a structured way to decide if AI/ML belongs in their SIS context, a clear view of the standard-driven gaps they must address to remain compliant with IEC 61511, and a set of mitigation patterns that turn those gaps into an actionable engineering plan.

II. Considerations for the use of AI/ML

When choosing to use an AI/ML solution for a particular functionality of your system, this must be the result of a deliberate and justified decision made after establishing that the use of AI/ML is acceptable. Such a decision must be clearly documented and defensible to independent assessors as an approach that is warranted and that can meet safety expectations.

To make such a decision in a structured and defensible manner, several factors should be assessed. The functionality must be precisely scoped, with a clear rationale for selecting AI/ML over conventional deterministic methods, either because deterministic approaches cannot feasibly achieve the required objectives (e.g., performance, scalability, or adaptability to highly variable environments), or because the nature of the problem makes it prohibitively complex to define the correct output in advance. Such complexity could arise in high-dimensional contexts where the number of interacting variables is too great for explicit specification, yet an AI/ML model can learn to produce acceptable outputs from representative data (e.g., vision-based monitoring). AI/ML should be selected only when its safety can be assured to an acceptable level and when its benefits outweigh the added risks and assurance effort it introduces.

Feasibility must be established by confirming that sufficient, high-quality data is available and that the effort to develop, verify, and validate the model is practical for your organization. This will require well-thought-out performance and validation criteria which are, in general, difficult to identify for AI/ML systems that often fail in ways that are less predictable and less intuitive than traditional systems, which makes targeted testing and assurance activities more complex.

During design and development, it is critical to identify the system's safe states and ensure the AI/ML component is architecturally isolated or monitored to prevent faults from compromising critical safety functions. Defining appropriate human oversight is also essential, considering the difficulty of maintaining vigilance when operators interact with well-functioning automated systems. Finally, upon deployment, a comprehensive plan must be in place for ongoing monitoring, maintenance, and model updates.

To identify what needs to be done and determine the appropriate level of rigor when introducing AI/ML to SIS, the AI Safety Integrity Level (AI-SIL) framework [1] can be used; it offers a quantitative approach to assess whether an AI/ML solution is acceptable. This framework extends traditional safety integrity levels by integrating specific metrics for AI-enabled systems into the SIL level computation. This approach tailored for AI/ML components supports the elicitation of the assurance activities necessary to generate objective evidence demonstrating to independent assessor that safety expectations have been met.

III. Identifying Gaps in IEC 61511

Despite the growing interest in integrating AI/ML components into SIS, the path to doing so safely is obstructed by regulatory frameworks that were not designed with these technologies in mind. IEC 61511, the prevailing process sector functional safety standard, emphasizes deterministic logic, static configurations, and fully specifiable behaviors [2]. This concept underpins most of the standard's validation and verification expectations, yet it becomes problematic when applied to AI/ML models whose behavior can be inherently probabilistic and whose outputs are shaped by training data rather than fixed logic. While traditional systems rely on requirements-based testing, AI/ML components introduce concerns such as adversarial failure modes, input/output and functional insufficiencies, uncertainty across the input space, complex training datasets, and overall model robustness [3], none of which are addressed in IEC 61511.

Another critical omission is the lack of data-centric development lifecycle in IEC 61511. In AI/ML-Based systems, data is not just an input; it is part of the implementation. The properties, limitations, and biases of the training dataset directly affect safety outcomes. However, IEC 61511 provides no guidance for dataset specification, curation, coverage analysis, or traceability from training data to safety requirements, all of which are necessary for justifying the use of AI/ML within safety functions. IEC 61511 focuses on verification of software code and traceability to safety requirement specifications (SRS) and the only mention of datasets discusses data integrity and data order, which is of a different context, focused on input/output data rather than training data used during AI/ML development [2].

IEC 61511 also assumes that safety configurations are fixed once deployed, with any modifications controlled solely through Management-of-Change (MoC) procedures [2]. ISO/PAS 8800, by contrast, treats an AI-enabled systems as requiring continuous monitoring and calibration, i.e., it mandates run-time performance monitoring, drift detection, and clearly defined triggers for model retraining or rollback, all supported by evidence that each update preserves the existing safety argument [3].

Furthermore, IEC 61511's notion of failure, centered on common cause failures, hardware faults, and systematic design errors, is not sufficient to capture the broader range of failure modes associated with AI/ML systems [2]. These include systematic design errors such as performance insufficiencies, specification gaps, out-of-distribution (OOD) behavior, and silent degradation over time due to changes in the operating environment [3]. If these risks are not explicitly acknowledged and mitigated, the resulting safety argument will be fundamentally incomplete.

The integration of AI/ML also challenges the technology bias of IEC 61511, particularly when considering the use of trained models within or adjacent to the logic solver. The standard explicitly prohibits the use of artificial intelligence within SIL 2+ logic [2]. This raises practical questions about how to isolate, monitor, or otherwise control AI/ML components so that they do not compromise certified elements of the SIS.

Perhaps most importantly, IEC 61511 provides no structured approach for arguing safety in the presence of AI/ML. It prescribes documentation and validation requirements but lacks the scaffolding for structured safety argumentation that is increasingly necessary when dealing with AI-enabled systems [3]. Without such an approach, safety argumentations involving AI/ML risk become ad hoc, inconsistent, or unconvincing to regulators and independent assessors.

In short, IEC 61511 gives us a solid foundation for deterministic SIS design, but it leaves material blind spots namely data-centric development, post-deployment evolution, AI-specific failure modes, probabilistic behavior, and evidence-based structured argumentation, that become showstoppers once AI/ML enters the loop. ISO/PAS 8800 demonstrates that each gap can be closed with additional lifecycle activities, datasets, monitoring strategies and structured assurance cases, yet those practices must be adapted, not copied, into the process-sector context. The next section therefore turns from diagnosis to prescription: we outline concrete mitigation strategies, mapped to each gap, that preserve IEC 61511 compliance while enabling organizations to reap the benefits of AI/ML under a disciplined, defensible safety regime.

IV. Bridging the Gaps

Currently, using AI/ML in systems that would typically conform with IEC 61511 might render the entire system non-compliant with IEC 61511. In general, there are two strategies for arguing that a system using AI/ML could be compliant with IEC 61511.

The first is a “bottom-up” approach, where AI/ML software is treated as a component of a larger system. In this approach, IEC 61511 compliance is argued as normal for the non-AI/ML portion of the system, leaving a remaining gap for AI/ML components. To fill this gap, one must detail equivalent means to demonstrate that the objectives of IEC 61511 are satisfied for AI/ML components using alternate means. This strategy works best for non-AI/ML systems to be retrofitted with an AI/ML component, where only a small portion of the system’s functionality is allocated to AI/ML. This bottom-up approach focuses the safety argument on the safety of AI/ML components being retrofitted and supports the top claim that the integration of the AI/ML component within the greater SIS does not degrade the overall safety of the system.

The second strategy is a “top-down” approach. This approach treats the entire system as an AI/ML system. As IEC 61511 is not prescriptive enough with respect to AI/ML components, other similar standards must be used: e.g., adapting ISO/PAS 8800 for use with the system. This top-down approach argues that the entire system is safe for use using IEC 61511 principles augmented by ISO/PAS 8800 requirements. The key difference between the two strategies is that the “top-down approach” rethinks safety from a new viewpoint, rather than attempting to retrofit existing methods in a more piecemeal way.

Effective argumentation for the safety of either approach can be accomplished by creating a structured argumentation (e.g., Goal-Structuring Notation (GSN) or Eliminative Argumentation (EA)) where the safety engineering activities along with their justifications and resulting evidence can be captured in a structured, logical manner and traced to lifecycle requirements.

Regardless of the approach chosen, the following six gaps with IEC 61511 must be considered when AI/ML is used within a system described above, namely with respect to demonstrating safety of a system where:

1. The outputs of the system may be based on probability (e.g., ML models) rather than being deterministic (i.e., conventional software),
2. The quality and selection of training data plays a substantial role in the output of the system,
3. New or modified traceability methods might need to be used,
4. Changes may occur over time to AI/ML components (e.g., dynamic ML models) and/or their operating environment,
5. Failure modes of an AI/ML system differ from those of a non-AI/ML system, and
6. An AI/ML model might influence non-AI/ML components of a system.

The following six sections each discuss specific potential mitigations for the above six gaps respectively. It should be noted that this does not constitute a complete list of risks nor mitigations, and exact risks and mitigations will vary from system to system.

A. *Probabilistic Outputs from AI/ML Systems*

Many AI/ML systems can produce results based on a probabilistic approach. Due to this, the AI/ML system might produce unexpected outputs given certain input parameters, especially when compared to non-AI/ML systems, which in general are following more logical and code-centric approaches.

To handle the probabilistic nature of certain AI/ML systems, specific “AI/ML safety requirements” can provide the structure framed by normal requirements for non-AI/ML systems, but instead for AI/ML systems that might not always produce acceptably safe results. These AI/ML-based safety requirements can be probabilistic-based requirements [3], where their acceptability is expressed in the form of probabilities (e.g., the AI system produces safe results with an uncertainty rate of 1e-6), and their outputs are demonstrated to meet a certain level of statistical confidence (e.g., statistical significance; p-values).

Given the difficulty of defining fully enumerated requirements for AI/ML, particularly in high-dimensional problem spaces, metamorphic relationships can be specified to formalize expected transformations in input/output behavior. Metamorphic testing can then be used to systematically validate that these constraints hold under varied conditions [4].

Using AI/ML safety requirements might still allow for unsafe outputs, albeit at a probabilistically low rate. Logical reasoning or argumentation can fill this gap, and be constructed to account for such edge cases, etc. [3]. Logical argumentation can be constructed to provide evidence to support a claim that even if the AI/ML system produces a directly unsafe result, the unsafe result will not propagate to unsafe situation. For example, the use of a safety supervisor system can detect certain unsafe situations, e.g., out-of-bound values, outputs that change too quickly, or outputs that are inconsistent with other systems. Additionally, other methods such as ensemble methods (e.g., using conventional programming methods for one of the ensembled systems) could be argued to provide acceptable mitigation of unsafe situations.

Overall, evaluation methods exist that can be used to evaluate the impact of AI errors, leading to other potential mitigation methods.

B. Quality and Selection of Training Data

For many AI/ML systems, the quality of the system's outputs is dependent on the quality of the training data used. In this context, "quality" is used in place of various positive attributes of an AI/ML system, such as precision, reliability, robustness.

Common issues with datasets can often be mitigated by following a systematic process to define and manage datasets used throughout the system. This might include steps such as creating dataset requirements, designing the contents of the dataset, followed by implementing, verifying, validating, testing, and then managing the dataset [3].

While accomplishing these milestones, certain aspects of the dataset should be considered, e.g., accuracy, completeness, representativeness, independence of datasets, temporality, traceability, verifiability, etc. [3]. Considering these aspects, among other relevant dataset attributes, can help address many dataset insufficiencies.

C. Traceability for AI/ML Systems

Traceability in AI/ML systems refers to the ability to establish clear, documented links between each stage of the system's lifecycle, from high-level safety and functional requirements, through design and implementation, to verification, validation, testing, and maintenance activities. This ensures that every requirement can be traced forward to evidence showing it has been met and traced backward to the design and safety objectives it supports. For AI/ML systems, these traceability practices are expected to be largely consistent with those of non-AI/ML systems.

Where applicable, AI/ML systems also require traceability to be extended back to the actual training data used [3]. As mentioned above, the quality and selection of training data can directly affect system performance, and as such, it is essential to confirm that this data is acceptable and properly linked to the relevant AI/ML requirements. For example, if an AI/ML requirement specifies detecting anomalous vibrations in a mechanical system, but the training data contains no instances of anomalous vibrations, a traceability review will reveal this gap.

To accomplish this, dataset requirements should be defined in a way that allows them to be directly linked to the AI/ML safety requirements. These links can then be used as evidence to support that the relevant AI/ML safety requirements are met [3].

D. The Dynamic Nature of AI/ML Systems

In general, non-AI/ML systems are approved for a specific version that remains static until deliberately changed through a formal MoC process [2]. AI/ML systems, however, can exhibit dynamic behavior even without explicit updates, as their performance may be influenced over time by evolving environmental conditions, shifts in input data distributions, degradation or replacement of upstream components, or changes in system usage patterns. These factors can alter the model's behavior relative to its validated state, requiring proactive monitoring and management to ensure safety objectives remain valid.

To mitigate risks associated with this dynamic nature, it is essential to first understand the full operational context of the AI/ML system. This includes its interfaces with other systems and subsystems, the physical environment in which it operates, the conditions under which its functionality is triggered, the potential for distributional shifts in input data, and the model's behavior under abnormal or degraded conditions.

Changes to conditions are inevitable during the system's operational life, especially over extended periods. These can include environmental changes (e.g., temperature, humidity), operational shifts caused by maintenance, shutdowns, or emergency situations, as well as input data variations resulting from component replacements or recalibration of related subsystems. In some cases, these changes can be mitigated by implementing monitoring mechanisms that verify whether incoming data remains consistent with the distributions observed during training.

Ongoing re-evaluation of AI/ML safety requirements and supporting evidence is critical for maintaining safety throughout the system's lifecycle. This continuous assessment ensures that changes in operating conditions do not invalidate prior assurance, and that corrective actions can be taken promptly if performance deviates from safety expectations [3].

E. AI/ML-Specific Failure Modes

AI/ML systems often have unique failure modes compared to non-AI/ML systems. In traditional non-AI/ML systems, the transformation from input to output is typically governed by explicit, human-readable logic, allowing engineers to clearly see which input conditions produce specific outputs. By contrast, AI/ML systems, particularly those using neural networks, often lack this high level of interpretability for two key reasons. First, their input–output mapping is defined by highly complex, high-dimensional, and non-linear mathematical relationships, making it difficult for a human to mentally trace cause and effect. Second, the learned internal representations do not necessarily correspond to human-understandable concepts, meaning there is no clear human-interpretable link between the input features and the model's decision process. Together, these factors make it harder to predict or explain AI/ML behavior, which can lead to unexpected or non-intuitive failure modes.

At a high level, standard testing methods such as component and integration tests using pass/fail criteria are still applicable [3]. The main difference is that AI/ML system testing will often use alternative testing methods within these high-level testing categories to gain confidence in the system's safety.

Methods for testing AI/ML systems can include a range of approaches, such as statistical testing, metamorphic testing, K-way combinatorial testing, gradient-based search methods, synthetic test case generation, expert knowledge-based testing, adversarial testing, and robustness testing [3]. In our experience, metamorphic testing, K-way combinatorial testing, and expert knowledge-based testing have proven effective for AI/ML system safety analysis [4] [5].

F. *Influence of AI/ML Systems on Non-AI/ML Components*

An area of concern when using AI/ML system is its potential impact on interconnected non-AI/ML systems.

To mitigate the likelihood of the AI/ML system affecting other non-AI/ML systems, a strong understanding of the interconnections is important. Creating an AI/ML system definition detailing the system's functionality, interfaces (e.g., inputs and outputs), and requirements is a first step to understanding potential interconnections.

Additionally, implementing sufficient architectural mitigations, such as isolation strategies, could reduce the impact of AI/ML on interconnected systems.

AI system integration, verification, and validation testing, including the potential for virtual and physical testing [3] help increase confidence in the AI/ML system.

The framework introduced in Safety Integrity levels for Artificial Intelligence [1] can help categorize and provide high-level guidance on the use of AI/ML systems with respect to safety integrity levels (SIL). This AI-SIL framework provides guidance on how two attributes of an AI/ML system (entropy of inputs and non-determinism of outputs) affect level of rigor for scrutinizing AI/ML systems.

G. *AI/ML Safety Assurance Case(s)*

In addition to the above, it is essential that a full, system-level, top-down safety assurance case is developed and maintained, even if only for internal purposes, and kept current as the AI/ML system and/or its operating environment evolve. ISO/PAS 8800 stresses the importance of safety assurance cases for AI/ML systems and outlines structured approaches in its sections on "Assurance arguments for AI systems" and "Safety analysis of AI systems" [3]. These are not merely optional guidance, they are critical tools for coordinating safety analysis, organizing testing activities, and assembling evidence to demonstrate that risk from AI/ML systems is acceptably mitigated. A structured safety assurance case not only organizes evidence into a coherent, accessible form but also captures the logical reasoning that supports (or challenges) the adequacy of that evidence. This makes it easier to identify and address risks or design gaps that could otherwise go unnoticed. Smaller, component-level assurance cases can be created for individual AI elements when a "bottom-up" approach to system safety is taken.

Building AI assurance case(s) are expected complement or work in lieu of other safety analysis techniques such as FTA, FMEA, STPA, ETA, Bayesian networks, or HAZOP to demonstrate acceptability safety for systems, especially those using AI/ML.

V. Conclusion

This paper responds to the growing demand for AI/ML integration within SIS by laying out a practical roadmap for doing so responsibly under the discipline of IEC 61511. Its purpose is to identify the key considerations that must be addressed, to identify coverage gaps of current regulatory expectations, and to outline a framework that closes those gaps with credible assurance. The aim was not to promote AI/ML everywhere, nor to relax obligations, but to show how AI/ML can be adopted in ways that remain defensible to independent assessors.

At a high level, the paper frames AI/ML adoption as a decision that must be justified against conventional methods and proportionate to an organization's ability to develop, verify, validate, deploy, and monitor the technology over time. It emphasizes making safety objectives explicit, calibrating rigor to risk (e.g., through AI-SIL concepts), and plan from the outset for controlled operation and change management.

The analysis contrasted deterministic, code-centric expectations with AI/ML's data-driven, potentially probabilistic behavior; captured the implications for requirements, datasets, architecture, V&V, and post-deployment governance, and summarized them as mitigation patterns. Two complementary integration approaches were outlined, a bottom-up and a top-down approach, each with common threads for clear and testable AI/ML safety requirements, architectural isolation, datasets treated as lifecycle artifacts with traceability, AI appropriate V&V, and continuous performance monitoring, all with an auditable assurance case that ties every claim to evidence.

Taken together, these elements provide a coherent path for organizations that must reconcile IEC 61511 compliance with AI/ML capabilities. By making decisions explicit, engineering actions in line with risks, and maintaining a live assurance case, stakeholders can realize the benefits of AI/ML while preserving the engineering rigor, transparency of process, and safety of functionality that SIS demands.

VI. Bibliography

[1] S. Diemert, "Safety Integrity Levels for Artificial Intelligence," in *IEEE SafeComp*, 2023.

[2] International Elecrotechnical Commission, "IEC 61511: Functional Safety - Safety Instrumented Systems for the Process Industry Sector," IEC, 2004.

[3] International Organization for Standardization, "ISO/PAS 8800: Road Vehicles - Safety and Aritificial Intelligence," ISO, 20204.

[4] L. Millet, "Specifying Functional Safety Requirements for AI/ML in Terms of Metamorphic Relations," CSL, 2024.

[5] S. Diemert, "Challenging Autonomy with Combinatorial Testing," in *IEEE International Conference on Software Testing*, 2023.