Classification for Avionics Capabilities Enabled by Artificial Intelligence

Andreas Schweiger*, Bjoern Annighoefer†, Marina Reich*,Christoph Regli§, Yannick Moy¶, Thomas Soodt**, Alexis de Cacqueray*, Romaric Redon‖
*Airbus Defence and Space GmbH, †University of Stuttgart, §School of Engineering, Zurich University of Applied Sciences (ZHAW), ¶AdaCore, **German Aerospace Center (DLR) ‖Airbus SE,

***Abstract*—Artificial Intelligence extended the limits considerably of what is technically feasible. In avionics, stakeholders are also pushing AI. However, research results are usually confronted with restrictions in avionics: Safety and certification are often assumed as showstoppers. This is not quite precise, since stakeholders are approaching each other. We collect the demand for AI applications in avionics and develop a classification scheme for these. For each identified AI class the status quo is compiled. We conclude with an assessment of AI readiness and the identification of necessary research effort for AI in avionics.**

***Index Terms*—artificial intelligence, machine learning, perception, planning, reasoning, standards, certification, qualification**

I. Introduction

Artificial Intelligence (AI) pushed the limits considerably of what is technically feasible. Cutting-edge AI applications have been implemented in many domains: In our daily lives AI is used e.g. for face recognition to unlock smartphones, for digital assistants with voice recognition, or in smart homes. Substantial success is also achieved in e.g. health care, in smart factories, or for advancing autonomous driving. In avionics, of course, governments, funding agencies, industry, and academia are heavily pushing AI, as well. Respective applications involving AI can be deployed e.g. to drones, to air taxis, for crew (work load) reduction, for flight path efficiency optimization, for predictive maintenance, or for human-machine interaction. However, research results are usually confronted with restrictions in avionics: Safety, qualification, and certification are often assumed as showstoppers. This picture is not quite precise, since stakeholders are approaching each other: (i) There are different use cases for AI that have less stringent regulations. (ii) Standards for applying AI in avionics are on their way. (iii) Promising verification, testing, and safeguarding techniques for AI applications are under development. (iv) Due to the daily use of AI its public acceptance increases steadily.

Already in 1983, Klos et al. required AI to reduce complexity in the cockpit by an AI based electronic crew member [1]. AI systems at that time however, were mainly rule-based expert systems, such that the conclusion in terms of certification was: "[...] many of the algorithms required to implement [...AI...] are already in today's avionics".

Ten years later, technologies for AI-based pilots have been test-flown [2]. The tested systems showed superior performance and were deterministic, but were judged to be hard to certify due to their large input space as well as missing intermediate artifacts and qualified tools. In general, however, it was concluded that so-called deterministic knowledge-based systems can be certified according to the RTCA DO-178B objectives. Concerning neural networks (NN), the authors stated that "AI-based systems with learning capability are unlikely to appear in civil aircraft". The situation is different today. Gatti/Damine claim that AI is one of the three pillars of future avionics systems [3], and also Annighöfer et al. assess the integration of AI as one of the major current challenges in avionics [4]. Today AI majorly means Learning Enabled Components (LEC). Boeing [5] and Airbus [6] are actively and publicly carrying out research projects of AI applications with trained NNs including test flights. [7] report major advances in the validation, verification, and explainability of AI , but also identify fundamental incompatibilities of LECs with the accepted development and certification process of safety-critical aircraft systems: (i) the data (invisibly) contains the requirements instead of human written requirements; (ii) practically it is infeasible to achieve an equivalent level of confidence compared to RTCA DO-178C objectives for complex AI implementations by pure testing; (iii) supervising a complex AI system by certified backup controllers will not be possible for every application. However, the report makes optimistic recommendations on two concepts: (i) Formal verification of LECs by a theorem prover, which is possible if the input and output space is restricted, the LEC has limited complexity, and the pre conditions, post conditions, and the LEC itself can be mathematically expressed. (ii) Runtime assurance by certified backup controllers, which is possible if traditional algorithms are available for the problem at hand and can reliably judge the behavior of the LEC. Vidot et al. share the optimism on formal methods for the verification of LECs [8]. In addition, authorities have reacted: EASA released a first set of practically applicable objectives [9] for the assurance of AI based human assistance applications (EASA AI level 1, see Sec. IV-A). EASA's AI roadmap [10] states that certification of objectives for complex AI applications is ready by 2028. However, up to the point of the first certified AI system in avionics there are no Accepted Means of Compliance (AMC) as we are used to in avionics.

Future capabilities driven by AI formed a central aspect during the latest scientific workshop on *Avionics Systems and Software Engineering* (AvioSE, [11]). During the workshop OEMs, authorities, software developers, and pilots participated in a panel discussion. This paper picks this up and delivers the

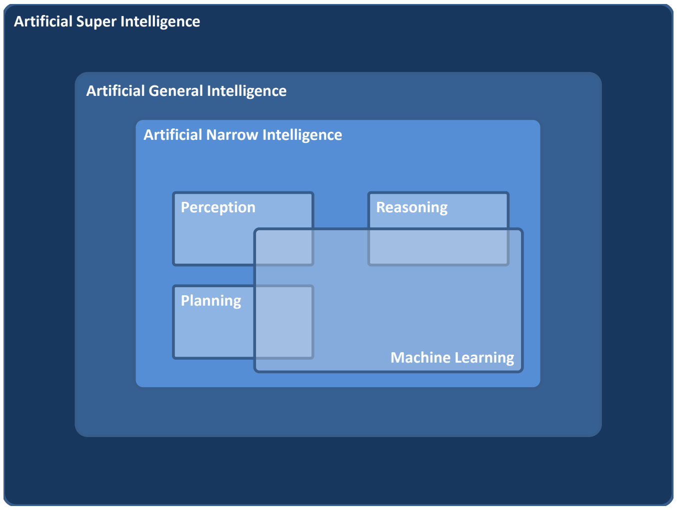

Fig. 1: AI Categories

following contributions: (i) An up-to-date review on applications, standards, and assurance methods of AI in the avionics context. (ii) A fine grained classification of AI applications, which leads to clear suggestions for their further development and certification. (iii) An up-to-date collection of open AI challenges.

II. AI Technology Definitions

According to [12, p. 2-5] a system exhibiting AI can be defined as **acting humanly**, **thinking humanly**, **thinking rationally**, and **acting rationally**. From a technical perspective we define the AI categories demonstrating these AI properties and implementing such systems as depicted in Fig. 1 and explained as follows:

Artificial Super Intelligence (ASI) is the most advanced form of AI. It refers to an intellect that is much smarter than the best human brains in practically every field, including scientific creativity, general wisdom, and social skills. This is said to be the last man-made technology, as any subsequent level of technological sophistication can be derived by this kind of AI itself. **Artificial General Intelligence** (AGI): Human level intelligence is applicable universally to a wide variety of problems and capable of general intelligent action. In other words, AGI seeks to emulate human-like intelligence on a hardware substrate. **Artificial Narrow Intelligence** (ANI): Machine intelligence that equals or exceeds human intelligence or efficiency at a specific task. This category is comprised of the domains **perception** (perceiving sensory inputs and extracting relevant information), **planning** (developing an action plan to reach a desired goal), **reasoning** (deducing logical implications from an observed state), and **machine learning** (ML) describing a form of problem solving, which is not explicitly programmed towards solving a specific problem instance, but rather infers solutions by processing a large set of examples. Perception, planning, and reasoning can be implemented using ML. Thus, the mentioned four areas have overlapping parts in Fig. 1. As the current research is focusing on the development of ANI, from which robust results are already available or expected in the short-term view, this paper is limited to this category, as well.

III. Demand for AI Applications in Avionics

The deployment of ANI in avionics follows one or several of these goals: (i) Replacing a function with a more sophisticated one. (ii) Enabling a function not feasible without ANI. (iii) Realizing an existing function with less effort. According to this this Sec. compiles the demand for ANI in avionics. First, general AI domains are reviewed according to possible applications in avionics. Second, the latest applications of AI involving avionics are summarized.

A. Transferring General AI Domains to Avionics

AI applications showed astonishing performance in applications, which were classified as almost infeasible a decade ago, namely Machine Perception (MP), Natural Language Processing (NLP), Control Theory (CT), Reasoning Systems (RS), and Data Analysis (DA). These domains have possible applications in the avionics context.

1) Machine Perception: MP is a provision of information similar to what a human would consider to judge and to react in certain situations. Whereas this definition holds true for most avionics systems with sensors, AI based MP became important, where the data is so complex that the information required cannot be retrieved reliably and efficiently with classical methods. Pretty prominent is *Object Recognition*, as a sub-category of Computer Vision, which is the identification of objects from image data. MP can also detect phenomena humans cannot perceive, e.g. infrared images, radar data, or LIDAR scans. From the perspective of avionics, AI based MP adds *artificial sensors* for information that could not be measured precisely before.

The identification of objects is assumed to be useful in several situations: On ground an assistance during taxiing for the confirmation of taxiways according to ATC instructions could prevent delays and incidents. Possible is, for instance, a camera-based localization and mapping approach and the automated *reading of taxiway/runway signs*. Another application is the automatic detection of the taxiway/runway centerline, in order to enable *automated taxiing, take-off, and landing*. Moreover, automated object detection could assure the appropriate *clearance from obstacles*. During flight, the *detection of other traffic* without transponder (e.g. Automatic Dependent Surveillance – Broadcast (ADS-B) or FLARM), i.e. non-cooperative participants, has the potential to reduce the crew workload and is a major enabler for autonomous air traffic. In the race towards unmanned piloted vehicles, the *sense-and-avoid* sensor is a considerable challenge. Most promising seems an (AI-)fused perception of optical, radar, and LIDAR data. Last, the *detection of visual landing cues* during approaches with low visibility could increase safety. Within the aircraft, object recognition could be deployed for the *detection of flight crew fatigue, stress, abnormal situations, and health issues*.

2) Natural Language Processing: NLP deals with understanding natural language. A recent benchmark showed that trained NNs can perform even better than humans [13]. In the cockpit NLP could be used for *automated processing for radio (ATC) calls* and for cross-checking with the corresponding crew actions (e.g. flight management system (FMS) entries, or changes in altitude, speed, and for tracks). In general, *voice-controlling in the cockpit* might simplify the aircraft management. The dynamic synthesis of speech could be used to provide the flight crew *detailed audio advice* in situations where other senses are distracted, which could provide assistance in critical and abnormal situations. Within the cabin, it is only a matter of time when passengers demand *voice-based control of their in-flight entertainment* system.

3) Reasoning Systems: RSs have the purpose to draw conclusions. Rule-based Expert Systems are already state-of-the-art, e.g. for the situation-adaptive system pages in the Airbus A320. However, recent advances in learning-based RSs (e.g. the Google phone call assistant [14]) show that almost human behaving RS might be possible (like the "electronic crew member" desired by [1]). An RS could *manage the checklists* for crew members and present these according to the current situation and supervise the proper execution. A virtual co-pilot could assist the pilot by a *second opinion during decision making* especially in critical situations. For instance, the virtual co-pilot could carry out a *what-if planning* if one engine is inoperative. It could calculate the ceiling and range objectively while the pilot is busy getting the situation under control. It could also pre-plan the all engine out gliding distance, suggest en-route alternate airports, and remind the pilot on the point of no return. A virtual co-pilot could also relief the pilot partially from air traffic management (ATM) by *accomplishing tasks from air navigation* services automatically. This could lead to an improved air transportation performance, e.g. by the possibility to reduced separation distances. Another task for the virtual co-pilot could be the *runway exit prediction*. On the path to *single or no pilot cockpit* there seems no way around automating those tasks and especially LECs seem to be a complementary method when trained by millions of flight data. In addition, AI based RSs could *improve warning systems*. For instance, wind shear detection and associated decision making and landing vs. go-around decision making. In general, the prediction of threats from terrain, traffic, and airspace data increases the *threat awareness* of the flight crew and helps to anticipate and prevent critical situations, e.g. by in-flight weather circumnavigation. Last, the steadily increasing complexity of aircraft systems, especially in abnormal situations, results in the demand for expert systems assisting in taking the right decision and for providing engineering support, for instance by autonomously *identifying failure root causes and mitigation strategies*. Reasoning Systems based on formal methods and theorem proving might *support the development of safety-critical systems*, but currently no airborne application is foreseen.

4) Planning Systems: PSs, also known as *Automated Planning and Scheduling*, comprise problems where a list of actions and execution times needs to be compiled to (optimally) achieve a higher-level goal. This can be accomplished with classical methods of (combinatorial) optimization, but also LECs are promising methods. Planning applications in avionics are primarily the planning and optimization of flight paths, e.g. an emission-optimal in-flight fuel planning by an optimal selection of cruise speed and flight level, which are dynamically adapted according to the situation. In general, flight path support, flight profile optimization, descent planning (e.g. continuous descent approach) are promising fields for the application of AI. Also, the complete execution of a flight or mission might be realized by LECs and leads to a virtual pilot. It is likely that such an AI pilot will in future outperform the human in at least some tasks, e.g. flight path precision, maneuver dynamics, and response time. This is especially important for military applications, but could also increase the efficiency of civil air transportation.

5) Control Theory: Control algorithms are a mandatory part of many aircraft functions to ensure numerical outputs of a system to be in a range of desired values and compensating for changing inputs, which works well if the input parameter range and dynamics are well-known and the system behavior does not change. The benefits envisioned by LECs are, first, better capabilities to cope with unknown situations, i.e. unknown changes in the system behavior or (partially) faulty inputs. Second, the controller performance might be increased by dynamically adapting to the current environment. This is also known as Adaptive Control. LECs have the potential to increase the performance and operation range of Adaptive Controllers and ease their development [15]. A flight control law that is able to *adapt to structural damages, degraded performance (e.g. due to airframe icing), load changes, or (partially) faulty sensors, actuators, and systems* without knowing the degradations beforehand, would increase safety and the level of autonomy. This might be a back-up system carrying out an *emergency landing* in case of crew incapacitation. Other applications are *upset recovery* or a proactive prevention of critical situations, e.g. by an *extended flight-envelope-protection* that takes into account fuel status, aircraft status, crew status, airspaces, and terrain. Last, NN based controllers show *superior performance* in existing applications or are easier to develop, because they are learned and not implemented. A prominent example is the optimization of flight-control laws.

6) Data Analysis: DA deals with the extraction of previously unknown, unquantifiable, or unusable information from large (heterogeneous) data sets. Advances in DA methods, massive available data, and high processing power created the field of data mining. Concerning avionics, the number of sensors increases steadily and a recording of massive amounts of data is state-of-the-art. During runtime DA methods could be used to create *virtual sensors* from values existing already. For instance, altitude, velocity, or orientation can be estimated from other values and provide an *additional degree of redundancy* or *sanity checking*. This is also called Analytic Sensors. Other examples are the *fuel tank quantity evaluation* or *icing detection* from regular flight data. The *detection and classification of failures* by finding abnormalities in data streams are additional applications. Furthermore, maintenance can be improved by *detecting indicators of uprising malfunc-*

tions, and deriving countermeasures before the malfunction occurs. Learning enabled DA could also support preventing cyber-attacks on the avionics system. DA methods could *distinguish between normal behavior and attack behavior*, by continuously monitoring data streams. In cooperation with reasoning system, countermeasures might automatically be deployed and their effectiveness is learned and considered for future reactions. Last, DA methods might assist the development process of avionics hardware and software. Development artifacts can be analyzed by AI and support information can be generated. For instance, the *quality of textual requirements* might be evaluated by AI, and *similar, malformed, duplicated, or contradicting requirements* might be identified. Moreover, *identifying possible errors or critical sections of source code* is possible [16].

B. Reported applications

In order to judge the importance of general AI domains and applications for avionics we review the most recent reports on applications in avionics:

- Within the *Intelligent Flight Control System* (IFCS) research project, NASA built a self-healing IFCS on top of fly-by-wire. The IFCS compensates changes in the control behavior of the aircraft, e.g. by structural damage. It uses pre-trained and online learning NNs. The IFCS was tested with an F-15 in flight and showed good capabilities in handling of abnormal configurations.
- *Advanced flight control laws* based on NNs have been proposed for fighter aircraft [17], UAVs [18], Drones [19], and CS-25 aircraft [20]. The motivation is commonly a better performance in changing dynamics or environments.
- Boeing's *Airpower Teaming System* (loyal wingman) uses LECs to *fly a fighter aircraft autonomously* or to support a crewed aircraft with maintaining a safe distance between aircraft [5]. AI controlled formation flights were successfully carried out with five small scale jets [21]. Test flights with the real fighter aircraft were carried out with a single fighter aircraft and a predefined route [22].
- DARPA carried out a benchmark for *AI controlled fighters* in a simulated environment, the so-called *AlphaDogfight trials* [23]. Heron Systems' *F-16 AI Agent* outperformed the other AI pilots and a human. It is based on an NN that was trained with dogfight data [24].
- Within the EU's FIVER project an *advanced FMS* was developed, which uses LECs. It is trained with topographic and meteorological data to derive the most energy-efficient flight trajectory [25].
- A vision-based runway detection LEC was developed by Airbus in the ATTOL project for an *autarkic automated take-off and landing* system. It was demonstrated successfully with an A350 for take-off. The sources are not clear about landings, but it is likely that theses have only been carried out virtually, because of potential catastrophic consequence of a malfunction. During demonstration flights safety-pilots were on board and ILS was available as a backup [6], [26].
- Airport navigation by voice-control was demonstrated by Nilfa et al. utilizing Google's speech-to-text engine in a simulation [27]. The demonstrator showed a robust understanding of commands followed by an auto-taxiing procedure. Image processing was used to hold the centerline and detect junctions. Safety was ensured by the pilots who could always stop auto-taxiing.
- An *optical tracking of the centerline* was used to demonstrate a runtime assurance architecture for LECs [28]. The runtime assurance allowed for go, slow, and stop intervention of the LEC, which was demonstrated successfully in a simulation.
- Chen et al. used NNs for the *identification of faults* in avionics hardware for rectifier units and improved the classification rate significantly [29].
- Adhikare et al. reported numerous ML based approaches for *predictive maintenance* of aircraft components [30]. They state a great opportunity for an increased operational reliability, but also a massive training data demand, which can probably only be fulfilled by artificially generated data.
- Garcia et al. defined five categories for applying AI to *aviation cyber-security* and conducted a literature review. Mainly anomaly detection and component fault identification show reasonable work and promising results.
- Lüttig et al. demonstrated an *automatic detection of peripherals* connected to Integrated Modular Avionics (IMA) modules that uses NNs to identify LEDs and electric motors based on their electric properties [31].
- Girard et al. used ML for *GPU performance approximation* in avionics hardware development [32].
- The upcoming KASIMIR project will investigate learning-based *AI Assistants in the development of safety-critical software and hardware* [33].

IV. AI Classification Scheme in Avionics

A. AI Autonomy Levels

AI is assumed to be a key enabler for system autonomy. For a device to operate autonomously , it needs to be capable of taking over several tasks typically assigned to a human operator. A widely adopted categorization of autonomy is presented in Tab. I.

Next to the autonomy levels introduced by [34], SAE International™ have specified the levels of driving automation in SAE J3016™ (Fig. 2). In addition, EASA have suggested a classification for AI/ML applications (Fig. 3, [10, p. 16], which was evolved in [9, p. 9]). [34] and SAE J3016 classify autonomy, while the EASA classification focuses on the role played by AI in the overall system, including the interaction with the human. We propose the mapping between these classifications as presented in Tab. II.

B. Certification Baseline

This Sec. describes the certification baseline which is necessary from the perspectives of both certification authorities and development organizations. The accepted path to airworthy

Level	Description
1	Human does the whole job up to the point of turning it over to the computer to implement.
2	Computer helps by determining the options.
3	Computer helps determine options and suggests one, which human need not follow.
4	Computer selects action and human may or may not do it.
5	Computer selects action and implements it if human approves.
6	Computer selects action, informs human in plenty of time to stop it.
7	Computer does whole job and necessarily tells human what it did.
8	Computer does whole job and tells human what it did only if human explicitly asks.
9	Computer does whole job and tells human what it did and it, the computer, decides he should be told.
10	Computer does whole job if it decides it should be done, and if so tells human, if it decides he should be told.

TABLE I: AI Autonomy Levels [34]

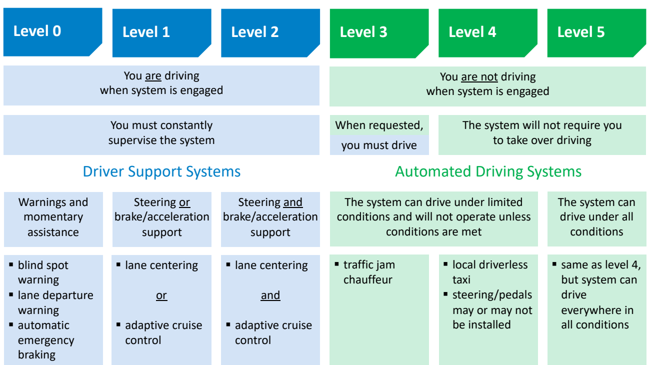

Fig. 2: SAE J3016 Levels of Driving Automation [35]

avionics applications follows the principles of intent, correctness, and innocuity [36], i.e. the certification process must ensure the safety of the function by proofing the correct and complete execution of the intended function and the absence of any harmful unintended function. Usually, this is achieved in a requirements based development process with exhaustive review, testing, and reporting activities as well as clear test coverage objectives. For the certification of safety-critical software for the deployment to aircraft, certification authorities (EASA, FAA) require particular development processes. These cover both hardware and software. For the purposes of this paper we focus on software only. The most important standard documents guiding the development process are AMC 2x.1309

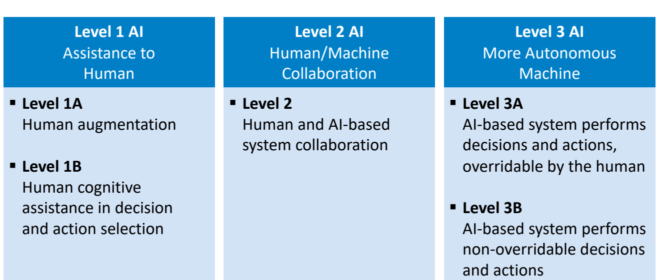

Fig. 3: EASA Classification of AI/ML Applications [9, p. 9]

AI Autonomy Level	SAE J3016™ Automation Level	EASA AI/ML Classification
1	0	1A
2	1	1A
3	1	1B
4	2	2
5	2	2
6	2	2
7	3	3A
8	4	3A
9	4	3A
10	5	3B

TABLE II: Mapping of AI/ML Levels

(Certification Specifications for Normal Aeroplanes/etc., Accepted Means of Compliance, System Design and Analysis), ARP 4754A (Guidelines for Development of Civil Aircraft and Systems), ARP 4761 (Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment), RTCA DO-178C (Software Considerations in Airborne Systems and Equipment Certification), RTCA DO-331 (Model-Based Development and Verification Supplement to DO-178C and DO-278A), RTCA DO-332 (Object Oriented Technology and Related Techniques Supplement to DO-178C and DO-278A), and RTCA DO-333 (Formal Methods Supplement to DO-178C and DO-278A).

An intermediate step towards a certification of AI based applications is already in place: It is possible to certify AI systems comprised of trained (deep) NNs, whose models have been frozen and tested, in order to demonstrate a deterministic behavior. However, this sacrifices the self-learning capability of AI systems in the operational environment. Moreover, this requires that the AI is testable with the required coverage and that (manually created) requirements exist that can be verified. Thus, an appropriate certification framework is required: The standards mentioned above form the baseline for the certification of AI technology as well, but need to be adapted [10], [37]. Existing FAA FAR Part 21 (Certification Procedures for Products and Articles) might be used as a framework for the development and certification of AI/ML solutions, where certain paragraphs like CS 25.1309 could be valid for the assessment of the safety of AI/ML-based systems [10]. However, current standards are not going to be adopted completely for the certification of AI based applications. Existing gaps need to be covered as required by EASA's AI Roadmap [10]. For this purpose, the international committee SAE G-34/EUROCAE WG-114 (Artificial Intelligence in Aviation) has been established in 2019 and is tasked with the development of the required standard documents for the deployment of AI technologies.

C. Classification Scheme

We propose to classify the applications involving AI in avionics as given in Tab. III. The applications can be put into six major domains from flight automation to development automation. The sub-domains comprise the most mentioned AI applications for avionics. In each domain, these are sorted according to the degree of autonomy required. In terms of learning, two categories are differentiated: First, offline trained AI, that got its behavior before its execution and behaves deterministically during runtime with respect to its input, and, second, online trained AI, which can adapt according to its input during runtime and changes its behavior, e.g. to improve its performance. In most cases, a higher degree of autonomy results in more severe malfunction consequences as well as NNs with online learning being used. The demand is judged to be rather urgent only for the application of sense-and-avoid and cyber-security detection. Most others are nice to have and a few are already implemented. The Technology Readiness Level (TRL) is low for applications with highest demand and autonomy level. We also assessed the need for certification and whether experience with the certification of such an application is existing. The degree of autonomy determines the constraints regarding the effort for the certification: The higher the autonomy level is, the more effort is required for the certification for the deployment of AI to aircraft.

We identify an urgent demand for AI based solutions where no other solution exists and the demand for a solution is prevalent. Most prominently this seems to be the case for unmanned vehicles that require an autarkic sense-and-avoid system. Currently, no classical technology seems sufficient. A certified solution is mandatory. In addition, autonomous flying by AI seems mandatory in the military domain, to stay on par with possible opponents. Certification is an issue, but currently does not seem to be the primary concern. In the civil domain autonomous flying has a strong commercial impact and is a game changer in aircraft operations, but we do not state an urgent demand today.

An application of LECs that could become mandatory but is hard to judge today, is autonomous cyber-threat protection. The non-predictive, ever-changing nature of cyber-threats make learning-based methods a possible tool to meet the cyber-security regulations CS-2x 1319 (Certification Specifications for Normal Aeroplanes/etc., Notice of Proposed Amendment 2019-01: Equipment, systems and network information security protection). However, the importance will depend on what level of effectiveness AI based cyber-security proves in the field.

The other applications, such as virtual assistance, predictive maintenance, flight path optimization, and cockpit voice control seem to be AI applications which increase the efficiency, comfort, safety, or added value of the aircraft, but are not mandatory. For most of them safety can be achieved without the certification of LECs.

V. AI Technology Status Quo

The classification scheme highlighted that at least two of the envisioned applications use methods that have not been certified successfully yet. Methods considering the (self-)learning nature of AI will be required. This Sec. summarizes the status quo of AI assurance methods in order to identify missing elements on the path to certified AI applications. We assume here the most challenging conditions of an AI used in a safety-critical function with catastrophic failure conditions. We focus on the qualification of the software since it is assumed that AI hardware is not special from the certification point-of-view. This does not mean that hardware accelerating AI algorithms is already completely certified. On the contrary, the certification for the deployment of e.g. GPGPUs with their considerable intrinsic parallelismis a particular challenge also know from the still missing certification of multi-core CPUs.

A. AI Trustworthiness Analysis

For the technical implementation of trust EASA have introduced the concept of **trustworthiness analysis** (including human-machine interaction, HMI, see Section IV-A), which includes the building blocks **Learning Assurance**, **AI Explainability**, and **AI Safety Risk Mitigation** [10, p. 16-20]. This analysis framework is to guide the development and evaluation of AI technology for certification: Learning assurance ensures the selection of correct and complete training data, of appropriate learning facilities (e.g. convolutional NNs) including their hyper parameters for configuration, and verification aspects for the inference phase (e.g. formal methods or Generative Adversarial Networks) for ML applications. Explainability (Explainable Artificial Intelligence, XAI) requires that humans are able to understand how e.g. an ML implementations is generating its results. XAI can be reached using the following approaches:

Layer-wise relevance propagation (LRP) denotes a "methodology that allows to visualize the contributions of single pixels to predictions for kernel-based classifiers over Bag of Words features and for multilayered neural networks. These pixel contributions can be visualized as heatmaps" [39, p. 1]. Thus, certain input is determined which contributes most to the provided result. This in turn helps an AI expert to focus on these aspects for understanding how an input determines output. The approach suggested by Bach et al. is suitable for NNs implemented for object recognition. Thus, methods for the XAI of other AI technologies are missing. In addition, there is no approach for the objective quality evaluation of the generated heatmap.

The **counterfactual method** helps to determine the impact of changes of the ML models and to determine the ramifications after editing input data [40]. Wexler et al. have presented a GUI based tool for this purpose. Although it can be applied to ML solutions in general, it is still necessary for an expert to assess the tool's findings. Thus, additional work for reducing the expertise level is necessary e.g. for identifying outliers and parts of the data for which the ML model needs improvements.

Local interpretable model-agnostic explanations (LIME) is a "technique that explains the predictions of any classifier in an interpretable and faithful manner, by learning an interpretable model locally around the prediction" [41]. Ribeiro et al. introduced this for the text and image domains. However, the approach needs to be extended for handling also speech recognition and object recognition in videos as well as other ML methods.

Risk mitigation suggests that an AI application is supervised by additional means to make sure that it is working according to its specification. For the certification of AI for its deployment to avionics the required trustworthiness needs to be established.

TABLE III: Classification of AI Applications in Avionics

Domain	Sub-domain	Learning	Autonomy level	Malfunction severity*****	AI methods	Demand	TRL*	Certification required	Certification experience
		online, offline	1-10 (general); 1a-3b (EASA)	NSE, MIN, MAJ, HAZ, CAT	Neural Network (NN), Classical ML** (CML), Rule-based (RB)	As soon as possible (ASAP), nice to have (NTH), state of the art (SOTA)	1-9	mandatory, preferred, not required	none, possible, done, N/A
Flight automation	Fully autonomous flight	online	10; 3b	CAT	NN	NTH/ASAP****	3	mandatory	none
	Sense-and-avoid	offline	7-10; 3a/b	CAT	NN	ASAP	3	mandatory	none
	Visual navigation***	offline	7-10; 3a/b	CAT	NN	NTH	3	mandatory	possible
	Partial task automation	offline	6-8; 3a	CAT	RB, NN,	NTH	N/A	mandatory	done
	Adaptive pilot assistant	online	2-5; 1a/b	NSE	NN,CML	NTH	N/A	preferred	N/A
	Rule-based pilot assistant	offline	2-4; 1a	NSE	RB	SOTA	9	preferred	N/A
Flight controls	Adaptive controller	online	10; 3b	CAT	NN, CML	NTH	7	required	none
	Efficient controller	offline	10; 3b	CAT	NN	NTH	7	mandatory	possible
NLP	Autonomous processing of ATC calls	offline	7-10; 3a/b	CAT	NN	NTH	3	required	none
	Assistance in processing ATC calls	offline	6-8; 3a	MIN	NN	NTH	5	mandatory	none
	Cockpit voice control	offline	5-6; 2	MAJ	NN	NTH	6	mandatory	possible
	Advanced audio advice	offline	3;1a	MIN	NN	NTH	7	preferred	N/A
	Voice-control for passengers	offline	2;1a	NSE	NN	NTH	7	not required	N/A
Cyber-security	Anomaly detection	online	2-3; 1a/b	NSE	NN	ASAP	3	not required	N/A
	Autonomous counter measure	online	5-10; 2-3a/b	NSE	NN	NTH	2	mandatory	none
Predictive maintenance	Learning predictors	online	3;1b	NSE	NN,CML	NTH	4	not required	N/A
	Pre-trained predictors	offline	3;1b	NSE	NN,CML	SOTA	6	not required	N/A
Development assistant	Qualified tool	offline	2-4;1b-2	CAT	NN	NTH	2	mandatory	none
	Unqualified support	online	2-4;1b-2	NSE	NN	SOTA	N/A	not required	N/A

* Technology Readiness Level according to [38]; ** Fuzzy logic, Decision tree, SVM, KNN, etc.; *** Runway detection, centerline detection, obstacle clearance, etc.;
**** Currently only military and UAV stakeholders have an ASAP demand. For civil air transportation it is judged to be NTH.
***** According to AMC 25.1309: No safety effect (NSE), minor (MIN), major (MAJ), hazardous (HAZ), catastrophic (CAT).

B. Runtime Assurance

Runtime Assurance is the application of architectural principles that make it possible to run an application in a critical context without trusting it, because its runtime behavior is continuously monitored, and control is transferred to a trusted application in case the untrusted application deviates from the set of allowed safe behaviors. One means for Runtime Assurance is the provision of redundancy [42] for safety-critical systems and of re-configuration after the occurrence of failures [43]. EASA suggested such an architecture as one way of mitigating the safety risk of AI applications [10]: monitoring of the output of the AI/ML and passivation of the AI/ML application with recovery through a traditional backup system (e.g. safety net). In that scenario, a backup and hot standby system is developed according to the current standards (Sec. IV-B). This system is checking the validity of the results of the AI algorithms. If a malfunction or unexpected results are detected, the AI component is deactivated and the backup is used instead. EASA suggest additional ways of mitigating the safety risk of AI applications:

i. The human operator is supposed to stay in command (human in command, HIC) or in the loop (HITL), where the the operator accepts AI suggestions or is observing the AI's decisions and is able to override them, respectively.
ii. The risk of creating wrong results by the AI is reduced by the encapsulation of ML algorithms in rule-based systems.
iii. The robustness and dependability of an AI component can be enhanced by adding an independent AI module checking the results of the component and releasing them only after the passed validity check.

Such a Runtime Assurance architecture is already described in [28] for monitoring the LEC detecting the runway. However, it was demonstrated that the taxiing speed needed to be slowed down or even set to zero due to the LEC not being capable any more to process the provided input data in time. While this approach might be an option for fail-passive applications, it is definitely excluded for fail-active applications. The monitor as part of a runtime assurance architecture can itself be based on AI for better accuracy and performance, as demonstrated in [44, RTSA] where the meta-controller is trained by reinforcement learning, providing here an example for item iii) above integrated in a runtime assurance architecture. An AI-based monitor does not solve the certification issue but complicates it by shifting it to an additional component.

C. Computer-aided (Formal) Verification for AI

Classical testing means are not sufficient for advanced AI applications: "Current test-based verification processes will never be sufficient to assess the behavior of adaptive systems" [37]. Considerable progress has been reached in the domain of formal verification during the previous years: The formal verification of software of considerable size has been demonstrated to be feasible [45]. Also the automation of corresponding verification tools has also seen major improvements regarding the fully automatic analysis of realistically sized software [46]. To conclude, even the formal verification of verification tools can be reached according to the current state-of-the-art. The challenge is to apply these for the verification of AI applications. Urban and Miné come to the conclusion ”that we are still far away from being able to verify the entire machine learning pipeline, which we argue is necessary to ensure the safe use of machine learning software in safety-critical applications” [47]. Urban and Miné suggest these future works:

i. **Data preparation**: Urban and Müller conclude that ”verification methods that detect [...] reused and duplicated data would be a valuable complement to existing approaches. More generally, approaches for tracking data provenance would provide important *traceability* guarantees” [48]. The authors require further progress for the ”inference of assumptions on the input data that are embedded in data preparation software guarantees”.
ii. **Model training**: Formal guarantees on the training process need to be enhanced. In particular, it is necessary to determine how the trained model evolves as an answer to certain input values. Furthermore, constraints on the training phase have to be developed, which accomplish the needed behavior regarding the provided input data.
iii. **Model deployment**: Previous effort is required for the verification of global robustness properties instead of local ones only. This means the verification of the model regarding any conditions and not only a limited input area. If the verification fails the source needs to be identified [49] for improving the model.

D. Tools

The ML domain is the one which is most mature, is in the focus of researchers, and is adopted widely. We give an overview of selected tools, which are useful for the development of ML applications. In this context we see the layers (i) distributed computing and (ii) AI frameworks. We compile a non-exhaustive list of prominent examples. These tools are representing the state-of-the-art, are widely adopted for the development of ML applications, and thus subject to permanent improvements:

i. Managing distributed computing: For the conduction of the learning phase a vast amount of training data is necessary. These can be managed only by using dedicated tools. **Kubeflow** (https://www.kubeflow.org/) manages ML development processes in heterogeneous infrastructures, which are based on Kubernetes (https://kubernetes.io/). Kubernetes is used for orchestrating containerized applications. Containers are often created using Docker (https://www.docker.com/). **Apache Spark™** (https://spark.apache.org/) is an analytics framework for enabling processing of huge data sets. **Apache™ Hadoop®** (https://hadoop.apache.org/) provides management facilities for processing large data sets in an distributed computing environment with several clusters.
ii. AI frameworks: **TensorFlow** (https://www.tensorflow.org/) is a library for general purpose ML. **Keras** offers an API for creating ML environments with a focus on readability by humans. It hides e.g. TensorFlow’s functionalities behind its API and thus enhances the programming model. **Caffe** (https://caffe.berkeleyvision.org/) adds an abstraction layer e.g. for using different AI accelerating hardware and renders in-depth knowledge of the hardware superfluous.

On top of these layers the actual ML model has to be selected and configured. The robustness of the developed AI application is still a challenge. For the creation and auditing of assurance cases the AdvoCATE tool for robust software engineering can be used [50], [51]. It provides the manual creation and editing of assurance arguments in the Goal Structuring Notation (GSN). Describing the task of autonomous software components using goals is well-known in software agent engineering [52] and thus exhibits already high maturity.

VI. AI Readiness Assessment

The state of LEC applications in avionics seems to be that an in-flight demonstration is feasible with sufficient safety nets, but the reported applications show that the trust is not high enough for scenarios with severe consequences. In the following an attempt is made to answer the primary questions of the applicability of AI in avionics.

A. What AI applications are ready?

i. Any AI application that does not require certification, because it cannot cause severe consequences on malfunction, i.e. the function is rated as DAL E.
ii. Any AI application that is fully testable, i.e. the number of inputs and states is of manageable size and it is computationally feasible to test each of them and it is developed and verified in accordance to the DO-178C. A subset of such a category are ML applications for which the input data set is limited to certain values during runtime.
iii. Runtime assurance is a technology that complies fully with the current certification regulations. The prerequisite is that the safety monitor and the backup function can be implemented by currently certifiable methods and are strictly segregated from the potentially unsafe AI. If the prerequisite holds, we see no reason, why not any kind of AI (offline and online learning) can be flown in such a “safety container”. We see parallels here to the strict segregation of low DAL partitions in IMA modules.

B. What AI applications are almost ready?

Formal verification has shown to be feasible for smaller AI implementations, e.g. control laws. If it is computationally feasible and the required pre and post conditions can be described and shown to be complete, formal verification should provide an acceptable means of compliance for the certification. RTCA DO-333 provides a baseline for that. This concerns the method of formal verification itself if carried out by an engineer, which will be infeasible in many cases. If theorem provers are used, they might need a tool qualification as long as the output cannot be checked by a human, which is likely to occur. A qualified theorem prover is not known to us. Moreover, the formal verification of AI currently lacks successfully certified examples.

C. What AI applications are not ready?

i. AI with intractable input that cannot be runtime assured, seems currently infeasible to certify. For instance, in object recognition and NLP the input space can never be completely tested or be meaningfully restricted. There are many counterexamples for fooling image recognition [53] and NLP [54]. These are usually triggered by generating adversarial examples and subjecting the NN to these: Tank images classified by the weather conditions instead of the actual tanks [55]; road sign detection distortion by special symbols [56]; fooling autonomous cars by generating a pattern in the image which renders the optical flow algorithm inoperative [57]; horse images classified by image watermarks and boats classified by water [58]. For image recognition and NLP approaches methods such as LRP or LIME provide insights into such effects and help to achieve a certain robustness, but currently these methods provide no meaningful quantity that can be used in certification, e.g. the probability of correct classification. This will be even worse for AI with inputs not easily sensible (see, hear, feel) by humans, e.g. as necessary for AI pilots. Although LRP and LIME are in principle applicable it will be difficult to draw conclusions from the results.

ii. AI with online-learning capabilities that cannot be runtime assured seem incompatible with certification regulations, because their behavior is not deterministic and depends on the environment the AI is confronted with in future.

D. What is necessary to make more use of AI in avionics?

i In order to enhance the robustness of ML applications such as object recognition and NLP, Generative Adversarial Networks [59, GAN] are used widely. In this approach possible input for training an NN is generated and assessed according to its validity and usefulness with two separate NNs before it is provided to the NN as input. Though this is promising, the convergence of GANs is still a challenge [60].

ii Online-learning AI could become airworthy if parts of the qualification activities can be carried out during the runtime of the system, i.e. enabling some sort of self-qualification also known as self-assurance. For instance, safety or security assessments could be carried out online if the system possesses sufficient knowledge about the intended function. This is tried in the PAFA-ONE project for self-organizing avionics modules [61], but having the self-assurance methods implemented in a certifiable way and having this accepted by the authority has by far not been accomplished.

iii Empirical reliability validation of AI methods with larger complexity such as image recognition might become possible with massive processing power and advanced (qualified) simulations or by testing the AI passively in real-world products for a long period of time.

VII. Conclusion and Outlook

Integrating AI into the avionics engineering environment is not a novel idea, in fact it is the title of an article published in 1986 [62], in a context where AI meant mainly expert systems. In our current context, AI subsumes especially (online) LECs, but similar concerns regarding the applicability of using more complex critical applications remain. We conclude that the most demanded AI applications today are sense-and-avoid, virtual (safety) pilot, and cyber-security protection. These applications if using AI are certifiable today as long as rule-based approaches or runtime assurance are used. When using NNs, formal methods seem to become AMC for certification. Currently, they suffer, however, from insufficient performance, trust, and experience. Especially online learning methods will need new methods. One possibility is self-assurance, but certifiable self-assurance methods are not available today. Though considerable challenges have been identified, there are also applications with small development effort and high benefit, i.e. those applications without safety effect, with runtime assurance, or being fully testable.

References

[1] L. Klos, J. Edwards, and J. Davis, "Artificial intelligence - an implementation approach for advanced avionics," in 4th Computers in Aerospace Conference. American Institute of Aeronautics and Astronautics, 1983.

[2] L. Harrison, P. Saunders, and P. Saraceni, "Artificial intelligence and expert systems for avionics," in DASC. IEEE, 1993.

[3] M. Gatti and A. Damien, "AI, connectivity and cyber-security in avionics," in 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE, 2019.

[4] B. Annighöfer, M. Halle, A. Schweiger, M. Reich, C. Watkins, S. H. VanderLeest, S. Harwarth, and P. Deiber, "Challenges and ways forward for avionics platforms and their development in 2019," in DASC. IEEE, 2019.

[5] Boeing, "Boeing airpower teaming system," 2020. [Online]. Available: https://www.boeing.com/defense/airpower-teaming-system/

[6] W. Bellamy, "Airbus concludes attol project that featured 'world-first' automated takeoffs and landings," Jun. 2020. [Online]. Available: https://www.aviationtoday.com/2020/06/29/airbus-concludes-attol-project-featured-world-first-automated-takeoffs-landings/

[7] AFE 87 Project Members, "Final Report AFE 87 - Machine Learning," Aerospace Vehicle Systems Institute, Tech. Rep., 2020. [Online]. Available: https://avsi.aero/wp-content/uploads/2020/06/AFE-87-Final-Report.pdf

[8] G. Vidot, C. Gabreau, I. Ober, and I. Ober, "Certification of embedded systems based on machine learning: A survey," 2021.

[9] EASA, "First usable guidance for level 1 machine learning applications - a deliverable of the easa ai roadmap," Tech. Rep. 1.0, 2021. [Online]. Available: https://www.easa.europa.eu/sites/default/files/dfu/easa_concept_paper_first_usable_guidance_for_level_1_machine_learning_applications_-_proposed_issue_01_1.pdf

[10] European Union Aviation Safety Agency, "Artificial intelligence roadmap — a human-centric approach to ai in aviation," Tech. Rep. 1.0, 2020.

[11] S. Götz, L. Linsbauer, I. Schaefer, and A. Wortmann, Eds., Proceedings of the Software Engineering 2021 Satellite Events. CEUR Workshop Proceedings, 2021.

[12] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 2nd ed. Pearson Education Limited, 2003.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018.

[14] Y. Leviathan and Y. Matias, "Google duplex: An ai system for accomplishing real-world tasks over the phone," May 2018. [Online]. Available: https://ai.googleblog.com/2018/05/duplex-ai-system-for-natural-conversation.html

[15] O. Malik, "Amalgamation of adaptive control and AI techniques: applications to generator excitation control," Annual Reviews in Control, vol. 28, no. 1, pp. 97–106, 2004.

[16] M. Pradel and K. Sen, "Deepbugs: A learning approach to name-based bug detection," 2018.

[17] B. S. Kim and A. J. Calise, "Nonlinear flight control using neural networks," Journal of Guidance, Control, and Dynamics, vol. 20, no. 1, pp. 26–33, 1997.

[18] T. Krüger, P. Schnetter, R. Placzek, and P. Vörsmann, "Fault-tolerant nonlinear adaptive flight control using sliding mode online learning," Neural Networks, vol. 32, pp. 267–274, 2012.

[19] S. Bansal, A. K. Akametalu, F. J. Jiang, F. Laine, and C. J. Tomlin, "Learning quadrotor dynamics using neural network for flight control," in CDC. IEEE, 2016.

[20] S. Heyer, D. Kroezen, and E.-J. V. Kampen, "Online adaptive incremental reinforcement learning flight control for a CS-25 class aircraft," in AIAA Scitech 2020 Forum. American Institute of Aeronautics and Astronautics, 2020.

[21] G. Waldron, "Boeing test sees five ai-enabled jets teamed up," Dec. 2020. [Online]. Available: https://www.flightglobal.com/defence/boeing-test-sees-five-ai-enabled-jets-teamed-up/141402.article

[22] M. de Git, "Boeing loyal wingman uncrewed aircraft completes first flight," Mar. 2021. [Online]. Available: https://boeing.mediaroom.com/news-releases-statements?item=130834

[23] DARPA, "Alphadogfight trials foreshadow future of human-machine symbiosis - virtual finale showcases ai's impressive abilities in simulated f-16 aerial combat," Aug. 2020. [Online]. Available: https://www.darpa.mil/news-events/2020-08-26

[24] F. Wolfe, "Heron systems shows future potential of ai in win over top f-16 weapons instructor," Aug. 2020. [Online]. Available: https://www.aviationtoday.com/2020/08/24/heron-systems-shows-future-potential-ai-win-top-f-16-weapons-instructor/

[25] Clean Sky, "Fiver feeds data to the 'brains of an aircraft'," 2021. [Online]. Available: https://www.cleansky.eu/CS2/fiver-feeds-data-to-the-brains-of-an-aircraft

[26] AIRBUS, "Airbus demonstrates first fully automatic vision-based take-off," Jan. 2020. [Online]. Available: https://www.airbus.com/newsroom/press-releases/en/2020/01/airbus-demonstrates-first-fully-automatic-visionbased-takeoff.html

[27] A. A. Nielfa, R. Q. Campos, and Q. Yang, "Autonomous taxiing for aircraft - the future of aircraft taxiing in aerodromes," in PEGASUS Student Conference, 2021.

[28] D. Cofer, "Run-time assurance architecture for learning-enabled systems," High Confidence Software and Systems Conference, 2020. [Online]. Available: https://cps-vo.org/hcss2020/cofer

[29] S. Chen, H. Ge, J. Li, and M. Pecht, "Progressive improved convolutional neural network for avionics fault diagnosis," IEEE Access, vol. 7, pp. 177 362–177 375, 2019.

[30] P. Adhikari, H. G. Rao, and M. Buderath, "Machine learning based data driven diagnostics & prognostics framework for aircraft predictive maintenance," in 10th International Symposium on NDT in Aerospace, 2018.

[31] B. Lüttig, B. Schwämmle, and B. Annighöfer, "Using neural networks to identify wired peripherals connected to integrated modularavionics hardware," in DASC, 2021.

[32] S. R. Girard, V. Legault, G. Bois, and J.-F. Boland, "Avionics graphics hardware performance prediction with machine learning," Scientific Programming, vol. 2019, pp. 1–15, 2019.

[33] Jan. 2022. [Online]. Available: https://www.researchgate.net/project/KASIMIR-AI-based-assistance-systems-for-safety-critical-product-development-and-measurement-technology-for-commissioning-and-repair

[34] T. B. Sheridan, Humans and Automation: System Design and Research Issues. Human Factors and Ergonomics Society, 2002.

[35] "Levels of Driving Automation," SAE International, Standard, 2019.

[36] M. DeWalt and G. F. McCormick, "Technology independent assurance method," in DASC. IEEE, 2014.

[37] S. Bhattacharyya, D. Cofer, D. J. Musliner, J. Mueller, and E. Engstrom, "Certification Considerations for Adaptive Systems," University of Zurich, Department of Informatics, Tech. Rep. NASA/CR–2015-218702, 2015.

[38] Technology readiness levels (TRL); Extract from Part 19 - Commission Decision C(2014)4995, European Union Std., 2014. [Online]. Available: https://ec.europa.eu/research/participants/data/ref/h2020/wp/2014_2015/annexes/h2020-wp1415-annex-g-trl_en.pdf

[39] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation," PLoS ONE, vol. 10, p. e0130140, 2015.

[40] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas, and J. Wilson, "The What-If Tool: Interactive Probing of Machine Learning Models," IEEE Transactions on Visualization and Computer Graphics, vol. 26, no. 1, pp. 56–65, 2020.

[41] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 1135–1144. [Online]. Available: https://doi.org/10.1145/2939672.2939778

[42] L. Sha, J. Lehoczky, and M. Bodson, "The simplex architecture: Analytic redundancy for software fault tolerance," in Proc. of the First International Workshop of Responsive Control Systems, Nice, France, 1991.

[43] M. Bodson, J. Lehoczky, R. Rajkumar, L. Sha, D. Soh, M. Smith, and J. Stephan, "Control reconfiguration in the presence of software failures," in Proceedings of 32nd IEEE Conference on Decision and Control, 1993, pp. 2284–2289 vol.3.

[44] C. Lazarus, J. G. Lopez, and M. J. Kochenderfer, "Runtime safety assurance using reinforcement learning," in DASC, 2020, pp. 1–9.

[45] T. Bourke, L. Brun, P. Dagand, X. Leroy, M. Pouzet, and L. Rieg, "A formally verified compiler for Lustre," in PLDI. ACM, 2017, pp. 586–601.

[46] D. Beyer, "Automatic Verification of C and Java Programs: SV-COMP 2019," in Tools and Algorithms for the Construction and Analysis of Systems, TACAS 2019, 2019, pp. 133–155.

[47] C. Urban and A. Miné, "A Review of Formal Methods applied to Machine Learning," 2021.

[48] C. Urban and P. Müller, "An Abstract Interpretation Framework for Input Data Usage," in Programming Languages and Systems, A. Ahmed, Ed. Cham: Springer International Publishing, 2018, pp. 683–710.

[49] C. Deng and P. Cousot, "Responsibility Analysis by Abstract Interpretation," in Static Analysis, B.-Y. E. Chang, Ed. Cham: Springer International Publishing, 2019, pp. 368–388.

[50] E. Denney and G. Pai, "Tool support for assurance case development," Automated Software Engineering Journal, 2017.

[51] NASA, "AdvoCATE: Assurance Case Automation Toolset," 2021. [Online]. Available: https://ti.arc.nasa.gov/tech/rse/research/advocate/

[52] M. Wooldridge, An Introduction to MultiAgent Systems, 2nd ed. Wiley, 2011.

[53] D. Heaven, "Why deep-learning ais are so easy to fool," nature, 2019. [Online]. Available: https://www.nature.com/articles/d41586-019-03013-5

[54] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits, "Is bert really robust? a strong baseline for natural language attack on text classification and entailment," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 05, pp. 8018–8025, Apr. 2020. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/6311

[55] G. Branwen, "The neural net tank urban legend," Aug. 2019. [Online]. Available: https://www.gwern.net/Tanks

[56] C. Sitawarin, A. N. Bhagoji, A. Mosenia, M. Chiang, and P. Mittal, "Darts: Deceiving autonomous cars with toxic signs," 2018.

[57] A. Ranjan, J. Janai, A. Geiger, and M. J. Black, "Attacking optical flow," 2019.

[58] S. Bach, A. Binder, G. Montavon, K.-R. Müller, and W. Samek, "Analyzing classifiers: Fisher vectors and deep neural networks," 2015.

[59] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in Advances in Neural Information Processing Systems, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014. [Online]. Available: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

[60] L. Mescheder, A. Geiger, and S. Nowozin, "Which Training Methods for GANs do actually Converge?" in Proceedings of the 35th International Conference on Machine Learning, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 3481–3490. [Online]. Available: http://proceedings.mlr.press/v80/mescheder18a.html

[61] B. Annighöfer, J. Reinhart, M. Brunner, and B. Schulz, "The concept of an autonomic avionics platform and the resulting software engineering challenges," in 2021 International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS). IEEE, May 2021.

[62] Stenerson, "The workstation: Integrating AI into avionics engineering environment," Computer, vol. 19, no. 2, pp. 88–91, 1986.