OPEN ACCESS
IOP Publishing | ISECAEP
Journal of Reliability Science and Engineering
J. Reliab. Sci. Eng. **2** (2026) 022001 (24pp)
https://doi.org/10.1088/3050-2454/ae524c

Topical Review

ChatGPT-like large language models for testing and verification of autonomous intelligent systems: a systematic review

Dun Li, Ruiguan Lin*, Zisheng Wang and Yan-Fu Li*

Department of Industrial Engineering, Tsinghua University, 100084 Beijing, People's Republic of China

E-mail: linruiguan@tsinghua.edu.cn and liyanfu@tsinghua.edu.cn

Received 14 August 2025, revised 28 January 2026
Accepted for publication 15 March 2026
Published 31 March 2026

Abstract

This paper provides a systematic review of how ChatGPT-like large language models (LLMs) contribute to the testing and verification of autonomous intelligent systems (AIS). Building upon recent advances in generative reasoning, this study integrates evidence from 120 peer-reviewed works to examine four key domains: test scenario generation, vulnerability detection, formal verification, and real-time monitoring. Comparative analysis across fuzz testing, symbolic execution, and reinforcement learning highlights how LLMs improve automation, semantic coverage, and adaptability while revealing limitations in benchmark completeness, interpretability, and resource efficiency. The review introduces structured tables summarizing representative datasets, domain-specific applications, and comparative insights between traditional and LLM-based testing approaches. Key challenges-including benchmarking gaps, explainability deficits, and ethical risks-are analyzed alongside emerging research directions such as hybrid verification frameworks and data quality enhancement. This work aims to bridge conceptual and practical gaps between AI safety engineering and large-model reasoning, offering a reference roadmap for integrating LLMs into future AIS verification pipelines.

Keywords: ChatGPT, large language models, autonomous intelligent systems, testing, verification, AI safety

* Authors to whom any correspondence should be addressed.

1. Introduction

Autonomous intelligent systems (AIS) are AI-driven agents capable of perceiving, reasoning, and acting autonomously in dynamic environments, with increasing deployment in domains such as mobility, healthcare, and defense [1–4]. By integrating advanced sensors, machine learning algorithms, and real-time data processing, AIS can tackle complex tasks that exceed the limits of conventional rule-based systems. However, their reliance on probabilistic decision-making and interaction with unpredictable environments introduces significant safety, reliability, and robustness challenges [5]. As AIS become increasingly embedded in safety-critical contexts, the demand for rigorous security testing and formal verification frameworks is more pressing than ever.

Traditional methods-including white-box testing, black-box testing, model checking, and formal verification-have proven effective for identifying flaws in conventional autonomous systems that operate under well-defined, deterministic logic [6]. However, the applicability of these methods becomes substantially limited when extended to AIS, which typically operate in open and dynamic environments and exhibit complex, non-deterministic behaviors driven by heterogeneous sensory inputs, data-driven learning components, and uncertain decision-making processes [7–11]. Figure 1 shows the difference between rule-based autonomous systems and AIS. While the former typically adopt deterministic logic that supports formal verification, AIS systems rely heavily on neural networks, introducing stochasticity and requiring more adaptive and semantically aware testing methodologies.

By combining contextual understanding, reasoning, and text generation, large language models (LLMs) enable a range of AIS testing capabilities, including diverse scenario generation, natural language-based behavior interpretation, automated code verification, and simulation of human-like interactions [12–16]. In addition, many labor-intensive verification tasks can be automated through LLMs, significantly reducing the time and resources required for comprehensive testing. Recent surveys have explored the integration of LLMs into autonomous driving, particularly emphasizing human-like behaviors and decision-making processes [17]. Tian *et al* [18] categorize recent LLM and vision-language model (VLM) applications into LLM-only, VLM-driven, and hybrid designs, and outline key challenges in multimodal alignment, interpretability, and safety, alongside directions for improving robustness and real-world deployment. In contrast, this review adopts a task-oriented perspective and focuses on the use of ChatGPT-like LLMs for core testing and verification tasks in AIS, covering conversational LLMs, agent-based frameworks across fuzz testing, symbolic execution, formal-method-assisted analysis, and benchmark design. The surveyed literature was collected through structured searches on major academic databases, including IEEE Xplore and Web of Science, using task-oriented keywords related to LLMs, testing, and verification of AIS, with studies selected based on their relevance to AIS testing and verification tasks rather than application domains.

Specifically, this review aims to systematically examine the application of ChatGPT-like LLMs in AIS testing and verification. The contributions are as follows:

- A comprehensive review of ChatGPT-like LLMs is provided in the context of AIS testing and verification.
- Key use cases, such as test case generation, automated code review, and real-time fault detection, are analyzed.
- Challenges and limitations in applying LLMs to real-world AIS testing tasks are discussed.
- Future research directions are proposed to further enhance the role of LLMs in AIS verification.

The structure of this review is shown in figure 2. Section 2 introduces the fundamental principles of LLMs and AIS. Section 3 explores their applications in test generation, vulnerability detection, system verification, and adaptive monitoring. Section 4 focuses on benchmarking and evaluation methodologies. Section 5 presents domain-specific applications of ChatGPT-like LLMs for AIS verification. Section 6 discusses open challenges and potential future research directions, followed by a conclusion in section 7.

2. Fundamental principles of ChatGPT-like LLMs and AIS

This section presents the fundamental principles of AIS and LLMs, focusing on the operational mechanisms and the corresponding testing and verification methods. For clarity and consistency, the abbreviations used throughout this review are listed in table 1.

2.1. Introduction to AIS

AIS are systems that make decisions and perform tasks autonomously, relying on complex algorithms, sensor data, and environmental perception [19]. Formally, AIS can be represented as a function $f: S \times A \rightarrow S$, where S is the state space and A is the action space. The system receives an initial state $s_0 \in S$ and selects an action $a_0 \in A$ based on a policy $\pi : S \rightarrow A$, resulting in a new state $s_1 = f(s_0, a_0)$. The process iterates as the system navigates through the environment. Due to the complexity and high-risk nature of AIS applications, testing and verification remain challenging [20]. Traditional methods such as simulation, formal verification, and rule-based analysis focus on ensuring that the policy π meets safety and performance criteria, but often fall short in handling real-world scenarios where the state space S is vast and includes unforeseen or rare conditions.

2.2. Development of LLMs

ChatGPT-like LLMs are built upon the Transformer architecture and trained on large-scale corpora to generate coherent, context-aware text. The core of Transformers is the self-attention mechanism, which models dependencies among tokens in an input sequence $\mathbf{X} = x_1, x_2, \ldots, x_n$

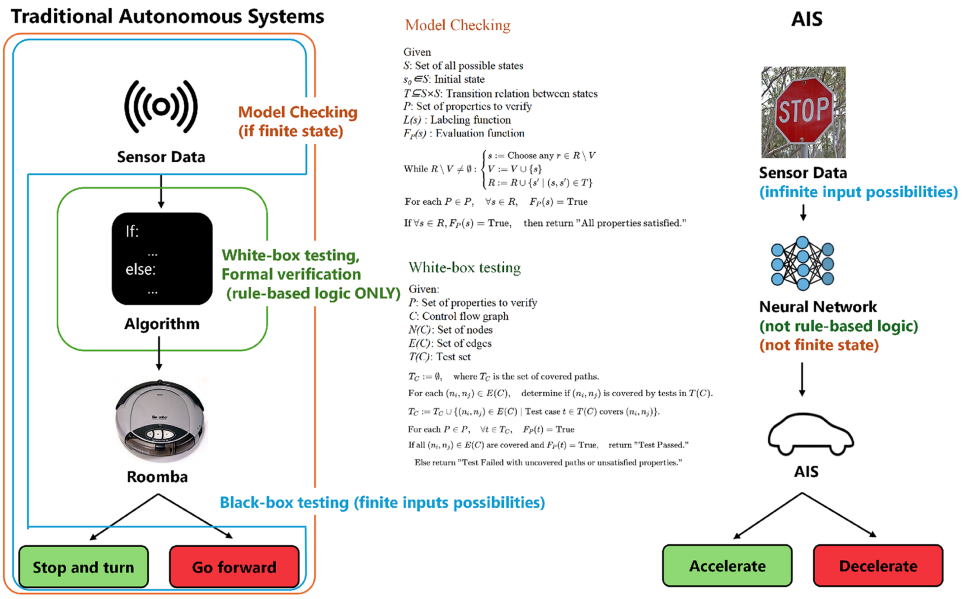


Figure 1. Comparison between traditional autonomous systems and AIS in terms of testing and verification approaches.

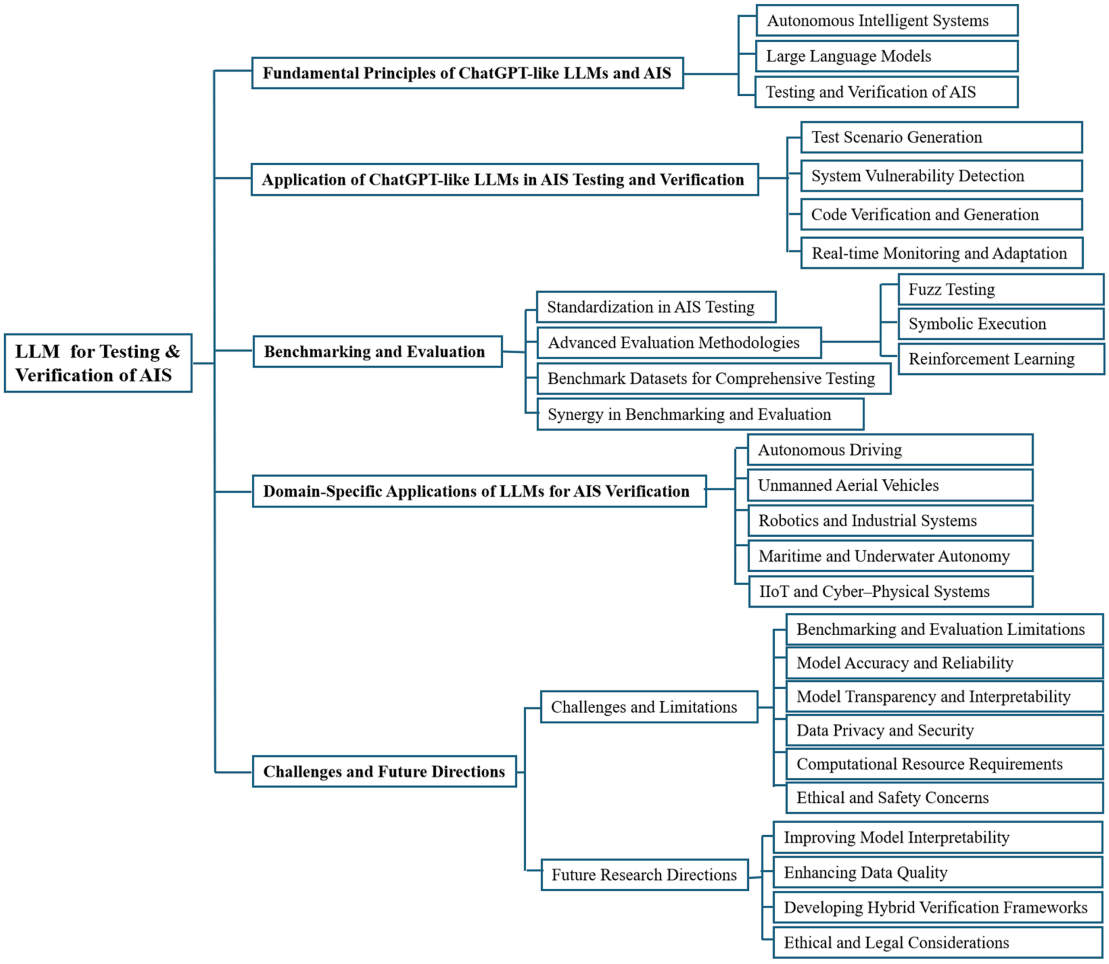


Figure 2. Structure of this review paper, covering seven major sections including Introduction, Principles, Applications, Benchmarking, Challenges, Future Directions, and Conclusion.

Table 1. Abbreviation glossary used in this review.

Abbreviation	Full term
AIS	Autonomous intelligent systems
LLM	Large language model
ChatGPT	Chat generative pre-trained transformer
VLM	Vision-language model
CoT	Chain-of-thought
RAG	Retrieval-augmented generation
RLHF	Reinforcement learning from human feedback
RL	Reinforcement learning
MoE	Mixture of experts
NLP	Natural language processing
CPS	Cyber-physical systems
IIoT	Industrial internet of things
DT	Digital twin
SITL	Software-in-the-loop
HITL	Hardware-in-the-loop
SR	Success rate
FAR	False alarm rate
FPS	Frames per second
BT	Behavior tree
SMC	Satisfiability modulo convex
GNN	Graph neural network
API	Application programming interface
SRL	Semantic role labeling
KG	Knowledge graph
S2R	Simulation-to-reality transfer
QoS	Quality of service

and produces a corresponding output representation $\mathbf{Y} = y_1, y_2, \ldots, y_m$. Through stacked self-attention and feed-forward layers, the model learns to selectively emphasize informative contextual cues, enabling effective long-range semantic reasoning.

LLMs have demonstrated strong performance across major NLP tasks such as text generation, translation, and question answering [21]. They are typically trained by maximizing the likelihood of predicting the next token given its preceding context:

$$\mathcal{L}(\theta) = \sum_{t=1}^{T} \log P(y_t \mid y_{<t}; \theta) \tag{1}$$

where θ denotes the model parameters and T is the sequence length. In the context of AIS testing, LLMs have evolved rapidly across multiple domains, driven by advances in attention optimization, instruction tuning, reinforcement learning from human feedback (RLHF), and retrieval-augmented generation (RAG). These continuous improvements have significantly enhanced contextual understanding, reasoning transparency, and adaptability in complex, high-stakes environments, including telecommunications, autonomous systems, and industrial monitoring. Representative ChatGPT-like models-including GPT-4 [22], Gemini 2.0 [23], Claude 3 [24], the DeepSeek family (e.g. DeepSeek-V2 [25], DeepSeek-V3 [26], DeepSeek-VL2 [27]), the Grok series by xAI (e.g. Grok-1.5 [28]), Mistral 7B/Mixtral 8×22B [29], LLaMA 3 [30], and Qwen 2 [31, 32])-exhibit diverse parameter scales, architectural optimizations, and alignment strategies, yet consistently demonstrate progress in systematic reasoning, safety alignment, and multimodal integration. The core characteristics and technical distinctions among these representative LLMs are summarized in table 2, which highlights the development institutions, model scales, and key innovations in reasoning and safety alignment.

Despite continuing progress, reasoning accuracy and stability remain constrained by data quality, model size, and architectural transparency, motivating ongoing exploration of scalable verifiability frameworks. Comprehensive surveys and empirical studies have analyzed these developments in depth, including Zhou *et al* [33], who provided a systematic review of LLM principles, architectures, and enabling techniques, and Zhang *et al* [34], who examined reasoning performance across arithmetic and logical benchmarks. Li *et al* [35] highlighted how LLMs, through in-context learning, fine-tuning, and RAG mechanisms, enhance interpretability and diagnostic performance in machine monitoring and fault diagnostics. Multimodal and time-series LLM architectures can effectively bridge language-driven reasoning with real-world operational data, offering promising directions for AIS testing and verification.

2.3. Testing and verification of AIS

Testing and verification are critical to ensure that AIS performs reliably and safely in diverse scenarios. Formal verification methods, such as model checking, help validate system correctness [36]. Testing focuses on evaluating the performance of the AIS under typical conditions, edge cases, and potential failure modes. The process can be modeled as an exploration of the state-action space $S \times A$, where the system's response $f(s,a)$ is evaluated for each state $s \in S$ and action $a \in A$ to determine compliance with predefined criteria. The coverage metric quantifies the extent of testing:

$$\text{Coverage} = \frac{|\{(s,a) \in S \times A : \text{tested}(s,a)\}|}{|S \times A|} \tag{2}$$

where $\text{tested}(s,a)$ indicates that the pair (s,a) has been evaluated.

In many existing studies, LLMs are primarily used to support testing and verification activities, for example by assisting with test scenario construction or interpretation of specifications, while the execution of formal verification is typically carried out by deterministic symbolic methods.

Complementarily, verification aims to ensure that the system behavior adheres to specified safety properties. Recent studies emphasize the value of combining formal verification with practical testing approaches. Ruospo *et al* [37] employed line-based testing to validate the correctness of AIS interactions, demonstrating its effectiveness in identifying protocol-level inconsistencies. Ferrando *et al* [38] proposed integrating static formal verification with run-time monitoring to detect and manage assumption violations dynamically, bridging the gap between theoretical soundness and real-world reliability.

Table 2. Representative ChatGPT-like large language models and core characteristics.

Model	Developer/institution	Release year	Parameters (Public/estimated)	Key features/technical highlights	References
GPT-4	OpenAI	2023	Estimated ∼1.8 T (MoE)	RLHF and instruction-tuning; strong reasoning and alignment. *Exact parameter count has not been officially disclosed.*	[22]
Gemini 2.0	Google DeepMind	2024	Not disclosed	Unified multimodal reasoning, extended context, agentic task orchestration.	[23]
Claude 3	Anthropic	2024	Not disclosed	Developed with Constitutional AI; high safety alignment and controllable behaviors.	[24]
DeepSeek family (V2/V3/VL2)	DeepSeek AI	2024	236B (MoE, publicly documented)	Efficient sparse expert routing; strong code and multimodal reasoning performance.	[25–27]
Grok Series (1.5)	xAI	2024	Estimated ∼314B (MoE)	Optimized for reasoning efficiency; conversational alignment for fast interactive inference. *Official model parameters have not been released publicly.*	[28]
Mistral 7B/Mixtral 8 × 22B	Mistral AI	2024	7B/176B (MoE, open-weight)	Sparse MoE architecture enabling efficient scaling and competitive open-weight performance.	[29]
LLaMA 3	Meta AI	2024	8B/70B (open-weight)	Improved tokenizer, multilingual support, extended context, strong instruction following.	[30]
Qwen 2 (Coder/VL)	Alibaba cloud	2024–2025	7B–72B (open-weight)	RAG-enhanced training; strong bilingual and domain-adaptive multimodal reasoning.	[31, 32]

3. Application of ChatGPT-like LLMs in AIS testing and verification

This section explores how LLMs support AIS testing through scenario generation, vulnerability detection, code verification, and real-time monitoring, enhancing the robustness, reliability, and adaptability of AIS in dynamic and unpredictable environments.

3.1. Overview of LLM-driven AIS testing

The traditional AIS testing framework relies on deterministic programs, including rule-based simulation, static code analysis, and predefined test suites [39]. In open-world environments with random behavior, multimodal inputs, and human-machine interaction, these methods face difficulties in scalability and generalization. In contrast, ChatGPT-like LLMs provide a generative and adaptive reasoning layer that understands, analyzes, and interacts with complex system behaviors [40]. By converting unstructured or multimodal data-logs, sensor readings, and operator feedback-into structured test representations, the verification process becomes more automated, semantically aware, and adaptive to diverse AIS scenarios [41].

From a holistic perspective, LLMs can be viewed as intelligent collaborative testers that work alongside traditional verification components to expand test coverage, resolve semantic ambiguity, and analyze anomalous behaviors. This collaboration enables the generation of diverse testing scenarios, supports the identification of system vulnerabilities, and facilitates continuous validation of both code and runtime behaviors [42, 43]. Therefore, LLM-driven AIS testing has evolved from a static, predefined workflow into a continuous and context-aware feedback loop, establishing the conceptual foundation of the LLM4Test paradigm and transforming the testing process into an adaptive, knowledge-driven, and interpretable verification cycle.

As shown in figure 3, the LLM4Test framework incorporates LLMs as intelligent collaborators within the AIS testing loop. System data, specifications, and execution traces are first abstracted into semantic representations, which allow LLMs to participate in multiple stages of testing and verification. Within this framework, LLMs support the construction of realistic simulation and stress-testing scenarios, assist in identifying system vulnerabilities through semantic reasoning over software and behavioral artifacts, contribute to code verification and generation in conjunction with formal analysis, and enable real-time monitoring and adaptation through continuous feedback. The following sections further elaborate on these roles and explain how LLMs contribute to different testing tasks across diverse AIS application contexts.

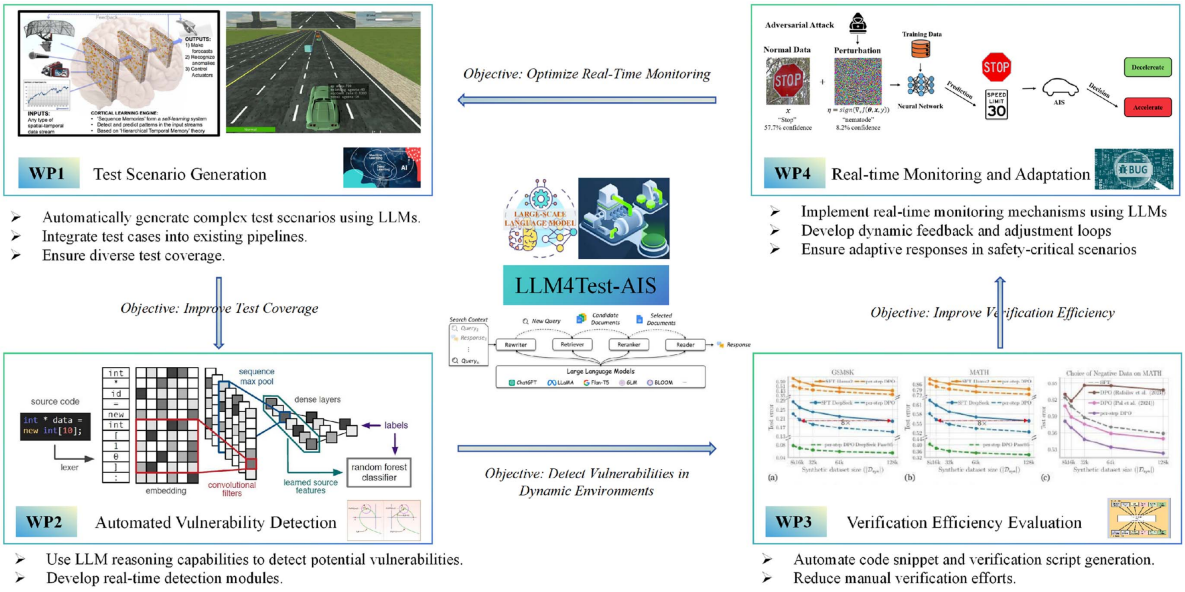


Figure 3. Overview of LLM4Test-AIS: Key areas where large language models contribute to autonomous intelligent systems testing, including scenario generation, vulnerability detection, code verification, and real-time adaptation.

3.2. LLM-enhanced verification techniques

3.2.1. Test scenario generation. One of the key applications of LLMs in AIS testing is the generation of diverse and complex scenarios that capture extreme cases and edge conditions beyond the reach of traditional methods [44, 45]. Let S represent the space of all possible scenarios and let $P(S)$ denote the probability distribution over this space. The goal is to generate a scenario $s \in S$ such that it maximizes the coverage of potential edge cases:

$$s^* = \arg\max_{s \in S} \text{Coverage}(s) \tag{3}$$

where $\text{Coverage}(s)$ is a function that measures how well the generated scenario s explores the extreme conditions and edge cases of the AIS.

3.2.2. System vulnerability detection. LLMs can significantly enhance the detection of potential vulnerabilities in AIS by simulating system behavior under various conditions. By integrating the chain-of-thought (CoT) method, LLMs can perform multi-step reasoning to understand complex system relationships and identify potential risks [46]. Formally, consider the following function: $V : S \times A \to 0, 1$, where $V(s,a) = 1$ indicates the presence of a vulnerability when the system is in state s and takes action a. The objective is to identify states and actions that maximize the likelihood of detecting vulnerabilities. As shown in figure 4, this process leverages neural network-based finite state abstraction combined with statistical model checking (SMC) techniques to classify safe and unsafe states and iteratively refine the verification process.

$$(s^*, a^*) = \arg\max_{s \in S, a \in A} P(V(s,a) = 1). \tag{4}$$

3.2.3. Code verification and generation. In the context of AIS development, LLMs can help developers verify code and generate new code snippets [47–49]. Let C be the space of all possible code segments, and let $E(c)$ be an error detection function that maps a code segment $c \in C$ to a binary result, where $E(c) = 1$ indicates the presence of a logical error or vulnerability in the code. The LLM can predict $E(c)$ by learning a mapping from code to error likelihood, denoted as $\hat{E}(c) = P(E(c) = 1 \mid c; \theta)$, where θ represents the parameters of the LLM. Besides, LLMs can generate new code snippets c' based on the current code context and target specifications, defined by $c' = \arg\max_{c \in C} P(c \mid \text{context}; \theta)$.

3.2.4. Real-time monitoring and adaptation. LLMs can be effectively integrated into real-time AIS monitoring systems, providing dynamic feedback and adaptive strategies crucial for navigating changing environments. By optimizing the monitoring and decision-making processes, LLMs help maintain the robustness and safety of AIS during real-time operations [50]. The monitoring function at time t, denoted as M_t, maps the observed state s_t to a set of actions A_t: $M_t : S \to A$. LLMs dynamically update M_t based on new observations and historical data, minimizing the risk R_t associated with the current state and action: $A_t^* = \arg\min_{A_t \in A} R_t(s_t, A_t)$.

3.3. Multimodal and interactive testing pipelines

Recent advances reveal a shift from task-specific evaluation toward integrated, multimodal verification pipelines where ChatGPT-like LLMs act as reasoning hubs that unify perception, planning, and control. Rather than passively generating outputs, these models increasingly participate in self-assessment and error attribution, redefining verification as a

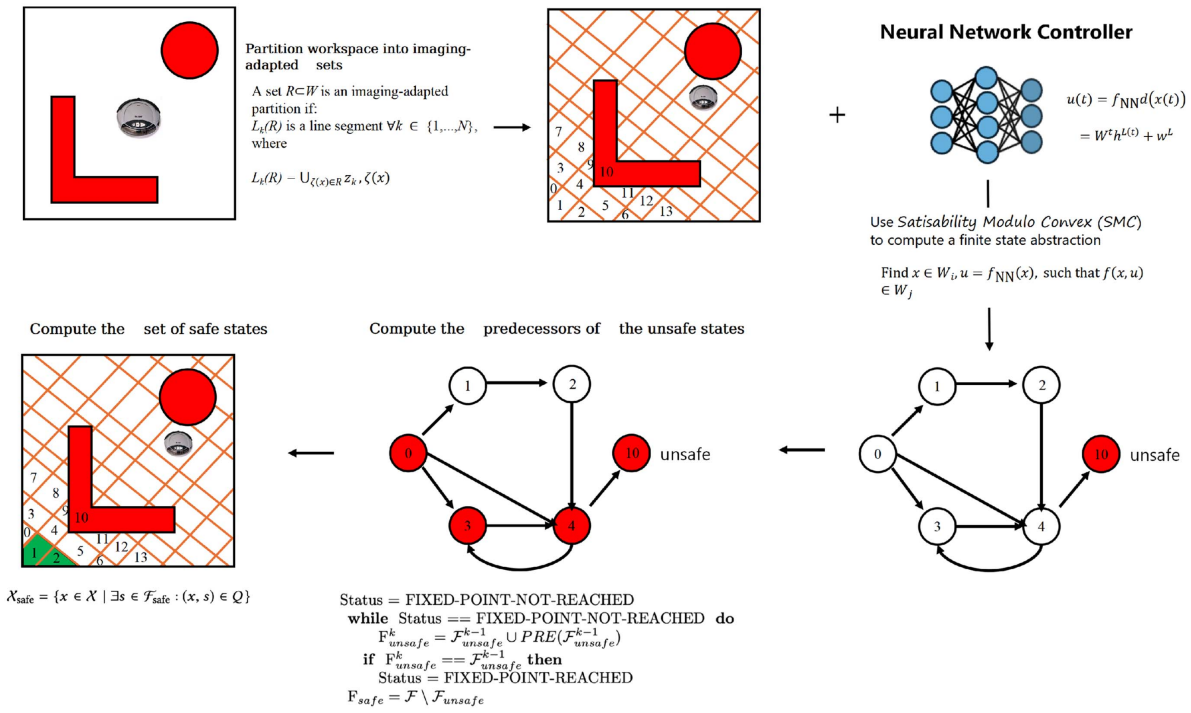


Figure 4. Finite state abstraction using neural networks and SMC. The process shows the identification of safe and unsafe states in the system and the iteration until a fixed point is reached for safe state verification.

continuous dialog between symbolic reasoning and empirical feedback. Cognitive analyses [51] suggest that, unlike human adaptive intelligence, current LLMs lack stable neural-semantic feedback loops, resulting in brittle interpretability and inconsistent verification across modalities. Empirical studies [52] further indicate that progress toward verifiable autonomy arises not from larger models but from tighter coupling among perception, reasoning, and adaptation-mirroring the evolution of human cognition. Within cyber-physical and IoT contexts, LLM agents now function as semantic coordinators that validate system behavior through contextual reasoning rather than static rule compliance [53]. Overall, AIS verification is moving from deterministic checks toward semantics-aware assurance-where the focus shifts from what systems do to how and why they reason, demanding hybrid oversight that blends human judgment with formalized verification logic.

4. Benchmarking and evaluation

This section highlights the critical role of benchmarking in AIS testing and underscores the need for standardized protocols, advanced evaluation methods, and comprehensive datasets to ensure the highest levels of safety and reliability.

4.1. Standardization in AIS testing

Standardization plays a critical role in ensuring the safety, reliability, and regulatory compliance of AIS. Traditional standards-such as ISO 26 262 for automotive functional safety [54], DO-178 C for airborne software certification, and the IEEE 29 119 software testing series [55]-establish rigorous, lifecycle-oriented verification frameworks emphasizing traceability, test coverage, and failure classification. However, the emergence of data-driven autonomy introduces verification challenges that conventional standards only partially address. As recent studies highlight [56, 57], applying ISO 26 262 and DO-178 C to neural or reinforcement learning (RL) components requires redefining verification evidence, uncertainty quantification, and explainability metrics. In this regard, several initiatives-such as the IEEE P7009 standard on fail-safe AI design and the ISO/IEC TR 5469 guidance on AI functional safety-are emerging to extend formal testing methodologies toward machine learning-enabled autonomy. Benchmarking and open testbeds further complement standardization by providing reproducible, measurable baselines for evaluating AIS behavior [58]. Yet, current efforts remain fragmented across sectors, lacking unified performance indicators for reasoning consistency, semantic interpretability, and adaptive safety.

4.2. Advanced evaluation methodologies

Evaluating AIS requires advanced methodologies that address system complexity and dynamic behavior. Traditional testing approaches often fail to capture the full range of operational scenarios, especially in unpredictable or high-risk environments.

4.2.1. Fuzz testing. Fuzz testing exposes failures in AIS by injecting irregular inputs into decision-making and sensor pipelines. While effective at low-level testing, traditional fuzzers struggle with semantically rich inputs, such as mission

Table 3. Comparison of traditional and LLM-based AIS testing methods.

Method	Strengths	Limitations
Traditional fuzzing	High coverage on syntax-level bugs	Poor semantic validity
LLM-assisted fuzzing	Context-aware seed generation	Needs prompt tuning, high variance

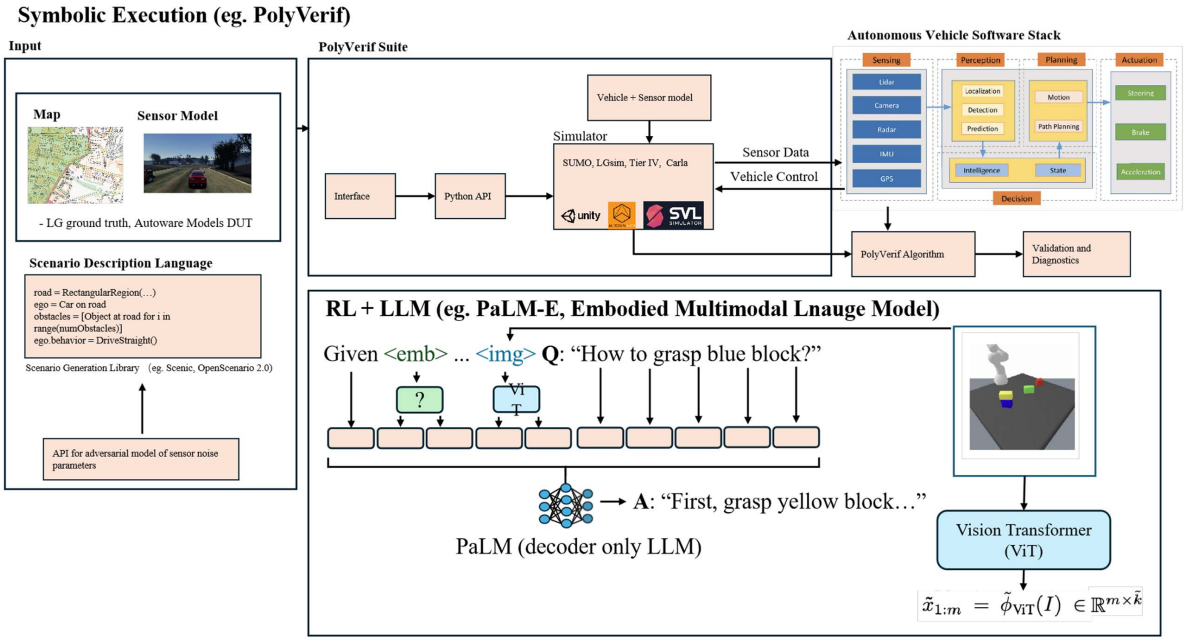


Figure 5. Symbolic execution framework in AIS showing interaction with vehicle models, sensor models, simulation environments (e.g. SUMO, Carla), and the PolyVerif algorithm for vulnerability identification.

scripts or multimodal sensor data. LLM-based fuzzers aim to bridge this gap through context-aware generation. LLM-based approaches enhance fuzzing by generating structured, semantically valid inputs that align with AIS behavior contexts [59]. Recent pipelines combine LLMs with graybox fuzzers to co-generate seeds, vulnerability proofs, and even patches [60]. These methods show the most gains when inputs follow strict grammar or domain-specific constraints, such as ROS messages or unmanned aerial vehicle (UAV) task planners. Compared to traditional fuzzers, LLMs offer better early-stage coverage when used in a feedback loop, especially under semantic constraints [61]. However, prompt-only seeds remain brittle and costly [62], reinforcing the value of hybrid setups with coverage feedback and validity checks. Existing benchmarks often overlook AIS-specific concerns such as safety trigger rates and state inconsistencies, highlighting the urgent need for standardized Software-in-the-loop (SITL)/hardware-in-the-loop (HITL) testbeds to fairly evaluate LLM-driven fuzzing in real-world scenarios. As summarized in table 3, prior work qualitatively indicates that LLM-assisted fuzzing is associated with improved semantic coverage compared to traditional methods, although prompt tuning and reasoning instability remain open challenges.

4.2.2. Symbolic execution. Symbolic execution systematically explores program paths using symbolic inputs, enabling formal detection of edge-case bugs and safety violations as shown in figure 5. Traditional symbolic engines often suffer from path explosion and solver bottlenecks. LLMs offer a complementary path by approximating symbolic reasoning in natural language or code representations, enabling faster generalization across codebases. For instance, LangProp [63] employs LLMs to optimize code performance and correctness through learned transformation patterns, while Liang *et al* [64] introduce dynamic self-correction in symbolic execution using prompt-based diagnostics. However, these methods trade off formal guarantees for scalability and speed, raising concerns about soundness. In autonomous systems, hybrid pipelines like PolyVerif [65] integrate symbolic logic with simulation platforms (e.g. Carla) to uncover policy-level failures. The automation provided by LLMs in symbolic execution is valuable, yet their lack of formal reasoning capabilities makes them more suitable as complements to, rather than replacements for, traditional techniques.

4.2.3. RL. RL facilitates closed-loop testing of AIS by simulating dynamic, interactive environments. Traditional RL-based test generation depends on handcrafted reward functions and large-scale simulation, which can be inflexible or costly in safety-critical scenarios. In parallel, recent studies have explored the use of generative AI models, such as ChatGPT and Gemini, for automated test case generation in software testing, highlighting their potential to improve coverage and reduce manual testing effort, which complements

Table 4. Datasets involving LLMs in AIS testing and verification.

Dataset name	Application scenario	Coverage	Ref	Dataset link
CARLA simulator	Autonomous vehicle testing and verification	Simulated urban environments	[4]	https://carla.org/
KITTI dataset	Computer vision for AVs	Real-world images, LIDAR, GPS	[4]	www.cvlibs.net/datasets/kitti/
Udacity self-driving	AD training and testing	Sensor data and driving scenarios	[4]	https://github.com/udacity/self-driving-car
MNIST dataset	DNN safety verification	Safe region detection in input space	[7]	http://yann.lecun.com/exdb/mnist/
ACAS Xu dataset	UAS safety verification	Verifying DNN models	[7]	https://arxiv.org/abs/1710.00486
highD dataset	Realistic traffic simulation	Vehicle trajectory recordings	[70]	www.highd-dataset.com/
SF110 dataset	Unit test case generation	2% line, 1% branch coverage	[71]	www.evosuite.org/experimental-data/sf110/
Fuzzing DL libraries	DL system fuzzing	TensorFlow API 66% coverage	[71]	https://doi.org/10.1145/3540250.3549085
BIG-bench	LLM evaluation	Diverse tasks	[72]	https://github.com/google/BIG-bench
LogicInference	Logical inference	Sequence-level accuracy	[72]	https://doi.org/10.48550/arXiv.2203.15099
SelfAware	LLM self-awareness	3369 QA samples	[72]	https://arxiv.org/abs/2207.08143
DIALFACT	Fact-checking dialogue	22 245 annotated claims	[72]	https://doi.org/10.48550/arXiv.2211.09110
ToolAlpaca	Tool learning for LLMs	426 tool uses	[73]	https://doi.org/10.48550/arXiv.2306.05301
Traj-LLM	AV trajectory prediction	Complex multi-agent scenes	[74]	https://github.com/chib2024/traj-llm
HumanEval	Code generation assessment	Code correctness checks	[75]	https://github.com/openai/human-eval

RL-based testing pipelines from a testing automation perspective [66]. Recent efforts integrate LLMs into the RL testing loop to enhance sample efficiency, reward design, and policy interpretability. For instance, the adaptive reward optimization framework [67] employs LLMs to adjust reward functions based on real-time feedback, enabling better adaptation to non-stationary behaviors. Similarly, LLMs have been used to synthesize behavior trees, promoting transparent decision structures for explainability and fault analysis [68]. However, LLMs generate behaviors without direct awareness of underlying physical dynamics, which can lead to discrepancies with real-world safety constraints. Although techniques such as retrieval-enhanced prompting enrich the model with domain-specific context, how to balance increased behavioral flexibility with dependable safety assurances remains an open question [69].

4.3. Benchmark datasets for comprehensive testing

To effectively evaluate AIS performance, the benchmark dataset must capture the complexity of the real world. High quality datasets can perform robustness analysis, reveal system vulnerabilities, and ensure comparability by covering various operating conditions, including rare and security critical edge situations. Table 4 summarizes representative datasets that support LLM-assisted AIS testing and verification, covering areas such as autonomous driving, safety verification, fuzz testing, logical reasoning, and code generation. However, many datasets were not originally designed for AIS assessment, which limits their effectiveness in capturing specific domain failure modes. For example, datasets such as CARLA, KITTI, and highD provide rich simulation or perception data but lack annotations for failure modes, safety triggers, or intervention thresholds, which are crucial for safety-critical testing. In contrast, benchmark datasets such as BIG bench and HumanEval provide valuable insights into general inference and coding capabilities, but have shortcomings in evaluating performance in physically grounded or time sensitive AIS scenarios.

4.4. Synergy in benchmarking and evaluation

The integration of standardized protocols, advanced evaluation methodologies, and diverse benchmark datasets forms a robust foundation for the verification of AIS. Within this unified framework, ChatGPT-like LLMs play a crucial role by enabling real-time anomaly detection, adaptive trend analysis, and dynamic response generation under complex operational conditions. Recent benchmarking frameworks increasingly combine quantitative and semantic metrics to assess reliability, interpretability, and safety compliance. Compared with traditional simulation- or rule-based

verification, LLM-augmented evaluation demonstrates superior generalization in unstructured environments and faster adaptation to edge cases. Empirical studies [76–78] indicate that integrating LLM reasoning with physics-based or digital-twin verification yields measurable improvements in behavioral consistency and fault recovery accuracy. As a result, LLM-enhanced benchmarking achieves improved edge-case coverage, higher predictive accuracy, and greater operational reliability across real-world environments. Furthermore, emerging AIS testbeds now incorporate human-in-the-loop evaluation and retrieval-augmented verification pipelines, allowing standardized yet flexible benchmarking of both model reasoning and decision integrity.

5. Domain-specific applications of LLMs for AIS verification

This section reviews how ChatGPT-like LLMs are applied across key autonomous domains to enhance the testing and verification of intelligent systems, including autonomous driving in section 5.1, UAVs in section 5.2, robotics in section 5.3, maritime autonomy in section 5.4, and the industrial internet of things in section 5.5.

5.1. Autonomous driving

Autonomous driving is a safety-critical domain that demands reliable decision-making under complex and uncertain conditions. At the decision layer, DriveGPT4 [79] encodes multimodal sensor inputs into descriptive prompts for direct control generation, while DriveGPT4-V2 [80] advances this approach by introducing a verifiable closed-loop framework that integrates perception, reasoning, and control via multi-view tokenization and expert-supervised decision heads. Complementary to these end-to-end strategies, DriveLLM [81] and DiLu [82] incorporate language-driven reasoning modules that verbalize hazard awareness and maneuver feasibility, offering interpretable checkpoints for semantic verification. However, a critical verification challenge persists: linguistic fluency does not equate to logical correctness, and ungrounded rationales may obscure unsafe behavior. Addressing this concern, SSuperLLM [83] repositions LLMs as runtime auditors-monitoring system behaviors against formal safety constraints-thus decoupling decision execution from compliance assurance and supporting certifiable autonomy.

Recent studies have further extended verification toward cognitive and human-centric dimensions. Fu *et al* [84] incorporated reflection and memory mechanisms into an LLM-based agent to assess self-correction and long-tail generalization, operationalizing cognitive traits such as introspection as measurable testing indicators. Talk2Drive [85] validated an LLM-driven interface through real-world trials, where GPT-4 translated both explicit (‘drive faster’) and implicit (“I am in a hurry’) verbal commands into executable control programs while learning user preferences. Although such personalization reduced human takeover rates by 65.2%, it also raised new verification concerns about bias and safety drift. Similarly, MiningLLM [86] extended the testing paradigm to structured industrial settings by defining prompt-based semantic standards-‘Who am I,’ ‘Where am I,’ and ‘What can I do’to align perception, decision, and control semantics under verifiable constraints. Overall, LLMs are increasingly reshaping AIS verification-no longer merely supporting autonomy but actively driving new verification paradigms. Architecturally, LLM-assisted testing in autonomous driving can be divided into end-to-end and modular frameworks, trading off coverage against interpretability and flexibility, while current benchmarks remain limited in assessing reasoning stability and long-horizon consistency. Table 5 compares key LLM-based frameworks in autonomous driving, highlighting their verification strategies and integration characteristics.

5.2. Unmanned aerial vehicles (UAVs)

UAVs constitute a fast-evolving domain of AIS testing and verification. At the mission and control layer, Ping *et al* [87] demonstrated that multimodal LLMs can translate speech, imagery, and task goals into coordinated swarm actions, while Sezgin [88] used RAG to quantitatively evaluate mission reliability via BLEU and cosine similarity metrics. To enhance adaptability, Tagliabue *et al* [89] proposed REAL, embedding GPT-4 reasoning into UAV control loops for resilience and fault recovery, enabling real-time gain tuning and emergency response. Duvvuru *et al* [90] extended this paradigm with an automated simulation testing system, where multiple GPT-4 agents generate, execute, and analyze PX4 test scenarios-achieving 50% higher coverage and 68% faster test setup than manual baselines. Similarly, Sautenkov *et al* [91] developed UAV-CodeAgents, a multi-agent ReAct-based reasoning framework integrating language and vision modules for mission generation, which reached a 93% success rate (SR) on fire-detection tasks.

At the perception and sensing level, LLMs are being incorporated to improve cross-modal reliability and interpretability. Cai *et al* [92] proposed LLM-Land, combining a BLIP vision encoder with a lightweight LLaMA model to enhance safe landing decisions under dynamic obstacles, achieving a 96% SR with minimal latency (1.4 s). Wu *et al* [93] introduced LPANet, where ChatGPT-guided textual embeddings progressively align semantic and spatial features across RGB-IR modalities, yielding a 4.6% mAP improvement. Similarly, Cai *et al* [94] proposed FlightGPT, which combines RL with vision-language reasoning to support more interpretable UAV navigation. Lin *et al* [95] introduced AirVista, a 3D spatial reasoning model that reports 96% qualitative accuracy in urban air mobility scenarios. Incorporating linguistic reasoning in these systems helps align semantic intent with perception outputs and supports multimodal perception verification. However, evaluations are still largely conducted in synthetic or simulation-based environments, which constrain the assessment of their reliability in real-world deployments. We report three key evaluation metrics: SR for task completion safety,

Table 5. LLM-based AIS in autonomous driving and the verification characteristics.

Model	LLM Role	LLM Model	Input Format	Output/Action	Verification Focus	Dataset	Advantages	Ref	Code
DriveLLM	Decision reasoning engine	GPT-3.5 / GPT-4 (via API)	Structured scene graph (e.g. traffic state, goals, obstacles)	Semantic driving directives (e.g. 'slow down', 'turn left')	Plan consistency, high-level reasoning transparency	Synthetic structured graph-action pairs (internal)	Modular integration; interpretable reasoning steps	[81]	https://github.com/yaodongC/DriveLLM
DriveGPT4	End-to-end driving policy planner	GPT-4	Multimodal sensor data translated into natural language prompts	Continuous control signals (e.g. steering, throttle, brake)	Prompt behavior traceability; output generalization across scenarios	BDD-X with LLM-augmented annotations	Unified architecture; natural language explainability	[79]	https://tonyxuqaq.github.io/projects/DriveGPT4
DriveGPT4-V2	Closed-loop end-to-end autonomous driving with online imitation learning	TinyLLaVA / Qwen-0.5B	Multi-view camera images and vehicle states	Target speed, angle, waypoints, and route points	Robustness and closed-loop safety verification	CARLA Longest6 benchmark (36 routes)	Achieves SOTA performance (DS 70, RC 91, IS 0.77)	[80]	https://drivegpt4-v2.github.io
DiLu	Chain-of-thought decision explainer	GPT-4-1106-preview	Traffic scenario images, route goals	Step-wise decision rationale + maneuver class	Multi-stage logic verification; interpretability of reasoning	nuScenes, LLaVA-generated reasoning sequences	Enhances human-like planning logic; supports step-by-step verification	[82]	https://github.com/PJLab-ADG/DiLu
SSuperLLM	Runtime safety supervisor	GPT-3.5 / GPT-4 (via OpenAI)	System logs, sensor states, behavioral specifications	Constraint satisfaction alerts (e.g. safe/unsafe)	Action traceability, standards compliance, deviation detection	Simple simulation with bicycle model and LQR controller	No control interference; enhances certification readiness	[83]	https://colab.research.google.com/drive/1tRLu1l-bK-zClOGACPyK3Oyzf7mtSxKb
DriveLikeAHuman	Human-like reasoning and reflection framework	GPT-3.5 + LLaMA-Adapter v2	Multimodal driving scenes via perception tools	Textual reasoning trace + control actions	Common-sense reasoning; consistency and memory-based verification	HighwayEnv, real-scene long-tail cases	Human-level interpretability; adaptive long-tail handling	[84]	https://github.com/PJLab-ADG/DriveLikeAHuman
Talk2Drive	Personalized autonomous driving with memory-enhanced LLM reasoning	GPT-4 (via ChatGPT API)	Verbal commands, contextual data, system messages	Executable Language Model Programs (LMPs) for vehicle control	Human-AI trust verification; personalization and safety verification	Real-vehicle field tests (highway, intersection, parking)	Reduces takeover rate by up to 65.2%; interpretable reasoning	[85]	https://github.com/PurdueDigitalTwin/Talk2Drive
MiningLLM	LLM-assisted smart mining and autonomous driving framework	GPT-4/LLaMA/BEV architectures	Multimodal mining data (text-image-sensor) with prompt-based supervision	Scene understanding, annotation, and risk evaluation outputs	Semantic consistency; safety and collaboration verification	BDD-X, NuPrompt, 2 K mining-prompt dataset	Specialized for Mining 5.0; improves zero-shot scene comprehension	[86]	—

false alarm rate (FAR) for robustness against erroneous detections, and frames per second (FPS) for real-time inference efficiency.

In parallel, recent studies have explored benchmarking and digital-twin-based verification to address issues of reproducibility and standardization. Yao *et al* [96] developed AeroVerse, a unified evaluation suite that defines five embodied UAV tasks and introduces GPT-4-based metrics for spatial reasoning and task planning, exposing persistent deficiencies in embodied generalization. Li *et al* [97] established UAVNLT, a benchmark for natural-language-guided tracking, offering a reproducible platform for spatial-semantic consistency testing. Wen *et al* [98] presented BiDGCNLLM, which fuses graph neural networks with LLM-based reasoning over digital-twin-augmented Remote ID data for airspace safety forecasting, achieving 94.3% accuracy with 28% fewer false alarms. Complementing these system-level tests, Khatiri *et al* [99] developed SALIENT to identify safety-critical defects in UAV software repositories, while Liu *et al* [100] released EAI-SIM, an open-source embodied simulation environment that enables reproducible LLM-in-the-loop verification across UAV and robotic platforms. In summary, UAV research is moving toward verification-centered design, where hybrid and digital-twin-based frameworks mitigate semantic ambiguity and physical grounding issues, while exposing trade-offs between end-to-end and modular LLM approaches and limitations of current benchmarks in assessing long-horizon reliability (table 6).

5.3. Robotics and industrial systems

Industrial robotics provides a rigorous testing ground for evaluating how ChatGPT-like LLMs can enhance the verification of autonomous behaviors under real-world constraints, as summarized in table 7. In inspection workflows, Tasneem and Pieters [101] integrated GPT-4 with robotic perception and human feedback to produce traceable inspection justifications and trigger reinspection when ambiguities arise. However, interpretability alone cannot guarantee reliability, as language reasoning must still obey formal safety constraints. Yang *et al* [102] addressed this issue through a 'Safety Chip' that filters LLM-generated commands using temporal-logic rules, blocking unsafe plans before execution. This separation between high-level reasoning and constraint enforcement illustrates how natural-language flexibility can coexist with provable rule compliance.

Beyond local inspection, LLMs are being tested within real-time industrial control and large-scale collaboration. Waseem *et al* [103] showed that GPT-4 can serve as a controller for robot-operated production lines, achieving reinforcement-learning-level throughput while maintaining interpretability and human auditability. Lykov *et al* [104] extended this concept to multi-robot ecosystems, where generative AI coordinates swarms of manipulators, UAVs, and 3D printers in a closed production loop-demonstrating self-validating workflows and scalable coordination under the 'Industry 6.0' paradigm. Rema *et al* [105] further explored task scheduling through LLM-based chatbots, which achieved competitive optimization but revealed instability in reasoning consistency and output reproducibility. Zhang *et al* [106] expanded LLM control to manipulators through a dual-loop design separating semantic planning from execution, enabling latency-aware verification across hardware layers. Rekik *et al* [107] demonstrated an LLM-based orchestrator that fuses speech, gesture, and intention cues for adaptive assembly collaboration, although response delay and prediction drift remain key verification challenges. At a higher cognitive level, Zhang *et al* [108] linked LLM reasoning with digital-twin verification to ensure virtual-physical consistency, while Oyekan *et al* [109] integrated ontologies and knowledge graphs to enhance explainability and ethical alignment in Industry 5.0 automation.

5.4. Maritime and underwater autonomy

Maritime and underwater environments represent some of the most verification-critical domains for AIS, where inherent uncertainties, sensor drift, and communication delays pose significant challenges to conventional control and RL approaches. Recent studies demonstrate that ChatGPT-like LLMs can function as semantic regulators, embedding advanced reasoning capabilities and interpretability into navigation, planning, and decision-making loops. Frameworks such as LLM4SAC [110], OceanChat [111], and OceanPlan [112] illustrate how language-guided policies effectively translate natural-language objectives into verifiable control trajectories and task plans, achieving enhanced convergence and operational transparency. Hybrid architectures-including LLM-guided USV planners [113] and LLM-RL AUV controllers [114]-further extend this paradigm by integrating symbolic reasoning with feedback-driven adaptation, although verification efficacy remains constrained by latency issues, data scarcity, and the absence of standardized safety ontologies.

At a higher abstraction level, verification scope is expanding from isolated control verification toward coordinated, explainable mission reasoning. Multi-agent frameworks such as Command-Agent [115], shared-autonomy [116], and RAG-KG underwater systems [117] integrate LLMs with knowledge graphs, behavior trees, and digital twins to evaluate collective consistency and ensure traceable decision logic. These architectures effectively bridge symbolic verification with human-aligned coordination; however, they predominantly rely on simulation environments and lack robust empirical Verification under real-world marine operational constraints. Complementary initiatives-such as AquaChat++ [118] and Word2Wave [119]-introduce intuitive human-language interfaces that enhance transparency in mission generation and monitoring, yet they also reveal emergent risks associated with linguistic ambiguity, underscoring the necessity for clearer delineation between semantic interpretation and control assurance.

Verification efforts are progressively extending beyond control and coordination to encompass perceptual and cognitive reliability. VLMs like Popeye [120] significantly improve interpretability in ship detection through multimodal semantic

Table 6. Representative LLM-based UAV frameworks and their verification characteristics.

Model	Purpose/verification focus	LLM Model	Input format	Output/action	Key advantages	Ref	Code
LLM-RAG UAV	Scenario-driven mission reliability evaluation using RAG-enhanced LLM	LLaMA 3.2 1B + RAG	Mission logs, environmental data	Mission-specific commands	96.2% accuracy; 120 ms latency	[88]	https://doi.org/10.3390/drones9030213
REAL	Resilient control and adaptive recovery verification	GPT-4	Logs, error codes, objectives	Adaptive mission-level decisions	Self-tuning; safe fallback actions	[89]	—
LLM-agent test	Automated simulation testing and fault-resilience verification	GPT-4	Test scripts, telemetry data	Flight metrics and reports	+52.3% coverage; +68% faster setup	[90]	https://github.com/UAVLab-SLU/AutoSimTestFramework
UAV-CodeAgents	Multi-agent mission planning via ReAct reasoning and vision-language coordination	Qwen2.5-72B/VL-32B	Satellite images, text commands	Coordinated routes and goals	93% success; dynamic adaptability	[91]	—
LLM-Land	Context-aware landing verification with LLM-MPC coupling	LLaMA 3.2 1B + BLIP	RGB, depth, telemetry	Adaptive landing control	96% success; 1.4 s latency	[92]	https://youtu.be/9yGEpqmCtdA
LPANet	Multimodal UAV object detection with LLM-guided feature alignment	ChatGPT + MPNet	RGB-IR images + captions	Aligned object boxes	+4.6% mAP; 36 FPS	[93]	—
FlightGPT	Interpretable navigation reasoning and generalization verification	Qwen2.5-VL-7B	Semantic maps, instructions	CoT-based navigation reasoning	+9.2% SR; open-source	[94]	https://github.com/Pendulumclock/FlightGPT
AirVista	3D spatial reasoning and human-in-the-loop safety verification	LLaVA-1.6 (LoRA)	RGB, depth, point clouds	Spatial task decomposition	96% qualitative accuracy	[95]	—
AeroVerse	Benchmark suite for UAV-agent training and embodied AI evaluation	GPT-4 + multimodal VLMs	Multimodal scene data	Multi-task reasoning outputs	5 UAV tasks; unified metrics	[96]	—
UAVNLT	Natural language-guided UAV tracking benchmark for spatial-semantic grounding	CLIP-based VLM	UAV videos, text prompts	Bounding boxes, relevance scores	Public dataset; baseline model	[97]	https://github.com/Lich-King000/UAVNLT
BiDGCNLLM	Graph-language hybrid model for UAM airspace safety forecasting	GPT-4 + BiDGCN	Remote ID logs, trajectories	Conflict prediction, explanations	94.3% acc; 28% FAR↓	[98]	—
SALIENT	ML-based UAV software safety concern identification tool	FastText	GitHub issues, PRs	Risk-level classifications	$F1 = 0.81$; fault triage	[99]	https://github.com/spanichella/SALIENT-TOOL
EAI-SIM	Open-source simulation for UAV and robotic verification	ChatGPT/PCM	Natural-language commands	Executable UAV control code	Multi-UAV; ROS2 + MAVLink APIs	[100]	https://github.com/PengICS/eai_sim

Table 7. LLM-enabled AIS verification in industrial robotics.

Model	LLM role	LLM model	Input	Output	Verification focus	Scenario/dataset	Ref	Code
Human-robot LLM inspector	Collaborative inspection agent	GPT-4 (OpenAI API)	Vision data, inspection scripts, human feedback	Inspection decisions; NL reports	Failure detection; decision traceability	Industrial visual inspection	[101]	https://github.com/CuriousLad1000/RoboSpection
Safety chip	Command constraint filter	GPT-3.5 / GPT-4	NL task prompts, robot states	Safe action plans via automata	Logic-rule compliance; violation prevention	Simulated household/tablet op	[102]	https://yzylmc.github.io/safety-chip/
LLM controller	Real-time production control	GPT-4 (pretrained)	Structured prompts, line states	Scheduling and control actions	Throughput optimization; interpretability	Serial production line	[103]	—
Closed-loop manipulator	Dual-loop real-time control	GPT-like multimodal model	Visual prompts, sensor feedback	Semantic + feedback control actions	Latency-aware verification; cross-device testing	Franka Emika robot tasks	[106]	—
LLM orchestrated HRC	Multimodal assembly collaboration	GPT-4 + LSTM intent predictor	Speech, gesture, GUI inputs	Context-driven task orchestration	Accuracy, latency, coordination verification	UR10e assembly tasks	[107]	—
Embodied industrial robotics	Framework for embodied intelligence	GPT-style LLMs	Multimodal inputs + DT states	Cognitive reasoning outputs	Cognitive physical consistency; HRC safety	Conceptual study (Industry 6.0)	[108]	https://github.com/jackyzengl/EIIR
Knowledge-augmented LLMs	Decision guidance for HRC	GPT2 + Ontology + KG	Textual manufacturing tasks	Knowledge-grounded reasoning plans	Interpretability scalability trade-off	Textile/electronics benchmarks	[109]	—
Industry 6.0 swarm system	Generative AI-driven multi-robot control	GPT-4o / Claude 3.5	NL design prompts, CAD data	Blueprints, STL files, control scripts	Workflow verification; task coordination	UR arms, UAVs, 3D printers	[104]	—
Chatbot Scheduler	Task scheduling for mobile robots	ChatGPT, Claude, Gemini 2.0	Job-shop text inputs	Task allocation schedules	Consistency and repeatability tests	110 synthetic scheduling cases	[105]	—

grounding, while Pei *et al* [121] demonstrate that only state-of-the-art models such as GPT-4o exhibit sufficient maritime domain knowledge and stable reasoning capabilities. As systematically summarized in table 8, although ChatGPT-like models increasingly facilitate semantics-aware assurance in maritime autonomy, persistent challenges-including non-deterministic behavior, inadequate physical grounding, and the scarcity of reproducible benchmarks-collectively indicate that interpretability alone remains insufficient for achieving certified safety standards.

5.5. Industrial Internet of Things and cyber-physical systems

Industrial Internet and cyber-physical systems (CPS) demand rigorous reliability and explainability, as large-scale interconnectivity and real-time coupling amplify cascading risks. Traditional rule- or graph-based verification pipelines struggle to capture semantic dependencies or adaptive reasoning in evolving topologies. ChatGPT-like LLMs redefine this process by embedding contextual understanding into verification loops, turning compliance checks into interpretable, reasoning-centered evaluation. Hu *et al* [123] proposed AS-LLM, which fuses reasoning models (DeepSeek-R1 and Qwen2.5-7B) with topology-aware rules to infer and validate industrial network relations, improving accuracy and interpretability under limited supervision. Ren *et al* [124] introduced IBRL-LLM, coupling LLM reasoning with RL for Industrial Internet of Things (IIoT) control and defining a 'reasoning fidelity' metric that aligns linguistic goals with verifiable actions. Together, these approaches move industrial verification beyond deterministic auditing toward semantically consistent, self-validating reasoning frameworks, as summarized in table 9.

6. Challenges and future directions

This section highlights AIS-specific challenges beyond general machine learning, outlining the gaps in current LLM-based methods and future directions to improve explainability, verifiability, and trustworthiness, as shown in figure 6.

6.1. Challenges and limitations

6.1.1. Lack of AIS-oriented benchmarks. Existing benchmarking frameworks provide limited insight into how LLMs perform in safety-critical and real-time AIS environments. Most current evaluations focus on linguistic accuracy or reasoning diversity, yet overlook essential dimensions such as simulation-to-reality fidelity, behavioral traceability, and semantic-physical consistency in tasks like motion control, fault diagnosis, or collaborative autonomy. Merten *et al* [77] similarly observed that current evaluations of LLMs for AIS data analysis concentrate on language-to-database translation and zero-shot reasoning, without addressing verifiable behavioral interpretation or maritime operational verification.

6.1.2. Reasoning reliability and context stability. LLMs have strong reasoning abilities but remain prone to context drift and temporal inconsistencies when performing long-horizon verification tasks. Reasoning instability is often associated with hallucination, and recent studies mitigate this issue by grounding LLM outputs with external knowledge, stabilizing inference through consistency-based strategies, or introducing lightweight external validation mechanisms. In AIS, decision-making loops continuously interact with physical system dynamics, and unstable reasoning can gradually accumulate into unsafe behaviors. Wang *et al* [125] and Tian *et al* [78] showed that hybrid LLM-VLM architectures improve trajectory reasoning but still face problems of uncertainty propagation and inference delay.

6.1.3. Transparency and semantic traceability. The black-box nature of LLMs impedes semantic traceability in AIS verification. Without interpretable reasoning, it is difficult to determine whether a model's decision path aligns with formal safety constraints or physical causality. Sobrín *et al* [126] and Buchmann *et al* [127] emphasize the need to embed deterministic reasoning layers and structured memory to enable post-hoc verification.

6.1.4. Data fidelity and domain grounding. The reliability of LLM-based AIS verification critically depends on the fidelity and grounding of multimodal data. General-purpose LLMs often lack alignment with real-world sensory, operational, or mission data, resulting in hallucinated or physically infeasible test scenarios. Zaki *et al* [128] and Lyons *et al* [129] observed that biased or incomplete datasets can propagate systemic errors throughout the verification process. Recent approaches emphasize domain-calibrated datasets, retrieval-augmented verification [130], and synthetic-to-real consistency protocols that constrain data generation within verifiable operational contexts.

6.1.5. Hybrid verification bottlenecks. Hybrid verification frameworks combining simulation, formal proof, and LLM reasoning face persistent scalability and coherence challenges. Tao *et al* [131] and Popescu [132] note that existing methods often validate reasoning outputs in isolation, lacking closed-loop consistency across perception-decision-action cycles. For instance, in autonomous driving, LLM-generated test scenarios may mismatch real-time vehicle dynamics in Carla simulations, creating verification gaps between semantic reasoning and physical execution. Similarly, industrial robotics frameworks combining GPT-4 planning with formal safety checkers encounter latency mismatches that disrupt real-time control. Emerging approaches address these issues by coupling symbolic safety models with probabilistic reasoning validators and HITL feedback, as demonstrated by Zheng *et al* [133] in cyber-physical systems, where LLM-generated scenarios are validated through model checking and simulation to align natural-language reasoning with physically grounded behaviors.

Table 8. Representative LLM-based frameworks for verification in maritime and underwater autonomy.

Model	LLM role	LLM model	Input	Output/action	Verification focus	Dataset/platform	Ref	Code
LLM4SAC	Semantic planner for RL docking	GPT-3.5/4	Visual states, goals	Docking trajectory, controls	Sim2Real reliability; semantic control	VRX sim + lake trials	[110]	https://github.com/RyanXu0428/LLM4SAC
Command-agent	Multi-agent command reasoning	DeepSeek-R1 + Qwen2.5-7B	Tactical logs, context	Plans, coord. actions, reports	Mission-level reasoning; reliability	Digital-twin warfare sim	[115]	—
OceanChat	Closed-loop task-motion planner	GPT-4	NL commands, states	Task seq., motion plans, replans	Closed-loop reliability; grounding	HoloEco + EcoMapper AUV	[111]	https://sites.google.com/view/oceanchat
LLM-guided mission planner	Symbolic USV planning w/ feedback	GPT-4	Goals, maps, controller fb	Symbolic plans, re-plans	Closed-loop + trace verification	MBZIRC Maritime (ROS2)	[113]	https://github.com/Muhayyuddin/llm-guided-mission-planning
Shared autonomy (Hull)	LLM+BT+DRL shared autonomy	LLaMA + DRL + BT	NL goals, fleet status	Goal parsing, BT exec., formation	Interpretability; HITL verification	Zeno AUV (lake) + ROS	[116]	—
RAG-KG underwater	Multi-AUV semantic verification	GPT-3.5 + RAG + KG	Mission scripts, taxonomy	Verified BTs, coordination	Semantic grounding; BT completeness	Stonefish (multi-AUV)	[117]	https://michele1996.github.io/rag_full_shared_autonomy.github.io/

(Continued.)

Table 8. (Continued.)

AquaChat++	Multi-ROV energy/fault-aware plans	GPT-4	NL tasks, battery, thrusters	Symbolic plans, adaptive control	Closed-loop; energy/fault tolerance	ROS-Gazebo + BlueROV2	[118]	—
OceanPlan	Hierarchical plan & replan	GPT-4/4 V	NL cmds, RGB, marine XML	Symbolic tasks, control policies	Uncertainty-robust closed-loop	HoloEco + EcoMapper	[112]	https://sites.google.com/view/oceanplan
LLM-PPAE AUV	3D path planning under currents	LLaMA	State vectors, ROMS flow	Action probs, RL control	Robustness; convergence; S2R	ROMS SCS simulator	[114]	—
Vision-LLM Pilot	Zero-shot VLM navigation	GPT-4o	Camera, env. data, goals	Nav commands, JSON plans	Reliability (zero-shot); latency	ROS2-Gazebo (WAM-V)	[122]	https://github.com/IOES-Lab/AI-Control-Room-VLLM
LLMs_Nav	Knowledge eval. for MASSs	GPT-4o/3.5, ERNIE, Qwen	OOW MCQs, prompts	Accuracy, latency, adherence	Knowledge-level verification	1500+ STCW MCQs (14 LLMs)	[121]	https://github.com/PeiDashuai/LLMs_Nav
Popeye	VLM for multisource ship detect.	CLIP-VLM + ViT	Optical/SAR/IR + text	Boxes, scores, text reports	Cross-modal consistency; perception	SSDD, HRSID, SAR-Ship	[120]	—
Word2Wave	NL mission programming for AUVs	T5-Small (+GPT data gen)	Speech/text mission cmds	Waypoints, mission maps	NL-mission consistency; usability	NemoSens trials + corpus	[119]	—

Table 9. Representative LLM-based frameworks for verification in IIoT and cyber-physical systems.

Model	LLM role	LLM model	Input format	Output/action	Verification focus	Dataset/platform	Ref	Code
AS-LLM	Relationship inference and reasoning verification for industrial networks	DeepSeek-R1 + Qwen2.5-7B (QLoRA fine-tuning)	BGP paths, AS metadata, and topology rules	Predicted AS relations with rationales	Reasoning interpretability; rule-grounded verification (Valley-Free)	RouteViews, RIPE, AS-Rank datasets	[123]	—
IBRL-LLM	Semantic reasoning and verification-guided reinforcement learning for IIoT	GPT-4 (semantic planner) + PPO/DDPG	Sensor data, task descriptions, and energy parameters	Optimized task scheduling and control actions	Reasoning-action consistency; energy and stability verification	IIoT testbed (industrial communication simulator)	[124]	—

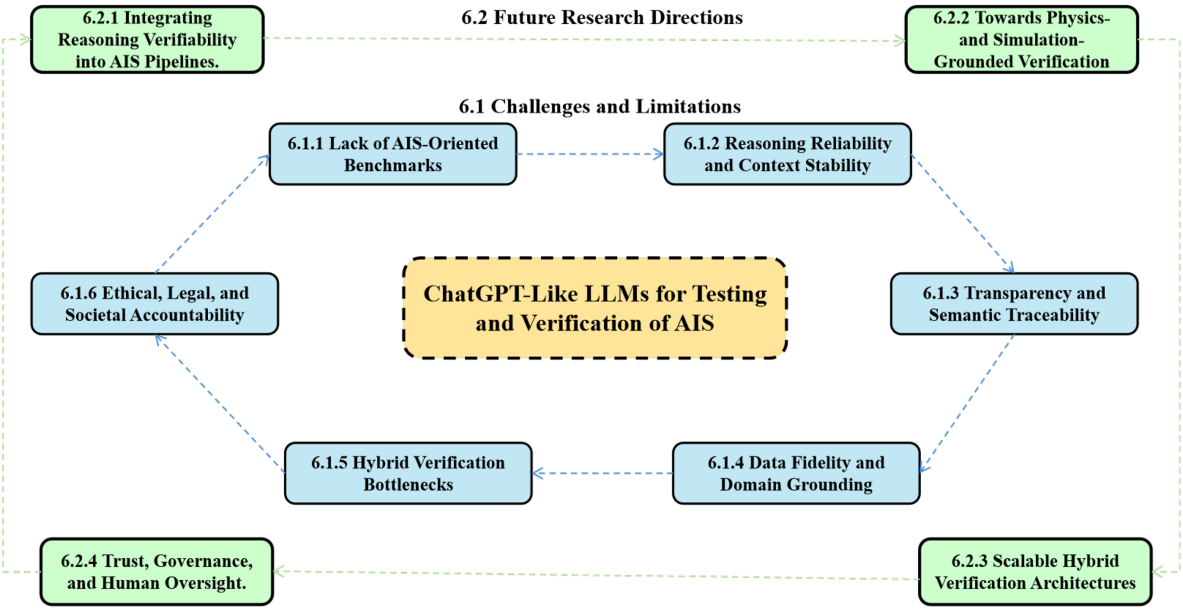


Figure 6. Challenges and future research directions for ChatGPT-like LLMs in testing and verification of autonomous intelligent systems.

6.1.6. Ethical, legal, and societal accountability. As LLMs assume quasi-decision-making roles in AIS testing, responsibility for erroneous or unsafe verification outputs becomes increasingly ambiguous. Hallucinated test cases or biased reasoning may lead to unsafe certifications. Kong *et al* [134] and Wu *et al* [135] warn that misalignment between model reasoning and human values can cause ethical and legal risks. Viewing LLM inference as probabilistic decision making can support traceable liability frameworks, while embedding legal reasoning into verification can ensure accountability and human oversight in AIS certification processes [136]. Ethical and legal risks in LLM-assisted testing vary across application domains. In safety-critical AIS, such as autonomous driving and industrial robotics, accountability mainly involves safety assurance and fault attribution, where opaque LLM reasoning can complicate post-incident analysis [137]. By contrast, human-facing AIS applications, including service and assistive robots, raise additional concerns regarding trust, privacy, and behavioral influence, highlighting the need for domain-aware testing and verification rather than a one-size-fits-all accountability framework [138].

6.2. Future research directions

6.2.1. Integrating reasoning verifiability into AIS pipelines. Reasoning verifiability should become a central objective in AIS testing. Each LLM-generated decision ought to be accompanied by semantic tags, quantified uncertainty, and causal explanations to support traceable evaluation. Developing reasoning fidelity metrics that measure the alignment between LLM inferences and real system behaviors

can promote standardized auditing mechanisms for intelligent verification. Future studies should also advance interpretable attention mechanisms, model distillation, and natural-language rationales to transform opaque reasoning into auditable and certifiable evidence [139].

6.2.2. Towards physics- and simulation-grounded verification. In AIS testing, ChatGPT-like LLMs must be grounded in simulation and digital-twin environments to ensure that generated scenarios and reasoning align with physical constraints and executable behavior. Coupling LLM-based test generation with physics-aware simulators and constraint checkers can transform abstract reasoning into measurable verification outcomes [140]. Persistent issues include unrealistic scenario generation and mismatches between the simulator-reality, which can be mitigated through schema-constrained prompting, temporal-logic guards, and calibration with real-world data [141].

6.2.3. Scalable hybrid verification architectures. Future AIS verification frameworks should combine formal verification, RL, and LLM-based reasoning into unified hybrid architectures. The goal is to establish bidirectional consistency-LLMs generating test hypotheses, symbolic engines proving constraints, and simulators validating outcomes-to form a closed-loop verification cycle that enhances interpretability and robustness. Zheng *et al* [133] applied this idea to cyber-physical systems, where LLMs generate domain-specific test scenarios validated through model checking and simulation. Jha *et al* [142] further showed that integrating LLMs into unified design-testing pipelines enhances interpretability and reduces manual intervention. Lu *et al* [143] proposed OmniTester, a multimodal LLM-driven scenario testing framework for autonomous vehicles, in which language-vision models generate semantically rich driving scenarios evaluated within simulation environments, improving scenario diversity and stress testing. Similarly, Zhou *et al* [144] developed a hierarchical test platform for VLM-integrated autonomous driving that combines vision-language reasoning with simulation-based validation across system layers, supporting scalable and practical verification workflows in safety-critical AIS.

6.2.4. Trust, governance, and human oversight. Finally, as LLMs gain autonomy in AIS verification, governance mechanisms must ensure transparent and human-aligned accountability. LLM-based verifiers should remain under human-in-the-loop supervision, enabling auditors to trace reasoning paths, override unsafe decisions, and retrain models when inconsistencies occur. Esposito *et al* [145] stressed that mission-critical governance demands collaboration among researchers, practitioners, and policymakers to balance innovation with regulatory compliance. Establishing unified governance frameworks that embed these principles will be essential for deploying ChatGPT-like systems in safety-critical verification environments.

7. Conclusion

This review provides a comprehensive synthesis of how ChatGPT-like LLMs enhance the testing and verification of AIS. Through comparative analysis across test scenario generation, vulnerability detection, formal verification, and real-time monitoring, it outlines the theoretical foundations, practical progress, and key integration pathways of LLM-driven testing frameworks. The discussion identifies four major dimensions-semantic generation, symbolic guidance, hybrid decision loops, and benchmark-oriented evaluation-that together define the current trajectory of intelligent and adaptive verification. While the incorporation of LLMs enables broader automation, improved test coverage, and enhanced interpretability compared to conventional methods, limitations persist in the completeness of benchmarks, the transparency of model behaviors, the scalability of computation, and the assurance of ethical reliability. Future progress depends on refining explainable reasoning mechanisms, improving data quality, establishing unified hybrid verification frameworks, and reinforcing ethical and legal governance. By consolidating recent advances and open challenges, this review offers a coherent foundation and forward-looking perspective for developing trustworthy and verifiable AIS. Moreover, as industrial sectors increasingly adopt LLM-assisted verification workflows, shared benchmark suites and standardized evaluation protocols will play a central role in ensuring reliability across heterogeneous deployment environments.

Data availability statement

No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflict of interest

The authors declare no conflict of interest.

ORCID iDs

Dun Li 0000-0002-1986-7144
Ruiguan Lin 0000-0003-2035-3770
Zisheng Wang 0000-0003-1722-2454
Yan-Fu Li 0000-0001-5755-7115

References

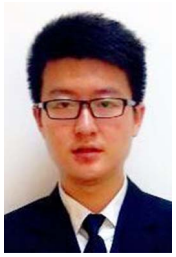
[1] Chen J, Sun J and Wang G 2022 From unmanned systems to autonomous intelligent systems *Engineering* **12** 16–19
[2] Leikas J, Koivisto R and Gotcheva N 2019 Ethical framework for designing autonomous intelligent systems *J. Open Innov.* **5** 18
[3] Long L N, Hanford S D, Janrathitikarn O, Sinsley G L and Miller J A 2007 A review of intelligent systems software for autonomous vehicles *2007 IEEE Symp. on*

Computational Intelligence in Security and Defense Applications (IEEE) pp 69–76
[4] Chen Y, Chen S, Zhang T, Zhang S and Zheng N 2018 Autonomous vehicle testing and validation platform: Integrated simulation system with hardware in the loop *2018 IEEE Intelligent Vehicles Symp. (IV)* (IEEE) pp 949–56
[5] Illiashenko O, Kharchenko V, Babeshko I, Fesenko H and Di Giandomenico F 2023 Security-informed safety analysis of autonomous transport systems considering AI-powered cyberattacks and protection *Entropy* **25** 1123
[6] Aghababaeyan Z, Abdellatif M, Briand L, Ramesh S and Bagherzadeh M 2023 Black-box testing of deep neural networks through test case diversity *IEEE Trans. Softw. Eng.* **49** 3182–204
[7] Păsăreanu C S, Gopinath D and Yu H 2019 Compositional verification for autonomous systems with deep learning components: white paper *Safe, Autonomous and Intelligent Vehicles* pp 187–97
[8] Wang H, Li Y-F and Ren J 2024 Machine learning for fault diagnosis of high-speed train traction systems: a review *Front. Eng. Manage.* **11** 62–78
[9] Dennis L A, Fisher M, Lincoln N K, Lisitsa A and Veres S M 2016 Practical verification of decision-making in agent-based autonomous systems *Autom. Softw. Eng.* **23** 305–59
[10] Zheng D, Fu X, Liu X, Xing L and Peng R 2024 Modeling and analysis of cascading failures in industrial internet of things considering sensing-control flow and service community *IEEE Trans. Reliab.* **74** 2723–37
[11] Li D, Wang H and Li Y-F 2024 Robust anomaly detection in unmanned ship systems based on large language models *ESREL 2024 Collection of Extended Abstracts Part 1* p 47
[12] Liu Y *et al* 2023 Summary of ChatGPT-related research and perspective towards the future of large language models *Meta-Radiology* **1** 100017
[13] Collins K M, Wong C, Feng J, Wei M and Tenenbaum J B 2022 Structured, flexible, and robust: benchmarking and improving large language models towards more human-like behavior in out-of-distribution reasoning tasks (arXiv:2205.05718)
[14] Xie C and Zou D 2024 A human-like reasoning framework for multi-phases planning task with large language models (arXiv:2405.18208)
[15] Hagendorff T, Fabi S and Kosinski M 2023 Human-like intuitive behavior and reasoning biases emerged in large language models but disappeared in ChatGPT *Nat. Comput. Sci.* **3** 833–8
[16] Li D, Wang H, Lin R, Li Y, Ye J, Miao D, Lin K and Zhang H 2024 Pretrained large models in telecommunications: a survey of technologies and applications *2024 IEEE Int. Conf. on Progress in Informatics and Computing (PIC)* (IEEE) pp 117–21
[17] Li Y, Katsumata K, Javanmardi E and Tsukada M 2024 Large language models for human-like autonomous driving: a survey *2024 IEEE 27th Int. Conf. on Intelligent Transportation Systems (ITSC)* (IEEE) pp 439–46
[18] Tian H, Reddy K, Feng Y, Quddus M, Demiris Y and Angeloudis P 2024 Large (vision) language models for autonomous vehicles: current trends and future directions *IEEE Trans. Intell. Transp. Syst.* **27** 187–210
[19] Chatila R and Havens J C 2019 The IEEE global initiative on ethics of autonomous and intelligent systems *Robotics and Well-Being* pp 11–6
[20] Acharya D B, Kuppan K and Divya B 2025 Agentic AI: autonomous intelligence for complex goals–a comprehensive survey *IEEE Access* **13** 18912–36
[21] Pandey A K and Roy S S 2024 Extractive question answering over ancient scriptures texts using generative AI and natural language processing techniques *IEEE Access* **12** 101197–209
[22] Achiam J *et al* 2023 Gpt-4 technical report (arXiv:2303.08774)
[23] Google DeepMind Team 2024 Introducing Gemini 2.0: the next generation of multimodal AI models (available at: https://blog.google/technology/google-deepmind/google-gemini-ai-update-december-2024/) (Accessed 23 October 2025)
[24] Anthropic 2024 https://www.anthropic.com/news/claude-3-family (Accessed 19 March 2026) Introducing the next generation of Claude
[25] Liu A *et al* 2024 Deepseek-v2: a strong, economical, and efficient mixture-of-experts language model (arXiv:2405.04434)
[26] Liu A *et al* 2024 Deepseek-v3 technical report (arXiv:2412.19437)
[27] Wu Z *et al* 2024 Deepseek-vl2: mixture-of-experts vision-language models for advanced multimodal understanding (arXiv:2412.10302)
[28] xAI Team 2024 Grok 1.5: advancing reasoning and efficiency in the xAI model family (available at: https://x.ai/news/grok-1.5) (Accessed 23 October 2025)
[29] Jiang A Q *et al* 2024 Mixtral of experts (arXiv:2401.04088)
[30] Meta AI Research Team 2024 The llama 3 herd of models (available at: https://ai.meta.com/research/publications/the-llama-3-herd-of-models/) (Accessed 23 October 2025)
[31] Hui B *et al* 2024 Qwen2. 5-coder technical report (arXiv:2409.12186)
[32] Bai S *et al* 2025 Qwen2. 5-vl technical report (arXiv:2502.13923)
[33] Zhou H *et al* 2024 Large language model (llm) for telecommunications: a comprehensive survey on principles, key techniques and opportunities *IEEE Commun. Surv. Tutor.* **27** 1955–2005
[34] Zhang H *et al* 2024 A careful examination of large language model performance on grade school arithmetic *Advances in Neural Information Processing Systems* vol 37 pp 46819–36
[35] Li Y-F, Wang H and Sun M 2024 ChatGPT-like large-scale foundation models for prognostics and health management: a survey and roadmaps *Reliab. Eng. Syst. Saf.* **243** 109850
[36] Clarke E, Garlan D, Krogh B, Simmons R and Wing J 2001 Formal verification of autonomous systems NASA intelligent systems program
[37] Ruospo A, Cantoro R, Sanchez E, Schiavone P D, Garofalo A and Benini L 2019 On-line testing for autonomous systems driven by risc-v processor design verification *2019 IEEE Int. Symp. on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)* (IEEE) pp 1–6
[38] Ferrando A, Dennis L A, Ancona D, Fisher M and Mascardi V 2018 Verifying and validating autonomous systems: towards an integrated approach *Runtime Verification: 18th Int. Conf., RV 2018, (Limassol, Cyprus, 10 November–13 November 2018)* (*Proc. 18*) (Springer) pp 263–81
[39] Wang X, Guo Y and Gao Y 2024 Unmanned autonomous intelligent system in 6g non-terrestrial network *Information* **15** 38
[40] Sandhya Devi R S and Varshni S D 2025 Embedded large language models for enhanced human-machine interface in autonomous vehicles *2025 Int. Conf. on Multi-Agent Systems for Collaborative Intelligence (ICMSCI)* (IEEE) pp 1143–50
[41] Thapa S and Adhikari S 2024 Leveraging ChatGPT-like large language models for Alzheimer's disease: enhancing care,

advancing research and overcoming challenges *Smart Healthcare Systems* (CRC Press) pp 265–75
[42] Yang Y, Zhang Q, Li Ci, Simões Marta D, Batool N and Folkesson J 2024 Human-centric autonomous systems with LLMs for user command reasoning *Proc. IEEE/CVF Winter Conf. on Applications of Computer Vision* pp 988–94
[43] Mahmud D, Hajmohamed H, Almentheri S, Alqaydi S, Aldhaheri L, Khalil R A and Saeed N 2025 Integrating LLMs with its: recent advances, potentials, challenges and future directions *IEEE Trans. Intell. Transp. Syst.* **26** 5674–709
[44] Chang C, Wang S, Zhang J, Ge J and Li Li 2024 LLMScenario: large language model driven scenario generation *IEEE Trans. Syst. Man Cybern.* **54** 6581–94
[45] Tang S, Zhang Z, Zhou J, Lei L, Zhou Y and Xue Y 2024 Legend: a top-down approach to scenario generation of autonomous driving systems assisted by large language models *Proc. 39th IEEE/ACM Int. Conf. on Automated Software Engineering* pp 1497–508
[46] González-Santamarta M A, Rodríguez-Lera F J, Manuel Guerrero-Higueras Angel and Matellán-Olivera V 2023 Integration of large language models within cognitive architectures for autonomous robots (arXiv:2309.14945)
[47] Xu F F, Alon U, Neubig G and Hellendoorn V J 2022 A systematic evaluation of large language models of code *Proc. 6th ACM SIGPLAN Int. Symp. on Machine Programming* pp 1–10
[48] Ma Y *et al* 2024 Lampilot: an open benchmark dataset for autonomous driving with language model programs *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 15141–51
[49] Nouri A, Cabrero-Daniel B, Fei Z, Ronanki K, Sivencrona H and Berger C 2025 Large language models in code co-generation for safe autonomous vehicles (arXiv:2505.19658)
[50] Luckcuck M 2023 Using formal methods for autonomous systems: five recipes for formal verification *Proc. Inst. Mech. Eng. O* **237** 278–92
[51] Grossberg S 2025 Neural network models of autonomous adaptive intelligence and artificial general intelligence: how our brains learn large language models and their meanings *Front. Syst. Neurosci.* **19** 1630151
[52] Lin Y, Wang X, Yang J and Wang S 2024 Core technology topic identification and evolution analysis based on patent text mining-a case study of unmanned ship *Appl. Sci.* **14** 4661
[53] Bhat A, Mondal A and Tripathy A 2025 LLM agents for Internet of Things (IoT) applications
[54] Palin R, Ward D, Habli I and Rivett R 2011 Iso 26262 safety cases: compliance and assurance *6th IET Int. Conf. on System Safety 2011* (IET) p B12
[55] Alaqail H and Ahmed S 2018 Overview of software testing standard ISO/IEC/IEEE 29119 *Int. J. Comput. Sci. Netw. Secur.* **18** 112–6
[56] Gosavi M A, Rhoades B B and Conrad J M 2018 Application of functional safety in autonomous vehicles using iso 26262 standard: a survey *SoutheastCon 2018* (IEEE) pp 1–6
[57] Song Q, Engström E and Runeson P 2021 Concepts in testing of autonomous systems: Academic literature and industry practice *2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN)* (IEEE) pp 74–81
[58] Madhavan R, Lakaemper R and Kalmár-Nagy T 2009 Benchmarking and standardization of intelligent robotic systems *2009 Int. Conf. on Advanced Robotics* (IEEE) pp 1–7
[59] Chen J and Lu S 2024 An advanced driving agent with the multimodal large language model for autonomous vehicles *2024 IEEE Int. Conf. on Mobility, Operations, Services and Technologies (MOST)* (IEEE) pp 1–11
[60] Sheng Z, Xu Q, Huang J, Woodcock M, Huang H, Donaldson A F, Gu G and Huang J 2025 All you need is a fuzzing brain: an LLM-powered system for automated vulnerability detection and patching (arXiv:2509.07225)
[61] Cheng Y, Kang H J, Shar L K, Dong C, Shi Z, Lv S and Sun L 2025 Towards reliable LLM-driven fuzz testing: vision and road ahead (arXiv:2503.00795)
[62] Black G, Vaidyan V and Comert G 2024 Evaluating large language models for enhanced fuzzing: an analysis framework for LLM-driven seed generation *IEEE Access* **12** 156065–81
[63] Ishida S, Corrado G, Fedoseev G, Yeo H, Russell L, Shotton J, Henriques J F and Hu A 2024 Langprop: a code optimization framework using large language models applied to driving *ICLR 2024 Workshop on Large Language Model (LLM) Agents*
[64] Liang X, Song S, Zheng Z, Wang H, Yu Q, Li X, Li R-H, Xiong F and Li Z 2024 Internal consistency and self-feedback in large language models: a survey (arXiv:2407.14507)
[65] Razdan R, İlhan Akbaş M, Sell R, Bellone M, Menase M and Malayjerdi M 2023 Polyverif: an open-source environment for autonomous vehicle validation and verification research acceleration *IEEE Access* **11** 28343–54
[66] Bandi A, Nukala H S T, Tatavarthi B and Boggavarapu A 2025 Automated test case generation for software testing using generative AI *Int. Conf. on Computers and Their Applications* (Springer) pp 78–87
[67] Chen Y, Ye Y, Chen Z, Zhang C and Ang M H 2024 Aro: large language model supervised robotics text2skill autonomous learning (arXiv:2403.15834)
[68] Li F, Wang X, Li B, Wu Y, Wang Y and Yi X 2024 A study on training and developing large language models for behavior tree generation (arXiv:2401.08089)
[69] Mitra C, Miroyan M, Jain R, Kumud V, Ranade G and Norouzi N 2024 Retllm-e: retrieval-prompt strategy for question-answering on student discussion forums *Proc. AAAI Conf. on Artificial Intelligence* vol 38 pp 23215–23
[70] Tian H, Reddy K, Feng Y, Quddus M, Demiris Y and Angeloudis P 2024 Enhancing autonomous vehicle training with language model integration and critical scenario generation (arXiv:2404.08570)
[71] Wang J, Huang Y, Chen C, Liu Z, Wang S and Wang Q 2024 Software testing with large language models: survey, landscape and vision *IEEE Trans. Softw. Eng.* **50** 911–36
[72] Guo Z *et al* 2023 Evaluating large language models: a comprehensive survey (arXiv:2310.19736)
[73] McIntosh T R, Susnjak T, Liu T, Watters P and Halgamuge M N 2024 Inadequacies of large language model benchmarks in the era of generative artificial intelligence (arXiv:2402.09880)
[74] Chib P S and Singh P 2024 LG-Traj: LLM guided pedestrian trajectory prediction (arXiv:2403.08032)
[75] Rasheed Z, Waseem M, Systä K and Abrahamsson P 2024 Large language model evaluation via multi AI agents: preliminary results (arXiv:2404.01023)
[76] Su J, Jiang C, Jin X, Qiao Y, Xiao T, Ma H, Wei R, Jing Z, Xu J and Lin J 2024 Large language models for forecasting and anomaly detection: a systematic literature review (arXiv:2402.10350)
[77] Merten G, Dejaegere G and Sakr M 2025 Using LLMs for analyzing AIS data (arXiv:2504.07557)
[78] Tian X, Gu J, Li B, Liu Y, Hu C, Wang Y, Zhan K, Jia P, Lang X and Zhao H 2024 Drivevlm: the convergence of

autonomous driving and large vision-language models (arXiv:2402.12289)
[79] Xu Z, Zhang Y, Xie E, Zhao Z, Guo Y, Wong K-Y K, Li Z and Zhao H 2024 DriveGPT4: interpretable end-to-end autonomous driving via large language model *IEEE Robot. Autom. Lett.*
[80] Xu Z, Bai Y, Zhang Y, Li Z, Xia F, Wong K-Y K, Wang J and Zhao H 2025 DriveGPT4-v2: harnessing large language model capabilities for enhanced closed-loop autonomous driving *Proc. Computer Vision and Pattern Recognition Conf.* pp 17261–70
[81] Cui Y, Huang S, Zhong J, Liu Z, Wang Y, Sun C, Li B, Wang X and Khajepour A 2023 Drivellm: charting the path toward full autonomous driving with large language models *IEEE Trans. Intell. Veh.* **9** 1450–64
[82] Wen L, Fu D, Li X, Cai X, Ma T, Cai P, Dou M, Shi B, He L and Qiao Y 2023 Dilu: a knowledge-driven approach to autonomous driving with large language models (arXiv:2309.16292)
[83] Katzourakis D 2025 Systems engineering for autonomous vehicles; supervising AI using large language models (ssuperllm) (arXiv:2501.10839)
[84] Fu D, Li X, Wen L, Dou M, Cai P, Shi B and Qiao Y 2024 Drive like a human: rethinking autonomous driving with large language models *2024 IEEE/CVF Winter Conf. on Applications of Computer Vision Workshops (WACVW)* (IEEE) pp 910–9
[85] Cui C, Yang Z, Zhou Y, Ma Y, Lu J, Li L, Chen Y, Panchal J and Wang Z 2024 Personalized autonomous driving with large language models: field experiments *2024 IEEE 27th Int. Conf. on Intelligent Transportation Systems (ITSC)* (IEEE) pp 20–27
[86] Li Y, Li L, Wu Z, Bing Z, Ai Y, Tian B, Xuanyuan Z, Knoll A C and Chen L 2024 Miningllm: towards mining 5.0 via large language models in autonomous driving and smart mining *IEEE Trans. Intell. Veh.* 1–12
[87] Ping Y *et al* 2025 Multimodal large language models-enabled UAV swarm: towards efficient and intelligent autonomous aerial systems (arXiv:2506.12710)
[88] Sezgin A 2025 Scenario-driven evaluation of autonomous agents: Integrating large language model for UAV mission reliability *Drones* **9** 213
[89] Tagliabue A, Kondo K, Zhao T, Peterson M, Tewari C T and How J P 2024 Real: resilience and adaptation using large language models on autonomous aerial robots *2024 IEEE 63rd Conf. on Decision and Control (CDC)* (IEEE) pp 1539–46
[90] Duvvuru V S A, Zhang B, Vierhauser M and Agrawal A 2025 LLM-agents driven automated simulation testing and analysis of small uncrewed aerial systems (arXiv:2501.11864)
[91] Sautenkov O, Yaqoot Y, Mustafa M A, Batool F, Sam J, Lykov A, Wen C-Y and Tsetserukou D 2025 UAV-CodeAgents: scalable UAV mission planning via multi-agent react and vision-language reasoning (arXiv:2505.07236)
[92] Cai S, Wu Y and Zhou L 2025 LLM-land: large language models for context-aware drone landing (arXiv:2505.06399)
[93] Wu W, Li C, Wang X, Luo B and Liu Q 2025 Large language model guided progressive feature alignment for multimodal UAV object detection (arXiv:2503.06948)
[94] Cai H, Dong J, Tan J, Deng J, Li S, Gao Z, Wang H, Su Z, Sumalee A and Zhong R 2025 FlightGPT: towards generalizable and interpretable UAV vision-and-language navigation with vision-language models (arXiv:2505.12835)
[95] Lin F, Tian Y, Wang Y, Zhang T, Zhang X and Wang F-Y 2024 Airvista: empowering UAVs with 3D spatial reasoning abilities through a multimodal large language model agent *2024 IEEE 27th Int. Conf. on Intelligent Transportation Systems (ITSC)* (IEEE) pp 476–81
[96] Yao F, Yue Y, Liu Y, Sun X and Fu K 2024 Aeroverse: UAV-agent benchmark suite for simulating, pre-training, finetuning, and evaluating aerospace embodied world models (arXiv:2408.15511)
[97] Li H, Liu X and Li G 2024 A benchmark for UAV-view natural language-guided tracking *Electronics* **13** 1706
[98] Wen Z, Zhao J, Zhang A, Bi W, Kuang B, Su Y and Wang R 2025 BiDGCNLLM: a graph–language model for drone state forecasting and separation in urban air mobility using digital twin-augmented remote ID data *Drones* **9** 508
[99] Khatiri S, Di Sorbo A, Zampetti F, Visaggio C A, Di Penta M and Panichella S 2024 Identifying safety–critical concerns in unmanned aerial vehicle software platforms with salient *SoftwareX* **27** 101748
[100] Liu G, Sun T, Li W, Li X, Liu X and Cui J 2024 EAI-sim: an open-source embodied AI simulation framework with large language models *2024 IEEE 18th Int. Conf. on Control & Automation (ICCA)* (IEEE) pp 994–9
[101] Tasneem O and Pieters R 2026 Human–robot collaborative visual inspection with large language models *Robot. Comput.-Integr. Manuf.* **98** 103154
[102] Yang Z, Raman S S, Shah A and Tellex S 2024 Plug in the safety chip: enforcing constraints for LLM-driven robot agents *2024 IEEE Int. Conf. on Robotics and Automation (ICRA)* (IEEE) pp 14435–42
[103] Waseem M, Bhatta K, Li C and Chang Q 2025 Pretrained LLMs as real-time controllers for robot operated serial production line (arXiv:2503.03889)
[104] Lykov A, Cabrera M A, Konenkov M, Serpiva V, Gbagbe K F, Alabbas A, Fedoseev A, Moreno L, Khan M H and Guo Z 2024 Industry 6.0: new generation of industry driven by generative AI and swarm of heterogeneous robots (arXiv:2409.10106)
[105] Rema C, Sousa A, Sobreira H, Costa P and Silva M F 2025 Exploring the potential of LLM-based chatbots for task scheduling in robot operations *2025 IEEE Int. Conf. on Autonomous Robot Systems and Competitions (ICARSC)* (IEEE) pp 45–51
[106] Zhang X, Yuan K, Xia L, Ma L, Liu H, Zhang X and Lyu Z 2025 LLM closed-loop application framework for industry manipulator system *Proc. 4th Int. Conf. on Computer, Artificial Intelligence and Control Engineering* pp 492–8
[107] Rekik K, Silva G, Bashir A and Müller R 2025 Multimodal interaction for human-robot collaboration in assembly: an LLM-enhanced approach *2025 IEEE 21st Int. Conf. on Automation Science and Engineering (CASE)* (IEEE) pp 1207–12
[108] Zhang C, Zhang C, Xu Z, Xie Q, Hou J, Feng P and Zeng L 2025 Embodied intelligent industrial robotics: concepts and techniques (arXiv:2505.09305)
[109] Oyekan J, Turner C, Bax M and Graf E 2025 Applying ontologies and knowledge augmented large language models to industrial automation: a decision-making guidance for achieving human-robot collaboration in industry 5.0 (arXiv:2505.18553)
[110] Xu C, Chu Y, Gao Q, Wu Z, Wang J, Yue Y, Dominik W and Zhu X 2025 Autonomous unmanned surface vehicle docking using large language model guide reinforcement learning *Ocean Eng.* **323** 120608
[111] Yang R, Hou M, Wang J and Zhang F 2023 Oceanchat: piloting autonomous underwater vehicles in natural language (arXiv:2309.16052)
[112] Yang R, Zhang F and Hou M 2024 Oceanplan: Hierarchical planning and replanning for natural language auv piloting in large-scale unexplored ocean environments *Proc. 18th Int. Conf. on Underwater Networks & Systems* pp 1–5

[113] Din M U, Akram W, Bakht A B, Dong Y and Hussain I 2025 Maritime mission planning for unmanned surface vessel using large language model *2025 IEEE Int. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR)* (IEEE) pp 1–6
[114] Wen J, Li Z, Xi M and He J 2025 A LLM-assisted AUV 3D path planning scheme under ocean current interference via reinforcement learning *IEEE Internet Things J.* **12** 39185–96
[115] Zhang M, Kuang M, Shi H, Zhu J, Zhu J and Jiang X 2025 Command-agent: reconstructing warfare simulation and command decision-making using large language models *Defence Technol.* **56** 294–313
[116] Caissutti C, Gerbier E, Khorrambakht E, Marinelli P, Munafo A and Caiti A 2025 Shared autonomy through LLMs and reinforcement learning for applications to ship hull inspections (arXiv:2509.05042)
[117] Grimaldi M, Cernicchiaro C, Rua S R, El-Masri-El-Chaarani A, Buchholz M, Michael L, Rodriguez P R, Carlucho I and Petillot Y R 2025 Advancing shared and multi-agent autonomy in underwater missions: integrating knowledge graphs and retrieval-augmented generation (arXiv:2507.20370)
[118] Saad A, Akram W and Hussain I 2025 Aquachat++: LLM-assisted multi-ROV inspection for aquaculture net pens with integrated battery management and thruster fault tolerance (arXiv:2508.06554)
[119] Chen R, Blow D, Abdullah A and Islam Md J 2025 Word2wave: language driven mission programming for efficient subsea deployments of marine robots *2025 IEEE Int. Conf. on Robotics and Automation (ICRA)* (IEEE) pp 4107–14
[120] Zhang W, Cai M, Zhang T, Lei G, Zhuang Y and Mao X 2024 Popeye: a unified visual-language model for multi-source ship detection from remote sensing imagery *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **17** 20050–63
[121] Pei D, He J, Liu K, Chen M and Zhang S 2024 Application of large language models and assessment of their ship-handling theory knowledge and skills for connected maritime autonomous surface ships *Mathematics* **12** 2381
[122] Kim T-Y and Choi W-S 2025 Autonomous vehicle maneuvering using vision–LLM models for marine surface vehicles *J. Mar. Sci. Eng.* **13** 1553
[123] Hu X, Zhang J, Liu W and Ma Z 2025 As-LLM: an LLM-based framework for industrial autonomous system relationship inference *2025 Joint Int. Conf. on Automation-Intelligence-Safety (ICAIS) & Int. Symp. on Autonomous Systems (ISAS)* (IEEE) pp 1–6
[124] Ren Y, Zhang H, Richard Yu F, Li W, Zhao P and He. Y 2024 Industrial internet of things with large language models (LLMS): an intelligence-based reinforcement learning approach *IEEE Trans. Mobile Comput.* **24** 4136–52
[125] Wang Y, Jiao R, Lang C, Zhan S S, Huang C, Wang Z, Yang Z and Zhu Q 2023 Empowering autonomous driving with large language models: a safety perspective (arXiv:2312.00812)
[126] Sobrín-Hidalgo D, González-Santamarta M A, Guerrero-Higueras Angel M, Rodríguez-Lera F J and Matellán-Olivera V 2024 Explaining autonomy: enhancing human-robot interaction through explanation generation with large language models (arXiv:2402.04206)
[127] Buchmann R, Eder J, Fill H-G, Frank U, Karagiannis D, Laurenzi E, Mylopoulos J, Plexousakis D and Santos M Y 2024 Large language models: expectations for semantics-driven systems engineering *Data Knowl. Eng.* **152** 102324
[128] Zaki O, Dunnigan M, Robu V and Flynn D 2021 Reliability and safety of autonomous systems based on semantic modelling for self-certification *Robotics* **10** 10
[129] Lyons J B, Clark M A, Wagner A R and Schuelke M J 2017 Certifiable trust in autonomous systems: making the intractable tangible *AI Mag.* **38** 37–49
[130] Vaid A *et al* 2024 Generative large language models are autonomous practitioners of evidence-based medicine (arXiv:2401.02851)
[131] Tao Z, Lin T-E, Chen X, Li H, Wu Y, Li Y, Jin Z, Huang F, Tao D and Zhou J 2024 A survey on self-evolution of large language models (arXiv:2404.14387)
[132] Popescu N 2022 Safety verification and validation techniques for autonomous driving systems *J. Human. Appl. Sci. Res.* **5** 71–87
[133] Zheng Xi, Mok A K, Piskac R, Lee Y J, Krishnamachari B, Zhu D, Sokolsky O and Lee I 2024 Testing learning-enabled cyber-physical systems with large-language models: a formal approach *Companion Proc. 32nd ACM Int. Conf. on the Foundations of Software Engineering* pp 467–71
[134] Kong X, Braunl T, Fahmi M and Wang Y 2024 A superalignment framework in autonomous driving with large language models (arXiv:2406.05651)
[135] Wu T, He S, Liu J, Sun S, Liu K, Han Q-L and Tang Y 2023 A brief overview of ChatGPT: the history, status quo and potential future development *IEEE/CAA J. Autom. Sinica* **10** 1122–36
[136] Wang L *et al* 2024 A survey on large language model based autonomous agents *Front. Comput. Sci.* **18** 186345
[137] Mann S P, Jiehao J S, Latham S R, Savulescu J, Aboy M and Earp B D 2025 Development of application-specific large language models to facilitate research ethics review (arXiv:2501.10741)
[138] Corfmat M, Martineau J T and Régis C 2025 High-reward, high-risk technologies? An ethical and legal account of AI development in healthcare *BMC Med. Ethics* **26** 4
[139] Acharya K, Velasquez A and Song H H 2024 A survey on symbolic knowledge distillation of large language models *IEEE Trans. Artif. Intell.* **5** 5928–48
[140] Van Noorden R 2023 ChatGPT-like AIS are coming to major science searches *Nature* **620** 258
[141] Lu Y, Tian Y, Bi Y, Chen B and Peng X 2024 Diavio: LLM-empowered diagnosis of safety violations in ads simulation testing *Proc. 33rd ACM SIGSOFT Int. Symp. on Software Testing and Analysis* pp 376–88
[142] Jha C K *et al* 2025 Large language models (LLMs) for verification, testing and design *2025 IEEE European Test Symp. (ETS)* (IEEE) pp 1–10
[143] Lu Q, Wang X, Jiang Y, Zhao G, Ma M and Feng S 2025 Omnitester: multimodal large language model driven scenario testing for autonomous vehicles *Autom. Innov.* **8** 1–15
[144] Zhou Y, Cui C, Peng J, Yang Z, Lu J, Panchal J, Yao B and Wang Z 2025 A hierarchical test platform for vision language model (VLM)-integrated real-world autonomous driving *ACM Trans. Internet Things* (https://doi.org/10.1145/3769867)
[145] Esposito M, Palagiano F, Lenarduzzi V and Taibi D 2025 On large language models in mission-critical it governance: are we ready yet? *2025 IEEE/ACM 47th Int. Conf. on Software Engineering: Software Engineering in Practice (ICSE-SEIP)* (IEEE) pp 504–15



Dun Li received a PhD degree in computer science from Institut Polytechnique de Paris, France, in 2025. He is currently a Postdoctoral Researcher with the Department of Industrial Engineering, Tsinghua University, China. His research interests include large language models, Industrial Internet of Things, digital twin, and system reliability.

Ruiguan Lin received a PhD degree in engineering from Nanjing University of Aeronautics and Astronautics, China, in 2024. He is currently a Postdoctoral Researcher and Assistant Researcher with the Department of Industrial Engineering, Tsinghua University, China. His research interests include intelligent operation and maintenance of high-end equipment, reliability assessment and management for civil aviation systems, and maintenance decision-making for civil aircraft structures.

Zisheng Wang received a BS degree in Mechanical Engineering from Northeastern University, Shenyang, China, in 2018, and a PhD degree in Mechanical Engineering from Huazhong University of Science and Technology, Wuhan, China, in 2023. He is currently an Assistant Research Fellow with the Department of Industrial Engineering, Tsinghua University, China, supported by the Shuimu Tsinghua Scholar Talent Program. His research interests include intelligent monitoring and maintenance for high-end equipment and multimodal large language models.

Yan-Fu Li (Senior Member, IEEE) was a Faculty Member with the Laboratory of Industrial Engineering, CentraleSupélec, University of Paris-Saclay, Gif-sur-Yvette, France, from 2011 to 2016. He is currently a Professor with the Department of Industrial Engineering, Tsinghua University, Beijing, China. He has led or participated in several research projects supported by the European Union, France, and Chinese governmental funding agencies, as well as various industrial partners. He has authored or coauthored more than 100 publications in international journals, conference proceedings, and books. His research interests include reliability, availability, maintainability, and safety (RAMS) assessment and optimization for industrial systems. Dr. Li is an Associate Editor of *IEEE Transactions on Reliability*.