NeuroStrata: Harnessing Neurosymbolic Paradigms for Improved Design, Testability, and Verifiability of Autonomous CPS

Xi Zheng
Macquarie University
Sydney, Australia
james.zheng@mq.edu.au

Ziyang Li
University of Pennsylvania
Philadelphia, USA
liby99@seas.upenn.edu

Ivan Ruchkin
University of Florida
Gainesville, USA
iruchkin@ece.ufl.edu

Ruzica Piskac
Yale University
New Heaven, USA
ruzica.piskac@yale.edu

Miroslav Pajic
Duke University
Durham, USA
miroslav.pajic@duke.edu

ABSTRACT

Autonomous cyber-physical systems (CPSs) leverage AI for perception, planning, and control but face trust and safety certification challenges due to inherent uncertainties. The neurosymbolic paradigm replaces stochastic layers with interpretable symbolic AI, enabling determinism. While promising, challenges like multisensor fusion, adaptability, and verification remain. This paper introduces **NeuroStrata**, a neurosymbolic framework to enhance the testing and verification of autonomous CPS. We outline its key components, present early results, and detail future plans.

CCS CONCEPTS

• Software and its engineering → Software verification and validation; **• Computer systems organization → Embedded and cyber-physical systems**; **• Computing methodologies →** *Machine learning*; **• Theory of computation →** *Formal languages and automata theory*.

KEYWORDS

AI-based Systems, Cyber-Physical Systems, Neurosymbolic AI, Testing, Verification

ACM Reference Format:
Xi Zheng, Ziyang Li, Ivan Ruchkin, Ruzica Piskac, and Miroslav Pajic. 2025. NeuroStrata: Harnessing Neurosymbolic Paradigms for Improved Design, Testability, and Verifiability of Autonomous CPS. In *Companion Proceedings of the 33rd ACM Symposium on the Foundations of Software Engineering (FSE '25), June 23–27, 2025, Trondheim, Norway*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 CONTEXT, MOTIVATION, AND AIMS

The integration of machine learning (ML) into CPS has driven innovations in autonomous vehicles [43, 47, 48], delivery drones [2, 21, 52], and robotic surgeries [11, 33, 35]. While ML enhances autonomy and intelligence, its uncertain and brittle nature, as witnessed in softmax-based classifications and regression-based control, undermines traditional formal verification, necessitating novel solutions. *Neurosymbolic approaches* combine symbolic reasoning with neural learning, addressing challenges in autonomous CPSs by enabling co-training of neural components with symbolic logic through probabilistic logic programming and differentiable reasoning [10, 29, 32, 34]. Advanced methods, such as program synthesis using Domain-Specific Languages (DSLs), demonstrate promise by supporting both deterministic and probabilistic programs [5, 18, 36]. However, significant bottlenecks persist, including unseen data in real-world deployments [19], multi-sensor fusion challenges [42], and the lack of neural component verification. *Determinism at the decision level* of perception and mission planning is also missing, which can enable the applicability of decades-long formal verification and testing techniques. Ensuring system-level safety and liveness requires *systematic co-design* of perception, planning, and control — a critical aspect missing in current methods.

This paper summarizes state-of-the-art neurosymbolic paradigms, using autonomous driving as a case study to highlight gaps in adaptability to unseen environments, multi-sensor complexity, systematic validation of neural components, and *determinism at the decision level*, critical for reliable autonomous CPS.

To address these issues, we put forward **NeuroStrata** — a neurosymbolic framework for designing and assuring autonomous CPSs. Our vision integrates neurosymbolic distillation and corner-case test generation using LLMs to enable data-driven specification mining, top-down synthesis of symbolic and neurosymbolic components, and runtime bottom-up adaptation via program induction. This approach evolves symbolic programs dynamically for decision-making in perception and planning/control modules. Building on prior work, we aim to transform the testability and verifiability of autonomous CPSs through the neurosymbolic paradigm.

2 MOTIVATING SYSTEM AND THE STATE-OF-THE-ART

State-of-the-art autonomous CPSs, such as autonomous driving systems (ADS), are typically built on middleware frameworks like Robot Operating System (ROS) with proprietary extensions (e.g., Baidu's CyberRT) to reduce message latency [16, 30, 31, 37, 51]. As shown in Figure 1, these systems integrate perception, prediction, planning, and control modules. The perception module processes multi-modal sensor data (e.g., LiDAR, cameras, IMU) for obstacle

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

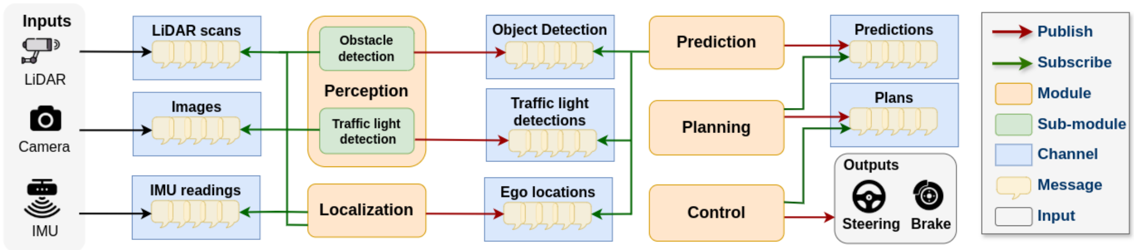

Figure 1: Motivating system: an industrial-strength software stack for autonomous driving [14].

detection, traffic light recognition, and localization. The prediction module forecasts dynamic object trajectories, while the planning module computes the ego vehicle's trajectory. The control module generates actuation commands, such as steering and braking. AI components are pervasive across these modules, enabling object detection, trajectory prediction, and real-time control.

Recent advancements in verifying machine learning models and testing learning-enabled CPS have utilized methods like NNV star sets [46], Sherlock [17], Reluplex [27], and Branch-and-Bound [7]. While promising, these methods often fall short in providing comprehensive safety and liveness guarantees, particularly when dealing with the complexity and scale of the real-world, multi-modal CPS as shown in the motivating system above. System-level testing [13, 28, 44] and robustness analyses [8, 12, 45] offer progress but still rely on probabilistic methods to assure probabilistic systems [4, 9], which can weaken the guarantees [3].

In parallel, *neurosymbolic paradigms* aim to combine neural and symbolic approaches — either through differentiable logic programming to produce differentiable yet deterministic outputs [29, 38] or through program induction to synthesize deterministic programs by observing limited input-output pairs [5, 18, 36]. However, they face notable limitations. The differentiable logic programming paradigm struggles with hard-coded mappings between neural component outputs and logic program inputs, as well as the complexity of implementing differentiable logic programming capabilities. These constraints limit its ability to support multi-modal sensors and the sophisticated logics required for autonomous systems, which demand more automated and expressive guidance. Similarly, the program induction paradigm relies on Domain-Specific Languages (DSLs) for guidance, which are often hard-coded and lack the flexibility needed for multi-modal sensor integration and advanced reasoning [5, 18, 36]. Furthermore, program induction may not be sufficient to replace all pure AI components in such systems.

Recent work on *multimodal neurosymbolic systems* integrates visual and auditory signals but relies on simplistic symbolic rules, unsuitable for the diverse sensor modalities of autonomous CPS, such as LiDARs, radars, and cameras [24]. A Neural State Machine (NSM) approach combines vision and language reasoning but suffers from scalability issues, manual scene graph construction, and limited interpretability due to missing source code [26]. These limitations underscore the need for refined co-design to meet the stringent requirements of autonomous CPS.

The above limitations emphasize the pressing need for *deterministic testing and verification approaches* in autonomous CPS, particularly in safety-critical domains like autonomous driving. For instance, perception modules in autonomous vehicles, which rely on uncertain or stochastic processes such as (Bayesian) neural networks for object detection, often fail in "long tail" scenarios where the inputs deviate from training data (covariate shift). Predictable and deterministic approaches, incorporating reasoning layers, can adapt to such unseen scenarios by leveraging symbolic logic to ensure robust decision-making and mitigate failures caused by stochastic uncertainties. This adaptability is crucial for guaranteeing safety and reliability across perception, prediction, and planning modules in dynamic and complex real-world environments [39, 40, 50].

3 NEUROSTRATA: OUR VISION FOR HIERARCHICAL NEUROSYMBOLIC FRAMEWORK FOR AUTONOMOUS SYSTEMS

To address the challenges of designing, testing, and verifying autonomous CPS, we propose a new **neurosymbolic framework**, **NeuroStrata**, tailored to the unique requirements of such systems. As shown in Figure 2, NeuroStrata combines neural adaptability with symbolic reasoning to enforce formal specifications across hierarchical DSLs that capture underlying safety and liveness properties. The framework structures *Perception* and *Planning & Control* capabilities into high-level (symbolic-only) and middle- and low-level (neurosymbolic) modules. It ensures runtime reliability and adaptation via a two-phase process: *top-down synthesis*, propagating symbolic specifications to neurosymbolic modules, and *bottom-up adaptation*, where neurosymbolic outputs refine symbolic programs.

Modules. At design time, *Specification Mining*, built on neurosymbolic distillation [1, 6, 41], extracts formal safety and liveness specifications from training datasets. To cover more diverse safety and liveness violations and out-of-distribution scenarios beyond existing training data, we leverage recent work using large language models to analyze multi-modal sensor data [15, 51], such as front-facing cameras in vehicles, to generate additional real-world crashes and unusual cases from various angles. These specifications are propagated hierarchically across the system. In the perception stack, a high-level *Scene Graph* encodes semantic relationships and interactions between objects (e.g., "pedestrian crossing road"), represented as differentiable, adaptable programs that can be verified using formal tools like theorem provers. The middle-level *Semantic Map* encodes spatial and semantic information such as road layouts and drivable areas, ensuring consistency with the scene graph via symbolic rules. The low-level *Sensor Fusion and Signal Processing* integrates multi-modal sensor data (e.g., LiDAR, cameras, GPS) while enforcing constraints on accuracy and consistency, leveraging neurosymbolic reasoning for fusion and processing. Similarly,

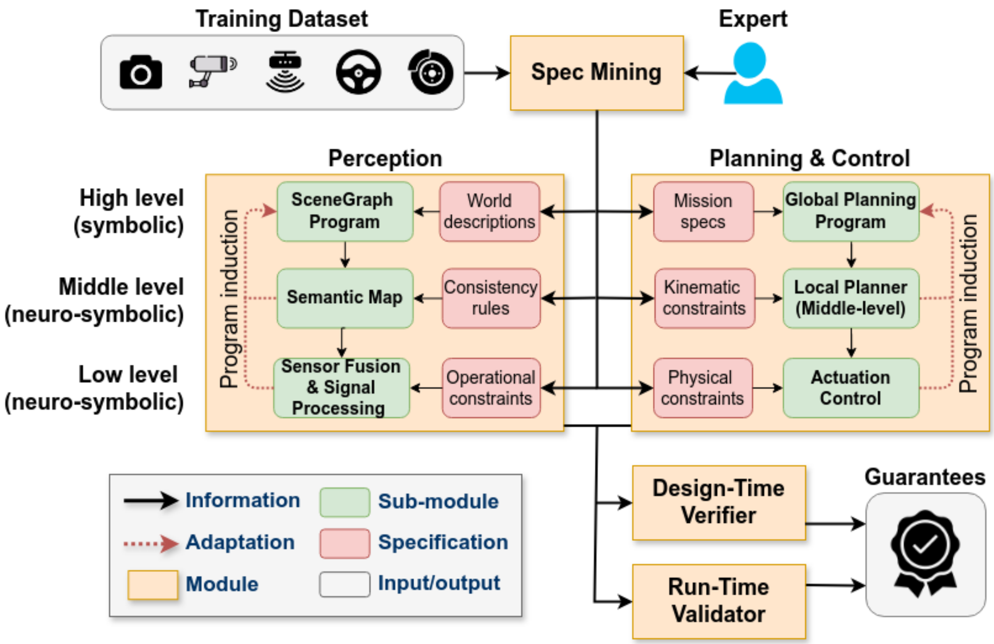

Figure 2: Proposed Vision for NeuroStrata: Hierarchical Neurosymbolic Programming for Autonomous CPS

the planning and control stack follows a hierarchical structure. The high-level *Global/Mission Planner* synthesizes deterministic programs to achieve overall system objectives, verified with formal methods such as theorem proving. The middle-level *Local Planner* generates short-term trajectories that align with global plans while adapting to local changes, guided by symbolic reasoning. The low-level *Actuation Control* converts trajectories into control commands (e.g., steering angle, throttle) and ensures compliance with constraints using runtime verification techniques.

Specifications. For *perception*, high-level specifications govern system-wide context awareness, such as ensuring that pedestrians and vehicles do not spatially overlap in the scene graph or that all objects adhere to semantic relationships. Middle-level specifications enforce localized consistency, such as aligning lane boundaries with the semantic map and ensuring that detected objects are positioned correctly within the road layout. Low-level specifications address operational constraints, such as maintaining sensor fusion accuracy within a 0.1-meter error margin and ensuring consistent integration of multi-modal sensor data. For *planning/control*, high-level specifications ensure system-wide safety and mission compliance, such as requiring the vehicle to remain within designated route bounds throughout its journey. Middle-level specifications enforce trajectory-level constraints, such as avoiding obstacles within a 2-meter radius or maintaining smooth transitions between trajectory points. Low-level specifications govern detailed actuation control, such as keeping the steering angle within physical limits and ensuring the stability of throttle and braking in response to control inputs. These hierarchical specifications for perception and planning/control ensure an integrated and reliable system design.

Adaptation. During runtime, NeuroStrata dynamically adapts its perception and planning modules to real-time data while maintaining formal specification compliance. For perception, sensor data flows upward through the hierarchy, where outputs from the low-level sensor fusion are validated against middle-level semantic map constraints, and updates propagate to the high-level scene graph. It evolves dynamically using differentiable program induction, compacting, and adapting specifications as needed. For planning and control, high-level mission planners adjust strategies based on changing conditions, while differentiable and adaptable control programs refine global plans and compact themselves in response to system data. Middle- and low-level components, such as local planners and actuation control, remain guided by symbolic reasoning to ensure safety and alignment with global objectives. This integration enables simultaneously adaptable and formally validated behavior throughout the system.

Guarantees. NeuroStrata ensures reliability through a hybrid validation framework. High-level deterministic programs, such as scene graphs and mission planners, are validated using formal verification tools like model checking and theorem proving. Middle- and low-level neurosymbolic components, such as semantic maps and sensor fusion, are guided by symbolic constraints and validated using white-box testing, runtime monitors, and error propagation analysis (e.g., approximate reachability verification [20] and conformance checking [23]). Together, this framework bridges the gap between deterministic high-level programs and adaptive, data-driven neuro-components, thus providing formal guarantees across all three levels of the hierarchy.

4 EARLY RESULTS AND FUTURE PLAN

We conducted a preliminary case study to investigate a key **Research Question (RQ)**: "can neurosymbolic reasoning complement neural-network training to align with underlying specifications?". We also outline our future plans, along with the potential challenges and proposed solutions.

4.1 Assessing neurosymbolic reasoning to align neural network training with specifications

In this study, we investigate the capability of differentiable neurosymbolic reasoning to align perceptual neural networks with specifications. We explore the application of a high-level visual perception system trained by aligning its output with specifications. In this application, the goal is to infer spatio-temporal scene graphs (STSG) from videos (e.g., ones taken by ego-centric cameras), where the scene graphs must align with a given spatio-temporal specification. Figure 3 depicts a specification for a traffic scene which is described in natural language but then formalized into a temporal logic formula. Notice that the specification consists of logical symbols like *exists* (∃), *and* (∧), *not* (¬), and *finally* (◇). A neurosymbolic approach to solving this task comprises of a neural model for STSG extraction and a differentiable symbolic component for aligning the predicted STSG with the given specification. Being differentiable, the loss computed from the alignment process can be used to supervise the neural model. We evaluated our work on three datasets: OpenPVSG [49], 20BN-Something-Something [22], and MUGEN [25], each with diverse temporal properties. Our approach outperforms current baselines on downstream tasks while offering explainability[1]. This specification alignment provides high confidence in our top-down synthesis approach and in guiding neural training with our specifications.

4.2 Future Research Plan

To advance NeuroStrata, we propose a *six-step future plan* with concrete steps to address challenges at each stage.

First, *generating diverse training datasets* will leverage recent advancements in model-based testing that utilize LLMs to analyze multi-modal sensor data [15, 51]. The multi-faceted challenge lies in ensuring the generated datasets are diverse, representative of real-world scenarios, and capable of addressing edge cases. Solutions can be tailored around prior work by accessing diverse sensor datasets and logs, leveraging advanced multi-modal LLMs, and integrating domain-specific constraints with iterative refinement based on industrial partner feedback.

Second, *designing suitable DSLs* is essential for capturing hierarchical and semantically rich specifications. These DSLs enable experts to encode operational constraints for sensor fusion, signal processing, and physical actuation control. Challenges include ensuring the DSLs are intuitive for domain experts while expressive enough to handle complex requirements. Solutions involve co-designing DSLs with autonomy and robotics specialists, developing language automation, and designing usable visual interfaces. By providing a bridge between formal methods and practical application, these DSLs empower experts to play an active role in system design.

[1]Reference to the full report was anonymized for the review process.

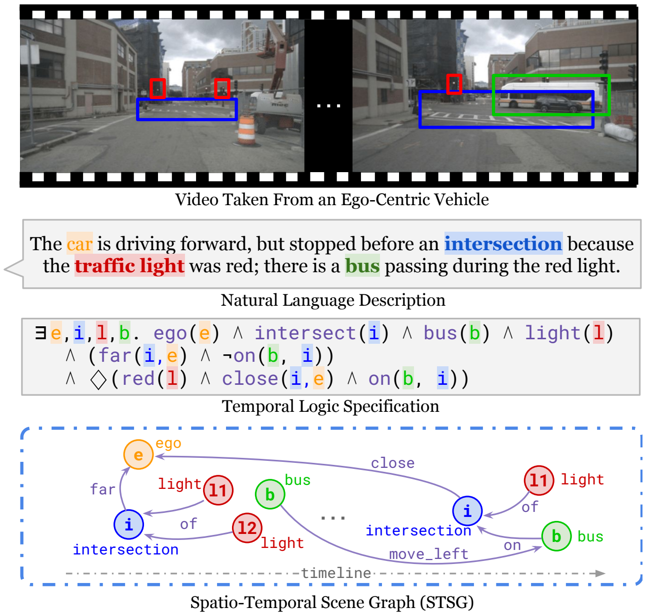

Figure 3: Aligning STSG with natural language description via temporal logic specifications.

Third, *developing a specification mining module* based on neurosymbolic distillation will extract formal safety and liveness specifications from training datasets and LLM interactions. Aligning mined specifications with real-world requirements and handling noisy/hallucinated data are key challenges. Hybrid approaches that combine symbolic reasoning with neural embeddings, as well as active learning techniques, can iteratively refine the mined specifications to ensure accuracy and physical grounding.

Fourth, for design-time synthesis and verification, we will enforce *multi-level specifications* for perception and planning/control modules, leveraging the hierarchical structure defined in our DSL. Modular architectures will enable scalable top-down synthesis of symbolic and neurosymbolic components, while formal verification ensures compliance with specifications. Parallelized processes and adaptive abstraction techniques will address scalability challenges, ensuring robustness across diverse scenarios and high-dimensional inputs.

Fifth, for runtime adaptation and validation, we will develop mechanisms to *dynamically refine symbolic programs* for real-world changes while ensuring specification compliance. Inspired by program induction approaches like DreamCoder [18], NeuroStrata will iteratively refine symbolic representations using real-time data. Challenges include maintaining computational efficiency and real-time guarantees. To address these, we will optimize runtime validators, integrate lightweight symbolic reasoning for faster adaptation, and implement efficient runtime verification to ensure reliability and compliance with minimal overhead. These advancements will enable NeuroStrata to adapt to dynamic environments and evolving operational conditions.

Finally, for *industrial deployment*, NeuroStrata will be applied to autonomous driving systems, delivery drones, cargo drones, and passenger aircraft — as facilitated by our partners. Key challenges include seamless integration into existing systems, adherence to stringent safety standards, and building trust among stakeholders. Solutions include close collaborative projects, iterative deployment in increasingly open environments, and the creation of comprehensive documentation and training programs to facilitate adoption.

5 CONCLUSION

This paper explores the potential and challenges of neurosymbolic paradigms for designing, testing, and verifying autonomous CPS. We propose **NeuroStrata**, a framework enabling top-down synthesis of symbolic and neurosymbolic components for perception and planning/control, and bottom-up adaptation of symbolic programs for real-time decisions. Early results validate neural alignment with specifications. We outline challenges and solutions for implementing **NeuroStrata**, aiming to bridge theoretical advancements and practical applications, transforming autonomous CPS testing and verification in real-world scenarios.

REFERENCES

[1] Farhan Fuad Abir. 2024. Neuro-Symbolic Distillation of Reinforcement Learning Agents. (2024).

[2] Amazon Prime Air 2023. *AmazonPrimeAir.* Retrieved Aug 3, 2023 from https://shorturl.at/otyU3

[3] Thom Badings, Licio Romao, Alessandro Abate, and Nils Jansen. 2023. Probabilities Are Not Enough: Formal Controller Synthesis for Stochastic Dynamical Models with Epistemic Uncertainty. In *Proc. of AAAI 2023*. https://doi.org/10.48550/arXiv.2210.05989 arXiv:2210.05989 [cs, eess].

[4] Apurva Badithela, Tichakorn Wongpiromsarn, and Richard M. Murray. 2023. Evaluation Metrics of Object Detection for Quantitative System-Level Analysis of Safety-Critical Autonomous Systems. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 8651–8658. https://ieeexplore.ieee.org/abstract/document/10342465/

[5] Katrine Bjørner, Samuel Judson, Filip Cano, Drew Goldman, Nick Shoemaker, Ruzica Piskac, and Bettina Könighofer. 2023. Formal XAI via syntax-guided synthesis. In *International Conference on Bridging the Gap between AI and Reality*. Springer, 119–137.

[6] Paul J Blazek, Kesavan Venkatesh, and Milo M Lin. 2024. Automated discovery of algorithms from data. *Nature Computational Science* 4, 2 (2024), 110–118.

[7] Rudy R Bunel et al. 2018. A unified view of piecewise linear neural network verification. *NeurIPS* 31 (2018).

[8] Feiyang Cai and Xenofon Koutsoukos. 2020. Real-time out-of-distribution detection in learning-enabled cyber-physical systems. In *ICCPS*. IEEE, 174–183. https://doi.org/10.1109/ICCPS48487.2020.00024

[9] Matthew Cleaveland, Ivan Ruchkin, Oleg Sokolsky, and Insup Lee. 2022. Monotonic Safety for Scalable and Data-Efficient Probabilistic Safety Analysis. In *ACM/IEEE 13th Intl. Conf. on Cyber-Physical Systems (ICCPS)*. 92–103. https://doi.org/10.1109/ICCPS54341.2022.00015

[10] William W Cohen, Fan Yang, and Kathryn Rivard Mazaitis. 2017. Tensorlog: Deep learning meets probabilistic dbs. *arXiv preprint arXiv:1707.05390* (2017).

[11] Davinci 2023. *Da Vinci Surgical System.* Retrieved Aug 3, 2023 from http://www.intuitivesurgical.com/

[12] Yao Deng et al. 2020. An analysis of adversarial attacks and defenses on autonomous driving models. In *PerCom*. IEEE, 1–10. https://doi.org/10.1109/PerCom45495.2020.9127389

[13] Yao Deng et al. 2022. A declarative metamorphic testing framework for autonomous driving. *TSE* (2022). https://doi.org/10.1109/TSE.2022.3206427

[14] Yao Deng et al. 2022. Scenario-based test reduction and prioritization for multi-module autonomous driving systems. In *FSE*. 82–93. https://doi.org/10.1145/3540250.3549152

[15] Yao Deng et al. 2023. TARGET: Traffic Rule-based Test Generation for Autonomous Driving Systems. *arXiv preprint arXiv:2305.06018* (2023).

[16] Yuankai Deng, Ting Zhang, Guang Lou, Xiangyu Zheng, Jin Jin, and Qing-Long Han. 2021. Deep learning-based autonomous driving systems: A survey of attacks and defenses. *IEEE Transactions on Industrial Informatics* 17, 12 (2021), 7897–7912.

[17] Souradeep Dutta et al. 2019. Sherlock-a tool for verification of neural network feedback systems: demo abstract. In *HSCC*. 262–263.

[18] Kevin Ellis, Catherine Wong, et al. 2021. Dreamcoder: Bootstrapping inductive program synthesis with wake-sleep library learning. In *Proceedings of the 42nd acm sigplan international conference on programming language design and implementation*. 835–850.

[19] Matt Fredrikson, Kaiji Lu, Saranya Vijayakumar, Somesh Jha, Vijay Ganesh, and Zifan Wang. 2023. Learning Modulo Theories. https://doi.org/10.48550/arXiv.2301.11435 arXiv:2301.11435 [cs].

[20] Yuang Geng, Jake Brandon Baldauf, Souradeep Dutta, Chao Huang, and Ivan Ruchkin. 2024. Bridging Dimensions: Confident Reachability for High-Dimensional Controllers. In *Proc. of the International Symposium on Formal Methods (FM)*. Milano, Italy. https://doi.org/10.48550/arXiv.2311.04843 arXiv:2311.04843 [cs].

[21] Google Wing 2023. *Google Wing.* Retrieved Aug 3, 2023 from http://www.wing.com/

[22] Raghav Goyal, Samira Ebrahimi Kahou, et al. 2017. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*. 5842–5850.

[23] P. Habeeb, Lipsy Gupta, and Pavithra Prabhakar. 2024. Approximate Conformance Checking for Closed-Loop Systems With Neural Network Controllers. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 43, 11 (Nov. 2024), 4322–4333. https://doi.org/10.1109/TCAD.2024.3445813 Conference Name: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

[24] Ijaz Ul Haq, Khalid Mahmood Malik, and Khan Muhammad. 2024. Multimodal Neurosymbolic Approach for Explainable Deepfake Detection. *ACM Trans. Multimedia Comput. Commun. Appl.* 20, 11 (Sept. 2024), 341:1–341:16. https://doi.org/10.1145/3624748

[25] Thomas Hayes, Songyang Zhang, Xi Yin, et al. 2022. Mugen: A playground for video-audio-text multimodal understanding and generation. In *European Conference on Computer Vision*. Springer, 431–449.

[26] Aman Jain, Anirudh Reddy Kondapally, Kentaro Yamada, and Hitomi Yanaka. 2024. Neuro-Symbolic Reasoning for Multimodal Referring Expression Comprehension in HMI Systems. *New Generation Computing* (2024), 1–20.

[27] Guy Katz et al. 2017. Reluplex: An efficient SMT solver for verifying deep neural networks. In *CAV*. Springer, 97–117.

[28] Guanpeng Li et al. 2020. Av-fuzzer: Finding safety violations in autonomous driving systems. In *2020 IEEE 31st international symposium on software reliability engineering (ISSRE)*. IEEE, 25–36.

[29] Ziyang Li, Jiani Huang, and Mayur Naik. 2023. Scallop: A language for neurosymbolic programming. *Proceedings of the ACM on Programming Languages* 7, PLDI (2023), 1463–1487.

[30] Linfeng Liang, Yao Deng, et al. 2023. RLaGA: A Reinforcement Learning Augmented Genetic Algorithm For Searching Real and Diverse Marker-Based Landing Violations. *arXiv preprint arXiv:2310.07378* (2023).

[31] Guannan Lou, Yao Deng, Xi Zheng, Mengshi Zhang, and Tianyi Zhang. 2022. Testing of autonomous driving systems: where are we and where should we go?. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 31–43.

[32] Zhen Lu, Imran Afridi, Hong Jin Kang, Ivan Ruchkin, and Xi Zheng. 2024. Surveying neuro-symbolic approaches for reliable artificial intelligence of things. *Journal of Reliable Intelligent Environments* (July 2024). https://doi.org/10.1007/s40860-024-00231-1

[33] Mako 2023. *Mako Robotic-Arm.* Retrieved Aug 3, 2023 from https://t.ly/VLEwP

[34] Robin Manhaeve, Sebastijan Dumančić, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. 2021. Neural probabilistic logic programming in DeepProbLog. *Artificial Intelligence* 298 (2021), 103504.

[35] Mazor 2023. *Mazor Robotics.* Retrieved Aug 3, 2023 from http://www.medtronic.com/

[36] Emilio Parisotto, Abdel-rahman Mohamed, Rishabh Singh, Lihong Li, Dengyong Zhou, and Pushmeet Kohli. 2016. Neuro-symbolic program synthesis. *arXiv preprint arXiv:1611.01855* (2016).

[37] Aditya Prakash et al. 2021. Multi-modal fusion transformer for end-to-end autonomous driving. In *CVPR*. 7077–7087.

[38] Mukund Raghothaman, Sulekha Kulkarni, Richard Zhang, Xujie Si, Kihong Heo, Woosuk Lee, and Mayur Naik. [n. d.]. DIFFLOG: Beyond Deductive Methods in Program Analysis. ([n. d.]).

[39] Sanjit A Seshia. 2017. Compositional verification without compositional specification for learning-based systems. *UC Berkeley* (2017), 1–8.

[40] Shai Shalev-Shwartz et al. 2017. On a formal model of safe and scalable self-driving cars. *arXiv preprint arXiv:1708.06374* (2017).

[41] Suraj Singireddy, Precious Nwaorgu, Andre Beckus, Aden McKinney, Chinwendu Enyioha, Sumit Kumar Jha, George K Atia, and Alvaro Velasquez. 2023. Automaton Distillation: Neuro-Symbolic Transfer Learning for Deep Reinforcement Learning. *arXiv preprint arXiv:2310.19137* (2023).

[42] Mark Strickland, Georgios Fainekos, and Heni Ben Amor. 2018. Deep Predictive Models for Collision Risk Assessment in Autonomous Driving. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 4685–4692. https://doi.org/10.1109/ICRA.2018.8461160 ISSN: 2577-087X.

[43] Tesla Autopilot 2023. *Tesla.* Retrieved Aug 3, 2023 from http://www.tesla.com/autopilot/

[44] Haoxiang Tian et al. 2022. MOSAT: finding safety violations of autonomous driving systems using multi-objective genetic algorithm. In *FSE*. 94–106. https://doi.org/10.1145/3540250.3549100

[45] Yuchi Tian et al. 2018. Deeptest: Automated testing of deep-neural-network-driven autonomous cars. In *ICSE*. 303–314. https://doi.org/10.1145/3180155.3180220

[46] Hoang-Dung Tran et al. 2020. Verification of deep convolutional neural networks using imagestars. In *CAV*. Springer, 18–42.

[47] Uber ATG 2023. *Uber.* Retrieved Aug 3, 2023 from https://shorturl.at/mSV59

[48] Waymo 2023. *Waymo.* Retrieved Aug 3, 2023 from http://www.waymo.com/

[49] Jingkang Yang, Wenxuan Peng, Xiangtai Li, et al. 2023. Panoptic video scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18675–18685.

[50] Eleni Zapridou et al. 2020. Runtime verification of autonomous driving systems in CARLA. In *RV*. Springer, 172–183.

[51] Xi Zheng, Aloysius K Mok, Ruzica Piskac, Yong Jae Lee, Bhaskar Krishnamachari, Dakai Zhu, Oleg Sokolsky, and Insup Lee. 2024. Testing learning-enabled cyber-physical systems with large-language models: A formal approach. In *Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering*. 467–471.

[52] Zipline 2023. *Zipline.* Retrieved Aug 3, 2023 from http://www.flyzipline.com/