Using Deep Reinforcement Learning And Formal Verification in Safety Critical Systems: Strategies and Challenges

Satyam Sharma[1], Muhammad Abdul Basit Ur Rahim[2], Shahid Hussain[3],
Muhammad Rizwan Abid[4], Tairan Liu[5]
[1,2]Computer Engineering & Computer Science, California State University, Long Beach, CA, USA
[3]Computer Science and Software Engineering, Penn State University, Behrend, PA, USA
[4]Computer Science, Florida Polytechnic University, Lakeland, FL, USA
[5]Mechanical and Aerospace Engineering, California State University, Long Beach, CA, USA
satyamsharma.-sa@csulb.edu, Muhammad.AbdulBasitUrRahim@csulb.edu,
shussain@psu.edu, mabid@floridapoly.edu, tairan.liu@csulb.edu

Abstract—Deep Reinforcement Learning (DRL) is critical in modern Artificial Intelligence (AI), powering innovations from gaming to autonomous vehicles. As DRL continues its rapid ascent, ensuring its systems are both trustworthy and effective is crucial. This research focuses on different DRL techniques and the challenges faced in real-life scenarios. The paper also describes various formal verification techniques and the challenges related to their application. It sheds light on the different frameworks and tools that can enhance the credibility of systems. We performed an extensive literature survey to present the existing methodologies, tools, and frameworks. The analysis systematically reviews and categorizes various formal verification techniques and frameworks employed in DRL. The insights garnered from this study are anticipated to foster an enriched understanding of the processes and contribute to decision-making in Safety Critical Systems using DRL and verification.

Keywords–Reinforcement Learning, Formal Methods, Formal Verification, Deep Reinforcement Learning, Safety Critical Systems, Decision making

1. Introduction

AI has greatly shifted from static decision-making models to adaptive learning systems. A new era in which computational agents refined their behavior depending on rewards and penalties was introduced by reinforcement learning. Deep Reinforcement Learning was created due to the fusion of deep learning methods and reinforcement learning (RL), significantly boosting an agent's capacity for comprehension and interaction in complex circumstances. This innovation made DRL appealing in various applications, especially in safety-critical systems like healthcare, autonomous vehicles, and robotics[1].

The dynamism and sophistication of DRL also usher in myriad challenges. From output uncertainties to real-time decision-making ambiguities, the complexities of DRL systems often cast shadows that can hide potential vulnerabilities [2]. However, formal verification can be our guide[8]. As this review will explain, various verification strategies have been developed and implemented to our help, making sure that DRL systems work with optimal reliability. The significance of this lies in ensuring the accuracy of a model's predictions and ensuring that safety-critical systems, where inaccuracies can lead to catastrophic consequences, operate faultlessly.

The Survey highlights the challenges faced using Deep Reinforcement Learning and ensures decision-making. Some tools and methods are introduced that are made just for this purpose. Some issues might stop people from using these tools everywhere. Understanding these issues is essential because it tells us where more work is needed in DRL research. This paper can inspire more research by highlighting these problems and the missing pieces in current DRL checks.

Research Motivation: DRL has carved a significant niche in robotics and healthcare, instigating advancements in robotic cognition and operational efficiency. [3] Mentions how DRL optimizes robotic control systems, empowering robots with enhanced decision-making and adaptability. As robots permeate various sectors, the integration of DRL augments their ability to maneuver complex, unpredictable environments. However, an exigent issue - the escalating use of DRL in robots calls for the need for robust regulatory measures[4]. Ensuring that these advanced robotic systems operate within defined safety and ethical parameters.

Despite the use of DRL in robotics, a wide gap exists in the literature that systematically addresses the attendant challenges and proposes comprehensive solutions. There is a need for discussions about the practical, ethical, and safety challenges of DRL in real-world applications that are relatively sparse[4]. On the other hand, formal verification emerges as a potential tool to validate and assure the safety and performance of DRL applications. Nevertheless, the incorporation of formal verification is riddled with complexities, including the dynamic nature of DRL environments and the nuanced, multifaceted challenges of ensuring exhaustive verification in such settings [5].

This paper explores the landscape of deep reinforcement learning and its formal verification, starting with a foundational background in section 2. The research methodology used to accumulate the contents this paper highlights is mentioned in section 3. This sets the stage by

2693-9371/23/$31.00 ©2023 IEEE
DOI 10.1109/QRS-C60940.2023.00112

explaining the evolution and significance of DRL. The paper then transitions to the techniques integral to DRL in section 4 and the importance of formal verification discussed in section 5. Highlighted next are the challenges inherent to RL/DRL and the intricacies of its formal verification in section 6. Section 7 gives an overview of current tools and frameworks for DRL verification. The paper continues with a discussion and future work in section 8 and a conclusion in section 9.

2. Background

Reinforcement Learning is a sub-field of artificial intelligence that focuses on training machine learning models to make a sequence of decisions. Along with supervised and unsupervised learning, reinforcement learning is one of the three fundamental machine learning paradigms. The model learns to achieve a goal in a complex, uncertain environment by interacting with the environment and receiving feedback in the form of rewards or penalties. The RL model, also known as an agent, learns a policy - a mapping from states to actions - that maximizes the cumulative reward over time [6].

Deep Reinforcement Learning is used when dealing with high-dimensional, continuous action spaces, DRL integrates deep learning and reinforcement learning. DRL uses neural networks to approximate the reward function and/or the policy[7]. Two main types of DRL are Batch RL, where the agent learns from a fixed batch of experiences, and Online RL, where the agent learns while interacting with the environment.

Formal verification is a process used in software engineering to check whether a system meets a given set of specifications or properties. The main principle behind formal system analysis is to construct a computer-based mathematical model of the given scenario and formally verify, within a computer, that this model meets rigorous specifications of intended behavior[8].

Safety-Critical Systems are systems where precision is given the most importance. Formal verification is pivotal in automation endeavors involving safety-critical systems, particularly where valuable machinery is employed [9]. Various formal verification techniques consider the system's dynamic environment, accounting for factors such as moving targets [10]. Deep reinforcement learning and formal verification can be combined to create new frameworks that exploit their complementary strengths. This synergy offers a potentially effective way to raise the trustworthiness and security of AI-driven systems.

However, fully realizing such a framework will require significant additional research. The challenges include managing massive state and action spaces, adjusting to changing formal verification specifications, enhancing training processes, and providing necessary tools to construct environments that efficiently integrate standard formal verification approaches. This is especially important in standard-governed businesses where the adoption of models may need authorities' approval.

3. Research Methodology

3.1 Database and Search String

An extensive literature search was conducted utilizing reputed databases including Springer, IEEE Xplore, Arxiv, AAAI, and Science Direct. The search string employed was *((“Formal Verification” OR “Scalable Verification” OR ”Model Checking” OR “Formal Methods”) AND (“Deep Reinforcement Learning” OR “Reward Function Design” OR “Deep Learning” OR “Agent based Learning” OR “Reinforcement”))* to ensure the capture of a broad yet relevant set of literature.

3.2 Selection Criteria

Papers were evaluated based on title and abstract, ensuring relevance to the formal verification of RL systems. Included works provided case studies or detailed discussions on model checking and property specification. Recent papers that were Papers proposing different RL frameworks without a focus on formal verification were excluded. Emphasis was placed on including recent publications to ensure the incorporation of the latest insights, methodologies, and developments in the field of formal verification in RL systems.

3.3 Review Process

A set of pertinent questions guided the review process, probing into the nature of formal verification performed, the case studies used, identified challenges in formal verification, and its effectiveness in the context of RL systems.

3.4 Data Extraction

Data were organized using Excel sheets, categorizing extracted information on the effectiveness of formal verification, proposed strategies for safety assurance, the type of approach used for formal verification, and the challenges associated with integrating formal verification with deep learning or reinforcement learning.

3.5 Analysis

Papers were analyzed to extract diverse challenges and techniques, prioritizing those with high frequency and relevance. An additional exploration via Google searches supplemented the identification of prominent frameworks and tools for the verification of DRL networks. Each identified challenge and technique was rigorously evaluated to ensure its pertinence and contribution to the overarching narrative of the survey.

4. Techniques of Deep Reinforcement Learning

Deep Reinforcement Learning brings together the concepts from deep learning and reinforcement learning to build agents that can learn to make sequences of decisions by interacting with a complex environment. The introduction of deep learning to reinforcement learning has mainly been to cope with high dimensional input spaces, particularly those encountered in tasks like game playing, robotics, and autonomous driving. The following subsections highlight the foundational techniques in DRL.

4.1 Q-Learning with Neural Networks

At the heart of many DRL algorithms is the Q-learning algorithm. Q-learning seeks to learn the value of taking an action in a state, often represented as $Q(s, a)$. When combined with deep learning, neural networks are trained to approximate the Q-values, leading to the famous Deep Q-Network (DQN) algorithm [11].

4.2 Policy Gradient Methods

Unlike Q-learning, which learns the value of actions, policy gradient methods directly learn the policy function that maps states to actions. This method works by optimizing the policy in the direction that increases expected rewards [12]. Deep neural networks can be used to represent and optimize complex policy functions.

4.3 Actor-Critic Methods

Actor-critic methods combine the benefits of both value-based and policy-based methods. While the actor component is responsible for selecting actions, the critic evaluates those actions. The combination allows for a more stable and efficient learning process [13].

4.4 Experience Replay

To improve the stability and efficiency of DRL, experience replay stores past experiences in a replay buffer. The agent then samples mini-batches from this buffer to update the neural network, breaking the temporal correlations and reusing past experiences [14].

4.5 Exploration Strategies

Strategies like epsilon-greedy, softmax action selection, and upper confidence bound (UCB) methods balance the exploration and exploitation trade-off in DRL [6].

4.6 Target Networks

In algorithms like DQN, the rapid updates to Q-values can lead to oscillations or divergence. To tackle this, a separate network, the target network, is used to compute the target Q-values, which gets updated less frequently than the primary network [7].

4.7 Deep Deterministic Policy Gradient (DDPG)

DDPG is an off-policy algorithm that is particularly well-suited for continuous action spaces. It uses a deterministic policy gradient approach with concepts like target networks and experience replay [15].

These techniques form the building blocks of DRL. They have enabled agents to achieve superhuman performance in various tasks. However, integrating deep learning with reinforcement learning has also introduced new challenges, which we discuss in the subsequent sections.

5. Techniques for Verification

Deep Reinforcement Learning systems require rigorous verification methods to ensure reliability when applied to safety-critical domains. This section highlights several foundational techniques and strategies for formal verification in DRL systems.

5.1 Model Checking

Model Checking forms the crux of many verification procedures. It exhaustively explores all possible states of a system to ascertain if it meets a specified property.

- **Probabilistic Model Checking**: Evaluates the probabilities of certain behaviors or states being reached in systems with inherent randomness.
- **Non-Probabilistic Model Checking**: Assesses systems without accounting for randomness, ensuring deterministic behaviors align with specified properties.
- **Interval Analysis**: A method to provide bounds on uncertainties, especially useful when exact values are elusive or when dealing with systems having continuous variables [33].

Machine learning techniques are being leveraged to augment model-checking processes' capabilities significantly. [30].

5.2 Linear Temporal Logic (LTL)

LTL is a symbolic logic that permits assertions about the future of paths. For DRL, it can serve to model and verify temporal properties of RL agents, such as ensuring an agent eventually reaches a goal.

- **Responsibility-Sensitive Safety (RSS) Model**: A model emphasizing safety by defining a set of rules which, if followed, guarantees safe behavior [41].
- **Linear Temporal Logic (LTL) Modeling**: Uses LTL to represent and enforce traffic or safety rules, ensuring agents do not violate them[42].

5.3 Satisfiability Modulo Theories (SMT)

SMT extends the classical satisfiability problem by incorporating background theories like arithmetic. DRL can check the consistency of various system behaviors with their specifications [33], [36].

5.4 Linear Equation

A mathematical statement equates two expressions.

- **Automated Linear Equation Solving**: Used for verifying input-output behaviors of systems, ensuring that for given inputs, the outputs remain as expected [39].

5.5 Mixed Integer Linear Programming (MILP)

A method to solve optimization problems where some variables can only take integer values. For DRL systems:

- **Linear Programming and Relaxation Strategies**: Incorporates MILP to derive optimal strategies or behaviors under certain constraints [37], [38].

- **Deep Imitation Learning (DIL)**: DIL learns the behavior of an initial controller, inherently embedding safety constraints during the learning process. This approach ensures that the derived controller produces safe and expected output actions for any given input state, acting as a supplementary method for input-output behavior verification[32].

5.6 Exact Methods and Over-Approximation-Based Approaches

These strategies focus on the precision of verification.

- **Combination Strategies**: Merge both exact methods and over-approximation-based techniques to balance between precision and computational feasibility[36].
- **MDP and Automata Strategies**: Involves creating abstractions of the system dynamics using Markov Decision Processes (MDP) and then constructing automata for verification[35], [40].
- **Justified Speculative Control**: An automated approach leveraging predefined mapping rules to speculate on actions for certain states or conditions. Its rule-based nature ensures precision in predicting appropriate actions, streamlining the verification process [31], [34].

The selection of verification techniques largely depends on the specific requirements of the DRL system. These methods serve to identify potential flaws and fortify the confidence in deploying DRL in safety-critical applications.

6. Challenges

6.1 Challenges in RL/DRL Systems

Reinforcement Learning, though promising, confronts various challenges. The challenges outlined are intricately tied to design, verification, and application of reinforcement learning systems. These can be defined within reinforcement learning in Safety Critical Systems.

- **CD1 - Partial Observability:** In many real-world scenarios, an agent cannot observe the entire state of the environment, leading to partial observability. This makes it challenging to make optimal decisions based solely on current observations [16].
- **CD2 - Environment Modeling:** Accurately modeling the environment is crucial for many DRL algorithms, especially model-based approaches. Inaccurate models can lead to suboptimal or even dangerous actions [17].
- **CD3 - Complexity of DRL Models:** The deep neural networks used in DRL can become highly complex, making them computationally expensive and hard to interpret [18].
- **CD4 - Achieving Transparency in DRL:** Interpreting and understanding the decisions made by DRL agents is non-trivial due to the black-box nature of deep networks. This makes achieving transparency a challenge [19].
- **CD5 - Multi-Agent Coordination:** When multiple agents interact in a shared environment, coordinating their actions to achieve global objectives becomes challenging [20].
- **CD6 - Continuous Action Spaces:** Dealing with continuous actions, as opposed to discrete ones, complicates the optimization process and introduces challenges in policy representation [15].
- **CD7 - Sample Efficiency:** Training DRL agents often require a large number of samples. Reducing the number of samples needed without compromising performance is a significant challenge [21].
- **CD8 - Safe Exploration:** Ensuring that an agent explores the environment safely without causing harm or getting into unrecoverable states is crucial, especially in real-world scenarios [22].
- **CD9 - Reward Shaping:** Defining appropriate reward functions to guide the agent toward desired behavior can be subtle and challenging. Incorrect reward shaping can result in unintended behaviors [23].
- **CD10 - Exploration vs Exploitation Dilemma:** Agents must balance the act of exploring new strategies and exploiting known ones. Achieving this balance is a longstanding challenge in RL [6].
- **CD11 - Delayed Reward Problem:** Actions taken by agents may have consequences that manifest much later, making it challenging to associate actions with their outcomes [24].
- **CD12 - Credit Assignment Problem:** Determining which actions were responsible can be non-trivial when rewards or penalties are received, especially in scenarios with delayed rewards [7].
- **CD13 - Over-fitting to the Training Environment:** DRL agents can become too specialized in their training environments, performing poorly when exposed to slightly different scenarios [18].
- **CD14 - Validation of Safety Properties:** Ensuring that DRL agents adhere to safety constraints during training and deployment is a challenge, especially given the complexity of the models involved [25].
- **CD15 - Resilience to Adversarial Attacks:** Like other deep learning models, DRL agents can be vulnerable to adversarial attacks, where slight input perturbations can lead to drastically different and potentially unsafe actions [26].
- **CD16 - Assuring Generalization Capability:** Ensuring that a DRL agent can generalize its learned policy to new, unseen situations is crucial for many applications, especially those in dynamic environments [27].
- **CD17 - Handling Model-Data Mismatch:** Differences between the model's assumptions and real-world data can lead to performance degradation or unexpected behaviors in DRL agents [28].
- **CD18 - Real-Time Decision Making:** For applications that require real-time responses, like robotics or autonomous vehicles, DRL agents must make decisions within tight time constraints, adding another layer of complexity [29].

These challenges highlight the ongoing requirement for persistent exploration and advancement within the domain of RL and DRL. They also emphasize the necessity for implementing formal verification methodologies to protect the efficacy of RL systems.

6.2 Challenges of Formally Verifying a DRL System

Based on our survey, various researchers have explored and experimented with strategies in RL/DRL. Throughout their endeavors, they've shed light on several challenges. Though diverse across different domains, these challenges converge into some commonly observed themes. Here, we present a consolidated overview of these mentioned challenges.

- **CF1 - Resilience to Defense Mechanisms:** DRL systems must be resilient against various defense strategies, ensuring their proper functioning even when subjected to unforeseen disturbances or attacks [34], [38], [37], [53], [52].
- **CF2 - Balancing Search Guarantees with Computation Times:** Achieving optimal results often demands extensive searches. However, this needs to be balanced with computational efficiency to be practically viable [39], [36].
- **CF3 - Complexity:** The inherent complexity of DRL algorithms makes them challenging to analyze and verify[33], [38], [37], [50].
- **CF4 - Limited Computational Efficiency of SMT Solvers:** Satisfiability Modulo Theories (SMT) solvers are vital for verification. Still, their computational demands can sometimes be limiting for large-scale DRL systems [33], [38], [37].
- **CF5 - Non-linearity:** DRL models often possess non-linear characteristics that can complicate the verification process [37].
- **CF6 - Scalability:** As DRL systems grow and become more intricate, the verification processes must scale without an exponential increase in complexity [49], [33], [39], [52], [36].
- **CF7 - Integration:** Integrating DRL systems with other systems or platforms poses verification challenges, especially concerning compatibility and performance.
- **CF8 - Environmental Complexity:** The diverse and dynamic environments in which DRL systems operate add complexity to the verification task [54].
- **CF9 - Continuous State and Action Spaces:** DRL systems often work in continuous spaces, complicating the task of exhaustive verification [59].
- **CF10 - Formalism:** Establishing a formal structure or methodology for DRL systems is challenging, given their dynamic nature [57].
- **CF11 - Precision vs Scalability Trade-off:** As verification processes become more precise, they may become less scalable, and vice versa [36].
- **CF12 - Need for Robustness Verification:** It's essential to ensure that DRL systems are functionally correct and robust against various uncertainties [36], [36].
- **CF13 - Adversarial Attacks:** DRL systems can be targets for adversarial attacks, where slight input perturbations can lead to significant deviations in behavior [37].
- **CF14 - Training Process Uncertainties:** The uncertainties inherent in the training process can lead to unexpected behaviors during real-world deployments [55].
- **CF15 - Evaluating Trained Policies:** Properly evaluating and verifying the policies learned by DRL agents is crucial for safety-critical applications [58].
- **CF16 - State Space Explosion:** The exponential growth in the state space, especially in complex environments, makes exhaustive verification nearly impossible [40], [52].
- **CF17 - Learning-based Synthesis:** Integrating learning processes into system synthesis adds another layer of complexity to the verification task [39].
- **CF18 - Verification of Multiple Networks:** As DRL systems might consist of multiple neural networks, verifying their collective behavior becomes challenging [49].

7. Frameworks and Tools for Verification

As Deep Reinforcement Learning continues to be paramount in safety-critical domains, numerous frameworks and tools have been developed to address its challenges. These tools are geared towards ensuring robustness, safety, and verification in DRL applications.

7.1 COOL-MC

COOL-MC is a comprehensive tool integrating reinforcement learning and model checking, allowing for an intertwined verification and learning process. This ensures that the RL models align with predefined specifications throughout the learning phase [47].

7.2 whiRL 2.0

The whiRL 2.0 tool capitalizes on techniques such as k-induction and employs semi-automated invariant inference, ensuring that the RL models' behaviors remain within desired boundaries [49].

7.3 Reluplex

Reluplex is a specialized algorithm for verifying neural networks that employ the ReLU (Rectified Linear Unit) activation function. This makes it particularly relevant for networks used in various DRL systems, ensuring their behaviors align with desired safety specifications [43].

7.4 TRAINIFY

TRAINIFY presents a unique combination of CEGAR (Counterexample-Guided Abstraction Refinement) driven training with a verification framework. This integrated approach ensures that DRL systems are trained optimally and verified for safety concurrently [48].

7.5 Safe Reinforcement Learning for CPSs

This framework integrates formal modeling and verification into the RL process for Cyber-Physical Systems (CPSs). Such integration ensures that RL models catering to CPSs are not just optimal but also verifiably safe [50].

7.6 NEURODIFF

NEURODIFF focuses on the differential verification of neural networks, utilizing fine-grained approximation techniques. This approach guarantees that slight changes or perturbations to the neural network don't cause undesired behaviors [52].

7.7 Task Space Approach

This method provides provable safety guarantees for deep reinforcement learning when applied to robotic manipulations in human-centric environments. Ensuring safety in such scenarios is crucial, and the task space approach tailors its verification techniques accordingly [51].

7.8 VerifAI

VerifAI is a comprehensive tool dedicated to formally designing and scrutinizing AI-driven systems. Its capability to tackle the complexities of DRL systems brings it to the forefront of verification tools in the AI safety domain. [44]

7.9 Sherlock

Sherlock is engineered explicitly for the safety verification of deep neural networks. Its robust framework guarantees that these networks, pivotal in numerous DRL applications, function within safe parameters, eliminating unforeseen adverse behaviors. [45]

7.10 Neurify

Neurify introduces an innovative neural network verification approach, banking on a linear approximation strategy. Such an approach ensures the robustness and safety of neural networks in dynamic DRL environments. [46]

8. Discussion and Future Direction

The domain of Deep Reinforcement Learning melded with formal verification, stands at an exciting crossroads of challenges and potential breakthroughs. There have been survey papers that highlighted research in the area of DRL and Formal verification. Still, this paper stands out as it reflects the new advancements made in the field and introduces tools and frameworks to automate some of the mundane processes [60]. The paper also discusses the challenges of using reinforcement learning and integrating the Formal Verification processes with DRL systems.

While academic advancements in DRL and its verification are laudable, the real litmus test lies in its application within practical scenarios. Gaining insight into the tangible challenges when deploying these strategies in real-world systems can shape our academic pursuits. What best practices are industry professionals leaning toward? Are there any in-field methodologies that overshadow others in terms of Capability? These are the pressing questions that could redefine the academia-industry connection.

Implications of this study extend to both the academic and industrial realms. The identified challenges and proposed solutions can inform the development of more refined DRL applications, leading to safer, more reliable systems. Moreover, our findings can catalyze collaborative efforts in the fields of Cybersecurity, IOT, Health Care, and Manufacturing. These strategies hold the potential to enhance the safety protocols in robotics and could significantly pave the way for a more controlled and systematic development of artificial intelligence applications.

A Systematic Literature Review (SLR) stands out as a valuable tool to uncover the state of DRL verification within operational setups. Such an endeavor can shed light on current industry norms, tangible challenges, and adopted resolutions. By fostering dialogues with professionals in the field and scouring academic contributions, a nuanced perspective can be pieced together for DRL verification's real and theoretical facets.

9. Conclusion

In the rapidly evolving domain of Deep Reinforcement Learning (DRL), the need for adequate formal verification becomes ever more pressing. This paper provided a comprehensive exploration of DRL and the challenges associated with its verification, offering insights into its evolution, techniques, and applications across various domains. Key challenges, from environmental complexities to the intricacies of evaluating trained policies, were brought to the fore, emphasizing the multifaceted nature of DRL systems.

A curated collection of tools and frameworks vital for verifying DRL was showcased, offering a lens into the resources available to researchers and industry professionals. The increasing incorporation of DRL in safety-critical applications makes these tools indispensable.

In conclusion, while the challenges in formal verifying DRL systems are substantial, they are not invincible. It's important to note that the applicability of the challenges and tools mentioned is specific to the domains and use cases cited herein. Readers are encouraged to carefully evaluate the relevance of these findings based on the nuances of their unique situations and challenges. Through a combination of research findings and industry practices bolstered by an empirical review, we can navigate these barriers. With its depth and breadth, this paper is poised to significantly shape the decision-making perspectives of its audience, promoting well-informed choices in the intricate world of DRL and its verification.

References

[1] Zhu, Zhuangdi, et al. "Transfer learning in deep reinforcement learning: A survey." IEEE Transactions on Pattern Analysis and Machine Intelligence (2023).

[2] Li, Yuxi. "Deep reinforcement learning: An overview." arXiv preprint arXiv:1701.07274 (2017).

[3] Kober, Jens, J. Andrew Bagnell, and Jan Peters. "Reinforcement learning in robotics: A survey." The International Journal of Robotics Research 32.11 (2013): 1238-1274.

[4] Tai, Lei, et al. "A survey of deep network solutions for learning control in robotics: From reinforcement to imitation." arXiv preprint arXiv:1612.07139 (2016).

[5] Alshiekh, Mohammed, et al. "Safe reinforcement learning via shielding." Proceedings of the AAAI conference on artificial intelligence. Vol. 32. No. 1. 2018.

[6] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," MIT press, 2018.

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., “Human-level control through deep reinforcement learning,” Nature, vol. 518, no. 7540, pp. 529–533, 2015.

[8] O. Hasan and S. Tahar, “Formal verification methods,” in Encyclopedia of Information Science and Technology, Third Edition, IGI Global, 2015, pp. 7162–7170.

[9] Abdul Basit Ur Rahim, M., Ahsan Ur Raheem, M., Sohail, M.K. et al. Formal verification of reconfigurable systems. Soft Comput (2023). https://doi.org/10.1007/s00500-023-08272-z

[10] M. Abdul Basit Ur Rahim, Q. Duan and E. Al-Shaer, ”A Formal Analysis of Moving Target Defense,” 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 2020, pp. 1802-1807, doi: 10.1109/COMPSAC48688.2020.00050.

[11] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., ”Playing atari with deep reinforcement learning,” *arXiv preprint arXiv:1312.5602*, 2013.

[12] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, ”Policy gradient methods for reinforcement learning with function approximation,” in *Advances in neural information processing systems*, 2000, pp. 1057-1063.

[13] V. R. Konda and J. N. Tsitsiklis, ”Actor-critic algorithms,” in *Advances in neural information processing systems*, 2000, pp. 1008-1014.

[14] L.-J. Lin, ”Self-improving reactive agents based on reinforcement learning, planning and teaching,” *Machine learning*, vol. 8, no. 3-4, pp. 293-321, 1992.

[15] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, ”Continuous control with deep reinforcement learning,” *arXiv preprint arXiv:1509.02971*, 201

[16] M. J. Hausknecht and P. Stone, ”Deep Recurrent Q-Learning for Partially Observable MDPs,” in Proceedings of the 2015 AAAI Fall Symposium Series, 2015.

[17] M. Deisenroth, G. Neumann, and J. Peters, ”A Survey on Policy Search for Robotics,” in Robotics and Autonomous Systems, vol. 54, 2011, pp. 388-402.

[18] A. Zhang, Z. Cui, and J. Xu, ”A survey on Deep Reinforcement Learning,” in Journal of Artificial Intelligence Research, vol. 69, 2019, pp. 2075-2101.

[19] M. T. Ribeiro, S. Singh, and C. Guestrin, ”Why Should I Trust You?” Explaining the Predictions of Any Classifier,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

[20] R. Lowe, Y. Wu, A. Tamar, J. Harb, O. P. Abbeel, and I. Mordatch, ”Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments,” in Advances in Neural Information Processing Systems, 2017.

[21] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, ”Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in International Conference on Machine Learning, 2018.

[22] J. Garcia and F. Fernandez, ”A Comprehensive Survey on Safe Reinforcement Learning,” Journal of Machine Learning Research, vol. 16, no. 1, pp. 1437-1480, 2015.

[23] A. Y. Ng, D. Harada, and S. Russell, ”Policy invariance under reward transformations: Theory and application to reward shaping,” in Proceedings of the Sixteenth International Conference on Machine Learning, 1999.

[24] M. C. Mozer, ”A focused backpropagation algorithm for temporal,” in Proceedings of the 1998 connectionist models summer school, 1998.

[25] A. Wachi, Y. Sui, Y. Yue, and M. Ono, ”Safe exploration and optimization of constrained MDPs using Gaussian processes,” in Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[26] X. Huang, M. Kwiatkowska, S. Wang, and M. Wu, ”Safety verification of deep neural networks,” in International Conference on Computer Aided Verification, 2017.

[27] K. Cobbe, O. Klimov, C. Hesse, T. Kim, and J. Schulman, ”Quantifying generalization in reinforcement learning,” arXiv preprint arXiv:1812.02341, 2018.

[28] D. Kang, Y. Sun, D. Beatson, and R. Tomioka, ”Generalization and Equilibrium in Generative Adversarial Nets (GANs),” in International Conference on Machine Learning, 2019.

[29] L. Tai, J. Zhang, M. Liu, and W. Burgard, ”A virtual-reality environment for robot-assisted human walking training,” in Robotics Research, 2017.

[30] A. Besbas, L. Belaiche, S. Slatnia, L. Kahloul and M. Khalgui, ”Machine learning Solutions to Model Checking: A Brief Literature Review,” 2022 International Symposium on iNnovative Informatics of Biskra (ISNIB), Biskra, Algeria, 2022, pp. 1-4, doi: 10.1109/ISNIB57382.2022.10076160.

[31] N. Fulton and A. Platzer, “Safe Reinforcement Learning via Formal Methods: Toward Safe Control Through Proof and Learning”, AAAI, vol. 32, no. 1, Apr. 2018.

[32] X. He, ”Building Safe and Stable DNN Controllers using Deep Reinforcement Learning and Deep Imitation Learning,” 2022 IEEE 22nd International Conference on Software Quality, Reliability and Security (QRS), Guangzhou, China, 2022, pp. 775-784, doi: 10.1109/QRS57517.2022.00083.

[33] D. Corsi, E. Marchesini, A. Farinelli and P. Fiorini, ”Formal Verification for Safe Deep Reinforcement Learning in Trajectory Generation,” 2020 Fourth IEEE International Conference on Robotic Computing (IRC), Taichung, Taiwan, 2020, pp. 352-359, doi: 10.1109/IRC.2020.00062.

[34] A. Pore et al., ”Safe Reinforcement Learning using Formal Verification for Tissue Retraction in Autonomous Robotic-Assisted Surgery,” 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),

Prague, Czech Republic, 2021, pp. 4025-4031, doi: 10.1109/IROS51168.2021.9636175.

[35] A. Nikou, A. Mujumdar, V. Sundararajan, M. Orlic and A. V. Feljan, "Safe RAN control: A Symbolic Reinforcement Learning Approach," 2022 IEEE 17th International Conference on Control & Automation (ICCA), Naples, Italy, 2022, pp. 332-337, doi: 10.1109/ICCA54724.2022.9831850.

[36] A. Baninajjar, K. Hosseini, A. Rezine and A. Aminifar, "SafeDeep: A Scalable Robustness Verification Framework for Deep Neural Networks," ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 2023, pp. 1-5, doi: 10.1109/ICASSP49357.2023.10097028.

[37] A. Dethise, M. Canini and N. Narodytska, "Analyzing Learning-Based Networked Systems with Formal Verification," IEEE INFOCOM 2021 - IEEE Conference on Computer Communications, Vancouver, BC, Canada, 2021, pp. 1-10, doi: 10.1109/INFOCOM42981.2021.9488898.

[38] X. Wang, J. Peng, S. Li and B. Li, "Formal Reachability Analysis for Multi-Agent Reinforcement Learning Systems," in IEEE Access, vol. 9, pp. 45812-45821, 2021, doi: 10.1109/ACCESS.2021.3060156.

[39] G. Hains, A. Jakobsson and Y. Khmelevsky, "Towards formal methods and software engineering for deep learning: Security, safety and productivity for dl systems development," 2018 Annual IEEE International Systems Conference (SysCon), Vancouver, BC, Canada, 2018, pp. 1-5, doi: 10.1109/SYSCON.2018.8369576.

[40] Yamagata, Yoriyuki, et al. "Falsification of cyber-physical systems using deep reinforcement learning." IEEE Transactions on Software Engineering 47.12 (2020): 2823-2840.

[41] Rong, Jikun, and Nan Luan. "Safe reinforcement learning with policy-guided planning for autonomous driving." 2020 IEEE International Conference on Mechatronics and Automation (ICMA). IEEE, 2020.

[42] M. Hasanbeig, Y. Kantaros, A. Abate, D. Kroening, G. J. Pappas and I. Lee, "Reinforcement Learning for Temporal Logic Control Synthesis with Probabilistic Satisfaction Guarantees," 2019 IEEE 58th Conference on Decision and Control (CDC), Nice, France, 2019, pp. 5338-5343, doi: 10.1109/CDC40024.2019.9028919.

[43] G. Katz, C. Barrett, D. L. Dill, K. Julian, and M. J. Kochenderfer, "Reluplex: An efficient SMT solver for verifying deep neural networks," in Proc. Computer Aided Verification. Springer, 2017, pp. 97-117.

[44] D. Fremont, X. Yue, S. A. Seshia, and A. D. Dragan, "VerifAI: A toolkit for the formal design and analysis of artificial intelligence systems," in Proc. International Conference on Computer Aided Verification. Springer, 2019, pp. 68-76.

[45] W. Xiang, H. Dutta, and N. M. Kochenderfer, "Sherlock: Efficient and sound safety verification for deep neural networks," in Proc. Workshop on Computer Safety, Reliability, and Security. Springer, 2018, pp. 18-32.

[46] W. Xiang, H. Dutta, and N. M. Kochenderfer, "Neurify: An efficient and robust tool for verifying feedforward neural networks," in Proc. International Conference on Automated Software Engineering. IEEE, 2020, pp. 114-125.

[47] Gross, Dennis, et al. "COOL-MC: a comprehensive tool for reinforcement learning and model checking." International Symposium on Dependable Software Engineering: Theories, Tools, and Applications. Cham: Springer Nature Switzerland, 2022.

[48] Jin, Peng, et al. "Trainify: A cegar-driven training and verification framework for safe deep reinforcement learning." International Conference on Computer Aided Verification. Cham: Springer International Publishing, 2022.

[49] Amir, Guy, Michael Schapira, and Guy Katz. "Towards scalable verification of deep reinforcement learning." 2021 formal methods in computer-aided design (FMCAD). IEEE, 2021.

[50] Yang, Chenchen, et al. "Safe Reinforcement Learning for CPSs via Formal Modeling and Verification." 2021 International Joint Conference on Neural Networks (IJCNN). IEEE, 2021.

[51] Thumm, Jakob, and Matthias Althoff. "Provably safe deep reinforcement learning for robotic manipulation in human environments." 2022 International Conference on Robotics and Automation (ICRA). IEEE, 2022.

[52] Paulsen, Brandon, et al. "Neurodiff: scalable differential verification of neural networks using fine-grained approximation." Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering. 2020.

[53] P. S. N. Mindom, A. Nikanjam, F. Khomh, and J. Mullins, "On Assessing The Safety of Reinforcement Learning algorithms Using Formal Methods," in *CoRR*, vol. abs/2111.04865, 2021.

[54] Mirchevska, Branka, et al. "High-level decision making for safe and reasonable autonomous lane changing using reinforcement learning." 2018 21st International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2018.

[55] M. Everett, B. Lütjens and J. P. How, "Certifiable Robustness to Adversarial State Uncertainty in Deep Reinforcement Learning," in IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 9, pp. 4184-4198, Sept. 2022, doi: 10.1109/TNNLS.2021.3056046.

[56] Garg, Arpit, et al. "Comparison of deep reinforcement learning policies to formal methods for moving obstacle avoidance." 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2019.

[57] Capobianco, Giovanni, et al. "A Methodology for Real-Time Data Verification Exploiting Deep Learning and Model Checking." 2019 IEEE International Conference on Big Data (Big Data). IEEE, 2019.

[58] Marchesini, Enrico, Davide Corsi, and Alessandro Farinelli. "Benchmarking safe deep reinforcement learning

in aquatic navigation." 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2021.

[59] Everett, Michael, Björn Lütjens, and Jonathan P. How. "Certifiable robustness to adversarial state uncertainty in deep reinforcement learning." IEEE Transactions on Neural Networks and Learning Systems 33.9 (2021): 4184-4198.

[60] Landers, Matthew, and Afsaneh Doryab. "Deep Reinforcement Learning Verification: A Survey." ACM Computing Surveys (2023).