Shielding Atari Games with Bounded Prescience

Extended Abstract

Mirco Giacobbe
University of Oxford
mirco.giacobbe@cs.ox.ac.uk

Mohammadhosein Hasanbeig
University of Oxford
hosein.hasanbeig@cs.ox.ac.uk

Daniel Kroening*
Amazon, Inc.
daniel.kroening@magd.ox.ac.uk

Hjalmar Wijk
University of Oxford
hannes.hjalmar.wijk@cs.ox.ac.uk

ABSTRACT

We present the first explicit-state method for analysing and ensuring the safety of DRL agents for Atari games. Our method only requires access to the emulator. We give a suite of 42 properties that characterise "safe behaviour" for 31 games. We evaluate the safety of the best available DRL agents which, as our experiments show, violate most of our properties. We propose a countermeasure that implements shielding using bounded explicit-state exploration. Our method improved their overall safety, producing the safest DRL agents for Atari games currently available.

KEYWORDS

Safe AI; Deep Reinforcement Learning; Atari Games

ACM Reference Format:
Mirco Giacobbe, Mohammadhosein Hasanbeig, Daniel Kroening, and Hjalmar Wijk. 2021. Shielding Atari Games with Bounded Prescience: Extended Abstract. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

Deep reinforcement learning (DRL) combines neural networks with reinforcement learning (RL) and, capitalising on recent advances in both technologies, has been successfully employed in many areas of artificial intelligence, from playing games against humans to controlling robots in the physical world [3, 19, 38, 44]. A setup of this kind consists of an agent and a neural network that automatically learns to interact with the environment by maximizing rewards received as consequence of its actions [13, 23]. DRL has demonstrated super-human capabilities in numerous applications, notably, the game of Go [38], and is now used in safety-critical domains such as autonomous driving [27]. While DRL agents perform well most of the time, the question of whether unsafe behaviour may occur in corner cases is an open problem. Safety analysis answers the question of whether the environment can possibly steer the system into an undesirable state or, dually, whether the agent can guarantee that the system remains within a set of safe states (an invariant) in which nothing bad happens [15, 20, 31]. We discuss the safety of popular DRL methods for one of the most challenging benchmarks: the Atari 2600 console games.

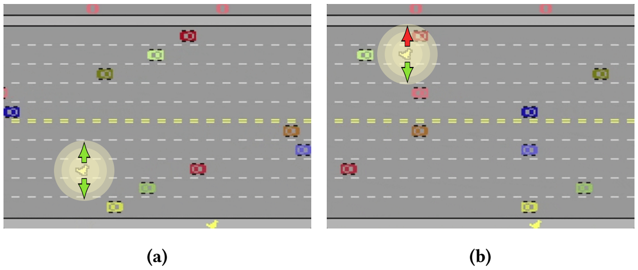
(a) (b)

Figure 1: The effect of a bounded-prescience shield on game `Freeway`. In (a), both actions 'up' and 'down' are safe thus allowed; in (b), action 'up' is unsafe thus blocked by the shield.

Games for the classic Atari 2600 console feature low-resolution graphics and small memory footprints. They are simple when compared with contemporary games, yet offer a broad variety of scenarios including many that are difficult for modern AI [9, 32, 34, 41]. Macroscopically, diversity in the game mechanics challenges the generality of the machine learning method; microscopically, diversity in the outcome of multiple identical plays, i.e., the *non-determinism* in the game, challenges the robustness of the trained agent. Many Atari games exploit variations in the response time of the human player for differentiating runs. The Arcade Learning Environment (ALE) creates this diversity by randomly injecting no-ops, skipping frames, or repeating agent actions [21, 32]. On one hand, this prevents overfitting the agent but, on the other hand, implies that there is no guarantee that an agent works all of the time—the scores that we use to rank training methods are averages. Agents are trained for strong average-case performance.

The application of DRL in safety-critical applications, by contrast, requires worst-case guarantees, and we expect a safe agent to maintain *safety invariants*. To evaluate whether or not state-of-the-art DRL delivers safe agents we specify a collection of properties that intuitively characterize safe behaviour for a variety of games, ranging from generic properties such as "don't lose lives" to game-specific ones such as avoiding particular obstacles. Figure 1 illustrates the property "duck avoids cars" in the game Freeway. In the scenario in Fig. 1a this property is maintained regardless of the action chosen by the agent whereas the scenario given in Fig. 1b offers the possibility of violating it. We conjecture that satisfying our properties is beneficial for achieving a high score, and therefore study whether

---

[*] The work reported in this paper was done prior to joining Amazon.

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

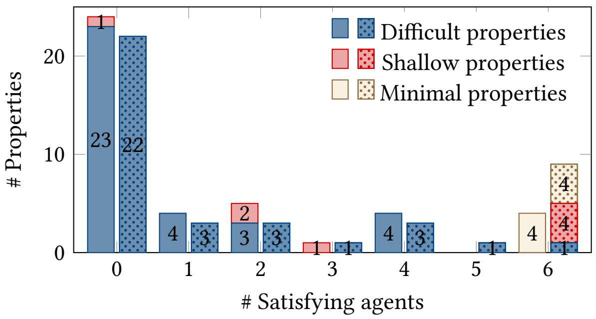

Figure 2: Properties grouped by number of satisfying agents before (w/o dots) and after BPS (with dots).

neural agents trained using best-of-class DRL methods learn to satisfy these invariants.

The safety of DRL has been studied from the perspective of verification, which determines whether a trained agent is safe as-is, and that of synthesis, which alters the learning or the inference processes in order to obtain a safe-by-construction agent [2, 5, 6, 8, 15]. Verification methods for neural agents have borrowed from constraint satisfaction or abstract interpretation [10, 16, 18, 24, 26, 39, 42, 43]. These symbolic approaches either reason about the safety of neural networks in isolation, e.g., vulnerability to adversarial attacks [40], or require a symbolic representation of the environment; unfortunately, these are unsuitable to Atari games because their mechanics are hidden inside the Stella emulator (i.e., the core of ALE). We circumvent this limitation by adopting an *explicit-state* verification strategy [12] that only requires access to the emulator.

Our method explicitly enumerates all traces induced by the agent after every non-deterministic initialisation of the game. Meanwhile, it labels all visited states as safe or unsafe using custom labelling functions that observe lives counts, rewards, and also the generated screen frames. We specified labelling functions for 42 non-trivial properties for 31 games. We evaluated the safety w.r.t. our properties on 6 agents trained using different technologies, i.e., A3C [33], DQN [35], IQN [14], and Rainbow [22]. As seen in Fig. 2 all agents violate 24 of our properties, whereas only the 4 *minimal* properties (properties satisfied more than half the time by random agents) are satisfied by all. Surprisingly, properties that are intuitively difficult for humans, e.g., “don’t die”, are satisfied by some agents, whereas many of the *shallow* properties which require no planning or foresight (e.g. “don’t walk out of bounds”) are violated.

To improve the overall safety of DRL agents w.r.t. our properties, we build *shields* using our explicit-state labelling and exploration technique. Ensuring safety amounts to constraining the traces of the system to those that are admissible by the safety property. Methods that act on the training phase modify the optimization criterion or the exploration process in order to obtain neural agents that naturally act safely [15]. Methods of this kind typically require information about the environment, e.g., in the form of teacher advice [37]. To the best of our knowledge, naturally safe agents have never been trained for Atari games. On the other side of the spectrum, shielding enables the option of fixing unsafe agents at inference phase only, introducing a third actor—the shield—that takes over control when necessary and with minimal interference [2]. A shield leverages the fact that safety properties are usually easy to satisfy, in contrast to the main objective of the task.

Shields formally guarantee that a model environment satisfies a safety property, regardless of the agent’s actions. Shielding has been applied to models defined as finite state machines, timed automata, dynamical systems, and multi-agent systems [2, 4, 7, 28, 30, 46]. Unfortunately, complete models of the environment are not always available, and this is also the problem for the Atari games. To overcome this limitation, shields are usually computed over an abstract model that is learned from samples of environment behaviour [1, 11, 25]. However, this has not been applied to the Atari games. We investigate the benefit of shielding for the Atari games using an arguably simpler approach: we shield the agents from taking actions that lead to unsafe outcomes within some bounded foresight of the future, which we obtain using explicit-state exploration. We thus study a form of shielding that acquires knowledge about the environment online, while it runs [29, 36].

The idea of a bounded search from the current state is commonplace. Like a rudimentary chess-playing computer, our method considers every combination of moves ahead of time—up to some bound—before taking an action [45]. We augment agents with *bounded-prescience shields* (BPSs) which, during execution, restrict the admissible actions to those that are necessarily safe within this prescience bound. At every step, a BPS enumerates all traces from the current state for a bounded number of extra steps and labels each of them as safe or unsafe; then, it invokes the agent and chooses the next action whose traces are all labelled as safe and whose agent score is the highest. As seen in Fig. 2, our method ensured satisfaction of shallow properties for all agents. Notably, it also fixed some properties that we consider difficult and that were satisfied by most but not all non-deterministic executions using the original agent.

Summarising, our contribution is threefold. First, we enrich the Atari games with the first comprehensive library of safety specifications. Second, we implement an explicit-state safety checker for the Arcade Learning Environment and discover that current DRL algorithms consistently violate most of our safety properties. Third, we implement a shielding method that, by exploiting a bounded foresight of the future, improves the safety of existing agents w.r.t. a set of simple yet critical properties, without interfering with their main objective. To the best of our knowledge, our method has produced the safest DRL agents for Atari games currently available.

The full version of this paper is available on arXiv [17]. The implementation and the experimental setup are on GitHub[1].

ACKNOWLEDGMENTS

This work is in part supported by UK NCSC, the HICLASS project (113213), a partnership between the Aerospace Technology Institute (ATI), Department for Business, Energy & Industrial Strategy (BEIS) and Innovate UK, and by the Future of Humanity Institute, Oxford.

---

[1] https://github.com/HjalmarWijk/bounded-prescience

REFERENCES

[1] Parand Alizadeh Alamdari, Guy Avni, Thomas A. Henzinger, and Anna Lukina. 2020. Formal Methods with a Touch of Magic. In *FMCAD*. IEEE, 138–147.

[2] Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. 2018. Safe Reinforcement Learning via Shielding. In *AAAI*. AAAI Press, 2669–2678.

[3] Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. 2017. Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Process. Mag.* 34, 6 (2017), 26–38.

[4] Osbert Bastani. 2020. Safe Reinforcement Learning with Nonlinear Dynamics via Model Predictive Shielding. *CoRR* abs/1905.10691 (2020).

[5] Roderick Bloem, Krishnendu Chatterjee, Karin Greimel, Thomas A. Henzinger, Georg Hofferek, Barbara Jobstmann, Bettina Könighofer, and Robert Könighofer. 2014. Synthesizing robust systems. *Acta Informatica* 51, 3-4 (2014), 193–220.

[6] Roderick Bloem, Krishnendu Chatterjee, Thomas A. Henzinger, and Barbara Jobstmann. 2009. Better Quality in Synthesis through Quantitative Objectives. In *CAV (LNCS, Vol. 5643)*. Springer, 140–156.

[7] Roderick Bloem, Peter Gjøl Jensen, Bettina Könighofer, Kim Guldstrand Larsen, Florian Lorber, and Alexander Palmisano. 2020. It's Time to Play Safe: Shield Synthesis for Timed Systems. *CoRR* abs/2006.16688 (2020).

[8] Roderick Bloem, Barbara Jobstmann, Nir Piterman, Amir Pnueli, and Yaniv Sa'ar. 2012. Synthesis of Reactive(1) designs. *J. Comput. Syst. Sci.* 78, 3 (2012), 911–938.

[9] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. OpenAI Gym. *CoRR* abs/1606.01540 (2016).

[10] Rudy Bunel, Ilker Turkaslan, Philip H. S. Torr, Pushmeet Kohli, and Pawan Kumar Mudigonda. 2018. A Unified View of Piecewise Linear Neural Network Verification. In *NeurIPS*. 4795–4804.

[11] Steven Carr, Nils Jansen, and Ufuk Topcu. 2020. Verifiable RNN-Based Policies for POMDPs Under Temporal Logic Constraints. In *IJCAI*. ijcai.org, 4121–4127.

[12] Edmund M. Clarke, Orna Grumberg, Daniel Kroening, Doron Peled, and Helmut Veith. 2018. *Model Checking, Second Edition.* MIT Press.

[13] Balázs Csanád Csáji. 2001. *Approximation with Artificial Neural Networks.* Master's thesis. Faculty of Sciences, Eötvös Loránd University, Hungary.

[14] Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. 2018. Implicit quantile networks for distributional reinforcement learning. *arXiv preprint:1806.06923* (2018).

[15] Javier García and Fernando Fernández. 2015. A Comprehensive Survey on Safe Reinforcement Learning. *J. Mach. Learn. Res.* 16, 1 (Jan. 2015), 1437–1480.

[16] Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri, and Martin T. Vechev. 2018. AI2: Safety and Robustness Certification of Neural Networks with Abstract Interpretation. In *IEEE Symposium on Security and Privacy*. IEEE Computer Society, 3–18.

[17] Mirco Giacobbe, Mohammadhosein Hasanbeig, Daniel Kroening, and Hjalmar Wijk. 2021. Shielding Atari Games with Bounded Prescience. *CoRR* abs/2101.08153 (2021).

[18] Sumathi Gokulanathan, Alexander Feldsher, Adi Malca, Clark W. Barrett, and Guy Katz. 2019. Simplifying Neural Networks with the Marabou Verification Engine. *CoRR* abs/1910.12396 (2019).

[19] Shixiang Gu, Ethan Holly, Timothy P. Lillicrap, and Sergey Levine. 2017. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. *2017 IEEE International Conference on Robotics and Automation (ICRA)* (2017), 3389–3396.

[20] Mohammadhosein Hasanbeig, Alessandro Abate, and Daniel Kroening. 2020. Cautious Reinforcement Learning with Logical Constraints. In *AAMAS*. International Foundation for Autonomous Agents and Multiagent Systems, 483–491.

[21] Matthew J. Hausknecht and Peter Stone. 2015. The Impact of Determinism on Learning Atari 2600 Games. In *AAAI Workshop: Learning for General Competency in Video Games*.

[22] Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. 2018. Rainbow: Combining Improvements in Deep Reinforcement Learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[23] Kurt Hornik. 1991. Approximation capabilities of multilayer feedforward networks. *Neural Networks* 4, 2 (1991), 251–257.

[24] Xiaowei Huang, Marta Kwiatkowska, Sen Wang, and Min Wu. 2017. Safety Verification of Deep Neural Networks. In *CAV (1) (LNCS, Vol. 10426)*. Springer, 3–29.

[25] Nils Jansen, Bettina Könighofer, Sebastian Junges, Alex Serban, and Roderick Bloem. 2020. Safe Reinforcement Learning Using Probabilistic Shields. In *Concurrency Theory (CONCUR)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

[26] Guy Katz, Clark W. Barrett, David L. Dill, Kyle Julian, and Mykel J. Kochenderfer. 2017. Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks. In *CAV (1) (LNCS, Vol. 10426)*. Springer, 97–117.

[27] B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A. Al Sallab, Senthil Yogamani, and Patrick Pérez. 2020. Deep Reinforcement Learning for Autonomous Driving: A Survey. arXiv:2002.00444 [cs.LG]

[28] Bettina Könighofer, Florian Lorber, Nils Jansen, and Roderick Bloem. 2020. Shield Synthesis for Reinforcement Learning. In *ISoLA (1) (LNCS, Vol. 12476)*. Springer, 290–306.

[29] Bettina Könighofer, Julian Rudolf, Alexander Palmisano, Martin Tappler, and Roderick Bloem. 2020. Online Shielding for Stochastic Systems. *CoRR* abs/2012.09539 (2020).

[30] Shuo Li and Osbert Bastani. 2020. Robust Model Predictive Shielding for Safe Reinforcement Learning with Stochastic Dynamics. In *ICRA*. IEEE, 7166–7172.

[31] Matt Luckcuck, Marie Farrell, Louise A. Dennis, Clare Dixon, and Michael Fisher. 2019. Formal Specification and Verification of Autonomous Robotic Systems: A Survey. *ACM Comput. Surv.* 52, 5 (2019), 100:1–100:41.

[32] Marlos C. Machado, Marc G. Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. 2018. Revisiting the Arcade Learning Environment: Evaluation Protocols and Open Problems for General Agents. *J. Artif. Int. Res.* 61, 1 (Jan. 2018), 523–562.

[33] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*. 1928–1937.

[34] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. 2013. Playing Atari with Deep Reinforcement Learning. *CoRR* abs/1312.5602 (2013).

[35] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529–533.

[36] Stefan Pranger, Bettina Könighofer, Martin Tappler, Martin Deixelberger, Nils Jansen, and Roderick Bloem. 2020. Adaptive Shielding under Uncertainty. *CoRR* abs/2010.03842 (2020).

[37] William Saunders, Girish Sastry, Andreas Stuhlmüller, and Owain Evans. 2018. Trial without Error: Towards Safe Reinforcement Learning via Human Intervention. In *AAMAS*. ACM, 2067–2069.

[38] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy P. Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. 2017. Mastering the game of Go without human knowledge. *Nat.* 550, 7676 (2017), 354–359.

[39] Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin T. Vechev. 2019. An abstract domain for certifying neural networks. *Proc. ACM Program. Lang.* 3, POPL (2019), 41:1–41:30.

[40] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *ICLR (Poster)*.

[41] Marin Toromanoff, Émilie Wirbel, and Fabien Moutarde. 2019. Is Deep Reinforcement Learning Really Superhuman on Atari? *CoRR* abs/1908.04683 (2019).

[42] Hoang-Dung Tran, Stanley Bak, Weiming Xiang, and Taylor T. Johnson. 2020. Verification of Deep Convolutional Neural Networks Using ImageStars. In *CAV (1) (LNCS, Vol. 12224)*. Springer, 18–42.

[43] Hoang-Dung Tran, Xiaodong Yang, Diego Manzanas Lopez, Patrick Musau, Luan Viet Nguyen, Weiming Xiang, Stanley Bak, and Taylor T. Johnson. 2020. NNV: The Neural Network Verification Tool for Deep Neural Networks and Learning-Enabled Cyber-Physical Systems. In *CAV (1) (LNCS, Vol. 12224)*. Springer, 3–17.

[44] Oriol Vinyals, Igor Babuschkin, Wojciech M. Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H. Choi, Richard Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, Laurent Sifre, Trevor Cai, John P. Agapiou, Max Jaderberg, Alexander Sasha Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom L. Paine, Çaglar Gülçehre, Ziyu Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy P. Lillicrap, Koray Kavukcuoglu, Demis Hassabis, Chris Apps, and David Silver. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nat.* 575, 7782 (2019), 350–354.

[45] Norbert Wiener. 1961. *Cybernetics or Control and Communication in the Animal and the Machine.* MIT Press.

[46] Wenbo Zhang and Osbert Bastani. 2019. MAMPS: Safe Multi-Agent Reinforcement Learning via Model Predictive Shielding. *CoRR* abs/1910.12639 (2019).