Intelligent and Dependable Decision-Making Under Uncertainty

Nils Jansen(✉)

Radboud University Nijmegen, Nijmegen, The Netherlands
nilsjansen123@gmail.com

Abstract. This talk highlights our vision of foundational and application-driven research toward safety, dependability, and correctness in artificial intelligence (AI). We take a broad stance on AI that combines formal methods, machine learning, and control theory. As part of this research line, we study problems inspired by autonomous systems, planning in robotics, and industrial applications. We consider reinforcement learning (RL) as a specific machine learning technique for decision-making under uncertainty. RL generally learns to behave optimally via trial and error. Consequently, and despite its massive success in the past years, RL lacks mechanisms to ensure safe and correct behavior. Formal methods, in particular formal verification, is a research area that provides formal guarantees of a system's correctness and safety based on rigorous methods and precise specifications. Yet, fundamental challenges have obstructed the effective application of verification to reinforcement learning. Our main objective is to devise novel, data-driven verification methods that tightly integrate with RL. In particular, we develop techniques that address real-world challenges to the safety of AI systems in general: Scalability, expressiveness, and robustness against the uncertainty that occurs when operating in the real world. The overall goal is to advance the real-world deployment of reinforcement learning.

1 Synopsis: Robust and Dependable Artificial Intelligence

Artificial intelligence (AI) is a disruptive force. Most major technology companies employ or develop AI, and with growing applications in fields like healthcare [37], transportation [48,68], game playing [51], finance [9], or robotics in general [44], it is entering our everyday lives. We can expect that our societal and technological involvement with AI will only intensify in the future. Such tight interaction with AI requires serious safety and correctness considerations. Recently, the field of safety in AI has triggered a vast amount of research with several seminal works defining their view on this area [4,25,58,61].

Can Formal Verification Help to Ensure AI Safety? The area of formal methods offers structured and rigorous ways to reason about the correctness

N. Jansen—This work was supported by the ERC Starting Grant 101077178 (DEUCE).

M. Chechik et al. (Eds.): FM 2023, LNCS 14000, pp. 26–36, 2023.
https://doi.org/10.1007/978-3-031-27481-7_3

of a system. Techniques range from model learning [66], over testing [36], to formal verification [24]. As an example for the application of verification in AI, solving techniques like SAT or SMT [11] help to assess the robustness of neural networks [30,33,41]. A specific verification technique is *model checking* [10,19]. For a fixed system model, a plethora of methods assert the system's correctness regarding *formal specifications*. The rigor of model checking suggests it is natural to employ model checking to prove the correctness of AI systems.

We focus on a specific branch of AI, namely *decision-making under uncertainty* [45]. Intelligent AI agents typically operate in unknown or unpredictable environments, coping with contextual changes at runtime or incompleteness of information. This unpredictability leads to the problem that the outcome of decisions made by an agent is *uncertain*. *Reinforcement learning* (RL) [64] agents make decisions under uncertainty via the exploration of potentially unknown environments. The area of *safe RL* [2,27] aims to restrict the behavior of an agent with respect to safety, or with respect to more general correctness constraints.

Several shortcomings towards the potential deployment of RL in critical environments remain. Specifically, we identify the following three main challenges to the state-of-the-art in formal verification and its application for safe RL:

- Scalability to **high-dimensional problems**,
- Providing correctness guarantees in **continuous spaces**, and
- effective handling of **uncertainty**.

Indeed, common approaches and case studies for safe RL employ idealized settings with a low number of dimensions that contribute to a problem. Most approaches assume discretized state spaces instead of realistic continuous settings. Currently employed simplistic notions of uncertainty may lead to incorrect behavior, and RL agents are often trained without any notion of safe behavior under uncertainty [72]. Finally, standard safety notions cannot express sophisticated task or correctness specifications.

The state-of-the-art leaves the aforementioned three challenges largely unaddressed. Our approaches to fundamentally overcome these restrictions employ a particularly tight integration of verification and learning. We see the data-driven nature not as a threat to effective and rigorous verification, but embrace the inherent access to state-of-the-art machine learning and exploit its flexibility.

Finally, to demonstrate the practical applicability of our work, we use the QComp [31] and Arch-Comp [1] competitions, and for more AI-related benchmarks, the OpenAI gym [53] and Google Deepmind's AI Safety Gridworlds [47]. Towards industrial demonstrators, we use, for instance, case studies from predictive maintenance, such as [42].

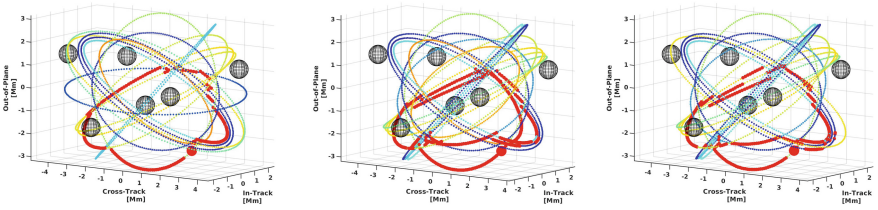
How to Make Intelligent Decisions Under Uncertainty? Various types and applications of uncertainty play a central role in our research. Uncertainty has been "largely related to the lack of predictability of some major events or stakes, or a lack of data" [5]. To name a few, there is uncertainty (1) in technological, social, environmental, or financial factors in the *business literature* [60],

(2) about sensor imprecisions and lossy communication channels in *robotics* [65], and (3) about the expected responses of a *human* operator in decision support systems [45]. The level of uncertainty affects the capabilities of AI systems that have to make decisions [3,45]. In particular, for strict safety requirements, decisions must be *verifiably robust* against uncertainty. Such considerations require precise knowledge about the nature of uncertainty.

Model checking for AI systems necessitates dedicated models. Markov decision processes (MDPs) capture sequential decision-making problems for agents operating in uncertain environments [57]. Sensor limitations may lead to partial observability of the system's current state, giving rise to partially observable Markov decision processes (POMDPs) [40]. While mature model checking tools like PRISM [46], Storm [22], or Uppaal [21] provide efficient synthesis or verification methods for MDPs, the situation is different for POMDPs. Policy synthesis for POMDPs is a hard problem, both from the theoretical and the practical perspective [50]. For infinite- or indefinite-horizon problems, computing an optimal policy is undecidable [49]. Optimal action choices depend on the whole observation history, requiring an infinite amount of memory.

If precise probabilities are not known, *uncertainty models* employ so-called uncertainty sets of probabilities. Uncertain MDPs (uMDPs) use, for example, *probability intervals* or *likelihood functions* [23,28,52,56,69–71,73]. Similar extensions exist for uPOMDPs, where uncertainty also affects the observation model [12,13,20,34,62].

A Motivating Example: Spacecraft Motion Planning. Consider a spacecraft motion planning system which serves as decision support for a human operator [26,32]. This system delivers advice on switching to a different orbit or avoiding close encounters with other objects in space. The spacecraft orbits the earth along a set of predefined natural motion trajectories (NMTs) [43]. While the spacecraft follows its current NMT, it does not consume fuel. We introduced the underlying uncertain POMDP model in [20]. The figure to the right depicts three models that differ only in the level of uncertainty (low, medium, high). Black spheres are the objects, and the colored lines depict NMTs. The thick red line indicates a trajectory of the spacecraft including orbit switches along the NMTs. A policy requires *robustness* against uncertainty, and *memory* to predict



(a) Low uncertainty (b) Med uncertainty (c) High uncertainty

Fig. 1. Robust spacecraft motion planning.

the location of the spacecraft based on its past trajectory (Fig. 1). The figure shows that *more uncertainty causes less-informed decisions*, as policies need to be more conservative.

2 Research Highlights

In the following, we discuss a number of results that are in line with the aforementioned research challenges to combining formal verification, AI systems, and reinforcement learning.

2.1 Reliable Neural Network Controllers for Autonomous Agents

Summary. These results are part of the publications [16–18]. Machine learning methods typically train recurrent neural networks (RNN) to effectively represent POMDP policies that can efficiently process sequential data. However, it is hard to verify whether the POMDP driven by such RNN-based policies satisfies safety constraints, for instance, given by temporal logic specifications. We propose a novel method that combines techniques from machine learning with the field of formal methods: training an RNN-based policy and automatically extracting a so-called finite-state controller (FSC) from the RNN. Such FSCs offer a convenient way to verify temporal logic constraints. Implemented on a POMDP, they induce a Markov chain. Probabilistic verification methods can efficiently check whether this induced Markov chain satisfies a temporal logic specification. Our method exploits this diagnostic information from verification to either adjust the complexity of the extracted FSC or improve the policy by performing focused retraining of the RNN. We synthesize policies that satisfy temporal logic specifications for POMDPs with up to millions of states, three orders of magnitude larger than comparable approaches.

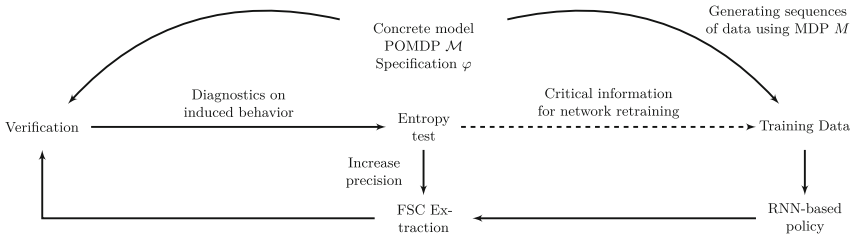


Fig. 2. Summary flowchart of the RNN-based refinement loop.

Our Approach: Learning and Verification. We combine the effectiveness of RNN-based representations from machine learning with the provable guarantees that are at the heart of formal verification. In a nutshell, we train RNN-based policy representations from sequences of data, to find candidate policies that might ensure an agent satisfies a temporal logic specification.

The central technical problem is: How to close the loop between training an RNN-based policy and efficiently verifying for a candidate policy? First, FSCs [39,54] encode memory in a finite automata-style fashion. For an FSC and a POMDP, formal verification methods like model checking are able to efficiently compute the probability of satisfying a specification [10]. We tightly integrate formal verification and machine learning towards three key steps: (1) extracting an FSC from an RNN-based policy, (2) verifying this candidate FSC for the POMDP against a temporal logic specification, and (3) if needed, either refining the FSC or generating more training data for the RNN. For an overview, see Fig. 2.

2.2 Learning Uncertainty Models

Summary. This result is part of the publication [63]. In data-driven applications, deriving precise probabilities from (limited) data introduces statistical errors that may lead to unexpected or undesirable outcomes. Consequently, we aim to learn uncertain MDPs (uMDPs) that use so-called uncertainty sets in the transitions, accounting for such limited data. Efficient implementations in tools like PRISM compute robust policies for uMDPs that provably adhere to formal specifications, like safety constraints, under the worst-case instance in the uncertainty set. We continuously learn the transition probabilities of an MDP in a robust anytime-learning approach that combines a dedicated Bayesian inference scheme with the computation of robust policies. In particular, our method (1) approximates probabilities as intervals, (2) adapts to new data that may be inconsistent with an intermediate model, and (3) may be stopped at any time to compute a robust policy on the uMDP that faithfully captures the data so far. Similarly, our method is capable of adapting to changes in the environment. We show the effectiveness of our approach and compare it to robust policies computed on uMDPs learned by the UCRL2 reinforcement learning algorithm.

Our Approach: Learning an MDP from Data. We propose an iterative learning method that uses uMDPs as intermediate models and is able to *adapt to new data* which may be inconsistent with prior assumptions. The Bayesian *anytime-learning approach* employs intervals with linearly updating conjugate priors [67], and can iteratively improve upon a uMDP that approximates the true MDP we wish to learn. The key features of our learning method are:

– *An anytime approach.* At any time, we may stop the learning and compute a robust policy for the uMDP that the process has yielded thus far, together with the worst-case performance of this policy against a given specification. This performance may not be satisfactory, e. g., the worst-case probability to reach a set of critical states may be below a certain threshold. We continue

learning towards a new uMDP that more faithfully captures the true MDP due to the inclusion of further data. Thereby, we ensure that the robust policy gradually gets closer to the optimal policy for the true MDP.
– *Specification-driven.* Our method features the possibility to learn in a task-aware fashion, that is, to learn transitions that matter for a given specification. In particular, for reachability or expected reward (temporal logic) specifications that require a certain set of target states to be reached, we only learn and update transitions along paths toward these states. Transitions outside those paths do not affect the satisfaction of the specification.
– *Adaptive to changing environment dynamics.* When using linearly updating intervals, our approach is adaptive to changing environment dynamics. That is, if during the learning process the probability distributions of the underlying MDP change, our method can easily adapt and learns these new distributions.

2.3 Robust Control for Dynamical Systems Under Uncertainty

Summary. These results are part of the publications in [6–8]. We provide probably correct controllers for dynamical systems that operate in noisy environments, where the uncertainty can be both aleatoric and epistemic. In particular, we consider environments where stochastic disturbances in the environment are not necessarily Gaussian, and external uncertainty may be caused by factors such as uncertain system parameters. In our work, no explicit representation of a noise distribution is necessary, but we only assume sampling access to the environment. Using the so-called scenario approach, we provide probabilistic guarantees on reach-avoid properties, that is, safely reaching a target while avoiding unsafe regions of the state space. At the heart of our approach is an abstraction of the dynamical system into an uncertain MDP. We show that a robust policy for this finite-state model carries guarantees on the performance of the analogous controller in the dynamical system.

Our Approach: Probabilities Are Not Enough. We consider stochastic dynamical models with continuous state and action spaces, under aleatoric and epistemic uncertainty. More precisely, aleatoric uncertainty captures natural randomness (i.e., stochasticity) in the outcome of transitions, while epistemic uncertainty is in particular modeled by parameters that are not precisely known [59].

– *PAC guarantees on abstractions.* We show that both probabilities and nondeterminism can be captured in the probability intervals of an uncertain MDP. We use sampling methods from scenario optimization [14] and show that, with a predefined confidence probability, the uncertain MDP correctly captures both aleatoric and epistemic uncertainty.
– *Correct-by-construction.* For the uncertain MDP, we compute a *robust optimal policy* that maximizes the worst-case probability of satisfying the reach-avoid specification. This policy is automatically translated to a *provably-correct feedback controller* for the original, continuous model 'on the fly'. This means that, by construction, the PAC guarantees on the uncertain MDP carry over

to the satisfaction of the specification for the continuous model, thus solving the problem stated above.
- *Contributions.* We develop the first abstraction-based, formal controller synthesis method that simultaneously captures epistemic and aleatoric uncertainty for continuous-state/action models. We provide results on the PAC-correctness of obtained uncertain MDP abstractions, and guarantees on the synthesized controllers for a reach-avoid specification.

2.4 Safe Deep Reinforcement Learning

Summary. These results are part of the publications in [15,29,35,38,55]. A common approach to safe reinforcement learning is to employ a so-called shield that forces an RL agent to select only safe actions. However, for adoption in various applications, one must look beyond enforcing safety and also ensure the applicability of RL with good performance. We extend the applicability of shields via tight integration with state-of-the-art deep RL, and provide an extensive, empirical study in challenging, sparse-reward environments under partial observability. We show that a carefully integrated shield ensures safety and can improve the convergence rate and final performance of RL agents. We furthermore show that a shield can be used to bootstrap state-of-the-art RL agents: they remain safe after initial learning in a shielded setting, allowing us to disable a potentially too-conservative shield eventually.

Our Approach: Shielding in Deep Reinforcement Learning. Our study demonstrates the following effects of shielding in a partially observable setting.

- *Shield construction:* We discuss several approaches to effectively construct and compute a shield in environments that exhibit various sources of uncertainty.
- *Safety during learning:* Exploration is only safe when the RL agent is provided with a shield. Without the shield, the agent makes unsafe choices even if it has access to the state estimation. Even an unshielded *trained agent* still behaves unsafe sometimes.
- *RL convergence rate:* A shield not only ensures safety, but may also significantly improve the convergence rate of modern RL agents by avoiding spending time to learn unsafe actions. Other knowledge interfaces like state estimators do help to a lesser extent.
- *Bootstrapping:* Due to the improved convergence rate, shields are a way to bootstrap RL algorithms, even if they are overly restrictive. RL agents can learn to mimic the shield by slowly disabling the shield.
- *Tool support:* We provide an open source tool called COOL-MC[1] that features a tied integration between state-of-the-art RL in OpenAI gym [53] and the Storm model checker [22].

[1] Available at https://github.com/LAVA-LAB/COOL-MC.

Acknowledgements. The approaches presented in this talk are the results of fruitful and enjoyable collaborations with a number of co-authors, in particular: Alessandro Abate, Thom S. Badings, Bernd Becker, Roderick Bloem, Steven Carr, Murat Cubuktepe, Dennis Gross, Sebastian Junges, Joost-Pieter Katoen, Bettina Könighofer, David Parker, Guillermo A. Pérez, Hasan A. Poonawala, Licio Romao, Sanjit Seshia, Alex Serban, Thiago D. Simão, Mariëlle Stoelinga, Marnix Suilen, Ufuk Topcu, and Ralf Wimmer.

References

1. Abate, A., et al.: ARCH-COMP18 category report: stochastic modelling. In: ARCH@ADHS. EPiC Series in Computing, vol. 54, pp. 71–103. EasyChair (2018)
2. Alshiekh, M., Bloem, R., Ehlers, R., Könighofer, B., Niekum, S., Topcu, U.: Safe reinforcement learning via shielding. In: AAAI. AAAI Press (2018)
3. Amato, C.: Decision-making under uncertainty in multi-agent and multi-robot systems: planning and learning. In: IJCAI, pp. 5662–5666. ijcai.org (2018)
4. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., Mané, D.: Concrete problems in AI safety. CoRR abs/1606.06565 (2016)
5. Argote, L.: Input uncertainty and organizational coordination in hospital emergency units. Adm. Sci. Q., 420–434 (1982)
6. Badings, T.S., Abate, A., Jansen, N., Parker, D., Poonawala, H.A., Stoelinga, M.: Sampling-based robust control of autonomous systems with non-Gaussian noise. In: AAAI (2022). To appear
7. Badings, T.S., Romano, L., Abate, A., Jansen, N.: Probabilities are not enough: Formal controller synthesis for stochastic dynamical models with epistemic uncertainty. In: AAAI (2023)
8. Badings, T.S., et al.: Robust control for dynamical systems with non-gaussian noise via formal abstractions. J. Artif. Intell. Res. (2023)
9. Bahrammirzaee, A.: A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems. Neural Comput. Appl. **19**(8), 1165–1195 (2010). https://doi.org/10.1007/s00521-010-0362-z
10. Baier, C., Katoen, J.P.: Principles of Model Checking. The MIT Press, Cambridge (2008)
11. Biere, A., Heule, M., van Maaren, H., Walsh, T. (eds.): Handbook of Satisfiability Frontiers in Artificial Intelligence and Applications, vol. 185. IOS Press, Amsterdam (2009)
12. Bry, A., Roy, N.: Rapidly-exploring random belief trees for motion planning under uncertainty. In: ICRA, pp. 723–730. IEEE (2011)
13. Burns, B., Brock, O.: Sampling-based motion planning with sensing uncertainty. In: ICRA, pp. 3313–3318. IEEE (2007)
14. Campi, M.C., Garatti, S.: Introduction to the scenario approach. SIAM (2018)
15. Carr, S., Jansen, N., Junges, S., Topcu, U.: Safe reinforcement learning via shielding under partial observability. In: AAAI (2023)
16. Carr, S., Jansen, N., Topcu, U.: Verifiable RNN-based policies for POMDPs under temporal logic constraints. In: IJCAI, pp. 4121–4127. ijcai.org (2020)
17. Carr, S., Jansen, N., Topcu, U.: Task-aware verifiable RNN-based policies for partially observable Markov decision processes. J. Artif. Intell. Res. **72**, 819–847 (2021)

18. Carr, S., Jansen, N., Wimmer, R., Serban, A.C., Becker, B., Topcu, U.: Counterexample-guided strategy improvement for POMDPs using recurrent neural networks. In: IJCAI, pp. 5532–5539. ijcai.org (2019)
19. Clarke, E.M., Henzinger, T.A., Veith, H., Bloem, R.: Handbook of Model Checking, vol. 10. Springer, Cham (2018)
20. Cubuktepe, M., Jansen, N., Junges, S., Marandi, A., Suilen, M., Topcu, U.: Robust finite-state controllers for uncertain POMDPs. In: AAAI, pp. 11792–11800. AAAI Press (2021)
21. David, A., Jensen, P.G., Larsen, K.G., Mikučionis, M., Taankvist, J.H.: Uppaal Stratego. In: Baier, C., Tinelli, C. (eds.) TACAS 2015. LNCS, vol. 9035, pp. 206–211. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46681-0_16
22. Dehnert, C., Junges, S., Katoen, J.P., Volk, M.: A **storm** is coming: a modern probabilistic model checker. In: Majumdar, R., Kuncak, V. (eds.) CAV 2017. LNCS, Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63390-9_31
23. Delahaye, B., Larsen, K.G., Legay, A., Pedersen, M.L., Wasowski, A.: Decision problems for interval Markov chains. In: Dediu, A.-H., Inenaga, S., Martín-Vide, C. (eds.) LATA 2011. LNCS, vol. 6638, pp. 274–285. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21254-3_21
24. Drechsler, R.: Advanced Formal Verification. Kluwer Academic Publishers, Dordrecht (2004)
25. Freedman, R.G., Zilberstein, S.: Safety in AI-HRI: challenges complementing user experience quality. In: AAAI Fall Symposium Series (2016)
26. Frey, G.R., Petersen, C.D., Leve, F.A., Kolmanovsky, I.V., Girard, A.R.: Constrained spacecraft relative motion planning exploiting periodic natural motion trajectories and invariance. J. Guid. Control. Dyn. **40**(12), 3100–3115 (2017)
27. Garcıa, J., Fernández, F.: A comprehensive survey on safe reinforcement learning. J. Mach. Learn. Res. **16**(1), 1437–1480 (2015)
28. Givan, R., Leach, S., Dean, T.: Bounded-parameter Markov decision processes. Artif. Intell. **122**(1–2), 71–109 (2000)
29. Gross, D., Jansen, N., Junges, S., Pérez, G.A.: COOL-MC: a comprehensive tool for reinforcement learning and model checking. In: Dong, W., Talpin, J.P. (eds.) SETTA 2022. LNCS, vol. 13649, pp. 41–49. Springer, Cham (2022)
30. Gross, D., Jansen, N., Pérez, G.A., Raaijmakers, S.: Robustness verification for classifier ensembles. In: Hung, D.V., Sokolsky, O. (eds.) ATVA 2020. LNCS, vol. 12302, pp. 271–287. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59152-6_15
31. Hahn, E.M., et al.: The 2019 comparison of tools for the analysis of quantitative formal models. In: Beyer, D., Huisman, M., Kordon, F., Steffen, B. (eds.) TACAS 2019. LNCS, vol. 11429, pp. 69–92. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17502-3_5
32. Hobbs, K.L., Feron, E.M.: A taxonomy for aerospace collision avoidance with implications for automation in space traffic management. In: AIAA Scitech 2020 Forum, p. 0877 (2020)
33. Huang, X., Kwiatkowska, M., Wang, S., Wu, M.: Safety verification of deep neural networks. In: Majumdar, R., Kunčak, V. (eds.) CAV 2017. LNCS, vol. 10426, pp. 3–29. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63387-9_1
34. Itoh, H., Nakamura, K.: Partially observable Markov decision processes with imprecise parameters. Artif. Intell. **171**(8), 453–490 (2007)

35. Jansen, N., Könighofer, B., Junges, S., Serban, A., Bloem, R.: Safe reinforcement learning using probabilistic shields (invited paper). In: CONCUR. LIPIcs, vol. 171, pp. 1–16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2020)
36. Jia, Y., Harman, M.: An analysis and survey of the development of mutation testing. IEEE Trans. Software Eng. **37**(5), 649–678 (2011)
37. Jiang, F., et al.: Artificial intelligence in healthcare: past, present and future. Stroke Vasc. Neurol. **2**(4) (2017)
38. Junges, S., Jansen, N., Seshia, S.A.: Enforcing almost-sure reachability in POMDPs. In: Silva, A., Leino, K.R.M. (eds.) CAV 2021. LNCS, vol. 12760, pp. 602–625. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-81688-9_28
39. Junges, S., et al.: Finite-state controllers of POMDPs using parameter synthesis. In: UAI, pp. 519–529. AUAI Press (2018)
40. Kaelbling, L.P., Littman, M.L., Cassandra, A.R.: Planning and acting in partially observable stochastic domains. Artif. Intell. **101**(1), 99–134 (1998)
41. Katz, G., Barrett, C., Dill, D.L., Julian, K., Kochenderfer, M.J.: Reluplex: an efficient SMT solver for verifying deep neural networks. In: Majumdar, R., Kunčak, V. (eds.) CAV 2017. LNCS, vol. 10426, pp. 97–117. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63387-9_5
42. Kerkkamp, D., Bukhsh, Z.A., Zhang, Y., Jansen, N.: Grouping of maintenance actions with deep reinforcement learning and graph convolutional networks. In: ICAART (2022). To Appear
43. Kim, S.C., Shepperd, S.W., Norris, H.L., Goldberg, H.R., Wallace, M.S.: Mission design and trajectory analysis for inspection of a host spacecraft by a microsatellite. In: 2007 IEEE Aerospace Conference, pp. 1–23. IEEE (2007)
44. Klingspor, V., Demiris, J., Kaiser, M.: Human-robot communication and machine learning. Appl. Artif. Intell. **11**(7), 719–746 (1997)
45. Kochenderfer, M.J.: Decision Making Under Uncertainty: Theory and Application. MIT press, Cambridge (2015)
46. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: verification of probabilistic real-time systems. In: Gopalakrishnan, G., Qadeer, S. (eds.) CAV 2011. LNCS, vol. 6806, pp. 585–591. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22110-1_47
47. Leike, J., et al.: AI safety gridworlds. arXiv preprint arXiv:1711.09883 (2017)
48. Levinson, J., et al.: Towards fully autonomous driving: Systems and algorithms. In: Intelligent Vehicles Symposium, pp. 163–168. IEEE (2011)
49. Madani, O., Hanks, S., Condon, A.: On the undecidability of probabilistic planning and infinite-horizon partially observable Markov decision problems. In: AAAI. pp. 541–548. AAAI Press (1999)
50. Meuleau, N., Peshkin, L., Kim, K.E., Kaelbling, L.P.: Learning finite-state controllers for partially observable environments. In: UAI, pp. 427–436. Morgan Kaufmann (1999)
51. Mnih, V., et al.: Playing atari with deep reinforcement learning. CoRR abs/1312.5602 (2013)
52. Nilim, A., El Ghaoui, L.: Robust control of Markov decision processes with uncertain transition matrices. Oper. Res. **53**(5), 780–798 (2005)
53. OpenAI Gym: (2018). http://gymlibrary.dev/
54. Poupart, P., Boutilier, C.: Bounded finite state controllers. In: Advances in Neural Information Processing Systems, pp. 823–830 (2004)
55. Pranger, S., Könighofer, B., Tappler, M., Deixelberger, M., Jansen, N., Bloem, R.: Adaptive shielding under uncertainty. In: ACC, pp. 3467–3474. IEEE (2021)

56. Puggelli, A., Li, W., Sangiovanni-Vincentelli, A.L., Seshia, S.A.: Polynomial-time verification of PCTL properties of MDPs with convex uncertainties. In: Sharygina, N., Veith, H. (eds.) CAV 2013. LNCS, vol. 8044, pp. 527–542. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39799-8_35
57. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley and Sons, Hoboken (1994)
58. Russell, S.J., Dewey, D., Tegmark, M.: Research priorities for robust and beneficial artificial intelligence. CoRR abs/1602.03506 (2016)
59. Smith, R.C.: Uncertainty Quantification: Theory, Implementation, and Applications, vol. 12. Siam, New Delhi (2013)
60. Sniazhko, S.: Uncertainty in decision-making: a review of the international business literature. Cogent Bus. Manage. **6**(1), 1650692 (2019)
61. Stoica, I., et al.: A Berkeley view of systems challenges for AI. CoRR abs/1712.05855 (2017)
62. Suilen, M., Jansen, N., Cubuktepe, M., Topcu, U.: Robust policy synthesis for uncertain POMDPs via convex optimization. In: IJCAI, pp. 4113–4120. ijcai.org (2020)
63. Suilen, M., Simão, T.D., Parker, D., Jansen, N.: Robust anytime learning of Markov decision processes. In: NeurIPS (2022)
64. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)
65. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. The MIT Press, Cambridge (2005)
66. Vaandrager, F.W.: Model learning. Commun. ACM **60**(2), 86–95 (2017)
67. Walter, G., Augustin, T.: Imprecision and prior-data conflict in generalized Bayesian inference. J. Stat. Theor. Pract. **3**(1), 255–271 (2009)
68. Wang, F.: Toward a revolution in transportation operations: AI for complex systems. IEEE Intell. Syst. **23**(6), 8–13 (2008)
69. Wiesemann, W., Kuhn, D., Rustem, B.: Robust Markov decision processes. Math. Oper. Res. **38**(1), 153–183 (2013)
70. Wolff, E.M., Topcu, U., Murray, R.M.: Robust control of uncertain Markov decision processes with temporal logic specifications. In: CDC, pp. 3372–3379. IEEE (2012)
71. Xu, H., Mannor, S.: Distributionally robust Markov decision processes. Math. Oper. Res. **37**(2), 288–300 (2012)
72. Zhang, J., Cheung, B., Finn, C., Levine, S., Jayaraman, D.: Cautious adaptation for reinforcement learning in safety-critical settings. In: ICML. Proceedings of Machine Learning Research, vol. 119, pp. 11055–11065. PMLR (2020)
73. Zhao, X., Calinescu, R., Gerasimou, S., Robu, V., Flynn, D.: Interval change-point detection for runtime probabilistic model checking. In: 35th IEEE/ACM International Conference on Automated Software Engineering. York (2020)