Research Statement

Kavosh Asadi

April 12, 2020

Half a century has passed since Moore's prediction pertaining to the exponential growth of computing resources [1], and advancements in computing have proven this prediction to be accurate thus far. The compute power that we could not dream of at the beginning of the millennium is now cheaply available in our so-called smartphones. But, why are our phones not truly smart yet?

Artificial intelligence (AI) has recently achieved numerous breakthroughs. From computer vision [2], to speech recognition [3], AI has created technologies that are influencing humanity in profound ways. These advancements are enabled by what is referred to as supervised learning, in which the computer learns patterns in some human-provided dataset to best imitate human expertise. While useful in many settings, supervised learning is limited in two ways: First, human expertise is costly, and so it is often difficult to collect the large datasets that are necessary for supervised learning. Second, and more importantly, a computer that is trained to imitate humans can never exceed the limited boundaries of human expertise.

There is an alternative paradigm that is becoming more popular. Known as interactive AI, in this paradigm the learning system trains to achieve a certain goal by interaction and through trial and error. Many problems could more naturally be formulated using this paradigm, because here the machine is not explicitly told what actions to take, but rather it is provided with some notion of reward that evaluates its taken actions. Using this evaluative feedback, the machine learns to optimize its behavior to progressively obtain more reward as learning proceeds. The study of this interaction is called reinforcement learning (RL).

Many questions arise when applying RL to real-world problems. Perhaps the most fundamental question pertains to generalization: real world is complex, large, and messy, so how can an RL agent learn to perform well in light of its access to limited amount of environmental interaction? A typical solutions is to employ function approximation, where the idea is to represent various ingredients of RL (such as value functions, policies, and models) using a parameterized function class. Training here consists of learning the parameter setting that best represents the ingredient of interest. Crucially, longstanding problems in RL including convergence guarantees, planning, model-learning, and exploration become significantly more challenging in the presence of function approximation.

As a concrete example, to guarantee any robustness for RL with function approximation, we need to ensure that our RL agent exhibits a convergent behavior. Stated differently, we desire for the agent to always converge to some solution. This does not mean that we know what the solution is a priori, but that there exists a well-defined solution and that our RL agent is capable of finding it. Some of my early research was gravitated towards identifying settings in which this basic notion of

convergence holds. Specifically, I have shown that this notion of convergence is missing in a class of model-free RL algorithms with the Boltzmann softmax operator [4]. I have developed a theory for an alternative softmax operator that is sound and convergent [4], but also performs well in large settings [5] in light of its connections to entropy regularization [6].

In many applications, such as robotics, the RL agent deals with an enormous or even a continuous action space [7]. In the continuous case, one could think of an action as a vector with d floating point numbers representing, say, the amount of force applied to each actuator of a robot. In these settings, finding an action that is optimal with respect to the learned value function can be difficult, yet identifying the action with the highest value is a core competency for learning and acting. In my research, I have developed function approximators that enable fast and accurate action maximization, but also allow for representing arbitrarily-complicated functions [8].

Another class of RL algorithms are those that explicitly learn a model of their environment. One primary goal of my research is to study these so called model-based algorithms in large state spaces. Specifically, in my research, I seek to understand the main principles underlying learning a good model of the environment [9, 10]. How should the model be used once it is learned? [11] And how can the agent be cognizant of the mistakes that its learned model makes, so as to avoid catastrophic consequences? [12]

Another key property of RL systems is the ability to build abstract representation of environments. I have been active in a research program that seeks to understand the underlying principles of learning state abstractions that facilitate future learning using simple algorithms [13]. These abstractions are often quite useful since they exhibit powerful generalization capabilities in some downstream tasks, and can solve problems by just retaining the relevant information about their environments [14].

More generally, my fundamental research philosophy is to prioritize ideas that are scalable in terms of compute, data, and memory. I usually proceed with a project only if I get an affirmative answer to scalability questions. Then, in order to explore the idea diligently, I find a clear and simple setting in which the idea could best be explored.

I believe this is a truly exciting time to do RL research, because we have just taken some small steps towards efficient and robust RL in large problems. But what kinds of problems do we need to solve next in order to make the biggest steps towards solving AI?

In my view the biggest hurdle is solving the exploration problem in large settings. Our RL algorithms need to be able to efficiently explore their world by minimizing their number of mistakes, while also capitalizing on their obtained knowledge for reward maximization. I am fascinated by the problem of exploration in metric spaces [15], where we can naturally assume some regularity properties on agent's states and action spaces [16]. I have takens small steps to this end by developing Q-learning-like algorithm for continuous control [8], and proving simulation lemmas for model-based RL based on Wasserstein [9], but the main work is left to be done.

Understanding planning and model-based control is another big hurdle between our algorithms and AI. How can humans plan so effectively and efficiently despite their inaccurate models, yet inaccuracies are so harmful in model-based RL? How can we learn and plan using partial models? How can we learn abstract states that are suitable for planning? How should RL agents be cognizant of modeling errors during planning? There are numerous open questions in this space.

Finally, I think time is long overdue for us to see real-world applications of RL. Our focus has almost exclusively been on simulated platforms and games. It is not bad to apply RL to simulation if that facilitates a platform to measure algorithmic improvements, but I see limited benefits in applying RL to even more games at this point. The use of RL in the game of Go has already demonstrated the fantastic potential of our RL technology in games [17], and so, it is time to move to other applications, ideally those with actual impact to our society. I am personally excited about dialog management, where it is reasonable to expect humans to provide some notion of reward signal at the end of the conversation [18, 19], thus making it a fascinating application for RL. Another promising area is robotics, where the agent needs to learn a complex mapping from its sensors to (often continuous and multi-dimensional) actions. In my view, safety, exploration, and planning are some of the most important problems when tackling robotics with RL.

References

[1] Gordon E Moore. Cramming more components onto integrated circuits. *Proceedings of the IEEE*, 86(1):82–85, 1998.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[3] Wayne Xiong, Lingfeng Wu, Fil Alleva, Jasha Droppo, Xuedong Huang, and Andreas Stolcke. The microsoft 2017 conversational speech recognition system. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5934–5938. IEEE, 2018.

[4] Kavosh Asadi and Michael L. Littman. An alternative softmax operator for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning ICML*, 2017.

[5] Seungchan Kim, Kavosh Asadi, Michael Littman, and George Konidaris. Deepmellow: removing the need for a target network in deep Q-learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI*, 2019.

[6] Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, 2019.

[7] Sebastian Thrun. Probabilistic robotics. *Communications of the ACM*, 45(3):52–57, 2002.

[8] Kavosh Asadi, Ronald E Parr, George D Konidaris, and Michael L Littman. Deep RBF value functions for continuous control. *arXiv preprint arXiv:2002.01883*, 2020.

[9] Kavosh Asadi, Dipendra Misra, and Michael L. Littman. Lipschitz continuity in model-based reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML*, 2018.

[10] Kavosh Asadi, Evan Cater, Dipendra Misra, and Michael L Littman. Equivalence between wasserstein and value-aware model-based reinforcement learning. In *FAIM Workshop on Prediction and Generative Modeling in Reinforcement Learning*, volume 3, 2018.

[11] Kavosh Asadi, Dipendra Misra, Seungchan Kim, and Michel L Littman. Combating the compounding-error problem with a multi-step model. *arXiv preprint arXiv:1905.13320*, 2019.

[12] Marc Deisenroth and Carl E Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pages 465–472, 2011.

[13] Kavosh Asadi, David Abel, and Michael L Littman. Learning state abstractions for transfer in continuous control. *arXiv preprint arXiv:2002.05518*, 2020.

[14] David Abel, Dilip Arumugam, Kavosh Asadi, Yuu Jinnai, Michael L Littman, and Lawson LS Wong. State abstraction as compression in apprenticeship learning. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, 2019.

[15] Sham Kakade, Michael J Kearns, and John Langford. Exploration in metric state spaces. In *Proceedings of the 20th International Conference on Machine Learning ICML*, 2003.

[16] Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, 2008.

[17] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

[18] Jason D Williams, Kavosh Asadi, and Geoffrey Zweig. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL*, 2017.

[19] Kavosh Asadi and Jason D Williams. Sample-efficient deep reinforcement learning for dialog control. *arXiv preprint arXiv:1612.06000*, 2016.