Master's Thesis Proposal

LTL Reward Specification in Constrained Markov Decision Processes

Jonathan Azpur

Contents

1 Introduction

Reinforcement Learning (RL) is a field of machine learning that focuses on training an intelligent agent on how it ought to act in order to achieve a set task in an unknown environment. The environment is typically assumed to be an unknown Markov decision process (MDP). Reinforcement Learning algorithms train an agent by making it explore the unknown environment (exploration) and learn which actions will maximize the cumulative reward provided by said environment (exploitation). This thesis will focus on improving two of the main concerns with reinforcement learning, safety and performance.

When training an agent in a MDP the main focus is to maximize the cumulative reward it receives. So, the exploration process typically does not concern itself with potential threats to the agent and therefore can not guarantee its safety. Constrained Markov Decision Processes (CMDPs) are a special form of MDPs that allow for the separation of safety specifications from the reward function by naturally encoding the safety concerns as constraints. They were first introduced by Altman [1999]. A CMDP will have the same components as an MDP plus a cost function and an upper bound for the expected cumulative constraint cost. In a CMDP the goal is to learn an optimal policy that maximizes the cumulative reward and whose cumulative cost is under the established upper bound.

In addition, intelligent agents do not have access to the reward function governing their environment. For the agents, the reward function is a black-box that will return an immediate reward given their current state. Therefore, finding an optimal policy using reinforcement learning can require a lot of exploration of the environment. Sometimes this can be of very high cost and in some cases not possible to perform, such as in physical environments. So, if the goal is for the agent to maximize cumulative reward, it would be helpful if they could have some knowledge of their governing reward function/s. One way to approach this limitation is by introducing the use of logic to supply prior knowledge to the agent. Icarte et al. [2018a] introduced Reward Machines (RM), a type of finite state machine that would allow the agent to be exposed to the structure of the reward function. The idea is for the RM to replace the original reward function in an MDP, transforming the MDP into a MDP with a Reward Machine (MDPRM). The MDPRM can then be exploited by a RL algorithm in a way that it will allow the agent to decompose their task and speed up the learning process. RMs can have some limitations in their ability to specify every type of reward-worthy behaviour. Instead one can use formal languages, such as Linear Temporal Logic (LTL), for this part of the process. Then, the reward machines can be constructed from said formal specification.

For my thesis, I propose the integration of Reward Machines into Constrained Markov Decision Processes. The goal is to develop a safer and more efficient solution to the constrained RL control problem. The ideas is to use Linear Temporal Logic (LTL) to translate the reward functions into formal rewards and use it to construct an equivalent Reward Machine. I will be using the RM to transform the CMDP into a new framework called CMDP with a Reward Machine (CMDPRM) and, to solve the CMDPRM I will be developing a new algorithm called Constrained Learning for RM (CLRM) that will leverage the RM to achieve a faster and safer learning process. The goal is to test and evaluate the proposed methodology in the Safety Gym benchmark suite developed by Ray et al. [2019], which consists of high-dimensional continuous control environments meant to measure the performance and safety of agents in Constrained Markov Decision Processes.

2 Related Work

2.1 Reinforcement learning, Logic and Reward Machines

The use of Logic to help solve Reinforcement Learning control problems has been a popular topic in the last several years. Most of the work has been focused on how it can be used to make reinforcement learning safer and assist reinforcement learning to improve its performance.

In terms of safety, Li and Belta [2019] proposed combining co-safe Truncated Linear Temporal Logic (scTLTL) with control Lyapunov functions to improve exploration, and to incorporate control barrier functions to safeguard the exploration and deployment process. They developed a learnable system that allows users to specify tasks and constraints. De Giacomo et al. [2019] explored the concept of safety through "Restraining Bolts", a device that restricts an agents actions when connected to its systems. The proposed system is built upon the idea of having two independent sets of features returned from the world, one to the agent and another one to the $\text{LTL}_f/\text{LDL}_f$ restraining specifications. The introduction of $\text{LTL}_f/\text{LDL}_f$ changes the whole framework to a Non-Markovian Decision Process (NMDP). In order to be able to solve the new framework the $\text{LTL}_f/\text{LDL}_f$ restraining formulas are transformed into deterministic finite state automata tracking the stage of satisfaction of the formulas. This enables transforming the NMDP into an equivalent MDP that can be solved with a RL algorithm. De Giacomo et al. [2020] propose a new learning framework that combines Imitation Learning (IL) with Restraining Bolts (RB). Imitation Learning consists of generating a reward function based on a set of traces captured by an expert agent. A limitation of IL is that the expert and the learner can be different types of agents, this could mean that the learner is not capable of interpreting the traces generated by the expert. To address this issue the authors propose the use of RB. De Giacomo et al. [2019] have proven that RB can help an RL agent learn from different representations that don't have explicit mappings. De Giacomo et al. [2020] use this principle to develop a new methodology that uses IL to produce a logical specification of the reward function and incorporate it into a Restraining Bolt to facilitate the agent's learning process. Hasanbeig et al. [2020] introduced the concept of safe padding, which allows an agent to learn an optimal policy while guaranteeing its safety. The main idea is for the agent to have limited knowledge of its own dynamics, it starts performing exploratory cautious actions, and gradually, in line with the growing confidence about the environment obtained from observations, the range of acceptably safe actions grows, and the uncertain component of the dynamics becomes known. This approach limits the exploration, so Hasanbeig et al. [2020] used LTL formulas in order to specify tasks to automatically provide reward shaping and task decomposition, and enable optimal learning with limited exploration.

Icarte et al. [2018b] proposed the use of advice to help guide the agent through a more efficient exploration, and they combine advice specified as an LTL formula, with a new version of the R-MAX RL algorithm which has the ability to be guided by said advice. Their results show that good advice seems to be able to reduce the exploration needed to learn a optimal policy. Icarte et al. [2018a] introduced the concept of Reward Machines, a type of finite state machine used to specifying reward and task decomposition. When paired with their Q-Learning for Reward Machines algorithm it can appropriately decomposes the tasks while simultaneously learn sub-policies for the different components and find better policies more quickly than other traditional Reinforcement Learning algorithms. Camacho et al. [2019] built on the work presented by Icarte et al. [2018a] by showing that a reward specified in any number of formal languages (LTL, LDL, Golog, PLTL, Regular Expressions, etc.) can be translated into a Reward Machine and how the reward machine can be exploited by a tailored q-learning algorithm to improve the sample efficiency compared to traditional reinforcement learning algorithms. Illanes et al. [2020] proposed combining high-level symbolic planning models and automated synthesis techniques with RL techniques. An approximated understanding of the environment can be characterized as a symbolic planning model, while leaving possibly complex low level aspects of the environment unspecified. The RL agent can improve sample efficiency as the high level plans can be used for transferring learning from previously learned policies, and the agent can learn complex low-level control policies as it relies on model-free RL to accommodate for all the information missing in the high level model. More work in this field has been done by Icarte et al. [2019b], Camacho and McIlraith [2019b], and Payani and Fekri [2020].

2.2 Constrained Reinforcement Learning

The application of constraints has been a focus of research in the field of reinforcement learning. They are a natural and widely relevant way for safety criteria to be formulated which makes it highly appealing when working on concerns of safety in RL control problems. In recent years there has been a surge of work focused on developing novel algorithms dedicated to solve constrained RL problems.

Achiam et al. [2017] introduced Constrained Policy Optimization (CPO), it was the first general-purpose policy search algorithm for constrained RL where at each iteration, guarantees of near-constraint satisfaction were achieved. Dalal et al. [2018] address the constrained RL problem by adding to the deep policy network a safety layer that analytically solves an action correction formulation per each state. Bohez et al. [2019] propose a new reinforcement learning technique that employs Lagrangian relaxation to learn the Lagrangian multipliers for the optimization and the parameters of a control policy that satisfies the constraints. In addition, they showed that one can meet constraints either in anticipation or in a per-step manner, and can also learn a single strategy that is able to trade between return and cost dynamically. Chow et al. [2019] presented a safe policy optimization algorithm based on a Lyapunov approach. The algorithm can use any standard policy gradient approach to train a deep neural network policy, while ensuring near-constraint satisfaction for each policy update by projecting either the policy parameter or the action on the set of feasible solutions induced by the state-dependent linearized Lyapunov constrains. More work in this field has been done by Saunders et al. [2017] and Wang et al. [2019].

3 Proposed Research

My thesis focuses on developing a safer and more efficient solution to the constrained RL control problem by using Linear Temporal Logic (LTL) formulas and Reward Machines with a new constrained learning algorithm developed specifically for Constrained Markov Decision Processes (CMDP). On one hand I am using LTL to specify the reward structure as a Reward Machine and use it to define a new framework called CMDP with RM (CMDPRM). On the other hand I am developing a novel Contrained Learning algorithm called Constrained Learning for RM (CLRM) that will use off-policy learning and is capable of solving the control RL problem in the CMDPRM.

3.1 CMDP with Reward Machines

In order to express a reward function that reflects complex reward-worthy behaviour I am using Linear Temporal Logic formulae to encode the complex task we want the agent to learn and use it to define the reward function as a Reward Machine which will also include the constraints provided by the CMDP. The RM is a type of finite state machine that will allow the agent to decompose the task (and specify multiple tasks) allowing it to learn an optimal policy while minimizing exploration. In order to be able to exploit the features of RMs in a CMDP I will have to map it to a CMDP with Reward Machines (CMDPRM) that contains all the elements of a CMDP plus the elements of a RM with a labelling function that will act as a liaison between them. The LTL formulas and the reward machines will also contain the constraints. In addition to returning the appropriate reward the reward machine will also return the appropriate constraint value when necessary. This will be used to learn the expected cost at every state in parallel to the expected reward. This will be the first time the advantages that Logic and Reward Machines bring have been leveraged to solve CMDPs.

3.2 Constrained Learning for Reward Machines

In order to solve the control problem defined by the CMDPRM I develop a tailored algorithm based on traditional constrained learning algorithm for RL called Constrained Learning for Reward Machines (CLRM). As mentioned earlier an advantage of Reward Machines is its ability to specify the structure of complex tasks. The goal is for CLRM to decompose the tasks represented by the RM and apply off-policy learning to learn sub-policies in parallel for the different components, this will allow the agent to learn better policies while minimizing the exploration. On top of that, during each iteration CLRM will define a set of safe states that the agent can transition to based on the expected cost that will also be learnt in parallel thanks to the reward machine.

4 Testing and Evaluation Methods

In order to prove that my novel approach is a viable solution for safety RL control problems I will be working with Open AI's Safety Gym benchmark suite developed by Ray et al. [2019]. It consists of an environment-builder that allows a user to create custom high-dimensional continuous control environments. One can mix and match from a wide range of agents, tasks, goals, and safety requirements, it is meant to be used to measure the performance and safety of agents in CMDPs.

The agents perceive the world through a robot's sensors and communicates with the world through its actuators. The available robots are Point, a basic 2D-plane confined robot, Car, a robot with two parallel wheels and Doggo, a bilateral symmetric quadrupedal robot.

Three unique tasks are currently provided by the Safety Gym environment-builder. The reward features can be customised to allow rewards to be either sparse or dense. The available tasks are Goal, a series of goal positions, Button, a set of highlighted buttons that are meant to be clicked in sequence, and Push, a box that has to be moved to a variety of goal location.

Safety Gym has five types of elements related to safety criteria. These elements can be mixed and matched freely. The available safety elements are Hazards, a danger area in the environment, Vases a set of objects that should be avoided, Pillars, immobile objects that should be avoided, Buttons, a fake button that act as a fake goal for the Button task that should not be clicked, and Gremlins, a moving object that should not be collided with.

4.1 Evaluation

In order to determine the viability of the proposed system I will evaluate the algorithm I am developing with other algorithms that are currently used in the RL field. To determine it's usefulness for safety concerns I will be comparing average episodic return, average episodic sum of costs and average cost over the training period. I will also be comparing the sampling efficiency of the algorithms.

5 Thesis Outline

1. Introduction
2. Related Work
   - 2.1. Reinforcement Learning and Formal Logic
   - 2.2. Constrained Reinforcement Learning
3. Background Knowledge
   - 3.1. Reinforcement learning
   - 3.2. Linear Temporal Logic
4. Reward Specification in Constrained Markov Decision Processes
   - 4.1. CMDP with Reward Machines
   - 4.2. Constrained Q-learning for Reward Machines
5. Experimental Evaluation
   - 5.1. Baseline Algorithms
   - 5.2. Test Set up
   - 5.3. Results and evaluation
6. Conclusion
7. References

References

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International Conference on Machine Learning*, pages 22–31. PMLR, 2017.

Eitan Altman. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.

Steven Bohez, Abbas Abdolmaleki, Michael Neunert, Jonas Buchli, Nicolas Heess, and Raia Hadsell. Value constrained model-free continuous control. *arXiv preprint arXiv:1902.04623*, 2019.

Alberto Camacho and Sheila A McIlraith. Learning interpretable models expressed in linear temporal logic. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 29, pages 621–630, 2019b.

Alberto Camacho, Rodrigo Toro Icarte, Toryn Q Klassen, Richard Anthony Valenzano, and Sheila A McIlraith. Ltl and beyond: Formal languages for reward function specification in reinforcement learning. In *IJCAI*, volume 19, pages 6065–6073, 2019.

Yinlam Chow, Ofir Nachum, Aleksandra Faust, Edgar Duenez-Guzman, and Mohammad Ghavamzadeh. Lyapunov-based safe policy optimization for continuous control. *arXiv preprint arXiv:1901.10031*, 2019.

Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

Giuseppe De Giacomo, Luca Iocchi, Marco Favorito, and Fabio Patrizi. Foundations for restraining bolts: Reinforcement learning with $\text{LTL}_f/\text{LDL}_f$ restraining specifications. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 29, pages 128–136, 2019.

Giuseppe De Giacomo, Marco Favorito, Luca Iocchi, and Fabio Patrizi. Imitation learning over heterogeneous agents with restraining bolts. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 30, pages 517–521, 2020.

Mohammadhosein Hasanbeig, Alessandro Abate, and Daniel Kroening. Cautious reinforcement learning with logical constraints. *arXiv preprint arXiv:2002.12156*, 2020.

Rodrigo Toro Icarte, Toryn Klassen, Richard Valenzano, and Sheila McIlraith. Using reward machines for high-level task specification and decomposition in reinforcement learning. In *International Conference on Machine Learning*, pages 2107–2116, 2018a.

Rodrigo Toro Icarte, Toryn Q Klassen, Richard Anthony Valenzano, and Sheila A McIlraith. Advice-based exploration in model-based reinforcement learning. In *Canadian Conference on Artificial Intelligence*, pages 72–83. Springer, 2018b.

Rodrigo Toro Icarte, Ethan Waldie, Toryn Klassen, Rick Valenzano, Margarita Castro, and Sheila McIlraith. Learning reward machines for partially observable reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 15523–15534, 2019b.

León Illanes, Xi Yan, Rodrigo Toro Icarte, and Sheila A McIlraith. Symbolic plans as high-level instructions for reinforcement learning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 30, pages 540–550, 2020.

Xiao Li and Calin Belta. Temporal logic guided safe reinforcement learning using control barrier functions. *arXiv preprint arXiv:1903.09885*, 2019.

Ali Payani and Faramarz Fekri. Incorporating relational background knowledge into reinforcement learning via differentiable inductive logic programming. *arXiv preprint arXiv:2003.10386*, 2020.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 2019.

William Saunders, Girish Sastry, Andreas Stuhlmueller, and Owain Evans. Trial without error: Towards safe reinforcement learning via human intervention. *arXiv preprint arXiv:1707.05173*, 2017.

Weixun Wang, Junqi Jin, Jianye Hao, Chunjie Chen, Chuan Yu, Weinan Zhang, Jun Wang, Xiaotian Hao, Yixi Wang, Han Li, et al. Learning adaptive display exposure for real-time advertising. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2595–2603, 2019.