Inferring and Obeying Norms in Temporal Logic

Extended Abstract

Daniel Kasenberg
Tufts University
Medford, Massachusetts
dmk@cs.tufts.edu

CCS CONCEPTS

• **Computing methodologies** → *Apprenticeship learning*; **Markov decision processes**; • **Theory of computation** → *Modal and temporal logics*;

KEYWORDS

Moral and social norms; temporal logic; Markov Decision Processes

ACM Reference Format:
Daniel Kasenberg. 2018. Inferring and Obeying Norms in Temporal Logic: Extended Abstract. In *HRI '18 Companion: 2018 ACM/IEEE International Conference on Human-Robot Interaction Companion, March 5–8, 2018, Chicago, IL, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3173386.3176914

1 INTRODUCTION

Robots and other artificial agents are increasingly being considered in domains involving complex decision-making and interaction with humans. These agents must adhere to human moral social norms: agents that fail to do so will be at best unpopular, and at worst dangerous. Artificial agents should have the ability to learn (both from natural language instruction and from observing other agents' behavior) and obey multiple, potentially conflicting norms.

One popular candidate solution to the problem of learning moral and social norms from behavior is *inverse reinforcement learing* (IRL) [10]. By observing the behavior of agents in stochastic environments, IRL algorithms can determine a reward function that "best explain" that behavior. IRL and reward-driven planning could solve the twin problems of learning (from observation) and obeying moral and social norms. IRL may, however, be inadequate for AI ethics due to the following challenges (as we have described in [2]):

(C1) Some temporally complex moral and social norms rely on information about the agent's past history that may not be encoded in the state space, and thus cannot be represented by a reward function; IRL cannot infer these norms.
(C2) Reward functions learned in one domain may not be easily transferable to other domains; and
(C3) It is often difficult to interpret the reward functions inferred by IRL; interpretability is a desirable property for AI ethics (e.g., for correction).

HRI '18 Companion, March 5–8, 2018, Chicago, IL, USA

ACM ISBN 978-1-4503-5615-2/18/03.
https://doi.org/10.1145/3173386.3176914

Our research seeks to overcome these challenges by allowing robots in stochastic domains, such as Markov Decision Processes (MDPs), to learn (by observing other agents' behavior) and obey moral and social norms represented in temporal logic.

2 RELATED WORK

Most "top-down" approaches to morality (those that explicitly represent moral injunctions and norms) employ deontic logics, some of which support complex temporal sequencing of actions and propositions (e.g. [1]). A number of papers address the challenge of conflicting norms in single- and multi-agent domains. Some (such as [11]) solve conflicts by directly modifying norms themselves; others (such as [3]) employ nonmonotonic logics that avoid many of the logical problems with conflicting obligations. These methods generally assume deterministic environments, and thus may not be well-equipped for probabilistic domains.

Our planning algorithm extends work employing LTL specifications in MDPs (e.g., [6]) to manage multiple conflicting norms. Our norm inference work extends [4], which infers temporal logic specifications from deterministic domains.

3 NORMS IN TEMPORAL LOGIC

Linear temporal logic (LTL) is a propositional logic augmented with the operators **X**, **G**, **F**, and **U**. $\mathbf{X}\phi$ means "in the next time step, ϕ"; $\mathbf{G}\phi$ means "in all present and future time steps, ϕ"; **F** means "in some present or future time step, ϕ"; and $\phi_1 \mathbf{U} \phi_2$ means "ϕ_1 will hold until ϕ_2 is true".

Encoding norms in LTL overcomes the challenges (C1)-(C3):

- LTL can represent temporally complex concepts that cannot be represented by (Markovian) reward functions. For example, [7] describe a robot elder care domain. The robot may be obligated to give care to the person, but only if the person consents to that care. The person's desire (or lack thereof) for care may be expressed only occasionally, and the robot is expected to remember the person's last preference. The desired behavior cannot be expressed as a reward function, but can be expressed by the following LTL statement:

$$\mathbf{G}((consentGiven \rightarrow ((\mathbf{X}careGiven)\mathbf{U}consentWithdrawn)) \wedge (consentWithdrawn \rightarrow ((\mathbf{X}\neg careGiven)\mathbf{U}consentGiven)) \quad (1)$$

- LTL statements may be transfered to new domains, with previously-unseen states and actions, as long as the set of propositions can be mapped onto the new states.
- Given the meanings of the propositions, an LTL statement may be easily interpreted. For example, the meaning of (1) is "whenever consent is given, care is given until consent

is withdrawn; whenever consent is withdrawn, care is not given until consent is given".

4 RESULTS

4.1 Planning to obey conflicting norms

We define a norm system $\mathcal{N}$ as a set of *weighted* norms (LTL statements):

$$\mathcal{N} = \{(w_1, \phi_1), \cdots, (w_n, \phi_n)\} \tag{2}$$

The weight w_i represents the importance the agent ascribes to ϕ_i.

Given a norm system $\mathcal{N}$, we have developed an algorithm allowing an agent to "maximally satisfy" $\mathcal{N}$ [9] . We define "maximally satisfying" a norm system in terms of a notion of *violation cost*.

Given an infinite agent behavior trajectory $\tau = s_0, a_0, s_1, a_1, \cdots$, we define the violation cost of the trajectory with respect to a norm ϕ as the (discounted) minimum number of time steps that must be *omitted* from τ in order for τ to satisfy ϕ. The violation cost of a trajectory with respect to a norm system is then the weighted sum of the violation costs with respect to the norms.

Each LTL norm can be shown to correspond to a deterministic Rabin automaton (DRA), a finite state machine over infinite words. Violation cost may be implemented by augmenting the DRA corresponding to each norm with self-loops (corresponding to omitting one time step), but causing the agent to incur a cost whenever these transitions are followed. When we compute the Cartesian product of these augmented DRAs with the original MDP (the *product MDP*), the expected violation cost is Markovian, and the problem may be solved with value iteration (there are some graph-theoretic caveats, which are explained in [9]). The optimal policy is non-stationary in the original MDP, but is stationary in the product MDP.

We have tested this approach in several (simulated) domains, and in each case found agent behavior to match intuitions about the "right" behavior in the domains.

4.2 Inferring norms from behavior

Given an MDP with known dynamics and a set $\tau^1, \cdots, \tau^m$ of agent trajectories, we developed an algorithm for inferring an LTL statement that "best explains" the observed trajectories [8]. The algorithm is based on the following principles:

- Simpler norms are preferred to more complex norms; and
- Norms that "specifically" describe behaviors are preferred. If a norm ϕ is hard to satisfy randomly, but is satisfied by trajectories $\tau^1, \cdots, \tau^m$, then ϕ explains $\tau^1, \cdots, \tau^m$ well.

This can be framed as a multi-objective optimization problem

$$\min_{\phi \in LTL} (Obj^S(\phi), Obj^X(\phi)) \tag{3}$$

where Obj^S measures formula complexity (in particular, $Obj^S(\phi) = \ell(\phi)$, the length of ϕ in symbols where each proposition, connective, and operator counts as one symbol). Obj^X measures this notion of "specific description" and is given by

$$Viol_\phi(\pi^o) - Viol_\phi(\pi^{rand}) \tag{4}$$

where $Viol$ is the notion of violation cost defined earlier, π^o is a (product-space) policy constructed from the trajectories $\tau^1_\otimes, \cdots, \tau^m_\otimes$, and π^{rand} is the uniformly random policy.

This problem may be solved using any approach for multi-objective optimization that can operate over the syntax of LTL (we used NSGA-II [5]). This yields a set of Pareto-efficient norms, from which the preferred norm may be selected.

We tested this approach in several simple domains, and found that it retrieved norms that well described the input trajectories.

5 FUTURE WORK

We aim to develop more efficient implementations, enabling real-time planning and inference (e.g., our planning algorithm takes exponential time in the number of norms). These steps will enable us to implement our algorithms on robots and perform human-subject experiments.

We aim to allow more sophisticated reasoning about norms by adding deontic operators to our formulation. We also aim to augment the logic to allow quantification over objects.

6 CONCLUSION

To interact in morally and socially acceptable ways with humans, robots will need to learn and obey moral and social norms. We seek to facilitate these capabilities using the language of temporal logic and the framework of Markov Decision Processes. We have initiated our pursuit of this goal by developing algorithms for planning to obey sometimes-conflicting norms in stochastic domains, and for inferring temporal logic norms from observed agent behaviors.

7 ACKNOWLEDGEMENTS

This project was supported in part by ONR MURI grant N00014-16-1-2278 and by NSF IIS grant 1723963.

REFERENCES

[1] Thomas Ågotnes, Wiebe Van Der Hoek, Juan A Rodríguez-Aguilar, Carles Sierra, and Michael Wooldridge. 2007. On the Logic of Normative Systems.. In *IJCAI*, Vol. 7. 1181–1186.

[2] Thomas Arnold, Daniel Kasenberg, and Matthias Scheutz. 2017. Value Alignment or Misalignment–What Will Keep Systems Accountable?. In *3rd International Workshop on AI, Ethics, and Society*.

[3] Mathieu Beirlaen, Christian Straßer, and Joke Meheus. 2013. An inconsistency-adaptive deontic logic for normative conflicts. *Journal of Philosophical Logic* (2013), 1–31.

[4] Daniil Chivilikhin, Ilya Ivanov, and Anatoly Shalyto. 2015. Inferring Temporal Properties of Finite-State Machine Models with Genetic Programming. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation*. ACM, 1185–1188.

[5] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation* 6, 2 (2002), 182–197.

[6] Xu Chu Dennis Ding, Stephen L Smith, Calin Belta, and Daniela Rus. 2011. LTL control in uncertain environments with probabilistic satisfaction guarantees. *IFAC Proceedings Volumes* 44, 1 (2011), 3515–3520.

[7] Daniel Kasenberg, Thomas Arnold, and Matthias Scheutz. 2018. Norms, Rewards, and the Intentional Stance: Comparing Machine Learning Approaches to Ethical Training. In *Proceedings of the First AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society*.

[8] Daniel Kasenberg and Matthias Scheutz. 2017. Interpretable Apprenticeship Learning with Temporal Logic Specifications. In *Proceedings of the 56th IEEE Conference on Decision and Control (CDC)*.

[9] Daniel Kasenberg and Matthias Scheutz. 2018. Norm conflict resolution in stochastic domains. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*.

[10] Andrew Y Ng, Stuart J Russell, et al. 2000. Algorithms for inverse reinforcement learning.. In *Icml*. 663–670.

[11] Wamberto W Vasconcelos, Martin J Kollingbaum, and Timothy J Norman. 2009. Normative conflict resolution in multi-agent systems. *Autonomous agents and multi-agent systems* 19, 2 (2009), 124–152.