Decision Theoretic Norm-Governed Planning

(Extended Abstract)

Luca Gasparini, Timothy J. Norman,
Martin J. Kollingbaum
Dept. of Computing Science
University Of Aberdeen, UK

Liang Chen
School of Computing and Engineering
University of West London, UK

ABSTRACT

We propose Normative Dec-POMDPs, a model of collective decision making in the presence of complex norms, with violations of norms classified according to their relative severity. We extend the PBPG algorithm in order to solve Normative Dec-POMDPs and propose a heuristic that improves its scalability without affecting the policy quality.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Coherence and coordination

Keywords

Norms, Coordination, Decision Theory

1. INTRODUCTION

Existing approaches to norm-directed collective decision making often consider limited representations of norms, such as labelling of states with violations. This is insufficient even for some of the most common types of norms, such as separation and binding of duties, and obligations with deadlines. Compliance with these norms must be evaluated based on an execution history, rather than on a single state. Moreover, violations are often modelled as a loss of utility on a real scale. This leaves the way open to significant fallacies in reasoning. Consider, for example, confidentiality levels for documents and assume that a punishment for disclosing a "Top Secret" document would be ten times as costly as that for the disclosure of a "Secret" document. Should we infer that 10 disclosures of secret documents is as severe as that of a top secret one? A more natural means to model norm violations in these cases is as a partial order, where violations lie at different qualitative levels of severity.

We develop mechanisms that enable a coalition of agents to coordinate their activity in order to maximize their compliance level. We represent this problem as an extension of the Decentralized Partially Observable MDP (Dec-POMDP), denoted as Normative Dec-POMDP (N-Dec-POMDP), which allows us to reason about decision making in the presence of complex norms with violations of varying severity.

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

2. NORMATIVE DEC-POMDPS

Dec-POMDPs model distributed decision making problems where multiple agents, each with a different view of the environment, must coordinate in order to maximise a joint-reward [3]. A Dec-POMDP is as a tuple, $\langle I, S, b^0, \{A_i\}, P, \{\Omega_i\}, O, R\rangle$ where I is a set of agents, and S a set of states; b^0 is a probability distribution over possible initial states; A_i the actions available to agent i and $\vec{a} = \langle a_1, \ldots, a_n\rangle$ a joint-action, one action for each agent; $P(s_j|s_i, \vec{a})$ is the probability that taking $\vec{a}$ in s_i will result in a transition to s_j; Ω_i are observations available to agent i and $\vec{\Omega}$ is the set of joint observations $\vec{o}$, one observation for each agent; $O(\vec{o} \mid \vec{a}, s_j)$ is the probability of observing $\vec{o}$ when taking $\vec{a}$ and transitioning to s_j; $R(s_i, \vec{a})$ is the reward obtained by performing $\vec{a}$ in s_i. A policy for agent i maps sequences of interleaved local actions and observations to local actions. Solving a Dec-POMDP means finding a joint-policy (one policy per agent) that maximises the expected total reward. Algorithms to approximately solve Dec-POMDPs rely on the fact that the transition probability depends only on the current state and joint-action (*Markovian property*). As discussed above, the evaluation of some complex norms depends on the execution history and is, therefore, non-Markovian. A number of norm formalisms (e.g. [2]) rely on a *normative configuation* to keep track of the evolution of norm instances (activation, expiration, etc.) as the system evolves. Algorithm 1 extends the state of a Dec-POMDP to include a normative configuration (nc_i) so that norm evaluation becomes Markovian. Let $\langle s_i, nc_i\rangle$ denote a normative state, consisting of the state of a Dec-POMDP (the factual state) and a normative configuration. We assume a function $viol(ns_i)$ that returns the violations detected in ns_i and a function upd_η that updates the normative configuration after a transition of the factual state. Let s^t and nc^t denote, respectively, the factual state and the normative configuration at time t, and η a set of norms. $upd_\eta(s^t, s^{t+1}, nc^t)$ denotes the normative configuration nc^{t+1} of the system after a transition from $\langle s^t, nc^t\rangle$ to a normative state with factual state s^{t+1}. The implementation of upd_η will depend on the chosen normative fomalism. We consider each possible execution, starting from every possible state (with an initial configuration nc_0), and update the configuration as we transition between states. We assume that observations depend only on factual states.

We now specify a partial order among norms and use it to define a utility function that captures the norms and their severity. Given two norms nd_k and nd_l, $nd_k \succ_n nd_l$ means that a violation of nd_k is more severe than any number of violations of nd_l. Let ϕ_k denote a violation of nd_k, and Φ_i

Algorithm 1 Generation of a N-Dec-POMDP

Input: $I, S, \{A_i\}, P, \{\Omega_i\}, O, \eta$
Output: NS_*, P_*, O_*

$toProc = \emptyset$, $NS_* = \emptyset$
for all $s_i \in S$ **do**
  add $\langle s_i, upd_\eta(s_i, s_i, nc_0)\rangle$ to NS_* and $toProc$
end for
while $toProc$ is not empty **do**
  $ns_i = \langle s_i, nc_i\rangle \leftarrow$ remove first element of $toProc$
  for all $\vec{a}_j \in A_0 \times \cdots \times A_n$ **do**
    for all $s_k \in S$ s.t. $P(s_k|s_i, \vec{a}_j) > 0$ **do**
      $ns_l \leftarrow \langle s_k, upd_\eta(s_i, s_k, nc_i)\rangle$
      $P_*(ns_l|ns_i, \vec{a}_j) \leftarrow P(s_k|s_i, \vec{a}_j)$
      if $ns_l \notin NS_*$ **then**
        $toProc \leftarrow toProc \cup \{ns_l\}$; $NS_* \leftarrow NS_* \cup \{ns_l\}$
      end if
    end for
    for all $\vec{o}_m \in \Omega_0 \times \cdots \times \Omega_n$ **do**
      $O_*(\vec{o}_m|\vec{a}_j, ns_i) \leftarrow O(\vec{o}_m|\vec{a}_j, s_i)$
    end for
  end for
end while

a violation set. We use this partial order to partially order violation sets. Φ_i is more severe than Φ_j ($\Phi_i \succ_v \Phi_j$) iff:

$$\exists \phi_k \in \Phi_i : \phi_k \notin \Phi_j \text{ and } \forall \phi_l \in \Phi_j : \phi_l \in \Phi_i \text{ or } nd_k \succ_n nd_l$$

Let Rk be a function that takes a violation set and gives its position in a ranking from the least to the most severe: (i) $Rk(\Phi_i) = 1$ if there is no $\Phi_j \in 2^\Phi$ such that $\Phi_i \succ_v \Phi_j$. (ii) $Rk(\Phi_i) = max\{Rk(\Phi_j) : \Phi_i \succ_v \Phi_j\} + 1$, otherwise. We refer to $Rk(ns_i) := Rk(viol(ns_i))$ as the *severity level* of ns_i. Bonet and Pearl [1] developed a qualitative theory of MDPs based on an order of magnitude approximation for utilities and probabilities. Given that ϵ represents a small unknown quantity, they define extended reals $\psi \in \Psi$ as infinite series $\psi = \sum_k c_k \epsilon^k$. They define operations over Ψ and, given a large parameter ρ, the magnitude of an extended real as $\|\psi^i\|_\rho := \sum_k |c_k| \rho^{-k}$. We use Ψ to represent the cost of a set of violations. Let mS be the maximum severity level. Equation 1 can be interpreted as the agents incurring in a higher cost for visiting states with higher severity.

$$\forall\ ns_i \in NS, \vec{a}_j \in \vec{A} : R(ns_i, \vec{a}_j) = -\epsilon^{mS - Rk(ns_i)} \quad (1)$$

3. SOLVING N-DEC-POMDPS

PBPG [3] may be adapted to solve N-Dec-POMDPs with extended reals rewards. PBPG heuristically identifies the beliefs that are most likely to be encountered and optimises policies against them. We propose the Most-Critical-States (MCS) heuristics that rely on the utility function structure to further restrict these beliefs and improve scalability. Given an initial state ns_i, MCS simulates the execution of the centralized MDP policy in order to identify reachable states ($R^t_{MDP}(ns_i)$) and estimate the probability of reaching each state ns_j ($pr_{ns_i}(ns_j)$). We use the MDP value function (V_{MDP}) to estimate what states are likely to lead to severe violations. Given a threshold $th \in \Psi$ and an initial state ns_i, $mc^{ns_i}_{th}$ is the set of $ns_j \in R^t_{MDP}(ns_i)$ s.t. the product of $V_{MDP}(ns_j)$ and $pr_{ns_i}(ns_j)$ is less than th. MdpMCS is the heuristics that selects beliefs b according to Equation 2. An improved heuristics (MixedMcs) uses a joint-policy to sample reachable states and V_{MDP} to evaluate them.

$$b(ns_j) = \begin{cases} \frac{pr_{ns_i}(ns_j)}{\sum_{ns_k \in mc^{ns_i}_{th}} pr_{ns_i}(ns_k)} & \text{if } ns_j \in mc^{ns_i}_{th} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Table 1: Generation and Policy Computation

		Standard		MCS		Generation		
Ag	**B**	**Value**	**Time[s]**	**Value**	**Time[s]**	**DP**	**NDP**	**Time[s]**
Initial Step								
2	2	-4.89e-4± 1.37e-3	34± 1	-4.89e-4± 1.37e-3	31± 1	223	2128	18
2	3	-1.98e-4± 3.38e-4	1415± 142	-3.32e-5± 1.23e-4	1041± 97	781	28360	738
3	2	-4.11e-6± 8.42e-6	1155± 81	-4.28e-6± 1.11e-5	794± 61	1141	8445	250
Improvement Step								
2	2	-1.79e-20± 2.91e-21	43± 1	-1.79e-20± 2.91e-21	37± 1	223	2128	18
2	3	-2.73e-20± 1.24e-23	1184± 143	-2.73e-20± 1.23e-23	813± 50	781	28360	738
3	2	-1.44e-21± 7.58e-24	1172± 65	-1.44e-21± 7.58e-24	827± 35	1141	8445	250

We evaluated our approach on the sea-guard scenario presented in [2], where a set of agents must surveil a restricted area and intercept any unauthorized access before a deadline. Four norms, specified in CÒIR [2], are included in the scenario. Agents observe their locations and, by monitoring the area, increase their chances of detecting unauthorized boats. Table 1 presents results for different instantiations of the scenario, with varying number of agents (**Ag**) and unauthorized boats (**B**). **Generation** shows the state-space size before (**DP**) and after (**NDP**) the expansion and the execution time for the generation. The left part of the table compares MCS with heuristics that do not use the severity-based pruning. The table shows average (over 20 runs) time for the computation of a policy and the magnitude ($\rho = 10^5$) of their values. **Initial Step** compares MdpMCS with a standard Mdp-based heuristics and **Improvement Steps** uses the same policy to evaluate MixedMcs against a standard Dec-POMDP based heuristics on the same scenarios. We tested these results for significance and observed significant difference in the execution times but not in the values.

4. CONCLUSION

We have presented N-Dec-POMDP, a formalism for modelling collective decision making in the presence of complex norms of varying severity. We represent norm violations using a qualitative decision theory, and adapted PBPG to solve N-Dec-POMDPs. Our heuristic, MCS, increases PBPG scalability while preserving policy quality.

Acknowledgements

This research was sponsored by Selex ES.

REFERENCES

[1] B. Bonet and J. Pearl. Qualitative mdps and pomdps: An order-of-magnitude approximation. In *Proc. of the 18th conference on Uncertainty in artificial intelligence*, pages 61–68. Morgan Kaufmann Publishers Inc., 2002.

[2] L. Gasparini et al. CÒIR : Verifying normative specifications of complex systems. In *Proc. of the 18h International Workshop on Coordination, Organisations, Institutions and Norms*, 2015.

[3] F. Wu, S. Zilberstein, and X. Chen. Point-based policy generation for decentralized POMDPs. In *Proc. of the 9th International Conference on Autonomous Agents and Multiagent Systems*, pages 1307–1314, 2010.