The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Preallocation and Planning Under Stochastic Resource Constraints

Frits de Nijs, Matthijs T. J. Spaan, Mathijs M. de Weerdt
{f.denijs, m.t.j.spaan, m.m.deweerdt}@tudelft.nl
Delft University of Technology, The Netherlands

Abstract

Resource constraints frequently complicate multi-agent planning problems. Existing algorithms for resource-constrained, multi-agent planning problems rely on the assumption that the constraints are deterministic. However, frequently resource constraints are themselves subject to uncertainty from external influences. Uncertainty about constraints is especially challenging when agents must execute in an environment where communication is unreliable, making on-line coordination difficult. In those cases, it is a significant challenge to find coordinated allocations at plan time depending on availability at run time. To address these limitations, we propose to extend algorithms for constrained multi-agent planning problems to handle *stochastic* resource constraints. We show how to factorize resource limit uncertainty and use this to develop novel algorithms to plan policies for stochastic constraints. We evaluate the algorithms on a search-and-rescue problem and on a power-constrained planning domain where the resource constraints are decided by nature. We show that plans taking into account all potential realizations of the constraint obtain significantly better utility than planning for the expectation, while causing fewer constraint violations.

Introduction

Planning for future uncertainties is an effective tool to increase the utility of a system of multiple agents. Particularly when the actions of agents are restricted by scarce resources, planning for resource usage is an important challenge that many authors have addressed (Adelman and Mersereau 2008; Agrawal, Varakantham, and Yeoh 2016; De Nijs, Spaan, and De Weerdt 2015; Gordon et al. 2012; Meuleau et al. 1998; Wu and Durfee 2010; Yoo, Fitch, and Sukkarieh 2012). These approaches have in common that they consider uncertainty in state transitions, while assuming full knowledge about future resource constraints.

However, resource capacity may itself be subject to uncertainty. For example, the amount of power produced from renewable sources such as wind turbines is a stochastic quantity (Klöckl, Papaefthymiou, and Pinson 2008). Similarly, when only a subset of agents participate in a traffic congestion control system, the non-participants contribute to congestion stochastically (De Weerdt et al. 2013). Another source of resource uncertainty may occur when an agent's consumption itself is stochastic (Mausam et al. 2005; Schaffer, Clement, and Chien 2005). Nevertheless, no earlier work has addressed multi-agent planning for such stochastic resources.

In several application domains where multiple constrained agents must coordinate their actions, there may be known fixed periods where communication between them is impossible (such as with non-geostationary satellites), unadvisable (such as in warfare), or too uncertain (as in hazardous environments). In other domains the required response time for actors maybe so short that planning and coordination needs to be done a priori, such as in robot soccer, high-frequency trading in multiple stock markets, or protection control in electricity distribution networks. In all of these situations, an approach is needed where coordinated policies are computed for a number of sequential decisions that are taken without further communication. Therefore, in this work we focus on *preallocation* algorithms, which compute policies for a given plan horizon by allocating resources to agents a priori, thereby effectively decoupling the agents' policies so that they can be computed and executed independently.

Decoupling necessarily introduces an error, as agents cannot respond to non-local realizations of uncertain transitions. In this work, however, we show how to permit effective decoupling even in the case of a tight and stochastic coupling constraint. We extend Multi-agent Markov Decision Processes (MMDPs) by a model of the resource constraint realizations in a separate, orthogonal part of the state space. This enables us to formulate novel approaches based on two state-of-the-art planning algorithms that can deal with deterministic resource constraints. These algorithms represent different solution categories: an optimal preallocation mixed-integer linear program (Wu and Durfee 2010) which restricts worst-case consumption, and the constraint relaxation approach Constrained Markov Decision Process (Altman 1999) restricting average consumption.

We evaluate the benefit of planning for stochastic resource constraints for both approaches by comparing to the state of the art—i.e., planning for the mean constraint level—on a coordinated search-and-rescue domain, demonstrating the need to handle stochastic resource constraints. Subsequently, we use a heater planning domain to demonstrate the scalability of the approximations and their reduced resource violation frequency in larger problems. We show that agents taking

into account all potential realizations of the resource limit obtain significantly better policies. Finally, we show that the number of resource violations further decreases with more frequent replanning.

Background

A Multi-agent Markov Decision Process (MMDP) models a system consisting of n cooperative agents operating under uncertainty (Boutilier 1996). Time in a finite-horizon MMDP is discretized into h time steps. At each step the state $\vec{\mathbf{s}}$ of the system describes all the relevant properties of all agents. We require that the set S of possible discrete states of the system is finite and known. A decision or action $\vec{\mathbf{a}} = \langle a_1, \ldots, a_n \rangle$ of the agents describes for each agent i the selected control input a_i. The finite set A contains all potential joint actions. For any given state-action pair, the transition function $T : S \times A \times S \to [0, 1]$ gives the probability of reaching potential future state $\vec{\mathbf{s}}'$. The performance of the agents is measured by a reward function $R : S \times A \to \mathbb{R}$ which assigns a real-valued instantaneous reward for every state-action pair. Tuple $\langle S, A, T, R, h \rangle$ fully specifies an MMDP.

The goal of planning for an (M)MDP is to compute the best action to take in order to obtain the highest possible expected value, as defined through the Bellman equation (1957). The optimal expected value function V is defined as

$$V[h, \vec{\mathbf{s}}] = \max_{\vec{\mathbf{a}} \in A} R\big(\vec{\mathbf{s}}, \vec{\mathbf{a}}\big), \qquad \forall \vec{\mathbf{s}}$$

$$V[t, \vec{\mathbf{s}}] = \max_{\vec{\mathbf{a}} \in A} \Big(R\big(\vec{\mathbf{s}}, \vec{\mathbf{a}}\big) + Q[t, \vec{\mathbf{s}}, \vec{\mathbf{a}}] \Big), \qquad 1 \leq t < h, \forall \vec{\mathbf{s}}$$

$$Q[t, \vec{\mathbf{s}}, \vec{\mathbf{a}}] = \sum_{\vec{\mathbf{s}}' \in S} T(\vec{\mathbf{s}}, \vec{\mathbf{a}}, \vec{\mathbf{s}}') \cdot V[t+1, \vec{\mathbf{s}}']. \qquad 1 \leq t < h, \forall \vec{\mathbf{s}}, \vec{\mathbf{a}}$$

A planner intends to find a policy $\pi : \{1, \ldots, h\} \times S \to A$ mapping states to actions that maximizes the expected value of the agents' rewards over the horizon. Given a policy π, we define the expected value V_π of following that policy analogously as

$$V_\pi[h, \vec{\mathbf{s}}] = R\big(\vec{\mathbf{s}}, \pi(t, \vec{\mathbf{s}})\big), \qquad \forall \vec{\mathbf{s}}$$

$$V_\pi[t, \vec{\mathbf{s}}] = R\big(\vec{\mathbf{s}}, \pi(t, \vec{\mathbf{s}})\big) + Q_\pi[t, \vec{\mathbf{s}}, \pi(t, \vec{\mathbf{s}})], \quad 1 \leq t < h, \forall \vec{\mathbf{s}}$$

$$Q_\pi[t, \vec{\mathbf{s}}, \vec{\mathbf{a}}] = \sum_{\vec{\mathbf{s}}'} T(\vec{\mathbf{s}}, \vec{\mathbf{a}}, \vec{\mathbf{s}}') \cdot V_\pi[t+1, \vec{\mathbf{s}}']. \qquad 1 \leq t < h, \forall \vec{\mathbf{s}}, \vec{\mathbf{a}}$$

An optimal policy π^* satisfying $V_{\pi^*}[t, \vec{\mathbf{s}}] = V[t, \vec{\mathbf{s}}]$ for all times t and states $\vec{\mathbf{s}}$ can be computed through dynamic programming over time (Puterman 1994).

In large multi-agent systems, the requirement that agents must be able to observe the state of the entire system can be too strict (Becker et al. 2004). This motivates viewing the problem as a *decentralized* MDP, in which the MMDP model is *factored* such that each agent i only observes its own part of the state space S_i, with the joint space becoming $S = \bigtimes_{i=1}^{n} S_i$. Then we can identify for each agent i its local state $\vec{\mathbf{s}}_i$ in the joint state (or action). When agents have such a factored structure, *and* additionally satisfy reward and transition independence, the model can be solved to optimality in a decentralized fashion by solving the individual agent MDP sub-problems.

However, this is not possible when the actions of the agents require resources that are constrained on the total amount consumed, as such constraints introduce dependencies between all agents. This forces optimal planners to consider all agents jointly, thereby invoking the curse of dimensionality because the joint state space grows exponentially with the number of agents. Therefore, decoupling is a common paradigm to solve resource-constrained factored MMDPs. It enables agent models to be planned individually while taking into account the effects of others on the resource constraints through proxy values such as a (Lagrangian Dual) cost of consumption (Gordon et al. 2012), the probability of successful consumption (De Nijs, Spaan, and De Weerdt 2015), or action frequency counts (Varakantham, Adulyasak, and Jaillet 2014). However, when resource constraints are uncertain, and therefore part of the transition model of the MMDP, these approaches result in a poor approximation of the true problem and many constraint violations.

Problem Definition

In this section we define stochastic resource constraints more formally. Then we introduce a generalization of a factored MMDP model, called a Stochastic Resource-Constrained Multi-agent Markov Decision Process (SRC-MMDP), where such a constraint is modeled as a separate part of the factored state space. In the remainder of the paper we then show how to deal with these tight interactions while keeping the rest of the planning problem decoupled.

Stochastic Resource Constraints

As running example, consider modeling an electricity grid (partially) powered by renewable sources such as wind and solar power. Because power grids require demand to be balanced with supply at all times, the fluctuating supply of these renewables must be buffered. This can be achieved by planning the demand of flexible devices such as heating, ventilation and air conditioning (HVAC) units, or of electric vehicle charging, taking into account the predicted production over time as well as the operational requirements of the device. Carpinon et al. (2010) show how predictive Markov Chain models of the near-future power production of wind farms can be constructed, which forms a *stochastic resource constraint* on the number of devices activated at each time step.

More formally, a stochastic resource constraint is a time and state dependent hard constraint on the allowed actions. The maximum amount a joint action is allowed to use of the resource is given by a real-valued resource limit function $L : \{1, \ldots, h\} \times S \to \mathbb{R}_+$. Each state-action pair may require zero or more units of the resource, specified through a resource usage function $U : S \times A \to \mathbb{R}$. Given a set of joint actions A, we define the set of safe actions in joint state $\vec{\mathbf{s}}$ at time t as

$$A_{t,\vec{\mathbf{s}}} = \big\{ \vec{\mathbf{a}} : \vec{\mathbf{a}} \in A \text{ and } U(\vec{\mathbf{s}}, \vec{\mathbf{a}}) \leq L(t, \vec{\mathbf{s}}) \big\}. \qquad (1)$$

Any action that is not a member of the safe set is a violation of the constraint. In the grid example, the set $A_{t,\vec{\mathbf{s}}}$ would contain all permutations of $\{0, 1, \ldots, L(t, \vec{\mathbf{s}})\}$ devices switched on. To ensure feasibility of the model, we require that the set

of safe actions is not empty, meaning that at least one safe action exists (e.g., all devices off).

SRC-MMDP

Two forms of uncertainty in the (satisfaction of) the stochastic constraint can be identified: endogenous uncertainty caused by the dynamics of the agents' models, and exogenous uncertainty induced by the stochastic resource constraint itself. In the context of our grid example, endogenous uncertainty may be caused by a house losing more temperature than expected, thereby lowering the time until a heater must be powered. Exogenous uncertainty comes from the uncertain realization of the wind speed. Therefore, for the subsequent definition of the Stochastic Resource-Constrained Multi-agent Markov Decision Process, we factorize the resource limit and its transition function to be separate from the agents' models.

Formally, we use S_L to indicate the state space for the resource limit, and let $T_L : S_L \times S_L \to [0, 1]$ describe the exogenous transition probabilities over this space, defining a Markov Chain. Furthermore, we define resource usage functions U_i for the individual agents in a straightforward manner, and overload $L(t, s_L)$ to mean $L(t, \vec{\mathbf{s}})$.

Definition 1 *A* Stochastic Resource-Constrained Multi-agent Markov Decision Process (SRC-MMDP) *is represented by a tuple $\langle S, A, T, R, h, U, L\rangle$ where $\langle S, A, T, R, h\rangle$ specifies a MMDP as defined above, and the agents as well as the resource constraint can be factored such that:*

$$\begin{aligned} S &= S_L \bigtimes_{i=1}^{n} S_i, \\ A &= \bigtimes_{i=1}^{n} A_i, \\ U(\vec{\mathbf{s}}, \vec{\mathbf{a}}) &= \sum_{i=1}^{n} U_i(s_L, s_i, a_i), \\ L(t, \vec{\mathbf{s}}) &= L(t, s_L), \\ T(\vec{\mathbf{s}}, \vec{\mathbf{a}}, \vec{\mathbf{s}}') &= T_L(s_L, s'_L) \prod_{i=1}^{n} T_i(s_i, a_i, s'_i). \end{aligned} \tag{2}$$

where $\vec{\mathbf{s}} = \langle s_L, s_1, \ldots, s_n\rangle$, $\vec{\mathbf{s}}' = \langle s'_L, s'_1, \ldots, s'_n\rangle$, and $\vec{\mathbf{a}} = \langle a_1, \ldots, a_n\rangle$. A *centralized* solution to an SRC-MMDP is a policy π for the MMDP $\langle S, A, T, R, h\rangle$ that furthermore is *safe*, i.e., for every state $\vec{\mathbf{s}}$ and time t, the chosen action $\pi(t, \vec{\mathbf{s}}) \in A_{t,\vec{\mathbf{s}}}$. To summarize, an SRC-MMDP generalizes the problem definition of (De Nijs, Spaan, and De Weerdt 2015; Wu and Durfee 2010) to include a stochastic model of the exogenous uncertainty in the resource constraint.

Decentralized Resource Decoupling

Unfortunately, the optimal solution to a general SRC-MMDP model requires communication, because the policy is conditioned on the state of all agents as well as the state of the resource limit. In our decentralized setting, the problem needs to be *decomposed* into n single-agent sub-problems, which we propose to do by augmenting the state space of each agent with the current limit (captured in the state feature S_L), so that the sub-problem of agent i becomes a tuple $\langle \bar{S}_i, A_i, \bar{T}_i, \bar{R}_i, h\rangle$ with components

$$\begin{aligned} \bar{S}_i &= S_L \times S_i, \\ \bar{T}_i\big(\langle s_L, s_i\rangle, a_i, \langle s'_L, s'_i\rangle\big) &= T_L(s_L, s'_L) \cdot T_i(s_i, a_i, s'_i), \\ \bar{R}_i\big(\langle s_L, s_i\rangle, a_i\big) &= R_i(s_i, a_i). \end{aligned} \tag{3}$$

Intuitively, this decomposition states that each agent is able to observe the phenomenon influencing their collective resource constraint, in addition to their own local state. By merging the constraint state into their individual state space, each agent is able to condition their own policy on their shared observations (Becker et al. 2004). In the power grid example, all the agents would receive the weather predictions, and have access to a wind speed sensor. This transformation polynomially increases the size of all MDPs, provided that the number of limit realizations is not itself exponential in the number of agents. To compute optimal policies for these decoupled sub-problems, we need to account for the effect of other agents on resource availability, or risk significant overconsumption.

Algorithms for SRC-MMDPs

In this section we show how this stochastic constraint decoupling can be implemented in two state-of-the-art preallocation algorithms. Both algorithms merge the decoupled agent sub-problems in a single 'master' problem of preallocating resources to agents during planning. Therefore, because the single-agent policies respect the allocations, merging them in a joint policy can be done without risk of conflicts, and thus these approaches can be used when communication is not possible, not reliable, or not desirable. These algorithms can be categorized in two groups: 1) a resource preallocation Mixed Integer Linear Program (MILP) which computes deterministic resource assignments that the agents respect in their policies (Wu and Durfee 2010), and 2) a Constrained MDP approach which relaxes the constraints to be sufficiently soft that they only need to be met in expectation (Altman 1999).

Wu and Durfee (2010) show that an optimal resource preallocation can be computed using a MILP. However, a major drawback of this approach is that it consists of a model having exponential run-time complexity growing in the number of agents, the horizon, the number of limit realizations, and the number of resource usage levels. Therefore, we also consider CMDPs (Altman 1999), which relax the preallocation to policies which meet their assigned (fractional) resource allocation in expectation, by allowing for stochastic policies. This can only be used in settings where a small and temporary violation of the constraint is not problematic. Briefly exceeding the supply constraints would be allowed in any robust power grid, as stochastic production is typically backed up by controllable fossil fuel-based generators and/or forms of storage such as batteries. Nevertheless, we would prefer to minimize the frequency of violations, since operating back-up generators and batteries is costly, and batteries need periods of overproduction to charge.

A naive approach to apply these algorithms to the stochastic constraint problem is to determinize the stochastic constraint and apply the algorithms directly. Given the stochastic

constraint Markov Chain, we compute the probability distribution over the limits $\mathrm{P}(s_L \mid t)$, starting from the known prior distribution $T_1(s_L)$, giving the expected constraint

$$E_L[t] = \sum_{s_L \in S_L} \mathrm{P}(s_L \mid t) L(t, s_L).$$

While we could have used other statistics, using for example the minimum realization may result in highly pessimistic policies when the worst-case outcome has a small likelihood.

Policies can be computed for the deterministic expected limit using the original algorithms. However, we expect that such a naive approach will not result in good policies; depending on the realized s_L, any policy using $E_L[t]$ either leaves resources unused, lowering expected value, or overconsumes resources, resulting in a constraint violation. By the stochastic nature of the constraint, we expect that *both* effects occur for policies planned for $E_L[t]$.

Therefore, we propose to modify the algorithms to explicitly reason about the realizations of the stochastic constraint.

Preallocation Mixed Integer Linear Program

We first present our extension of the optimal preallocation MILP encoding. This extended model is shown in Algorithm 1. The encoding contains variables $x^i_{t,s,a} \in [0, 1]$ which give the (unconditional) probability that action a is chosen in state s at time t by agent i. These variables are chained together by probability conservation constraints (5) and (6) encoding the transition function, which is initialized to a prior over the initial states $T_{1,i} : \bar{S}_i \to [0, 1]$. In the original binary consumption model of Wu and Durfee (2010) a binary variable per agent per time step encodes whether a (single) resource is allocated to that agent at that time, based on the a priori (estimated) resource availability. Conflict-free policies can be guaranteed by constraining the sum of binary consumption variables over all agents for the respective times by this resource availability.

However, we cannot simply generalize binary consumption to the *arbitrary consumption* that we model in SRC-MMDPs, and repeat the procedure, because consumption may differ per action, and we aim to guarantee that the allocated resources are sufficient for every action assigned a non-zero probability. We therefore introduce a binary variable $\hat{x}^i_{t,s,a}$ *for each action* to denote that the action has non-zero probability. Let furthermore Δ^i_{t,s_L} denote the resources preallocated to agent i at time t of resource state s_L. To ensure now that no policy uses an action that requires more than the resources allocated to an agent i, we include constraint (6). The total resource demand can then simply be bounded by the sum over all Δ through constraint (7), for each time step. Furthermore, to deal with multiple resource limit realizations, we repeat this for each of these.

Constrained MDPs

The framework of Constrained MDPs allows arbitrary linear constraints to be added to MDP models by encoding the constrained model as a linear program (Altman 1999). Instead of restricting the worst-case resource consumption as is done in the MILP, CMDPs restrict the *expected* resource consumption of all agents taken together to be less than the constraint in a state s_L at time t by requiring that

$$\sum_{i=1}^{n} \sum_{\bar{s} \in \bar{S}_i} \sum_{a \in A_i} x^i_{t,\langle s_L, s_i \rangle, a} \cdot U(s_i, a) \leq L(t, s_L).$$

The challenge in the stochastic-constraint case is to account for the fact that only one out of $|S_L|$ constraints will be 'active' at any time. By transforming the individual agent problems as defined in equation (3), we keep track of the active constraint through the state of the agents. Of course, the sum of all occupancy variables relating to a limit l_t will only sum to the unconditional probability that limit state l_t will be reached. Therefore the consumption limit l_t must be normalized to the probability it will be reached, defined as

$$\bar{C}(1, s_L) = T_1(s_L), \qquad \forall s_L \in S_L$$
$$\bar{C}(t+1, s'_L) = \sum_{s_L \in S_L} T_L(s_L, s'_L) \cdot \bar{C}(t, s_L).$$

Putting it all together, we obtain the linear program presented in Algorithm 2.

Algorithm 1 Resource allocation MILP for SRC-MMDP.

$$\max \sum_{i=1}^{n} \sum_{t=1}^{h} \sum_{\bar{s} \in \bar{S}_i} \sum_{a \in A_i} x^i_{t,\bar{s},a} \cdot \bar{R}_i(\bar{s}, a) \tag{4}$$

$$\text{s.t.} \sum_{a \in A_i} x^i_{t+1,\bar{s},a} = \sum_{\bar{s}' \in \bar{S}_i} \sum_{a' \in A_i} x^i_{t,\bar{s}',a'} \cdot \bar{T}_i(\bar{s}', a', \bar{s}) \quad \forall i, t, \bar{s} \in \bar{S}_i$$

$$\sum_{a \in A_i} x^i_{1,\bar{s},a} = T_{1,i}(\bar{s}) \qquad \forall i, \bar{s} \in \bar{S}_i \tag{5}$$

$$x^i_{t,\langle s_L, s_i \rangle, a} \leq \hat{x}^i_{t,\langle s_L, s_i \rangle, a} \qquad \forall i, t, s_L, s_i, a$$

$$\hat{x}^i_{t,\langle s_L, s_i \rangle, a} \cdot U_i(\langle s_L, s_i \rangle, a) \leq \Delta^i_{t,s_L} \qquad \forall i, t, s_L, s_i, a \tag{6}$$

$$\sum_{i=1}^{n} \Delta^i_{t,s_L} \leq L(t, s_L) \qquad \forall t, s_L \tag{7}$$

$$0 \leq x^i_{t,\bar{s},a} \leq 1,\ \hat{x}^i_{t,\bar{s},a} \in \{0, 1\} \qquad \forall i, t, \bar{s}, a$$

Discussion

We propose two new algorithms for solving multi-agent resource-constrained planning problems with a stochastic time-variable resource constraint, which compute policies that are executable without requiring communication. Both algorithms compute *optimal* policies, but with different conditions on the resource constraint satisfaction: the MILP (Algorithm 1) computes safe policies which never violate the constraints, while the CMDP LP (Algorithm 2) computes relaxed policies which satisfy the constraints in expectation, allowing for occasional resource constraint violations.

Algorithm 2 Constrained MMDP LP for SRC-MMDP.

$$\max \sum_{i=1}^{n} \sum_{t=1}^{h} \sum_{\bar{s} \in \bar{S}_{i,t}} \sum_{a \in A_i} x^i_{t,\bar{s},a} \cdot \bar{R}_i(\bar{s}, a) \tag{8}$$

$$\text{s.t.} \sum_{a \in A_i} x^i_{t+1,\bar{s},a} = \sum_{\bar{s}' \in \bar{S}_i} \sum_{a' \in A_i} x^i_{t,\bar{s}',a'} \cdot \bar{T}_i(\bar{s}', a', \bar{s}) \quad \forall i, t, \bar{s} \in \bar{S}_i$$

$$\sum_{a \in A_i} x^i_{1,\bar{s},a} = T_{1,i}(\bar{s}) \quad \forall i, \bar{s} \in \bar{S}_i \tag{9}$$

$$\sum_{i=1}^{n} \sum_{\bar{s} \in \bar{S}_i} \sum_{a \in A_i} x^i_{t,\bar{s},a} \cdot U(\bar{s}, a) \leq \bar{C}(t, s_L) \cdot L(t, s_L) \quad \forall t, s_L$$

$$0 \leq x^i_{t,\bar{s},a} \leq 1 \quad \forall i, t, \bar{s}, a$$

While the CMDP algorithm computes policies which are not completely safe, the trade-off is that the algorithm is tractable; because MILP solvers have exponential complexity in the number of integer variables, we expect that Algorithm 1 can only be applied to problems with a short planning horizon. Nevertheless, many problems with constraints are tolerant to occasional violations, motivating the use of Algorithm 2. However, this also raises the question to what degree we benefit from handling stochastic constraints explicitly. Therefore, in the experimental evaluation we explore the frequency of constraint violations compared to versions of the algorithms planning for the weighted mean constraint.

Because the agents may still be able to communicate from time to time, we also propose a replanning algorithm that updates agent policies with each communication, based on their current state. Because replanning incorporates new state information, we expect that the coordination between agents improves when they communicate more frequently, which should result in fewer constraint violations.

Experimental Evaluation

In this section we evaluate the effect of planning for stochastic resource constraints on a single time-step search and rescue domain and on a longer horizon energy demand planning problem. We compare the modified algorithms Preallocation MILP and CMDP, designated $\mathrm{P}(X = x)$, with their original versions planning for the *expected* resource limit, $\mathrm{E}(X)$. We expect two beneficial effects of the stochastic variants: 1) a better performance, and 2) fewer constraint violations.

Coordinated Search and Rescue Missions

First we consider a disaster response cooperative game as an illustrative domain. Consider a group of countries that collectively commits response teams in order to perform expensive search-and-rescue (SAR) operations that would be too costly to perform individually. Due to the urgent nature of crises, each country must individually decide its response level without time-consuming coordination. They do so in accordance with a single time-step policy that they agreed on beforehand (e.g., at the previous summit).

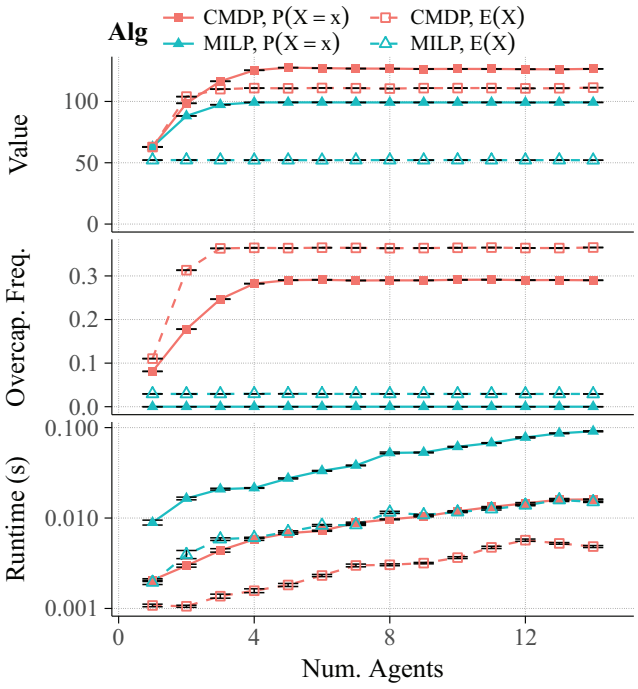

Figure 1: Realized mean and standard errors of the value, frequency of overcapacity, and runtime, comparing mean ($\mathrm{E}(X)$) with stochastic limit ($\mathrm{P}(X)$) on SAR problem.

The size of an operation that a country commits determines the cost to that country, while the sum of all committed operations influences the probability of successful rescues. The cost of an operation of size j is simply j, where $j \leq 4$, which we assume to be the politically acceptable maximum spending on rescue missions. The probability of retrieving a survivor using an operation of size 1 is given by p, which we assume to be 0.2. More generally, the number of survivors i rescued, as a function of the sum of operation sizes j is given by random variable W having probability distribution

$$\mathrm{P}\left(W = i \mid j\right) = \binom{j}{i} p^i (1-p)^{(j-i)}.$$

The reward for rescuing a survivor is 100.

In practice, the number of survivors that can be rescued is bounded by the number of people affected, which informs the stochastic constraint in this problem. Due to the high value of rescuing survivors, countries are incentivized to deploy all their resources in the first crisis in an uncoordinated setting. To retain some resources for future calamities, countries constrain their response to be sufficient for the size of the disaster. Because the size of an unexpected disaster can only be estimated when the disaster occurs, the number of potential survivors x is learned only at the time mission size must be determined. We assume that the probability density function on the number of potential survivors of any potential disaster is given by

$$\mathrm{P}(X = x) = \{0 : 0.05,\ 1 : 0.4,\ 2 : 0.3,\ 3 : 0.2,\ 4 : 0.05\}.$$

A centralized joint task force (without maximum operation size) would thus aim to optimize the following function $f_x(j)$ for each disaster size x.

$$f_x(j) = \sum_{i=1}^{j} \Big(100 \cdot \mathrm{P}\left(W = i \mid j\right) \cdot \min\left(x, i\right) \Big) - j.$$

Since this set of functions attains maximum value at $j = \{0, 14, 22, 30, 37\}$, for $x = \{0, 1, 2, 3, 4\}$ respectively, the joint task force should assign operation sizes to countries such that their sum operation size matches these values. However, when the countries do not have time to communicate their commitment, they must select their responses such that the *expected* sum is equal to the optimal. We compare the proposed coordination planning algorithms with versions that condition their response on the mean disaster size:
1) Deterministic preallocation MILP, E(x): mean disaster survival rate is ≈ 1.8 survivors; thus a mission size is selected such that the maximum number of survivors is at most 1.
2) Conditional preallocation MILP, P(x): depending on the potential number of survivors x, the mission response size is selected such that exactly x are rescued.
3) Deterministic preallocation CMDP, E(x): a mission size is selected such that in expectation 1.8 survivors are rescued.
4) Deterministic preallocation CMDP, P(x): a mission size is selected such that in expectation, x survivors are rescued.

Figure 1 presents the results, showing means and standard errors obtained, when each computed policy is sampled 100,000 times. We compute 100 policies per data point to obtain significance with respect to the runtime. The value reported is the *observed* value, given by the number of actual rescues minus the operational costs. As expected, the value obtained when planning for just the mean (results with $\mathrm{E}(X)$) is significantly less than the value obtained through taking into account the uncertainty in X for both algorithms (results denoted by $\mathrm{P}(X = x)$). Additionally, the frequency of deploying more successful operations than there are potential survivors (i.e., overcapacity) is also significantly smaller when planning for $\mathrm{P}(X = x)$ than for $\mathrm{E}(X)$. Planning for the stochastic limit increases the required time to plan policies significantly, however this does not change the scalability characteristics: the trends in the run time depending on the number of agents are the same. Comparing the behavior of the two different algorithms themselves, we observe that the MILP trades off overcapacity probability (i.e., almost none) for slightly reduced value and more significant runtime costs compared to the CMDP approach.

Planning Thermostatically Controlled Loads

Next we compare the same methods for planning how thermostatically controlled loads (TCLs) use a shared resource for a longer horizon. TCLs are electric devices for managing temperature, consisting of a controller and an electric heating or cooling element. TCLs typically control insulated systems, whose inertia gives the TCLs a degree of flexibility. This flexibility can be employed to buffer for the fluctuating supply of energy from renewable sources. By replacing the thermostat controller with a policy anticipating energy availability, we can unlock this flexibility while minimizing the impact

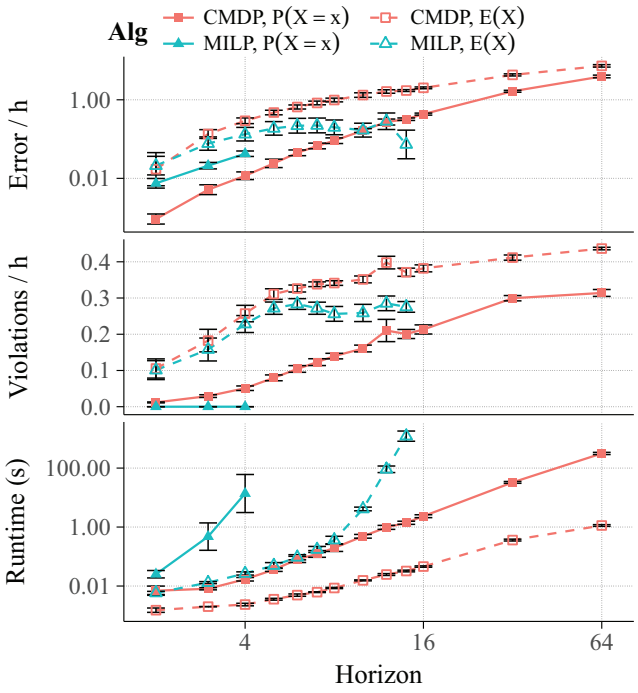

Figure 2: Realized mean and standard errors of the absolute error, violation frequency, and runtime, comparing mean ($\mathrm{E}(X)$) with stochastic limit ($\mathrm{P}(X)$) on TCL instances.

on thermal comfort (De Nijs, Spaan, and De Weerdt 2015). Because supply from renewable sources is typically not only fluctuating but also *uncertain*, this domain naturally exhibits stochastic constraints.

In our experiments we consider TCL problems with temperature ranges discretized into 25 states, and with agents having 4 actions, corresponding to switching a heater on for $\{0, 5, 10, 15\}$ out of 15 minutes per time step. The thermal parameters are based on reference insulation levels of houses equipped with heat-pumps. To model consumer behavior and build quality variation, we add small Gaussian noise to the parameters, resulting in a heterogeneous population of TCLs.

To obtain challenging instances of the TCL problem, we generate resource limit scenarios such that each scenario is *in expectation* sufficient to keep the temperature at the setpoint, but has realizations that are far from the mean. We randomly generate 10 such (deterministic) resource limit scenarios and merge them together in a Markov chain by allowing for a small probability of cross-over between scenarios.

For evaluating the quality of the proposed algorithms, we define an error measure by the distance of the results from a theoretical upper bound, which we obtain by computing *joint* (centralized) policies with Value Iteration (Puterman 1994). Because this algorithm has exponential complexity in the number of agents, we perform experiments for 3 agents and an increasing length of the horizon. Figure 2 presents the results, normalized by the horizon as each time step has potential to incur error, and each constitutes a new resource constraint that can be violated. The results show a similar

trend as in the search-and-rescue instances. Planning for the stochastic resource constraint $\mathrm{P}[X = x]$ increases the run-time of the algorithms as a result of the increase in number of states and constraints. This has the largest effect on MILP, which also has exponential worst-case complexity when planning for the mean constraint. For both algorithms we observe that planning for the stochastic constraint results in a significant increase in the quality of the solution, resulting both in lower error and in lower violation frequency.

Regarding scalability, we observe that the run-time measurements of CMDP form almost a straight line in the log-log plots in both experiments. We therefore conclude that this run time scales polynomially with the number of agents in the SAR domain (Figure 1) as well as with the length of the horizon in the TCL domain (Figure 2).

Re-planning TCLs In the TCL domain, an important practical concern is that the heat-pumps should continue to operate as normal when connectivity is briefly lost, for which preallocation algorithms are suitable. For such settings an approach is needed where coordinated policies are computed for a number of sequential decisions that are taken without further communication. However, we want to incorporate new information when agents can have an opportunity to communicate. Therefore we propose a re-planning algorithm that uses the previously described algorithms as subroutines and evaluate the effect of communication in the TCL domain.

Let $\hat{h} \leq h$ be the maximum time that agents may need to operate without communication, and let time t_c be any time step in which communication is possible, and at which point the agents are in state $\bar{s}_c$. Then, we adapt the algorithms as follows: the algorithm objective functions (4) and (8) are changed to range over the time from the communication point until the next sync is guaranteed to happen,

$$\sum_{i=1}^{n} \sum_{t=t_c}^{\min(t_c+\hat{h},h)} \sum_{\bar{s}\in\bar{S}_{i,t}} \sum_{a\in A_i} x^i_{t,\bar{s},a} \cdot \bar{R}_i(\bar{s},a),$$

while the initial conditions (5) and (9) are set to match the current state, $\sum_{a\in A_i} x^i_{t_c,\bar{s}_{c,i},a} = 1, \quad \forall i.$

In order to assess the effect of periodic coordination, we apply the re-planning algorithm to a TCL instance with horizon $h = 216$ (9 days in hours) and re-planning horizon $\hat{h} = 24$. We let the agents re-plan at a regular interval (the communication gap), and measure the number of violations as a function of the length of the interval. Figure 3 shows the results, with the horizontal lines representing the baseline case of coordinating only at the start. We observe that re-planning can greatly reduce the number of violations. However, more importantly, we also observe that planning for stochastic constraints is effective at reducing constraint violations even when agents only need to bridge gaps of 3 steps, demonstrating the practical value of our algorithms.

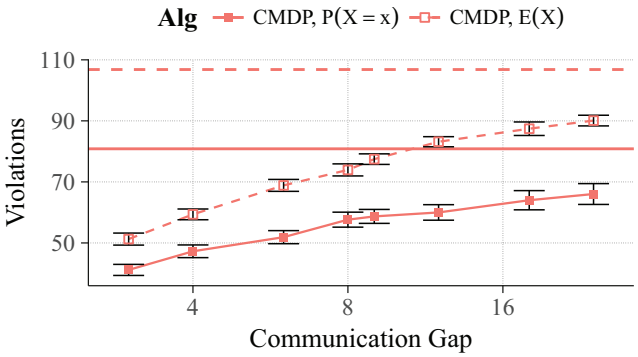

Figure 3: Effect of increasing the time between communication in the re-planning algorithm.

Related Work

Handling stochastic resource constraints has to our knowledge thus far been limited to scheduling under uncertainty, in which case there is only a single agent and a predefined set of activities (Fink et al. 2006). Even though stochastic resource constraints are not widely studied, there are several other works that attempt to address deterministic resource constraints through other means than decoupling. Meuleau et al. (1998) consider large-scale planning problems with instantaneous constraints; however their strategy for addressing them is to ignore them in the planning phase and only enforce them at execution time. Such an approach would require communication at execution time, which means that it could not be applied in a decentralized setting.

The literature on Decentralized (PO)MDPs provides algorithms that exploit the limited influence that agents might exert on each other (Oliehoek, Witwicki, and Kaelbling 2012). However, our global resource constraints prevent that agents can be easily decoupled using such models. Related to our re-planning algorithm are approaches that consider intermittent communication (Nair et al. 2004) or delayed communication (Spaan, Oliehoek, and Vlassis 2008; Oliehoek and Spaan 2012). These methods rely on a solution of the underlying Multi-agent POMDP which is exponentially-sized in the number of agents. Hence, scalability is poor and they are typically only demonstrated for two agents.

Conclusions

Stochastic resource constraints have not been widely studied in multi-agent planning under uncertainty, although they occur naturally in domains where the resource constraint is a natural process or results from unmodeled external influences. Multi-agent systems are additionally typically expected to operate decentrally for periods at a time, either because re-planning time exceeds decision time, or because of communication restrictions. In this work we show how stochastic resource constraints can be factored such that policies can still be effectively decoupled. To demonstrate this we extend two state-of-the-art decoupling algorithms for deterministic constraints to handle stochastic constraints: a Mixed-Integer Linear Program approach and Constrained MDPs.

In our experimental evaluation we observe that using our extensions to plan for stochastic constraints results in significantly better solutions than using the original algorithms to plan for the expectation of the limit. We show that these

results continue to hold when combined with an intermittent replanning scheme, which allows the system to operate with reduced violations over a longer horizon.

We observe that the CMDP and MILP algorithms have their individual drawbacks; the MILP has worst-case complexity exponential in the number of resource allocations, which grows with the number of agents, while CMDP solutions result in high probability of violations. Both drawbacks have been addressed by related work for which we intend to investigate the effect of our stochastic constraint setting in future work. Agrawal, Varakantham, and Yeoh (2016) present a Lagrangian decomposition of the MILP, which splits the problem into n subproblems through dual pricing of resource consumption. For CMDPs, De Nijs et al. (2017) present algorithms to bound the probability of violations, through reducing the resource capacities used in planning. We expect that the same technique can be applied here, because our approach does not change the underlying structure of the constraints, and constraint realizations are independent.

Acknowledgments

Support of this research by network company Alliander is gratefully acknowledged.

References

Adelman, D., and Mersereau, A. J. 2008. Relaxations of weakly coupled stochastic dynamic programs. *Operations Research* 56(3):712–727.

Agrawal, P.; Varakantham, P.; and Yeoh, W. 2016. Scalable greedy algorithms for task/resource constrained multi-agent stochastic planning. In *Proc. of the 25th Intl. Joint Conf. on Artificial Intelligence*, 10–16.

Altman, E. 1999. *Constrained Markov Decision Processes*. Stochastic Modeling. Chapman & Hall/CRC.

Becker, R.; Zilberstein, S.; Lesser, V.; and Goldman, C. V. 2004. Solving transition independent decentralized Markov decision processes. *Journal of Artificial Intelligence Research* 22:423–455.

Bellman, R. 1957. A Markovian decision process. *Journal of Mathematics and Mechanics* 6(5):679–684.

Boutilier, C. 1996. Planning, learning and coordination in multiagent decision processes. In *Proc. of the 6th Conf. on Theoretical Aspects of Rationality and Knowledge*, 195–210.

Carpinon, A.; Langella, R.; Testa, A.; and Giorgio, M. 2010. Very short-term probabilistic wind power forecasting based on Markov chain models. In *Intl. Conf. on Probabilistic Methods Applied to Power Systems*, 107–112. IEEE.

De Nijs, F.; Walraven, E.; Spaan, M. T. J.; and De Weerdt, M. M. 2017. Bounding the probability of resource constraint violations in multi-agent MDPs. In *Proc. of the 31st AAAI Conf. on Artificial Intelligence*, 3562–3568.

De Nijs, F.; Spaan, M. T. J.; and De Weerdt, M. M. 2015. Best-response planning of thermostatically controlled loads under power constraints. In *Proc. of the 29th AAAI Conf. on Artificial Intelligence*, 615–621.

De Weerdt, M. M.; Gerding, E. H.; Stein, S.; Robu, V.; and Jennings, N. R. 2013. Intention-aware routing to minimise delays at electric vehicle charging stations. In *Proc. of the 23rd Intl. Joint Conf. on Artificial Intelligence*, 83–89.

Fink, E.; Jennings, P. M.; Bardak, U.; Oh, J.; Smith, S. F.; and Carbonell, J. G. 2006. Scheduling with uncertain resources: Search for a near-optimal solution. In *Proc. of the 19th IEEE Intl. Conf. on Systems, Man and Cybernetics*, 137–144.

Gordon, G. J.; Varakantham, P.; Yeoh, W.; Lau, H. C.; Aravamudhan, A. S.; and Cheng, S. 2012. Lagrangian relaxation for large-scale multi-agent planning. In *Proc. of the IEEE/WIC/ACM Intl. Confs. on Web Intelligence and Intelligent Agent Technology*, 494–501.

Klöckl, B.; Papaefthymiou, G.; and Pinson, P. 2008. Probabilistic tools for planning and operating power systems with distributed energy storage. *Elektrotechnik und Informationstechnik* 125(12):460–465.

Mausam; Benazera, E.; Brafman, R.; Meuleau, N.; and Hansen, E. A. 2005. Planning with continuous resources in stochastic domains. In *Proc. of the 19th Intl. Joint Conf. on Artificial Intelligence*, 1244–1251.

Meuleau, N.; Hauskrecht, M.; Kim, K.; Peshkin, L.; Kaelbling, L. P.; Dean, T.; and Boutilier, C. 1998. Solving very large weakly coupled Markov decision processes. In *Proc. of the 15th National Conf. on Artificial Intelligence*, 165–172.

Nair, R.; Tambe, M.; Roth, M.; and Yokoo, M. 2004. Communication for improving policy computation in distributed POMDPs. In *Proc. of the 3rd Intl. Conf. on Autonomous Agents and Multi Agent Systems*, 1098–1105.

Oliehoek, F. A., and Spaan, M. T. J. 2012. Tree-based solution methods for multiagent POMDPs with delayed communication. In *Proc. of the 26th AAAI Conf. on Artificial Intelligence*, 1415–1421.

Oliehoek, F. A.; Witwicki, S. J.; and Kaelbling, L. P. 2012. Influence-based abstraction for multiagent systems. In *Proc. of the 26th AAAI Conf. on Artificial Intelligence*, 1422–1428.

Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc.

Schaffer, S. R.; Clement, B. J.; and Chien, S. A. 2005. Probabilistic reasoning for plan robustness. In *Proc. of the 19th Intl. Joint Conf. on Artificial Intelligence*, 1266–1271.

Spaan, M. T. J.; Oliehoek, F. A.; and Vlassis, N. 2008. Multiagent planning under uncertainty with stochastic communication delays. In *Proc. of the 8th Intl. Conf. on Automated Planning and Scheduling*, 338–345.

Varakantham, P.; Adulyasak, Y.; and Jaillet, P. 2014. Decentralized stochastic planning with anonymity in interactions. In *Proc. of the 28th AAAI Conf. on Artificial Intelligence*, 2505–2512.

Wu, J., and Durfee, E. H. 2010. Resource-driven mission-phasing techniques for constrained agents in stochastic environments. *Journal of Artificial Intelligence Research* 38:415–473.

Yoo, C.; Fitch, R.; and Sukkarieh, S. 2012. Probabilistic temporal logic for motion planning with resource threshold constraints. In *Proc. of Robotics: Science and Systems VIII*.