A Decision-Theoretic Characterization of Organizational Influences

Jason Sleight and Edmund H. Durfee
Computer Science and Engineering
University of Michigan
Ann Arbor, MI 48109
{jsleight,durfee}@umich.edu

ABSTRACT

Despite a large body of research on integrating organizational concepts into cooperative multiagent systems, a formal understanding of how organizations can influence agents' decisions remains elusive. This paper works toward such an understanding by beginning with a model of agent decision making based on decision-theoretic principles, and then examining the possible routes that organizational influences can take to affect that model. We show that alternative avenues of applying influences correspond to different prior notions of organizational control, and empirically demonstrate the impact that each can have on the quality and overhead of coordinated behavior. To do so, we must define the agents' baseline behavior (without a designed organization), and we present a methodology for initializing agents' models to comprise what amounts to an "uninformed" organization. Finally, we show how the specification of organizational influences in terms of components of a decision-theoretic agent creates opportunities for agents to compare actual events with predictions implied in the models, such that agents can reason about whether to change organizations. We demonstrate that this capability to question and change organizations can be valuable if used judiciously.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Coherence & co-ordination, multiagent systems*

General Terms

Design, performance

Keywords

Organization, organizationally adept agents

1. INTRODUCTION

Organizational structuring is a widely adopted and often powerful tool for coordinating large groups of people to achieve common goals effectively and efficiently, by giving each person guidance in how to make local decisions that are useful to the collective endeavor. Multiagent systems research has investigated how organizational concepts and strategies can be modeled and utilized by computational agents, showing that organizations can increase the expected performance of large-scale, cooperative multiagent systems [12, 6]. Research also suggests that organizational control becomes increasingly effective as the number of agents increases, the time horizon increases, the system complexity increases, the system resources decrease, and/or the performance goals increase [4]. That these issues arise in realistic application domains has driven research into how to encode pertinent organizational control and how to augment agent architectures to follow such control.

A point of departure in this paper is that we attack the question of what an organization is or could be, computationally, from the opposite direction. We begin with a model of agent decision making based on decision-theoretic principles, captured as decentralized partially observable Markov decision processes, Dec-POMDPs. Within this formal, well-defined decision framework, we then explore how various types of organizational control and influences can be captured in the different components of the framework, such as transition and reward functions. Hence, one contribution of this paper is a systematic and comprehensive enumeration of where organizational control can be applied, and how it can be formally manifested in decision-theoretic agents. We empirically evaluate how the embodiment of organizational influence in different Dec-POMDP framework components individually and collectively impacts the quality of agents' behavior and the costs of agents' reasoning.

Measuring performance improvements resulting from designing and following a good organization requires a baseline of performance without any organization. Our more principled formulation reveals, however, that defining such a baseline is problematic. A second contribution of this paper, therefore, is a methodology for forming baseline organizations for experimental comparisons.

Our third main contribution in this paper is to demonstrate how an explicit representation of organizational control in terms of components of a decision-theoretic framework captures statistical predictions about runtime behavior, which agents can use to decide how and when to change (or abandon) their organization. We build off of the abstract concept of an organizationally adept agent (OAA) [3] to formulate a more precise notion of an OAA that can compare actual experiences in its environment with the organization's predictions, and can (with other OAAs) adopt a better alternative organizational design. Our preliminary experiments show that this capability to question an organization's suitability

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.

rather than to follow it blindly can improve system-wide performance, but that responsiveness needs to be tempered by the costs of reorganization.

The remainder of this paper is structured as follows. In Section 2, we describe the decision-theoretic framework that our agents use and that, for the purposes of this paper, the organizational structure must work within. Then we turn to our first contribution, in Section 3, where we describe how organizational influence is manifested in each of the components and how the different manifestations capture prior organizational strategies in the literature. In Section 4 we make our second contribution by analyzing the space of baseline organizations to consider and justifying our choices for our experiments. Section 5 presents our empirical evaluation of the impact of different forms of organizational influence on the quality and costs of coordination. We then turn to a description of how a rudimentary form of organizational adeptness has been captured in our agents and present preliminary experiments illustrating the promise and potential costs of agents that can change organizations (Section 6). We conclude in Section 7 with a summary of the work presented here and of our ongoing efforts.

2. PROBLEM REPRESENTATION

We adopt a standard Dec-POMDP decision model [2], $\mathcal{M} = \langle \mathcal{N}, S, \alpha, A, R, P, \Omega, O, T \rangle$, where: $\mathcal{N}$ is the set of n cooperative agents; S is the (finite) set of global states; α is a probability distribution over initial global states; A is the (finite) set of possible joint actions; R is the joint reward function; P is the joint transition function; Ω is the (finite) set of possible joint observations; O is the joint observation function; and T is the finite time horizon. Given a full specification of the Dec-POMDP, an optimal joint policy, π^*, can be formulated in principle. In practice, however, finding such a policy for anything but very simple problems (with few agents and small state and action spaces) is intractable [2], and even if found, executing such a policy is problematic because it generally assumes that all agents have the same beliefs about the global state.

For these reasons, multiagent approaches to solving such problems often assume that each agent possesses a local view of the joint problem. As is customary in that work, we assume that state is factored: every state is represented using the same set of τ state features, such that $\forall s \in S, s = \langle f_1 \in F_1, \cdots, f_\tau \in F_\tau \rangle$, where F_j is the finite set of possible values for state feature j. Each agent i has a local state representation S_i consisting of a subset of the τ features. Agent i has a local decision model defined for this state space: $\mathcal{M}_i = \langle S_i, \alpha_i, A_i, R_i, P_i, \Omega_i, O_i, T_i \rangle$, where local rewards, transitions, actions, etc. are defined over the states in S_i. We further adopt the common assumption of local full observability (each agent i can exactly observe the values of all of its local state's features). Given these assumptions, the local decision model $\mathcal{M}_i$ of an agent i represents a local MDP, such that an agent can compute its (optimal) local policy π_i with respect to $\mathcal{M}_i$. The joint policy is then simply defined as $\pi = \langle \pi_1, \pi_2, ..., \pi_n \rangle$.

To illustrate a problem of this type, we use a simplified firefighting scenario, where firefighting agents and fires to be fought exist in a grid world (Figure 1). The global state consists of the locations of the agents and the locations and intensities of the fires. Figure 1 shows an initial global state, where the locations of agents A1 and A2 are shown,

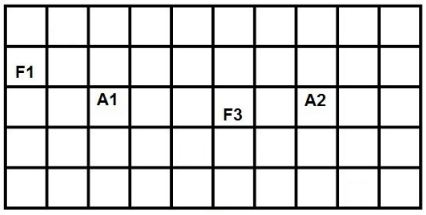

Figure 1: Example initial state of a 10×5 firefighting grid world domain. Ai indicates the position of agent i, and Fj indicates that there is a fire in that cell with intensity j.

along with positions of each fire Fx, where x is the current intensity of the fire in that position. Each agent has 6 actions: a NOOP action that makes no change to the world state; 4 possible movement actions (N, S, E, W) that move the agent one cell in the specified direction (and equates to a NOOP if there is no cell in that direction); and a fight-fire (FF) action that decrements by 1 the intensity of the fire in the agent's current location, if any and otherwise behaves like a NOOP. Joint actions are defined as the aggregation of the agents' local actions. Movement actions are independent (agents can occupy the same location), but FF actions are not: the intensity of a fire only decreases by 1 even if multiple agents simultaneously fight it. The joint reward for the agents in states prior to reaching T is the negative sum of the fires' intensities in that state. When the time horizon is reached, the problem episode ends, and the joint reward is 10 times the negative sum of the remaining fires' intensities, encouraging the agents to put all the fires out before the deadline.

An example of how agents might have local models of this joint model is the following. An agent's local state consists of its location and the locations and intensities of the fires. That is, it does *not* include the position of other agents. Hence, its local action space only includes its 6 actions, and its local transition model will only model how its local actions affect its local state. Its local reward function is the same as the global reward function; note that in this case the sum of the agents' local rewards will overestimate the true (negative) reward. Its local finite time horizon is identical to the global finite time horizon, and its local initial state distribution is calculated by directly mapping the initial distribution of global states into the local state space. Given such a local model, each agent will formulate a local policy that would fight the fires optimally if the agent were alone in the world. Note that, in general, the joint policy formed by the combination of these optimal local policies will not itself be optimal. For example, in Figure 1, both agents will be drawn to the high intensity fire first and redundantly fight it rather than dividing up to fight the two fires concurrently.

3. ORGANIZATIONAL INFLUENCE

As just illustrated, optimizing policies for local models of joint problems does not necessarily lead to optimal joint policies. Yet, as has been already discussed, centrally solving for an optimal joint policy is computationally infeasible and can lead to policies that rely on agents knowing the global state. The organizational approach that we examine here, therefore, focuses on modifying agents' local models such that the local policies that agents individually construct

will, in combination, result in better joint policies. We note that runtime communication to increase global awareness of agents' states and plans can also help improve coordination (e.g., help prevent firefighting agents from behaving redundantly), and could gainfully augment an organizational approach. However, in the remainder of this paper we assume no communication between agents in order to avoid confounding factors in our presentation and in our analysis of organizational influence's performance.

3.1 Organizational Design Space

We assert that the components of the Dec-POMDP model provide a way to systematically enumerate the dimensions of the organizational design space, at least for designs intended for decision-theoretic agents. Formally, let an organizational design be defined as $\Theta = \langle \theta_1, \cdots, \theta_n \rangle$ where $\theta_i = \langle S_{\theta_i}, \alpha_{\theta_i}, A_{\theta_i}, R_{\theta_i}, P_{\theta_i}, T_{\theta_i} \rangle$ is the local organizational model for agent i.[1] θ_i specifies the local state space, initial state distribution, action space, reward function, transition function, and finite time horizon (FTH) for agent i, when the agent is following organization Θ. We now step through each of the components and discuss how each could be used to introduce some commonly cited organizational influences.

Rewards: The idea of modifying local models to improve coordination is not new. In particular, a growing body of literature on *reward shaping* specifically looks at how agents' reward functions can be manipulated to bias agents into taking actions that benefit the collective [15, 10]. For example, reward shaping can lead an agent to establish conditions that have no (unshaped) local reward, but that enable other agents to then take actions that lead to high joint reward. In a similar spirit, Agogino and Tumer [1] have explored the process of designing agents' individual objective functions such that maximizing local rewards leads to maximizing a global objective function in expectation. Hence, one obvious dimension in the organizational design space is the space of alternative combinations of reward functions to assign to agents.

Transitions: It turns out, however, that changing each agent's local rewards alone might be insufficient to induce some forms of cooperative behavior. For example, consider the situation where one agent can establish a condition that enables another to take actions that ultimately lead to high reward. An organizational reward function can bias the first agent into establishing the condition; however, the second agent might not take useful precursor actions because its local model indicates that the condition is unlikely to be established by default. To induce the second agent into complementary behavior, the organizational designer needs to convey the expectation that, because of how the first agent's reward is shaped, the second agent should expect the condition to be (or become) established with high probability. The organization could give the second agent a modified transition function indicating that, given organizational influences elsewhere, the condition of interest is now more likely to be established. Note that the revised transition function summarizes the expectations without needing to be specific about the details; the second agent need not reason about how the first will establish the condition, or even which agent is establishing the condition.

[1]As mentioned, for simplicity we assume local state is fully observable. What follows can be extended to local partial observability with the usual impacts on complexity.

Hence, besides reward functions, transition function modification is another dimension of organizational design. While the example above points out how these can be correlated, even if agents' reward functions are left unchanged they could still benefit from improved transition functions, for example, by reflecting the tendencies that agents inherently have in affecting the states that others might face.

Actions: Without specialized optimizations during policy creation, organizational shaping of reward and/or transition components will not reduce the size of the agents' local policy spaces, but only their decisions about which of those policies are optimal. Redesigning some of the other components of an agent's decision model, however, can achieve another objective often attributed to organizational influence, which is to simplify an agent's reasoning. For example, the organizational designer might associate different roles with different agents and thus induce agents to specialize in the possible actions they will exercise. The designer can give agent i a reduced action specification $A_{\theta_i} \subseteq A_i$ that constrains its choices in some (or all) states. For example, in Figure 1, agent A1 might be prohibited from moving outside of an organizationally-dictated area of responsibility. Chosen well, such restrictions not only help agents pursue complementary policies, but simplify planning for each. Like reward shaping, encoding organizational influence as constraints on behavior is a familiar approach in the literature [6, 11].

States: In a factored state representation, the organizational designer could determine that there are features that an agent can sense that are unnecessary to represent given the organization. In our running firefighting example, for instance, the organizational designer might decide that some (distant) fires need not be modeled by an agent at all (because they are the responsibility of other agents), thereby simplifying its local decision problem. Further, the organizational designer might purposely augment an agent's local state representation with new features, where the designer has decided that those features are crucial to distinguishing between states that otherwise would look locally identical. Such augmentations must be done with caution, however, and if the designer includes such augmentations, it must also delineate the communication protocols and policies that would ensure an agent possesses up-to-date values for those features despite not being able to directly observe them. For instance, in our running example, to improve coordination the designer might insist that each firefighter tell the others which fire it is now working towards extinguishing. Establishing these types of commitments and conventions has proven useful [8], but this paper will only consider organizations that remove state features.

Initial State and FTH: Finally, an organization can also influence an agent's behavior through α_{θ_i} and T_{θ_i}. In the firefighting scenario, an organization could, for example, initially position the firefighters at particular locations and reflect the influence on initial state correspondingly. Similarly, by shaping the rewards, transitions, and actions of the various agents, the organizational designer might determine that the improved parallelism from coordination means that agents can safely reason over shorter time horizons. Alternatively, the designer might improve coordination by increasing T_{θ_i} for the agents, effectively asking them to be less myopic.

3.2 Related Work

In the preceding, we have stepped through the compo-

nents of a local decision model for a decision-theoretic agent, and described how an organizational designer could adjust a component to influence an agent's decisions. By adjusting the agents' components appropriately, an organizational designer can influence agents to make more complementary, globally-useful decisions, and in some cases also simplify the agents' local reasoning processes. As noted above, adjusting components like reward functions and action spaces have correspondences with familiar notions in the organizational structuring literature. However, prior work on implementing organizational influences within agents largely takes a top-down approach: given influences that a researcher's intuitions determine are pertinent, an agent architecture (such as a BDI architecture [3]) is extended to incorporate those influences. In contrast, the dimensions for organizational influence in this paper emerge from the bottom up, directly from the components of the principled decision-theoretic framework.

Much of the literature in multiagent organization design and specification concentrates on formulating organizational modeling languages (OMLs), such as MOISE$^+$ [13] and OMNI [14], among a variety of others. Though the specifics of these OMLs vary, they generally emphasize specifying an agent organization at an abstract level in terms of roles, role relationships/interactions, norms, etc. They also tend to be agnostic about how an agent would map the abstract specification into its internal reasoning processes. Hence, our work here complements that work, helping to bridge the gap between modeling and implementation by identifying opportunities and limitations in what OMLs can express that can be meaningfully mapped into influences over decision-theoretic agents.

4. BASELINE ORGANIZATION

In our preceding characterization of how an organizational designer influences an agent, the basic idea is that the design $\theta_i = \langle S_{\theta_i}, \alpha_{\theta_i}, A_{\theta_i}, R_{\theta_i}, P_{\theta_i}, T_{\theta_i} \rangle$ supplants the agent's "local" model $\mathcal{M}_i = \langle S_i, \alpha_i, A_i, R_i, P_i, T_i \rangle$. But where does an agent's (original) local model come from? Clearly, the performance improvements that an organizational design will make depends on how (dis)organized the agents are when following their initial local models. This means that we could show arbitrarily good performance improvements by initializing agents with arbitrarily bad local models.

This is a fundamental and under-addressed quandary in the artificial agent organizations research field. The combination of initial local models of agents essentially *do* comprise an organizational design. When assembling an agent system, agents might be selected based on the inherent alignment between their local models and the (organizational) biases of whomever is assembling the system. The actions agents are capable of, the states they can represent, their predispositions about what states are rewarding, etc. can all factor into decisions about which agents are included in the system.

Our evaluation of the improvement achievable by following a designed organization thus depends on defining a baseline organization. To develop as even-handed a baseline as possible, we advocate initializing local decision models by performing an uninformed mapping of the joint Dec-POMDP models into localized versions. In this way, the local models are perforce aligned with the global model, but they are not crafted to differentiate the roles and behaviors of the agents. Essentially, the philosophy is to endow each agent with a local model that directly makes the individual agent responsible for solving the global problem, to the extent its awareness and capabilities allow.

Specifically, our methodology for initializing agents' local models to provide an experimental baseline is as follows. First, we assume that the subset of state features directly observable to the agent defines its local state representation. Second, the action space of an agent is simply its component of the joint action space. Third, the local reward function is the same as the global reward function, except that any components involving features outside of the agent's local state representation are dropped, since the agent does not have values for those features. Fourth, the local transition model corresponds to the joint transition entries where the existence of other agents is moot. Finally, the initial local state distribution maps the global distribution into the local state space, and the local finite time horizon is identical to the global value. In the firefighting domain, the baseline organization is the local model as we described it in the last paragraph of Section 2.

While this method for creating a baseline model is still dependent on somewhat arbitrary decisions (e.g., which features are included in an agent's local state), the idea is that aspects that influence how an agent formulates a policy (what is rewarding, what might happen in the world, etc.) are aligned with the "true" global model but contain as little information as possible about what an agent might expect others to do in the world. We assume that it is up to an organizational designer to provide such information.

Despite our adoption of this uninformed-but-aligned baseline, we have recognized that other factors also influence the difference that organizational design can make. A simple example we've encountered is how the initial configuration of state can greatly affect whether the baseline organization is effective. In the firefighting domain, if we assume that the fires pop up across the space with uniform probability, then where should we assume firefighters begin? If we assume that they are uniformly distributed in the environment, then their local models (where they prefer fighting nearby fires) inherently lead to a good allocation of tasks (fires) to agents. If we assume that they all start in the same location, on the other hand, then the local models inherently lead to agents moving around *en masse* and yields no parallelism benefits.[2] Even randomly placing firefighters is not an answer, because distributing fires and agents in the same uniformly random way introduces its own bias. In our experiments described in Section 5.2, we present results from the two extreme environments: the agents beginning uniformly distributed; and the agents beginning clustered in the center of the grid world, which represents the best and worst case in expectation for the baseline organization respectively.

5. EVALUATION

We now turn to evaluating our claimed benefits of characterizing the organizational design space in terms of adjusting the components of an agent's decision-theoretic model. In this section, we use our simplified firefighting problem domain to investigate the effects of modifying each component individually, and in combination, as a step toward building an automated design algorithm.

[2]Note that if multiple firefighters on the same fire had a super-additive effect, instead of the sub-additive effect in our domain, then initially spreading out could be disadvantageous, while moving around in a pack might be beneficial.

The experiments that follow use the problem formulation already described (Section 2), in terms of state features, agents' actions, their transitions, and joint reward function. To test the degree to which an organizational design provides long-term benefit to a multiagent system, we run a fixed organizational design over a large number of randomly-generated problem instances, where each instance is an episode that begins with a randomized configuration of fires and ends when the time horizon is reached. By the luck of the draw, some problem instances might be well suited to one organization over another. We focus on aggregate performance over many episodes not only to smooth out the randomness of the instances but moreover to identify an organization's effectiveness over the long term, due to the assumption that organizational design has a high cost that is amortized over time. The measures of performance of interest are the expected joint reward and the planning overhead of the agents in each episode. A well-designed organization is one that improves joint reward while also simplifying each agent's local planning problem.

5.1 Comparison to Optimal

To be able to compute an upper-bound on performance (an optimal joint policy) against which to compare, we begin with problems in a simple 10×5 grid world with 2 cooperative agents and 2 fires, as illustrated in Figure 1. The distribution of fires' locations is uniformly random over the entire grid, and the fires' intensities are uniformly random over $\{1, 2, 3\}$; however, the agents always begin in the same locations (those in Figure 1). To speed up the tests without pruning any viable solutions, the finite time horizon is the maximal time either agent would require to put out both fires alone (varies per episode). To get a sense of the impact of different organizational designs, we tested three designs in addition to the baseline organization. One, called fullOverlapOrg, assigns both agents to be responsible for all fires in the entire grid. However, unlike the baseline organization where agents have no model of each other, fullOverlapOrg provides agents with improved transition models that reflect the possible activities of the other agent. Specifically, our organizational designer heuristically assumes that an agent will first fight the fire closest in its region, then the closest fire from there, and so on, until the time horizon. So, the organization adjusts the other agent's transition function to anticipate that some fires (on the other side of the grid) will have decreasing intensities even without fighting them itself, helping it refrain from rushing to distant high-intensity fires that will be addressed by someone else.

A second organizational design, called partitionOrg, partitions the locations, assigning responsibility for fires in the western 5×5 subgrid to A1, and the eastern subgrid to A2, removing actions from the agents' action spaces that would move them out of their regions. More generally, partitionOrg represents an assignment of each task to exactly one agent.

The third organization is called smallOverlapOrg, in which the 4 middle columns of the grid are in both agents' regions of responsibility. Like in partitionOrg, agents' action spaces are pruned so an agent doesn't consider moving out of its region, while like fullOverlapOrg, an agent has an adjusted transition function to reflect that fires in its local state space have a chance of going out without it fighting them.

To create the local policies, each agent uses its organizational model to create the reachable state space from the given initial state forward. It then uses CPLEX [7] to calculate the optimal local policy for the reachable state space using the linear program as formulated by Kallenberg [9].

Before describing our results, we have to address one more issue. Agents build policies that only consider states they could conceivably reach within the time horizon. Because an agent using the baseline organization models the world as if it is alone, its reachable state space does not include states where some fires' intensities decrease without it fighting them. Thus, when executing its policy it could reach an unexpected state. Rather than explode the state space by including low-probability transitions covering every possibility, in our experiments we simply assume that when an agent "falls off" its policy (reaches an unplanned state), it constructs a new policy going forward from its (unexpected) current state, and that this planning is instantaneous with respect to events in the world. (The world "waits" for the agent to replan.) While future work should treat this more realistically, for the purposes of our experiments this assumption favors less informed organizations (that fall off policy more frequently) more than informed ones, so the benefits of organizational design will be, if anything, understated. Finally, note that agents given improved transitions might still sometimes fall off policy, because the heuristics used in the transition functions are imperfect.

Our experiments are summarized in Table 1. We generated 1,500 episodes with random initial states and solved each using the 3 organizational designs (partitionOrg, smallOverlapOrg, and fullOverlapOrg), as well as the uninformed baseline organization. We also generated the optimal joint policy for each episode to compute the optimal attainable reward if the agents could afford the time to generate it and could also sense each others' positions. These results show that even simple organizational designs can improve rewards considerably compared to the baseline, but that overly restrictive organizations (partitionOrg) can degrade performance because the same agent too often must fight both fires. As one would expect, more restrictive organizations increasingly simplify agents' local decision problems. Moreover, note that all of the organizations decrease local computation over the baseline, because in the baseline both agents solve larger problems (putting out all the fires by themselves) than when they are informed (through the transition function) that they will have help.

Note that the fullOverlapOrg has greater global awareness than the other organizations; however, this increased awareness incurs greater computational costs. Because performance is basically inversely correlated with computation, we created a unified performance metric by adopting the standard methodology of having the agents sit idle at the start of execution while they create their policies. To do this, we convert the actual CPU time for policy creation into simulation time steps, and then force the agents to sit idle for that many time steps at the beginning of the episode (essentially performing NOOPs). Figure 2 presents the adjusted expected reward after accounting for computational costs as a function of the CPU time per simulation time step. These results confirm our intuitions that when computation is expensive (low c) paritionOrg is best due to its highly simplified decision process. Then as computation becomes cheaper (c increases), the more flexible organizations become superior, and finally when computation is very cheap, computing the optimal joint policy becomes best.

	Reward	Plan Time	Replans
Baseline	-15.97	86	1.32
ParitionOrg	-16.15	12	0.26
SmallOverlapOrg	-14.74	27	0.16
FullOverlapOrg	-14.70	70	0.14
Joint	-14.37	24558	0.00

Table 1: Mean experimental results for Section 5.1 for expected reward, CPU time to create initial policy (ms), and average number of times the replanning mechanism was invoked per agent per episode.

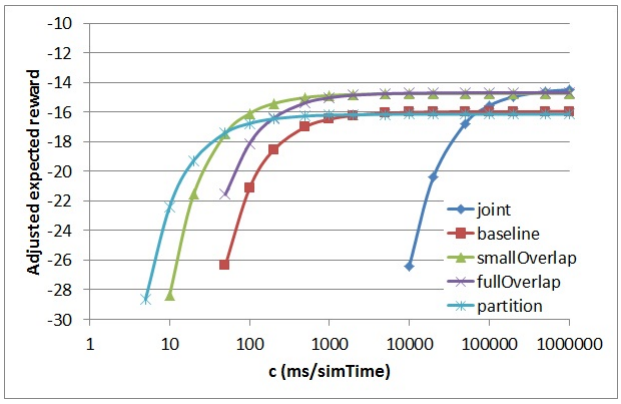

Figure 2: Adjusted rewards for Section 5.1 after accounting for computation time as a function of the CPU time per simulation time step.

5.2 Design Components

We now turn to isolating the impact of different dimensions of organizational design, corresponding to different components of the agents' decision models. For these experiments, we use larger environments with 10 cooperative agents and 10 fires on a 25×10 grid. Fires are still distributed uniformly randomly over the entire grid, with intensities drawn uniformly from $\{1, 2, 3\}$. As discussed at the end of Section 4, the initial locations of the agents can favor, or disfavor, some organizational designs. Thus, in these experiments, we consider two extreme cases of initial locations for the agents: where they are evenly spread around the environment; and where they are clustered at the center of the grid.

To understand the impact of designing along different dimensions, we implemented largely the same organizational structure using the different components. The structure inherits from the smallOverlapOrg in Section 5.1, narrowing agents' ranges of policies to consider while still providing them with some flexibility to load balance by having overlapping regions of responsibility. Specifically, the 25×10 grid is divided into 10 distinct 5×5 subgrids, one for each agent, to act as the agent's primary area of responsibility (PAR). In each (non-wall) direction, the subgrid is expanded by 3 cells to introduce overlap; conceptually, this is an agent's secondary area of responsibility (SAR). We implemented 5 organizations capturing this fundamental structure: **actionOrg** removes actions that take an agent out of its combined PAR and SAR; **stateOrg** removes features for states outside of the combined PAR and SAR; **rewardOrg** penalizes the agent with increasing severity for leaving its PAR (Manhattan distance from PAR squared); **transitionOrg** models how fires in the PAR and SAR might go out due to someone else's actions using the same heuristics as in Section 5.1 (and like stateOrg ignores more distant fires to curb state-space explosion resulting from the richer transition model); and **fullOrg** uses all of the dimensional levers just described.

We generated 100 random episodes (initial fire configurations), for each of the spread and clustered variations of agents' initial locations. For each episode, we ran each of the 5 organizations above, as well as the baseline organization. The problems were too large to compute optimal joint policies. Table 2 presents the results for these experiments.

These results illustrate many of the intuitions from Section 3.1. As others have discovered, reward shaping can be a powerful tool for increasing the expected joint reward; however, it does not generally reduce the agents' computational efforts. Shaping the transition functions can also yield a large increase in the expected reward; however, it substantially increases the agents' computational costs. Notice that organizations with improved transition functions replan during execution much less, indicating that if recovering from falling off policy incurs non-negligible cost, then transition shaping could be of critical importance. We also observe that constraining the agents' action or state spaces can greatly simplify the agents' decision problems and can also increase the expected joint reward. Finally, with fullOrg, we observe that the organizational influences in the components are not completely redundant, as it is largely possible to obtain the additive benefits found in each of the other organizations. The drop in expected reward as compared to transitionOrg is due to the shaped reward functions urging agents to quickly go their respective PARs rather than stop and fight fires along the way. However, also note that the computation time is drastically reduced, suggesting that the tradeoff would be beneficial unless computation is exceptionally cheap.

Finally, the reader may have noted that our experiments did not evaluate the impact of restructuring the other two components: the initial state distribution α_{θ_i}; and the time horizon T_{θ_i}. One could envision organizations that modify T_i to give agents specific roles for planning horizons, where some agents focus on the near-term and others on the long-term, though the organization would probably also focus an agent's action space A_{θ_i} on actions of a matched granularity. If α_i summarizes the exogenously-determined initial state, the designer can only map this into the agent's adjusted state space S_{θ_i}, as was implicitly done for the organizational variations above. However, as seen in the relative performance between the spread and clustered environments, if the organization can impose initial states on agents (spreading them out in anticipation of arising fire configurations), then this provides an additional lever for influencing collective performance.

6. ORGANIZATIONAL ADEPTNESS

As demonstrated in Section 5.1, an organizational designer confronts tradeoffs in deciding how tightly to influence the agents. If not tightly enough, agents might duplicate effort or work at cross purposes while, if too tightly, agents might load balance poorly or have tasks fall between the cracks. We assume the organizational designer can use a model of the expected problem distribution to form an organization that, in expectation, will work best. However, if its model is (or over time becomes) erroneous, the agents must decide how to refine, revise, or even abandon that organizational structure.

	Large Problems (Spread)			Large Problems (Clustered)		
	Reward	**Plan Time**	**Replans**	**Reward**	**Plan Time**	**Replans**
Baseline	-107.40	1646	7.89	-436.7	10912	0.00
RewardOrg	-91.45	1817	7.49	-242.0	11051	9.38
TransitionOrg	-85.14	14606	0.86	-222.5	10859	0.55
ActionOrg	-94.14	551	7.70	-264.5	621	8.56
StateOrg	-94.14	1237	2.60	-254.4	1588	1.50
FullOrg	-87.51	5476	0.88	-250.4	2652	1.02

Table 2: Mean experimental results for Section 5.2 for expected reward, CPU time to create initial policy (ms), and average number of times the replanning mechanism was invoked per agent per episode.

Following Corkill *et al.* [3], we refer to agents with this capability as *organizationally adept agents* (OAAs). As advocated elsewhere [5], agents need *operational control* capabilities to elaborate and refine organizational control guidelines. For example, agents with overlapping areas of responsibility can use operational control to resolve who is responsible for which tasks in the current situation. But operational control can be expensive (in computation, communication, delay, etc.), so organizations that more narrowly define the roles of each agent, and thus require less operational control, can be preferable. However, if the designer's assumptions about the problems that will be encountered are wrong, a narrower organization might leave too little latitude for operational refinement to meet coordination needs. An OAA should be able to compare the problems actually encountered to the organizational designer's expectations, and decide whether a change or abandonment of organization is warranted, thus allowing for narrower organizations to be utilized.

Our decision-theoretic formulation of organizational design provides a framework for agents to make such comparisons and decisions, and thus for a more formal characterization of what it means for an agent to be organizationally adept. For example, an OAA i can compare its organizational initial state distribution α_{θ_i} with the initial states it has actually witnessed over a series of episodes to detect mismatches. Similarly, i can recognize that, for example, the probabilities that fires will be be put out by others according to P_{θ_i} are not supported by statistics over observed transitions, or that states whose rewards have been shaped by the organization are seldom reachable.

When the expectations implied in the organizational structure stem from high-level assumptions the designer has about the problem domain, such as that fires will appear uniformly randomly through the entire region, the designer can annotate an organization with the assumptions on which its selection is conditioned. Our current decision-theoretic OAA architecture captures such annotations in terms of variables to monitor and expectations over their values. More formally, optionally along with its organizational specification θ_i, agent i can receive a set of monitor-variable and value-expectation pairs $\psi_i = \langle(\psi_{i_1}, v_{i_1})...(\psi_{i_m}, v_{i_m})\rangle$. (If none are provided, the OAA can still use the expectation implicit in θ_i.) Essentially, the annotated formulation indicates that, to the extent that the monitor-variables take on values consistent with expectations, the organization should be followed.

As a preliminary illustration of these OAA concepts, we use our 10-agent problem domain from Section 5.2 and consider two different models of how fires arise: having an increasingly higher probability of arising toward the east end of the grid; and having an increasingly higher probability of arising toward the west end. Note that the desired organizational behavior is significantly different between the two environments; in the eastEnvironment we would want to designate more agents to the eastern region (and *vice versa* for the westEnvironment). We designed a specialized organization for each case, which are analogous to fullOrg from Section 5.2 except that the PARs are non-uniformly sized to compensate for the biased fire distributions. For example, in the westOrg, 3 agents are responsible for the western 4 columns (4×3, 4×4, and 4×3 PARs). Working eastward, the PARs get progressively larger, starting with two 4×5 PARs (stacked vertically), then two 5×5 PARs, then two 6×5 PARs. Finally, a lone agent is responsible for the eastern edge with a 5×10 PAR. The eastOrg is a symmetric copy of the westOrg. Associated with each organization is a set of monitor variables informing each agent that the organizational designer expected one fire, on average, to be in its PAR (for that organization).

We provided the agents with both of these annotated organizations, in addition to fullOrg, which is weakly applicable both environments. The agents all initially adopt (based on the designer's directives) fullOrg to reflect the designer's uncertainty about the environment. As episodes are experienced, the agents track their monitor variables. They then jointly aggregate this observational evidence, e, and perform Bayesian inference to calculate the likelihood that each of the environments is the actual environment being observed, which are used to estimate the expected reward of following each available organization. The agents then collectively and greedily adopt the organization with the highest anticipated expected reward. Formally, they adopt Θ^*:

$$\Theta^* = \arg\max_{\Theta} E[R|\Theta, e] - c(\Theta_c, \Theta)$$

$$E[R|\Theta, e] = \sum_j Pr(M_j|e) E[R|\Theta, M_j]$$

where $c(\Theta_c, \Theta)$ is the cost of switching from the current organization Θ_c to Θ. We assume there is no cost for remaining in the same organization, $\forall i\ c(\Theta_i, \Theta_i) = 0$. $Pr(M_j|e)$ is the likelihood of environmental model M_j being the actual model given e, which the agents calculate via Bayesian inference. $E[R|\Theta, M_j]$ is the expected reward of following organization Θ in M_j, which we assume is provided by the organizational designer in the annotations. For our experiments, we estimated $E[R|\Theta, M_j]$ by *a priori* simulating Θ on a training set of episodes created from M_j.

Our experiments present the agents with episode batches where the true environment model is selected uniformly randomly from the two environments every 20 episodes (all organizations face the same episodes in the same order).

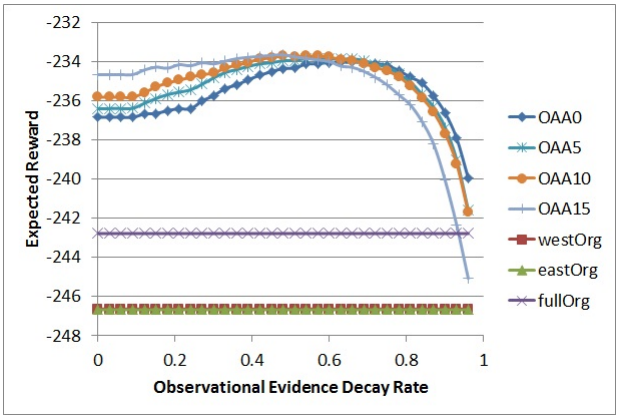

Figure 3: Expected reward as a function of the observational evidence decay rate.

Only at the end of each episode are the agents allowed to collectively adopt whichever organization they deem best. Since the true environment is dynamic, we allow the organizational designer to set a decay rate in the annotations, which the agents use to decay the importance of past monitor variable observations. We performed experiments with several organizations: statically using the east/west/fullOrg for every episode; and several parameter settings of the OAA process described above. OAAx refers to the OAA process above where the organizational switching cost is x.

Our results are summarized in Figure 3, which confirms several intuitions. Firstly, statically following either specialized organization performs poorly since they suffer when being used in the environment they were not intended for; however, statically following fullOrg makes a noticeable improvement by being weakly suited to both environments. Secondly, by allowing the agents to react to the shifting environment, the OAA capability (in general) can yield a large performance gain. Finally, if the organizational switching cost is low, the agents should maintain sufficient observational evidence history in order to prevent the agents from switching organizations due to a transient episode, such as when an episode from the eastEnvironment happens to "look" like an episode from the westEnvironment due to unlikely fire locations.

7. CONCLUSIONS

In this paper we have presented a decision-theoretic framework that provides a systematic method for enumerating the possible ways in which an organization can influence agents' decision-making processes. We have intuitively described and empirically demonstrated how influencing the various Dec-POMDP components can both increase the agents' expected joint reward as well as simplify their local decision problems as compared to a baseline local model. Finally, in Section 6 we have shown how our organizational framework provides a more formal characterization of what organizational adeptness can mean compared to prior work and have provided preliminary empirical evidence of the benefits of OAA. In the future, we plan to expand the functionality of OAA; for example, rather than greedily reacting to current model likelihoods, the agents could make predictions about the ways the environment is changing and preemptively switch organizations. Additionally, we plan to investigate the effects of an agent reasoning unilaterally about its observational evidence and individually changing its organization (as opposed to a central decision process), as well as the possibility of gradually blending organizations together when switching as opposed to the all-or-nothing switching described in this paper. Finally, using the insights gained from Section 5, we plan to develop an automated organizational designer that can create organizations within our structured framework.

8. ACKNOWLEDGMENTS

We thank the anonymous reviewers for their thoughtful comments, and our collaborators at the University of Massachusetts for their consistently helpful feedback. This work was supported by NSF grant IIS-0964512.

9. REFERENCES

[1] A. K. Agogino and K. Tumer. Multi-agent reward analysis for learning in noisy domains. In *AAMAS*, pages 81–88, 2005.

[2] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research*, 27(4):819–840, 2002.

[3] D. Corkill, E. Durfee, V. Lesser, H. Zafar, and C. Zhang. Organizationally Adept Agents. In *COINS2011 Workshop at AAMAS*, 2011.

[4] D. D. Corkill and S. E. Lander. Diversity in Agent Organizations. *Object Magazine*, 8(4):41–47, 1998.

[5] E. H. Durfee and Y. p. So. The effects of runtime coordination strategies within static organizations. In *IJCAI*, pages 612–619, 1997.

[6] M. S. Fox, M. Barbuceanu, M. Gruninger, and J. Lin. An organizational ontology for enterprise modeling. In *Simulating organizations*, pages 131–152. MIT Press, Cambridge, MA, USA, 1998.

[7] IBM. IBM ILOG CPLEX, 2011. See http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/.

[8] N. R. Jennings. Commitments and conventions: The foundation of coordination in multi-agent systems. *The Knowledge Engineering Review*, 8(03):223–250, 1993.

[9] L. C. M. Kallenberg. *Linear Programming and Finite Markovian Control*. Mathematical Centre Tracts, 1983.

[10] A. Y. Ng, D. Harada, and S. Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *ICML*, pages 278–287, 1999.

[11] Y. Shoham and M. Tennenholtz. On social laws for artificial agent societies: off-line design. *Artificial Intelligence*, 73(1-2):231 – 252, 1995.

[12] Y. p. So and E. H. Durfee. Designing organizations for computational agents. In *Simulating Organizations*, pages 47–64. MIT Press, Cambridge, MA, USA, 1998.

[13] M. B. van Riemsdijk, K. V. Hindriks, C. M. Jonker, and M. Sierhuis. Formalizing organizational constraints: A semantic approach. In *AMMAS*, pages 823–830, 2010.

[14] J. Vázquez-Salceda, V. Dignum, and F. Dignum. Organizing multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 11:307–360, November 2005.

[15] D. Wolpert and K. Tumer. Optimal payoff functions for members of collectives. *Advances in Complex Systems*, 4(2/3):265–279, 2001.