Coordination in Multiagent Reinforcement Learning: A Bayesian Approach

Georgios Chalkiadakis
Dept. of Computer Science, Univ. of Toronto
Toronto, ON, M5S 3G4, Canada
gehalk@cs.toronto.edu

Craig Boutilier
Dept. of Computer Science, Univ. of Toronto
Toronto, ON, M5S 3H5, Canada
cebly@cs.toronto.edu

ABSTRACT

Much emphasis in multiagent reinforcement learning (MARL) research is placed on ensuring that MARL algorithms (eventually) converge to desirable equilibria. As in standard reinforcement learning, convergence generally requires sufficient exploration of strategy space. However, exploration often comes at a price in the form of penalties or foregone opportunities. In multiagent settings, the problem is exacerbated by the need for agents to "coordinate" their policies on equilibria. We propose a Bayesian model for optimal exploration in MARL problems that allows these exploration costs to be weighed against their expected benefits using the notion of value of information. Unlike standard RL models, this model requires reasoning about how one's actions will influence the behavior of other agents. We develop tractable approximations to optimal Bayesian exploration, and report on experiments illustrating the benefits of this approach in identical interest games.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Miscellaneous

General Terms

Algorithms

Keywords

multiagent learning, reinforcement learning, Bayesian methods

1. INTRODUCTION

The application of reinforcement learning (RL) to multiagent systems has received considerable attention [12, 3, 7, 2]. However, in multiagent settings, the effect (or benefit) of one agent's actions are often directly influenced by those of other agents. This adds complexity to the learning problem, requiring that an agent not only learn what effects its actions have, but also how to coordinate or align its action choices with those of other agents. Fortunately, RL methods can often lead to coordinated or equilibrium behavior. Empirical and theoretical investigations have shown that standard (single-agent) RL methods can under some circumstances lead to equilibria [3, 8, 2], as can methods designed to explicitly account for the behavior of other agents [12, 3, 7].

Multiagent reinforcement learning (MARL) algorithms face difficulties not encountered in single-agent settings: the existence of multiple equilibria. In games with unique equilibrium strategies (and hence values), the "target" being learned by agents is well-defined. With multiple equilibria, MARL methods face the problem that agents must coordinate their choice of equilibrium (and not just their actions).[1] Empirically, the influence of multiple equilibria on MARL algorithms is often rather subtle, and certain game properties can make convergence to undesirable equilibria very likely [3]. For obvious reasons, one would like RL methods that converge to desirable (e.g., optimal) equilibria. A number of heuristic exploration strategies have been proposed that in fact increase the probability (or even guarantee) that optimal equilibria are reached in identical interest games [3, 11, 10, 16].

Unfortunately, methods that encourage or force convergence to optimal equilibria often do so at a great cost. Coordination on a "good" strategy profile often requires exploration in parts of policy space that are very unrewarding. In such a case, the benefits of eventual coordination to an optimal equilibrium ought to be weighed against the cost (in terms of reward sacrificed while learning to play that equilibrium) [1]. This is simply the classic RL exploration-exploitation tradeoff in a multiagent guise. In standard RL, the choice is between: *exploiting* what one knows about the effects of actions and their rewards by executing the action that—given current knowledge—appears best; and *exploring* to gain further information about actions and rewards that has the potential to *change* the action that appears best. In the multiagent setting, the same tradeoff exists with respect to action and reward information; but another aspect comes to bear: the influence one's action choice has on the future action choices of other agents. In other words, one can exploit one's current knowledge of the strategies of others, or explore to try to find out more information about those strategies.

In this paper, we develop a model that accounts for this *generalized exploration-exploitation tradeoff* in MARL. We adopt a Bayesian, model-based approach to MARL, much like the single-agent model described in [4]. Value of information will play a key role in determining an agent's exploration policy. Specifically, the value of an action consists of two components: its estimated value given current model estimates, and the expected decision-theoretic *value of information* it provides (informally, the ability this information has to change future decisions). We augment both parts of this value calculation in the MARL context. The estimated value of an action given current model estimates requires predicting how

AAMAS'03, July 14–18, 2003, Melbourne, Australia.

[1] The existence of multiple equilibria can have a negative impact on known theoretical results for MARL [7, 8].

the action will influence the future action choices of other agents. The value of information associated with an action includes the information it provides about other agents's strategies, not just the environment model. Both of these changes require that an agent possess some model of the strategies of other agents, for which we adopt a Bayesian view [9]. Putting these together, we derive optimal exploration methods for (Bayesian) multiagent systems.

After reviewing relevant background and related work, we develop a general Bayesian model, and describe computational approximations for optimal—with respect to the tradeoff discussed above—exploration. We describe a number of experiments illustrating this approach, with our experiments focusing on identical interest games, since these have has been the almost exclusive target of recent research on heuristic exploration methods.

2. BACKGROUND

We begin with basic background on RL and stochastic games.

2.1 Bayesian Reinforcement Learning

We assume an agent learning to control a stochastic environment modeled as a Markov decision process (MDP) $\langle \mathcal{S}, \mathcal{A}, R, \Pr \rangle$, with finite state and action sets $\mathcal{S}, \mathcal{A}$, reward function R, and dynamics $\Pr$. The dynamics $\Pr$ refers to a family of transition distributions $\Pr(s, a, \cdot)$, where $\Pr(s, a, s')$ is the probability with which state s' is reached when action a is taken at s. $R(s, r)$ denotes probability with which reward r is obtained when state s is reached.[2] The agent is charged with constructing an optimal Markovian policy $\pi : \mathcal{S} \mapsto \mathcal{A}$ that maximizes the expected sum of future discounted rewards over an infinite horizon: $E_\pi[\sum_{t=0}^{\infty} \gamma^t R^t | S^0 = s]$. This policy, and its value, $V^*(s)$ at each $s \in \mathcal{S}$, can be computed using standard algorithms such as policy or value iteration.

In the RL setting, the agent does not have direct access to R and $\Pr$, so it must learn a policy based on its interactions with the environment. Any of a number of RL techniques can be used to learn an optimal policy. We focus here on *model-based RL* methods, in which the learner maintains an estimated MDP $\langle \mathcal{S}, \mathcal{A}, \widehat{R}, \widehat{\Pr} \rangle$, based on the set of experiences $\langle s, a, r, t \rangle$ obtained so far. At each stage (or at suitable intervals) this MDP can be solved (exactly or approximately).

Bayesian methods allow agents to incorporate priors and explore optimally. We assume some prior density P over possible dynamics D and reward distributions R, which is updated with each data point $\langle s, a, r, t \rangle$.[3] Letting H denote the (current) *state-action history* of the observer, we use the posterior $P(D, R|H)$ to determine an appropriate action choice at each stage. The formulation of [4] renders this update tractable by assuming a convenient prior. Specifically, the following assumptions are made: (a) the density P is factored over R and D; (b) $P(D)$ is the product of independent local densities $P(D^{s,a})$ for each transition distribution $\Pr(s, a, \cdot)$; and (c) each density $P(D^{s,a})$ is a Dirichlet.[4]

To model $P(D^{s,a})$ we require a Dirichlet parameter vector $\mathbf{n}^{s,a}$ with entries $n^{s,a,s'}$ for each possible successor state s'. The expectation of $\Pr(s, a, s')$ w.r.t. P is given by $n^{s,a,s'} / \sum_i n^{s,a,s_i}$. Update of a Dirichlet is ~~a~~ straightforward: given prior $P(D^{s,a}; \mathbf{n}^{s,a})$ and data vector $\mathbf{c}^{s,a}$ (where $c_i^{s,a}$ is the number of observed transitions from s to s_i under a), the posterior is given by parameter vector $\mathbf{n}^{s,a} + \mathbf{c}^{s,a}$. Thus the posterior $P(D|H)$ can be factored into posteriors over local families, each of the form:

$$P(D^{s,a}|H^{s,a}) = z \Pr(H^{s,a}|D^{s,a}) P(D^{s,a}) \qquad (1)$$

where $H^{s,a}$ is the history of s, a-transitions updates are of Dirichlet parameters, and z is a normalizing constant.

The Bayesian approach has several advantages over other approaches to model-based RL. First, it allows the natural incorporation of prior knowledge. Second, approximations to optimal Bayesian exploration can take advantage of this model [4]. We elaborate on optimal exploration below in the MARL context.

2.2 Stochastic Games and Coordination

A *normal form game* is a tuple $G = \langle \alpha, \{A_i\}_{i \in \alpha}, \{U_i\}_{i \in \alpha} \rangle$, where α is a collection of agents, A_i is the set of actions available to agent i, and U_i is agent i's payoff function. Letting $A = \times A_i$ denote the set of *joint actions*, $U_i(a)$ denotes the real-valued utility obtained by i if the agents execute $a \in A$. We refer to any $\sigma_i \in \Delta(A_i)$ as a *mixed strategy*. A *strategy profile* σ is a collection of strategies, one per agent. We often write σ_i to refer to agent i's component of σ, and σ_{-i} to denote a reduced strategy profile dictating all strategies except that for i. We use $\sigma_{-i} \circ \sigma_i$ to denote the (full) profile obtained by augmenting σ_{-i} with σ_i. Let σ_{-i} be some reduced strategy profile. A *best response* to σ_{-i} is any strategy σ_i s.t. $U_i(\sigma_{-i} \circ \sigma_i) \geq U_i(\sigma_{-i} \circ \sigma_i')$ for any $\sigma_i' \in \Delta(A_i)$. We define $\mathit{BR}(\sigma_{-i})$ to be the set of such best responses. A *Nash equilibrium* is any profile σ s.t. $\sigma_i \in \mathit{BR}(\sigma_{-i})$ for all agents i.

Nash equilibria are generally viewed as the standard solution concept to games of this form. However, it is widely recognized that the equilibrium concept has certain (descriptive and prescriptive) deficiencies. One important problem (among several) is the fact that games may have multiple equilibria, leading to the problem of equilibrium selection. As an example, consider the simple two-player identical interest game called the *penalty game* [3], shown here in standard matrix form:

	$a0$	$a1$	$a2$
$b0$	10	0	k
$b1$	0	2	0
$b2$	k	0	10

Here agent A has moves $a0, a1, a2$ and B has moves $b0, b1, b2$. The payoffs to both players are identical, and $k < 0$ is some penalty. There are three pure equilibria. While $\langle a0, b0 \rangle$ and $\langle a2, b2 \rangle$ are the optimal equilibria, the symmetry of the game induces a coordination problem for the agents. With no means of breaking the symmetry, and the risk of incurring the penalty if they choose different optimal equilibria, the agents might in fact focus on the suboptimal equilibrium $\langle a1, b1 \rangle$.

Learning models have become popular as a means of tackling equilibrium selection [9, 6]. Assuming repeated play of some "stage game," these methods require an agent to make some prediction about the play of others at the current stage based on the history of interactions, and play the current stage game using these predictions. One simple model is *fictitious play* [14]: at each stage, agent i uses the empirical distribution of observed actions by other agents over past iterations as reflective of the mixed strategy they will play at the current stage; agent i then plays a best response to these estimated strategies. This method is known to converge (in various senses) to equilibria for certain classes of games.

Another interesting learning model is the Bayesian approach of Kalai and Lehrer [9]. In this model, an agent maintains a distribution over all strategies that could be played by other agents. This strategy space is not confined to strategies in the stage game, but allows for beliefs about strategies another agent could adopt for

[2] We will treat this distribution as if it has support over a finite set of possible values r, but more general density functions can be used.

[3] We write D for the family of distributions for notational clarity.

[4] We assume reward densities are modeled similarly, with a Dirichlet prior over reward probabilities for each s. Gaussian reward distributions [4] pose no serious complications.

the repeated game itself.[5] Standard Bayesian updating methods are used to maintain these beliefs over time, and best responses are played by the agent w.r.t. the expectation over strategy profiles.

Repeated games are a special case of *stochastic games* [15, 5], which can be viewed as a multiagent extension of MDPs. Formally, a stochastic game $G = \langle \alpha, \{A_i\}_{i\in\alpha}, \mathcal{S}, \Pr, \{R_i\}_{i\in\alpha}\rangle$ consists of five components. The agents α and action sets A_i are as in a typical game, and the components $\mathcal{S}$ and $\Pr$ are as in an MDP, except that $\Pr$ now refers to joint actions $a \in A = \times A_i$. R_i is a the reward function for agent i, defined over states $s \in S$ (pairs $\langle s, a\rangle \in S \times A$). The aim of each agent is, as with an MDP, to act to maximize the expected sum of discounted rewards. However, the presence of other agents requires treating this problem in a game theoretic fashion. For particular classes of games, such as zero-sum stochastic games [15, 5], algorithms like value iteration can be used to compute Markovian equilibrium strategies.

The existence of multiple "stage game" equilibria is again a problem that plagues the construction of optimal strategies for stochastic games. Consider another simple identical interest example, the stochastic game shown in Figure 3(a). In this game, there are two optimal strategy profiles that maximize reward. In both, the first agent chooses to "opt in" at state s_1 by choosing action a, which takes the agents (with high probability) to state s_2; then at s_2 both agents either choose a or both choose b—either joint strategy gives an optimal equilibrium.

Intuitively, the existence of these two equilibria gives rise to a coordination problem at state s_2. If the agents choose their part of the equilibrium randomly, there is a 0.5 chance that they miscoordinate at s_2, thereby obtaining an expected immediate reward of 0. On this basis, one might be tempted to propose methods whereby the agents decide to "opt out" at s_1 (the first agent takes action b) and obtain the safe payoff of 6. However, if we allow some means of coordination—for example, simple learning rules like fictitious play or randomization—the sequential nature of this problem means that the short-term risk of miscoordination at s_2 can be more than compensated for by the eventual stream of high payoffs should they coordinate. Boutilier [1] argues that the solution of games like this, assuming some (generally, history-dependent) mechanism for resolving these stage game coordination problems, requires explicit reasoning about the odds and benefits of coordination, the expected cost of attempting to coordinate, and the alternative courses of action.

Repeated games can be viewed as stochastic games with a single state. If our concern is not with stage-game equilibrium, but with reward accrued over the sequence of interactions, techniques for solving stochastic games can be applied to solving such repeated games. Furthermore, the points made above regarding the risks associated with using specific learning rules for coordination can be applied with equal force to repeated games.

2.3 Multiagent RL

In this section, we describe some existing approaches to MARL, and point out recent efforts to augment standard RL schemes to encourage RL agents in multiagent systems to converge to optimal equilibria. Intuitively, MARL can be viewed as the direct or indirect application of RL techniques to stochastic games in which the underlying model (i.e., transitions and rewards) are unknown. In some cases, it is assumed that the learner is even unaware (or chooses to ignore) the existence of other agents.

Formally, we suppose we have some underlying stochastic game $G = \langle \alpha, \{A_i\}_{i\in\alpha}, \mathcal{S}, \Pr, \{R_i\}_{i\in\alpha}\rangle$. We consider the case where each agent knows the "structure" of the games—that is, it knows the set of agents, the actions available to each agent, and the set of states—but knows neither the dynamics $\Pr$, nor the reward functions R_i. The agents learn how to act in the world through experience. At each point in time, the agents are at a known state s, and each agent i executes one of its actions; the resulting joint action a induces a transition to state t and a reward r_i for each agent. We assume that each agent can observe the actions chosen by other agents, the resulting state t, and the rewards r_i.

Littman [12] devised an extension of Q-learning for zero-sum Markov games called *minimax-Q*. At each state, the agents have estimated Q-values over joint actions, which can be used to compute an (estimated) equilibrium value at that state. Minimax-Q converges to the equilibrium value of the stochastic game [13]. Hu and Wellman [7] apply similar ideas—using an equilibrium computation on estimated Q-values to estimate state values—to general sum games, with somewhat weaker convergence guarantees. Algorithms have also been devised for agents that do not observe the behavior of their counterparts [2, 8].

Identical interest games have drawn much attention, providing suitable models for task distribution among teams of agents. Claus and Boutilier [3] proposed several MARL methods for repeated games in this context. A simple *joint-action learner* (JAL) protocol learned the (myopic, or one-stage) Q-values of joint actions. The novelty of this approach lies in its exploration strategy: (a) a fictitious play protocol estimates the strategies of other agents; and (b) exploration is biased by the *expected Q-value of actions*. Specifically, the estimated value of an action is given by its expected Q-value, where the expectation is taken w.r.t. the fictitious play beliefs over the other agents's strategies. When semi-greedy exploration is used, this method will converge to an equilibrium in the underlying stage game.

One drawback of the JAL method is the fact that the equilibrium it converges to depends on the specific path of play, which is stochastic. Certain equilibria can exhibit serious resistance—for example, the odds of converging to an optimal equilibrium in the penalty game above are quite small (and decrease dramatically with the magnitude of the penalty). Claus and Boutilier propose several heuristic methods that bias exploration toward optimal equilibria: for instance, action selection can be biased toward actions that form part of an optimal equilibrium. In the penalty game, for instance, despite the fact that agent B may be predicted to play a strategy that makes the $a0$ look unpromising, the repeated play of the $a0$ by A can be justified by optimistically assuming B will play its part of this optimal equilibrium. This is further motivated by the fact that repeated play of $a0$ would eventually *draw* B toward this equilibrium.

This issue of learning optimal equilibria in identical interest games has been addressed recently in much greater detail. Lauer and Riedmiller [11] describe a Q-learning method for identical interest stochastic games that explicitly embodies this optimistic assumption in its Q-value estimates. Kapetanakis and Kudenko [10] propose a method called FMQ for repeated games that uses the optimistic assumption to bias exploration, much like [3], but in the context of individual learners. Wang and Sandholm [16] similarly use the optimistic assumption in repeated games to guarantee convergence to an optimal equilibrium. We critique these methods below.

3. A BAYESIAN VIEW OF MARL

The spate of activity described above on MARL in identical interest games has focused exclusively on devising methods that ensure eventual convergence to optimal equilibria. In cooperative

[5] A strategy in the repeated game is any mapping from the observed history of play to a (stochastic) action choice. This admits the possibility of modeling other agents's learning processes.

games, this pursuit is well-defined, and in some circumstances may be justified. However, these methods do not account for the fact that—by *forcing* agents to undertake actions that have potentially drastic effects in order to reach an optimal equilibrium—they can have a dramatic impact on accumulated reward. The penalty game was devised to show that these highly penalized states can bias (supposedly rational) agents away from certain equilibria; yet optimistic exploration methods ignore this and blindly pursue these equilibria at all costs. Under certain performance metrics (e.g., average reward over an infinite horizon) one might justify these techniques.[6] However, using the discounted reward criterion (which all of these methods are designed for), the tradeoff between long-term benefit and short-term cost should be addressed.

This tradeoff was discussed above in the context of known-model stochastic games. In this section, we attempt to address the same tradeoff in the RL context. To do so, we formulate a Bayesian approach to model-based MARL. By maintaining probabilistic beliefs over the space of models and the space of opponent strategies, our learning agents can explicitly account for the effects their actions can have on (a) their knowledge of the underlying model; (b) their knowledge of the other agent strategies; (c) expected immediate reward; and (d) expected future behavior of other agents. Components (a) and (c) are classical parts of the single-agent Bayesian RL model [4]. Components (b) and (d) are key to the multiagent extension, allowing an agent to explicitly reason about the potential costs and benefits of coordination.

3.1 Theoretical Underpinnings

We assume a stochastic game G in which each agent knows the game structure, but not the reward or transition models. A learning agent is able to observe the actions taken by all agents, the resulting game state, and the rewards received by other agents. Thus an agent's experience at each point in time is simply $\langle s, a, \mathbf{r}, t\rangle$, where s is a state in which joint action a was taken, $\mathbf{r} = \langle r_1, \cdots, r_n\rangle$ is the vector of rewards received, and t is the resulting state.

A *Bayesian MARL agent* has some prior distribution over the space of possible models as well as the space of possible strategies being employed by other agents. These beliefs are updated as the agent acts and observes the results of its actions and the action choices of other agents. The strategies of other agents may be history-dependent, and we allow our Bayesian agent (BA) to assign positive support to such strategies. As such, in order to make accurate predictions about the actions others will take, the BA must monitor appropriate observable history. In general, the history (or summary thereof) required will be a function of the strategies to which the BA assigns positive support. We assume that the BA keeps track of sufficient history to make such predictions.[7]

The *belief state* of the BA has the form $b = \langle P_M, P_S, s, h\rangle$, where: P_M is some density over the space of possible models (i.e., games); P_S is a joint density over the possible strategies played by other agents; s is the current state of the system; and h is a summary of the relevant aspects of game history, sufficient to predict the action of any agent given any strategy consistent with P_S. Given experience $\langle s, a, \mathbf{r}, t\rangle$, the BA updates its belief state using standard Bayesian methods. The updated belief state is:

$$b' = b(\langle s, a, \mathbf{r}, t\rangle) = \langle P'_M, P'_S, t, h'\rangle \qquad (2)$$

Updates are given by Bayes rule: $P'_M(m) = z\Pr(t, \mathbf{r}|a, m)P_M(m)$ and $P'_S(\sigma_{-i}) = z\Pr(a_{-i}|s, h, \sigma_{-i})P_S(\sigma_{-i})$. And h' is a suitable update of the observed history (as described above). This model combines aspects of Bayesian reinforcement learning [4] and Bayesian strategy modeling [9].

To make belief state maintenance tractable (and admit computationally viable methods for action selection below), we assume a specific form for these beliefs [4]. First, our prior over models will be factored into independent local models for both rewards and transitions. We assume independent priors P^s_R over reward distributions at each state s, and $P^{s,a}_D$ over system dynamics for each state and joint action pair. These local densities are Dirichlet, which are conjugate for the multinomial distributions to be learned. This means that each density can be represented using a small number of hyperparameters, expected transition probabilities can be computed readily, and the density can be updated easily. For example, our BA's prior beliefs about the transition probabilities for joint action a at state s will be represented by a vector $\mathbf{n}^{s,a}$ with one parameter per successor state t. Expected transition probabilities and updates of these beliefs are as described in Section 2.1. The independence of these local densities is assured after each update.

Second, we assume that the beliefs about opponent strategies can be factored and represented in some convenient form. For example, it would be natural to assume that the strategies of other agents are independent. Simple fictitious play models could be used to model the BA's beliefs about opponent strategies (corresponding to Dirichlet priors over mixed strategies), allowing ready update and computation of expectations, and obviating the need to store history in the belief state. Similarly, distributions over specific classes of finite state controllers could also be used. We will not pursue further development of such models in this paper, since we use only simple opponent models in our experiments below. But the development of tractable classes of (realistic) opponent models remains an interesting problem.

We provide a different perspective on Bayesian exploration than that described in [4]. The value of performing an action a_i at a belief state b can be viewed as involving two main components: an expected value with respect to the current belief state; and its impact on the current belief state. The first component is typical in RL, while the second captures the *expected value of information* (EVOI) of an action. Since each action gives rise to some "response" by the environment that changes the agent's beliefs, and these changes in belief can influence subsequent action choice and expected reward, we wish to quantify the value of that information by determining its impact on *subsequent* decisions.

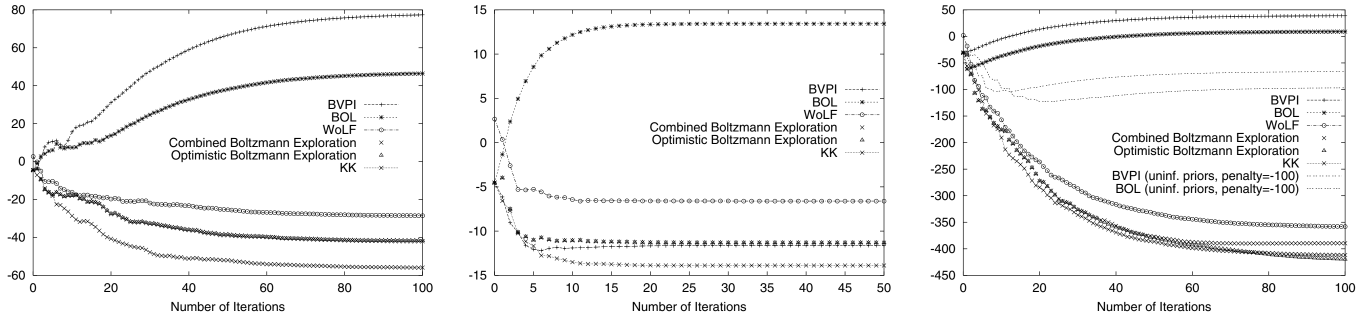
EVOI need not be computed directly, but can be combined with "object-level" expected value through the following Bellman equations over the belief state MDP:

$$\begin{aligned} Q(a_i, b) = \sum_{a_{-i}} \Pr(a_{-i}|b) \sum_{t} \Pr(t|a_i \circ a_{-i}, b) \\ \sum_{\mathbf{r}} \Pr(\mathbf{r}|a_i \circ a_{-i}, b)[r + \gamma V(b(\langle s, a, \mathbf{r}, t\rangle))] \qquad (3) \\ V(b) = \max_{a_i} Q(a_i, b) \qquad (4) \end{aligned}$$

These equations describe the solution to the POMDP that represents the exploration-exploitation problem, by conversion to a belief state MDP. These can (in principle) be solved using any method for solving high-dimensional continuous MDPs—of course, in practice, a number of computational shortcuts and approximations will be required (as we detail below). We complete the specification

[6] Even then, more refined measures such as bias optimality might cast these techniques in a less favorable light.

[7] For example, should the BA believe that its opponent's strategy lies in the space of finite state controller that depends on the last two joint actions played, the BA will need to keep track of these last two actions. If it uses fictitious play beliefs (which can be viewed as Dirichlet priors) over strategies, no history need be maintained.



(a) $k = -20$, $\gamma = 0.95$, uninf. priors (b) $k = -20$, $\gamma = 0.75$, uninf. priors (c) $k = -100$, $\gamma = 0.95$, inform. priors

Figure 1: Penalty Game Results

with the straightforward definition of the following terms:

$$\Pr(a_{-i}|b) = \int_{\sigma_{-i}} \Pr(a_{-i}|\sigma_{-i}) P_S(\sigma_{-i}) \quad (5)$$

$$\Pr(t|a,b) = \int_m \Pr(t|s,a,m) P_M(m) \quad (6)$$

$$\Pr(r|b) = \int_m \Pr(r|s,m) P_M(m) \quad (7)$$

We note that the evaluation of Eqs. 6 and 7 is trivial using the decomposed Dirichlet priors mentioned above.

This formulation determines the optimal policy as a function of the BA's belief state. This policy incorporates the tradeoffs between exploration and exploitation, both with respect to the underlying (dynamics and reward) model, and with respect to the behavior of other agents. As with Bayesian RL and Bayesian learning in games, no *explicit* exploration actions are required. Of course, it is important to realize that this model may not converge to an optimal policy for the true underlying stochastic game. Priors that fail to reflect the true model, or unfortunate samples early on, can easily mislead an agent. But it is precisely this behavior that allows an agent to learn how to behave well without drastic penalty.

3.2 Computational Approximations

Solving the belief state MDP above will generally be computationally infeasible. In specific MARL problems, the generality of such a solution—defining as it does a value for every possible belief state—is not needed anyway. Most belief states are not reachable given a specific initial belief state. A more directed search-based method can be used to solve this MDP for the agent's current belief state b. We consider a form of *myopic EVOI* in which only immediate successor belief states are considered, and their values are estimated without using VOI or lookahead.

Formally, myopic action selection is defined as follows. Given belief state b, the *myopic* Q-function for each $a_i \in A_i$ is:

$$Q_m(a_i, b) = \sum_{a_{-i}} \Pr(a_{-i}|b) \sum_t \Pr(t|a_i \circ a_{-i}, b)$$
$$\sum_{\mathbf{r}} \Pr(\mathbf{r}|a_i \circ a_{-i}, b)[r + \gamma V_m(b(\langle s, a, \mathbf{r}, t\rangle))] \quad (8)$$

$$V_m(b) = \max_{a_i} \int_m \int_{\sigma_{-i}} Q(a_i, s|m, \sigma_{-i}) P_M(m) P_S(\sigma_{-i}) \quad (9)$$

The action performed is that with maximum myopic Q-value. Eq. 8 differs from Eq. 3 in the use of the myopic value function V_m, which is defined as the expected value of the optimal action at the current state, assuming a fixed distribution over models and strategies.[8] Intuitively, this myopic approximation performs one step-lookahead in belief space, then evaluates these successor states by determining the expected value to BA w.r.t. a fixed distribution over models, and a fixed distribution over successor states.

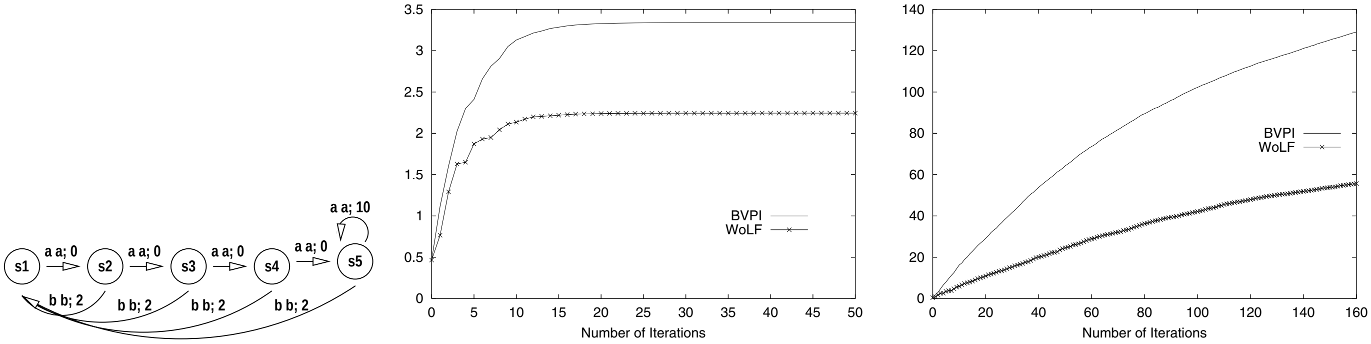
This computation involves the evaluation of a finite number of successor belief states—$A \cdot R \cdot S$ such states, where A is the number of joint actions, R is the number of rewards, and S is the size of the state space (unless b restricts the number of reachable states, plausible strategies, etc.). Greater accuracy can be realized with multistage lookahead, with the requisite increase in computational cost. Conversely, the myopic action can be approximated by sampling successor beliefs (using the induced distributions defined in Eqs. 5, 6, 7) if the branching factor $A \cdot R \cdot S$ is problematic.

The final bottleneck involves the evaluation of the myopic value function $V_m(b)$ over successor belief states. The $Q(a_i, s|m, \sigma_{-i})$ terms are Q-values for standard MDPs, and can be evaluated using standard methods, but direct evaluation of the integral over all models is generally impossible. However, sampling techniques can be used [4]. Specifically, some number of models can be sampled, the corresponding MDPs solved, and the expected Q-values estimated by averaging over the sampled results. Various techniques for making this process more efficient can be used as well, including importance sampling (allowing results from one MDP to be used multiple times by reweighting) and "repair" of the solution for one MDP when solving a related MDP [4].

For certain classes of problems, this evaluation can be performed directly. For instance, suppose a repeated game is being learned, and the BA's strategy model consists of fictitious play beliefs. The immediate expected reward of any action a_i taken by the BA (w.r.t. successor b') is given by its expectation w.r.t. its estimated reward distribution and fictitious play beliefs. The maximizing action a_i^* with highest immediate reward will be the best action at *all* subsequent stages of the repeated game—and has a fixed expected reward $r(a_i^*)$ under the myopic (value) assumption that beliefs are fixed by b'. Thus the long-term value at b' is $r(a_i^*)/(1-\gamma)$.

The approaches above are motivated by approximating the direct myopic solution to the "exploration POMDP." A different approach to this approximation is proposed in [4], which estimates the (myopic) value of obtaining *perfect information* about $Q(a, s)$. Suppose that, given an agent's current belief state, the expected value

[8] We note that $V_m(b)$ only provides a crude measure of the value of belief state b under this fixed uncertainty. Other measures could include the expected value of $V(s|m, \sigma_{-i})$.



(a) Chain World Domain

(b) $\gamma = 0.75$

(c) $\gamma = 0.99$

Figure 2: Chain World Results

of action a is given by $\overline{Q}(a, s)$. Let a_1 be the action with highest expected Q-value and state s and a_2 be the second-highest. We define the *gain* associated with learning that the true value of $Q(a, s)$ (for any a) is in fact q as follows:

$$gain_{s,a}(q) = \begin{cases} \overline{Q}(a_2, s) - q, & \text{if } a = a_1 \text{ and } q < \overline{Q}(a_2, s) \\ q - \overline{Q}(a_1, s), & \text{if } a \neq a_1 \text{ and } q > \overline{Q}(a_1, s) \\ 0, & \text{otherwise} \end{cases}$$

Intuitively, the gain reflects the effect on decision quality of learning the true Q-value of a specific action at state s. In the first two cases, what is learned causes us to change our decision (in the first case, the estimated optimal action is learned to be worse than predicted, and in the second, some other action is learned to be optimal). In the third case, no change in decision at s is induced, so the information has no impact on decision quality.

Adapted to our setting, a computational approximation to this *naive sampling* approach involves the following steps:

(a) a finite set of k models is sampled from the density P_M;

(b) each sampled MDP j is solved (w.r.t. the density P_S over strategies), giving optimal Q-values $Q^j(a_i, s)$ for each a_i in that MDP, and average Q-value $\overline{Q}(a_i, s)$ (over all k MDPs);

(c) for each a_i, compute $gain_{s,a_i}(Q^j(a_i, s))$ for each of the k MDPS; let *EVPI*(a_i, s) be the average of these k values;

(d) define the value of a_i to be $\overline{Q}(a_i, s) +$ *EVPI*(a_i, s) and execute the action with highest value.

This approach can benefit from the use of importance sampling and repair (with minor modifications). Naive sampling can be more computationally effective than one-step lookahead (which requires sampling and solving MDPs from *multiple* belief states). The price paid is approximation inherent in the perfect information assumption: the execution of joint action a does not come close to providing perfect information about $Q(a, s)$.

4. EXPERIMENTAL RESULTS

We have conducted a number of experiments with both repeated and stochastic games to evaluate the Bayesian approach. We focus on two-player identical interest games largely to compare to existing methods for "encouraging" convergence to optimal equilibria. The Bayesian methods examined are one-step lookahead (BOL) and naive sampling for estimating VPI (BVPI) described in Section 3. In all cases, the Bayesian agents use a simple fictitious play model to represent their uncertainty over the other agent's strategy. Thus at each iteration, a BA believes its opponent to play an action with the empirical probability observed in the past (for multistate games, these beliefs are independent at each state).

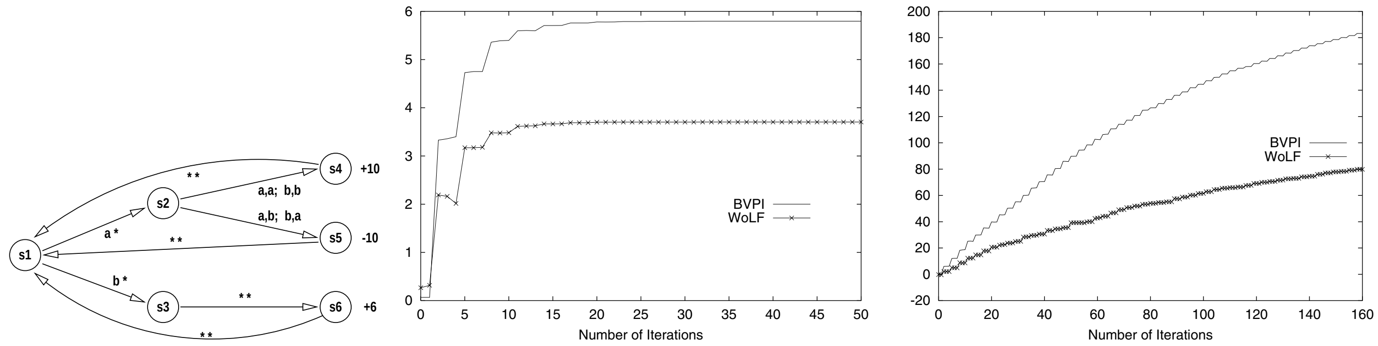
BOL is used only for repeated games, since these allow the immediate computation of expected values over the infinite horizon at the successor belief states: expected reward for each joint action can readily be combined with the BA's fictitious play beliefs to compute the expected value of an action (over the infinite horizon) since the only model uncertainty is in the reward. BVPI is used for both repeated and multi-state games. In all cases, five models are sampled to estimate the VPI of the agent's actions. The strategy priors for the BAs are given by Dirichlet parameters ("prior counts") of 0.1 or 1 for each opponent action. The model priors are similarly uninformative, with each state-action pair given the same prior distribution over reward and transition distributions (except for one experiment as noted).

We first compare our method to several different algorithms on a stochastic version of the penalty game described above. The game is altered so that joint actions provide a stochastic reward, whose mean is the value shown in the game matrix.[9] We compare the Bayesian approach to the following algorithms. KK is an algorithm [10] that biases exploration to encourage convergence to optimal equilibria in just these types of games. Two related heuristic algorithms that also bias exploration optimistically are the Optimistic Boltzmann (OB) and Combined OB (CB) [3] are also evaluated. Unlike KK, these algorithms observe and make predictions about the other agent's play. Finally, we test the more general algorithm WoLF-PHC [2], which works with arbitrary, general-sum stochastic games (and has no special heuristics for equilibrium selection). In each case the game is played by two learning agents of the same type. The parameters of all algorithms were empirically tuned to give good performance.

The first set of experiments tested these agents on the stochastic penalty game with k set to -20 and discount factors of 0.95 and 0.75. Both BOL and BVPI agents use uninformative priors over the set of reward values.[10] Results appear in in Figures 1(a) and (b) showing the total discounted reward accumulated by the learning agents, averaged over 30 trials. Discounted accumulated reward provides a suitable way to measure both the cost being paid in the attempt to coordinate as well as the benefits of coordination (or lack

[9] Each joint action gives rise to X distinct rewards.

[10] Specifically, any *possible* reward (for any joint action) is given equal (expected) probability in the agent's Dirichlet priors.



(a) The Opt In or Out Domain

(b) $\gamma = 0.75$

(c) $\gamma = 0.99$

Figure 3: "Opt In or Out" Results: Low Noise

thereof). The results show that both Bayesian methods perform significantly better than the methods designed to force convergence to an optimal equilibrium. Indeed, OB, CB and KK converge to an optimal equilibrium in virtually all of their 30 runs, but clearly pay a high price.[11] By comparison, when $\gamma = 0.95$, the ratio of convergence to an optimal equilibrium, the nonoptimal equilibrium, or a nonequilibrium is, for BVPI 25/1/4 and for BOL 14/16/0. Surprisingly, WoLF-PHC does better than OB, CB and KK in both tests. In fact, this method outperforms the BVPI agent in the case of $\gamma = 0.75$ (largely because BVPI converges to a nonequilibrium 5 times and the suboptimal equilibrium 4 times). WoLF-PHC converges in each instance to an optimal equilibrium.

We repeated this comparison on a more "difficult" problem with a mean penalty of -100 (and increasing the variance of the reward for each action). To test one of the benefits of the Bayesian perspective we included a version of BOL and BVPI with informative priors (giving it strong information about rewards by restricting the prior to assign (uniform) nonzero expected probability to the small range of truly feasible rewards for each action). Results are shown in Figure 1(c) (averaged over 30 runs). Wolf-PHC and CB converge to the optimal equilibrium each time, while the increased reward stochasticity made it impossible for KK and OB to converge at all. All four methods perform poorly w.r.t. discounted reward. The Bayesian methods perform much better. Not surprisingly, the agents BOL and BVPI with informative priors do better than their "uninformed" counterparts; however, because of the high penalty (despite the high discount factor), they converge to the suboptimal equilibrium most of the time (22 and 23 times, respectively).

We also applied the Bayesian approach to two identical-interest, multi-state, stochastic games. The first is a version of Chain World [4] modified for multiagent coordination, and is illustrated in Figure 2(a). The optimal joint policy is for the agents to do action a at each state, though these actions have no payoff until state s_5 is reached. Coordinating on b leads to an immediate, but smaller, payoff, and resets the process.[12] Unmatched actions $\langle a, b\rangle$ and $\langle b, a\rangle$ result in zero-reward self-transitions (omitted from the diagram for clarity). Transitions are noisy, with a 10% chance that an agent's action has the "effect" of the opposite action. The original Chain World is difficult for standard RL algorithms, and is made especially difficult here by the requirement of coordination.

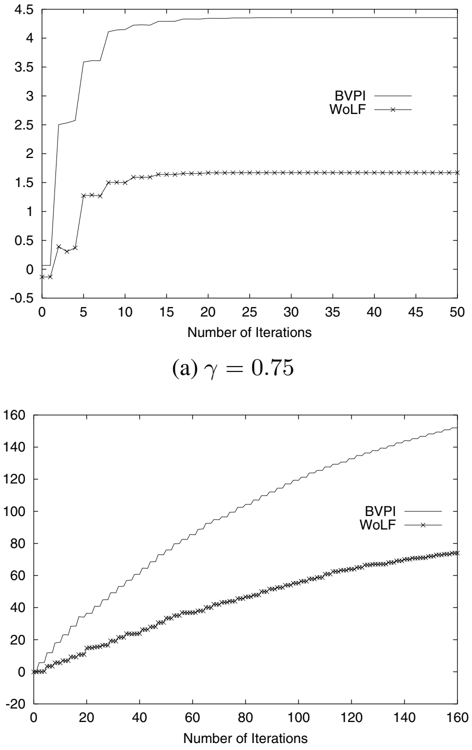
We compared BVPI to WoLF-PHC on this domain using two different discount factors, plotting the total discounted reward (averaged over 30 runs) in Figure 2(b) and (c). Only initial segments are shown, though the results project smoothly to 50000 iterations. BVPI compares favorably to Wolf-PHC in terms of online performance. BVPI converged to the optimal policy in 7 (of 30) runs with $\gamma = 0.99$ and in 3 runs with $\gamma = 0.75$, intuitively reflecting the increased risk aversion due to increased discounting. WoLF-PHC rarely even managed to reach state s_5, though in 2 (of 30) runs with $\gamma = 0.75$ it stumbled across s_5 early enough to converge to the optimal policy. The Bayesian approach manages to encourage intelligent exploration of action space in a way that trades off risks and predicted rewards; and we see increased exploration with the higher discount factor, as expected.

The second multi-state game is "Opt in or Out" shown in Figure 3(a). The transitions are stochastic, with the action selected by an agent having the "effect" of the opposite action with some probability. Two versions of the problem were tested, one with low "noise" (probability 0.05 of an action effect being reversed), and one with "medium" noise level (probability roughly 0.11). With low noise, the optimal policy is as if the domain were deterministic (the first agent opts in at s_1 and both play a coordinated choice at s_2), while with medium noise, the "opt in" policy and the "opt out" policy (where the safe move to s_6 is adopted) have roughly equal value. BVPI is compared to WoLF-PHC under two different discount rates, with low noise results shown in Figure 3(b) and (c), and high noise results in Figure 4(a) and (b). Again BVPI compares favorably to WoLF-PHC, in terms of dicounted reward averaged over 30 runs. In the low noise problem, BVPI converged to the optimal policy in 18 (of 30) runs with $\gamma = 0.99$ and 15 runs with $\gamma = 0.75$. The WoLF agents converged in the optimal policy only once with $\gamma = 0.99$, but 17 times with $\gamma = 0.75$. With medium noise, BVPI chose the "opt in" policy in 10 (0.99) and 13 (0.75) runs, but learned to coordinate at s_2 even in the "opt out" cases. Interestingly, WoLF-PHC always converged on the "opt out" policy (recall both policies are optimal with medium noise).

Finally, we remark that the Bayesian methods do incur greater computational cost per experience than the simpler model-free RL methods we compared to. BOL in particular can be intensive, taking up to 25 times as long to select actions in the repeated game experiments as BVPI. Still, BOL computation time per step is only half a millisecond. BVPI is comparable to all other methods on repeated games. In the stochastic games, BVPI takes roughly 0.15ms to compute action selection, about 8 times as long as WoLF in ChainWorld, and 24 times in Opt In or Out.

[11] All penalty game experiments were run to 500 games, though only the interesting initial segments of the graphs are shown.

[12] As above, rewards are stochastic with means shown in the figure.



(a) $\gamma = 0.75$

(b) $\gamma = 0.99$

Figure 4: "Opt In or Out" Results: Medium Noise

5. CONCLUDING REMARKS

We have described a Bayesian approach to modeling MARL problems that allows agents to explicitly reason about their uncertainty regarding the underlying domain and the strategies of their counterparts. We've provided a formulation of optimal exploration under this model and developed several computational approximations for Bayesian exploration in MARL.

The experimental results presented here demonstrate quite effectively that Bayesian exploration enables agents to make the tradeoffs described. Our results show that this can enhance online performance (reward accumulated while learning) of MARL agents in coordination problems, when compared to heuristic exploration techniques that explicitly try to induce convergence to optimal equilibria. This implies that BAs run the risk of converging on a suboptimal policy; but this risk is taken "willingly" through due consideration of the learning process given the agent's current beliefs about the domain. Still we see that BAs often find optimal strategies in any case. Key to this is a BA's willingness to exploit what it knows before it is very confident in this knowledge—it simply needs to be confident enough to be willing to sacrifice certain alternatives.

While the framework is general, our experiments were confined to identical interest games and fictitious play beliefs as (admittedly simple) opponent models. These results are encouraging but need to be extended in several ways. Empirically, the application of this framework to more general problems is important to verify its utility. More work on computational approximations to estimating VPI or solving the belief state MDP is also needed. Finally, the development of computationally tractable means of representing and reasoning with distributions over strategy models is required.

Acknowledgements

Thanks to Bob Price for very helpful discussion and the use of MARL software. This research was supported by the Natural Sciences and Engineering Research Council.

6. REFERENCES

[1] C. Boutilier. Sequential optimality and coordination in multiagent systems. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pages 478–485, Stockholm, 1999.

[2] M. Bowling and M. Veloso. Rational and convergent learning in stochastic games. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 1021–1026, Seattle, 2001.

[3] C. Claus and C. Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 746–752, Madison, WI, 1998.

[4] R. Dearden, N. Friedman, and D. Andre. Model-based bayesian exploration. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 150–159, Stockholm, 1999.

[5] J. A. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer Verlag, New York, 1996.

[6] Drew Fudenberg and David K. Levine. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1998.

[7] J. Hu and M. P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 242–250, Madison, WI, 1998.

[8] P. Jehiel and D. Samet. Learning to play games in extensive form by valuation. *NAJ Economics, Peer Reviews of Economics Publications*, 3, 2001.

[9] E. Kalai and E. Lehrer. Rational learning leads to Nash equilibrium. *Econometrica*, 61(5):1019–1045, 1993.

[10] S. Kapetanakis and D. Kudenko. Reinforcement learning of coordination in cooperative multi-agent systems. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, pages 326–331, Edmonton, 2002.

[11] M. Lauer and M. Riedmiller. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 535–542, Stanford, CA, 2000.

[12] M. L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the Eleventh International Conference on Machine Learning*, pages 157–163, New Brunswick, NJ, 1994.

[13] M. L. Littman and C. Szepesvári. A generalized reinforcement-learning model: Convergence and applications. In *Proceedings of the Thirteenth International Conference on Machine Learning*, pages 310–318, Bari, Italy, 1996.

[14] J. Robinson. An iterative method of solving a game. *Annals of Mathematics*, 54:296–301, 1951.

[15] L. S. Shapley. Stochastic games. *Proceedings of the National Academy of Sciences*, 39:327–332, 1953.

[16] X. Wang and T. Sandholm. Reinforcement learning to play an optimal nash equilibrium in team markov games. In *Advances in Neural Information Processing Systems 15 (NIPS-2002)*, Vancouver, 2002. to appear.