Analyzing Myopic Approaches for Multi-Agent Communication

Raphen Becker Victor Lesser Shlomo Zilberstein
University of Massachusetts, Amherst
Department of Computer Science
{raphen, lesser, shlomo}@cs.umass.edu

Abstract

Choosing when to communicate is a fundamental problem in multi-agent systems. This problem becomes particularly hard when communication is constrained and each agent has different partial information about the overall situation. Although computing the exact value of communication is intractable, it has been estimated using a standard myopic assumption. However, this assumption–that communication is only possible at the present time–introduces error that can lead to poor agent behavior. We examine specific situations in which the myopic approach performs poorly and demonstrate an alternate approach that relaxes the assumption to improve the performance. The results provide an effective method for value-driven communication policies in multi-agent systems.

1. Introduction

Deciding when to communicate is a fundamental challenge in multi-agent systems. Finding the optimal communication policy is usually intractable in decentralized problems when communication has a cost, ranging from NP-complete to NEXP-complete [7, 11]. This decision can be formulated as a value of information problem. The value of the information collected and disseminated can be measured by the difference between the improvement in the agents' performance and the costs associated with communication, regardless whether communication takes the form of state information, intentions or commitments. The optimal communication policy involves the agents choosing the communicative act at each step that maximizes the expected future utility, much like choosing an optimal action in an MDP.

Information value theory [9] is an important component of decision making, and it has been used to calculate the value of information in different settings, for example the expected value of computation [8]. However, even in the single-agent contexts where information value theory has been extensively used, finding the exact value is very difficult. The typical approach to dealing with this complexity is to approximate it with two myopic assumptions: each source of information is evaluated in isolation and they are evaluated with a 1-step horizon [10].

Others [5, 6, 13] have extended these myopic assumptions to multiple agents in order to generate communication policies. Frequently, however, the exact assumptions being made are not clearly stated. Additionally, a careful analysis of the impact of these assumptions on the quality of the resulting communication policy has not been made. While the myopic assumptions may be an appropriate way to approximate the value of information in the single-agent case, it is not obvious that they remain so for the multi-agent case.

This work attempts to improve the understanding of communication in multi-agent systems by examining the implications of the myopic assumptions. First, we clearly state the basic myopic assumptions and formally show how to compute the optimal communication policy given these assumptions. We then identify and describe two facets of the assumptions that introduce error, and we provide an improved way to compute the communication policy that compensates for this bias.

We perform our analysis of communication using the Transition Independent Decentralized MDP [1] as the underlying multi-agent framework extended to allow communication between the agents. We chose this model for several reasons. First, decision-theoretic models are a formal way of describing a problem and have natural definitions of optimality. Second, in this framework each agent has a different, local view of the world. Their actions are based on their own local views, so an agent does not know the actions of the other agents even if it knows the other agents' policies. Centralized models, like the MMDP [3], are not capable of (naturally) representing this decentralized view. Third, this model also has a known algorithm to find the optimal joint policy assuming zero communication. This guarantees that the results of the analysis are not due to interactions between two heuristics.

This model also isolates the effect communication has on the expected value of a problem by imposing a strict separation between domain-level actions and communicative acts. Most other decision-theoretic, multi-agent models allow domain-level actions to include implicit forms of communication [2, 6, 11], which make analysis difficult. Implicit communication occurs when one agent gains information about another agent's state through a non-communicative act. This communication is often represented within the transition function and is difficult to quantify. For example, a robot attempts to move forward and fails. The failure could be caused by the wheels spinning in place or by another robot sitting in front of it. Therefore, its failure to move forward changes its belief about the location of the other robot.

Many other researchers have studied different aspects of communication. Some have worked with algorithms not based on myopic assumptions, like Reinforcement Learning [4]. The advantage of using RL is that they do not need a complete model of the problem, but they do their learning online and potentially make very bad decisions until they learn better ones. Others have addressed different questions, like **what** the agents should communicate [12] instead of **when**.

Xuan and Lesser [14] have worked toward understanding communication as a way to reduce uncertainty. This work compliments and builds on their understanding of communication by using the value of information as a quantitative measure of the benefit of reducing uncertainty.

2. Problem Description

A myopic strategy for communication is a generic approach that can be applied to any multi-agent domain. We have chosen to illustrate this work using the Transition Independent DEC-MDP [1] extended to include communication.

The model is composed of n cooperative agents. Each agent i works on its own local subproblem that is described by an MDP, $\langle S_i, A_i, P_i, R_i \rangle$. The local subproblem for agent i is completely independent of the local subproblems for the other agents, and completely observable only by agent i. This means that at each step agent i takes action $a_i \in A_i$ and transitions from state $s_i \in S_i$ to $s'_i \in S_i$ with probability $P_i(s'_i|s_i, a_i)$ and receives reward $R_i(s'_i)$. The state of the world is just the collective local states of all of the agents.

At each time step each agent first performs a domain-level action (one that affects its local MDP) and then a communication action. The communication actions are simply *communicate* or *not communicate*. If at least one agent chooses to communicate, then **every** agent broadcasts its local state to every other agent. This synchronizes the world view of the agents, providing each agent complete information about the current world state. The cost of communication is $\mathcal{C}$ if at least one agent initiates it, and it is treated as a negative reward. Goldman and Zilberstein [7] show that for a fixed communication cost no other communication protocol can lead to a higher expected reward.

An optimal joint policy for this problem is composed of a local policy for each agent. Each local policy is a mapping from the current local state $s_i \in S_i$, the last synchronized world state $\langle s_1 ... s_n \rangle \in \langle S_1 ... S_n \rangle$, and the time T since the last synchronization to a domain-level action and a communication action, $\pi_i : S_i \times \langle S_1 ... S_n \rangle \times T \to A_i \times \{yes, no\}$. We will occasionally refer to domain-level policies and communication policies as separate entities, which is just a mapping to A_i and $\{yes, no\}$ respectively.

In addition to the individual agents accruing rewards from their local subproblems, the system also receives reward based on the joint states of the agents. This is captured in the global reward function $R : S_1 \times ... S_n \to \Re$. To the extent that the global reward function depends on past history it must be included in the local states of the agents just like the local rewards. The goal is to find a joint policy $\langle \pi_1 ... \pi_n \rangle$ that maximizes the global value function V, which is the sum of the expected rewards from the local subproblems and the expected reward the system receives from the global reward function.

Definition 1 *The global value function* $V(s_1 ... s_n) =$

$$\sum_{s'_1 ... s'_n} \prod_{i=1}^{n} P_i(s'_i|s_i, a_i) \left[\sum_{i=1}^{n} R_i(s'_i) + R(s'_1 ... s'_n) + V(s'_1 ... s'_n) \right]$$

To summarize, the class of problems we are dealing with can be defined by n MDPs, a global reward function R, and synchronizing communication between the agents with a fixed cost $\mathcal{C}$.

The complexity of the related decision problem to this class is NP-complete [7], which is low for a problem with communication. The key structure in the model that reduces the complexity to NP-complete is the synchronizing communication protocol. When any information is transferred between the agents it is complete information so only the last communication must be remembered. Without this, the agents must remember the entire history of communication to make correct decisions, which results in an exponential increase in the size of the policies and a doubly-exponential increase in the solution time.

2.1. Example Application

We illustrate this class of problems with the following multi-agent data collection example. This example can be viewed as an abstraction of many different types of data

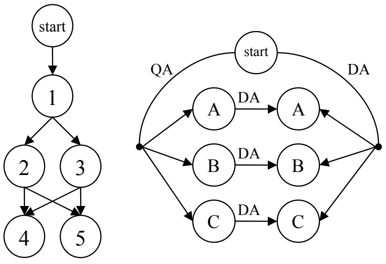

Figure 1. Graphical depiction of an example decision problem. (left) A partially ordered list of 5 sites. (right) A decision problem for one site with three potential classes.

collection problems, though we will present it as a rover exploration problem. Consider n rovers exploring a landscape and collecting data. Each rover has its own partially ordered list of sites it can visit, see Figure 1 (left). Each site contains a particular class of information. This class is not known *a priori*, instead the rover has a distribution over the classes for each site. See Figure 1 (right) for an example decision problem of a site with three classes of information. Each site has a similar decision problem associated with it. For example, the site could be an interesting rock formation. With 70% probability it could be (A) a sedimentary rock, 25% (B) an igneous rock, and 5% (C) a fossil. The value of discovering and collecting data from a fossil may be significantly higher than collecting data from yet another sedimentary rock.

When a rover arrives at a site it has two choices. First, it can gather the information through a Detailed Analysis (DA) without knowing what class of information it is collecting. Alternatively, the rover can perform a Quick Analysis (QA) to determine the class of information available at the site before choosing whether to collect the information. The rover is restricted from collecting information at every site due to limited resources, like time and battery power.

The value of a DA comes from the information collected. The value of a QA is that it consumes fewer resources than a DA and allows the rover to make a more informed decision. The system receives reward based on the total information collected by all of the rovers. Each class of information has a base value. If the information in a particular class is redundant then the total value for collecting that class more than once may be only slightly higher than the base value. Alternatively, a class could be complementary, in which case the value for two pieces of information may be greater than twice the base value. The values of the information are captured in the global reward function.

3. Basic Myopic Approach

Using a myopic algorithm is a common way of dealing with the complexity inherent in finding an optimal solution. We present an algorithm for determining when the agents should communicate. This algorithm is optimal assuming that it must be initiated by the current agent (agent i in the following description) and that the current step is the only time communication is possible. For clarity the equations are presented for two agents i and j, but the approach easily extends to n agents. The complexity results still include all n agents.

While the problems we are solving are distributed in nature (each agent chooses an action based on its own local view) the algorithm we present here computes offline the policies for each agent in a centralized location with a fully specified model of the problem, and the individual policies are given to the agents to follow. This does not trivialize the problem, nor does it reduce it to a single MDP since the solution found is still a decentralized solution. We chose this approach for two reasons. First, individual agents often lack the computational resources necessary to generate high quality solutions. Second, individual agents often lack a global view of the problem, which while not strictly necessary does simplify the solution process and reduces the communication between the agents (which has a cost).

The basic idea is that each agent follows the optimal policy assuming no future communication, which is obtained using the Coverage Set Algorithm (CSA) [1]. At each state, the agent chooses whether to communicate by computing the **Value of Communication** (VoC). If the $\mathrm{VoC} > 0$ then the agent initiates communication causing all of the agents to broadcast their local state. This synchronizes the local views of all of the agents to the world state. The agents then compute a new optimal policy assuming no future communication, using their synchronized world state as the starting state. The domain-level actions the agents take always come from this zero-communication policy.

The VoC from agent i's perspective depends on i's current local state s_i, the previous synchronized world state (or original starting state) $\langle s_i^0, s_j^0 \rangle$, and the time since the last synchronization t. It also implicitly depends on the optimal joint policy assuming zero communication that the agents have been following since the previous synchronization, $\langle \pi_i^0, \pi_j^0 \rangle$.

Definition 2 *The Value of Communication (VoC) is the difference between the expected value when communicating and the expected value for remaining silent.*

$$\mathrm{VoC}\left(s_i, \langle s_i^0, s_j^0 \rangle, t\right) = \sum_{s_j} P(s_j | s_j^0, t, \pi_j^0)\left[V^*(s_i, s_j) - \mathcal{C} - V(s_i, s_j)\right],$$

where $P(s_j | s_j^0, t, \pi_j^0)$ is agent i's belief about agent j's current local state, $V(s_i, s_j)$ is the expected value for *following the current local policy, and $V^*(s_i, s_j) - \mathcal{C}$ is the expected value if the agents communicate now and follow a new zero communication policy after synchronizing.*

The complexity of the VoC depends on the size of the local state space as well as the number of agents.

Theorem 1 *Computing the Value of Communication can be done in time polynomial in the number of local states and exponential in the number of agents.*

Proof. There are four components to computing the VoC that add to the complexity:

- $P(s_j|s_j^0, t, \pi_j^0)$ is the t-step transition function for agent j. Given the assumption that j will never initiate communication,
$$P(s_j|s_j^0, t, \pi_j^0) = \sum_{s'_j} P(s'_j|s_j^0, t-1, \pi_j^0) P(s_j|s_j, \pi_j^0).$$
This takes $O(|S_j|)$ if the values from $t-1$ were cached from a previous call to VoC and $O(|S_j|^2)$ to compute from scratch.

- $V(s_i, s_j)$ and $V^*(s_i, s_j)$ are both expected values (see Definition 1). The only difference is that they assume different domain-level policies. With dynamic programming they can be solved in time polynomial in the number of world states, which is exponential in the number of agents, $O(|S_i|^n)$.

- The difficult part of computing the VoC is finding the new optimal joint policy with no communication for the different possible world states. We observed that the CSA does not need to be run in its entirety each time. Instead, most of the computation can be cached and only the final step of the algorithm must be rerun for each world state. That step involves searching through a small set of policies for each agent for the optimal joint policy. This step takes time exponential in the number of agents.

- When there are $n > 2$ agents the summation in the VoC is over all possible local states of the other agents. The loop, therefore, must be repeated $O(|S_j|^{n-1})$ times. However, it is useful to note that $V^*(s_i, s_j) - V(s_i, s_j) \geq 0$ and therefore the summation can terminate as soon as it becomes greater than $\mathcal{C}$ instead of looping through all possible next states.

The net result is a complexity polynomial in the number of local states for the agents and exponential in the number of agents. □

A final point about the complexity is the number of times VoC must be executed to generate the joint communication policy. While the worst case appears to be quite large, $O(n|S|^{n+2})$, in practice it is not nearly that bad. The reason is that many of the combinations of variables are not reachable. For example, if communication is frequent, then the time since the last communication, t, will remain low. If communication is infrequent then the number of reachable synchronized world states $\langle s_i^0, s_j^0 \rangle$ remains low because the world state is only synchronized through communicating. Additionally, there will be substantial overlap in computation between calls to VoC and caching can greatly reduce the running time in practice.

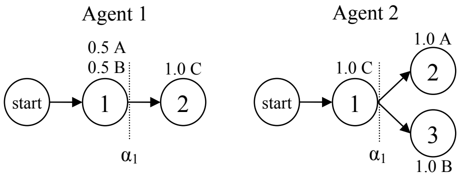

Figure 2. A simple example that illustrates how a simple model for the other agent introduces error.

4. Implications of the Myopic Assumption

The myopic assumption allows a simple, straightforward computation of the value of communication. While this may be a good assumption for the single agent case, there are additional implications that may not be readily apparent in a multi-agent setting. We examine these implications by identifying and analyzing two sources of error in the basic myopic approach, and for each we illustrate it with a simple example.

4.1. Modelling the Other Agents

The *Basic* myopic approach (Definition 2) assumes the simplest of models for the other agents–they never initiate communication. However, since every agent is following a communication policy based on computing the value of communication this is an inaccurate model. The first implication of an accurate model of the other agents is that not communicating itself becomes a form of communication. The distribution of states agent j can be in after t steps, $P(s_j|s_j^0, t, \pi_j^0)$, changes because j is known to not have passed through states in which it would have communicated.

The second implication is that at the current step, agent i may not need to initiate communication to acquire valuable information from agent j if j is already planning to initiate if it has the information. Figure 2 illustrates this with a simple example where agent 1 collects information valuable to

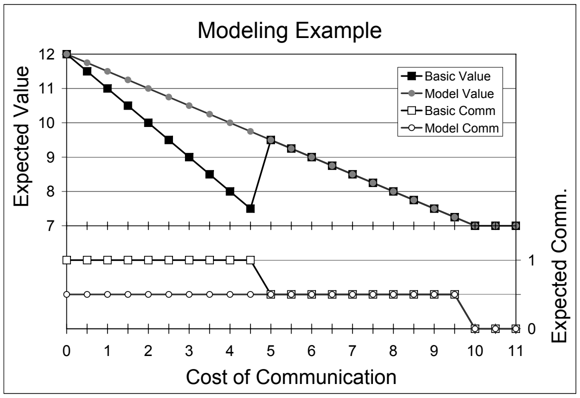

Figure 3. Performance comparison of the *Basic* and *Model* approaches.

		Agent 2				
		s_2^1	s_2^2	s_2^3	π_{1c}	VoC
Agent 1	s_1^1	**-1**	0	-1	*no*	-2
	s_1^2	**4**	**-1**	**-1**	***yes***	**2**
	s_1^3	**-2**	-1	1	*no*	-2
	π_{2c}	***yes***	*no*	*no*		
	VoC	**1**	-2	-1		

Figure 4. A Table M showing the expected gain in value for communicating for each world state.

agent 2. At site 1, agent 1 has an equal chance of collecting an A or a B. If both agents collect A's or B's the system receives reward 10. The system also receives a reward of 1 every time class C is collected. α_1 is the communication point of interest.

The initial zero-communication policy is for agent 2 to collect data from site 2. The only reason to communicate is if agent 1 collects a B, agent 2 needs to change its policy to go to site 3. Based on the initial policy, 50% of the time the agents will receive the maximum reward of 12 and 50% the minimum reward of 2. When agent 1 collects a B, its $\text{VoC} = -\mathcal{C} + 1.0[12-2] = -\mathcal{C} + 10$. As long as the cost $\mathcal{C} < 10$, agent 1 will initiate communication. Agent 2 does not know what agent 1 has collected, so its $\text{VoC} = -\mathcal{C} + 0.5[12-12] + 0.5[12-2] = -\mathcal{C} + 5$. When the cost of communication $\mathcal{C} < 5$ agent 2 will communicate because its $\text{VoC} > 0$. Half of the time this communication is unnecessary because agent 1 had collected an A. When $\mathcal{C} \geq 5$ it is no longer valuable for agent 2 to initiate the communication and their communication policies are optimal.

The *Basic* line in Figure 3 shows the performance of the basic myopic strategy. As the cost of communication increases from 4.5 to 5 it exhibits a jump in value. This undesirable behavior is caused by error introduced into the VoC by not accounting for the other agent's communication policy. This error can be removed from the approximation by computing an optimal **joint** communication policy for each step (assuming no future communication) instead of an optimal **local** communication policy.

To compute the optimal joint communication policy for the current step, the agents must maximize the expected value over all possible world states they could be in. They do this by creating a table M with rows representing the possible states of agent 1 and columns representing states of agent 2 for the current step (see Figure 4).[1] The elements in the table are the value of communicating in that world state weighted by the probability that it is the current world state,

$$M_{xy} = \tag{1}$$

$$P\left(s_1^x | s_1^0, t, \pi_1^0\right) P\left(s_2^y | s_2^0, t, \pi_2^0\right) \left[V^*\left(s_1^x, s_2^y\right) - \mathcal{C} - V\left(s_1^x, s_2^y\right)\right].$$

The *Basic* approach (π_{1c} and π_{2c} in Figure 4) represents building a communication policy for each agent by checking if the sum of a row or column is greater than 0. This strategy double counts certain elements in the table and can result in choosing a communication policy worse than not communicating at all! The expected value of a joint communication policy for one step is the sum of all entries in the table where communication happens (an entry is only counted once, even if both agents initiate communication). In the example table, the *Basic* policy given has a value of -1 (sum of the bold entries) because the valuable state $M_{2,1}$ was counted twice for determining the policies (once for each policy), but only once for determining the value of the table. If agent 2 did not communicate in s_1 then the value would be 2. Never communicating ($\pi_{ic} = \{no, no, no\}$) will always have a value of 0.

The optimal joint communication policy is the joint policy that maximizes the value of this table. Finding the optimal joint policy is exponential in the size of the table, while a simple hill-climbing algorithm can find a Nash equilibrium in polynomial time. The line labeled *Model* in Figure 3 optimizes this table to eliminate the error, resulting in the optimal policy for this example.

Creating the table costs no more than the original approach since each entry represents a reachable world state.

[1] This table does not correspond to the problem in Figures 2 and 3.

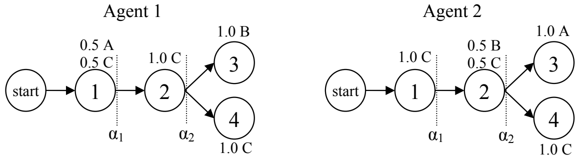

Figure 5. A simple example that illustrates how delaying communication can improve the expected value.

4.2. Myopic View of the Future

The second facet of the myopic assumption is that no agent will communicate in the future. This approximates the true value of communication by introducing error in two ways. The first is due to the greedy nature of the algorithm. When communicating immediately has a positive value, $\mathrm{VoC} > 0$, the agent communicates without considering whether the expected value would be even higher if it waited to communicate until a future step. To compensate, the agents can compute the value of (possibly) communicating after a 1-step delay:

$$\mathrm{VoC}_{delay}\left(s_i, \langle s_i^0, s_j^0\rangle, t\right) = \sum_{s_i'} P(s_i'|s_i, \pi_i^0) \times \max\left(0, \mathrm{VoC}\left(s_i', \langle s_i^0, s_j^0\rangle, t+1\right)\right).$$

The agent will initiate communication when its $\mathrm{VoC} > \mathrm{VoC}_{delay}$. This does not imply that the agent really will initiate communication in the next step because the same comparison will be made at that time to later steps. As long as the expected value for delaying one step is greater than the value of communicating immediately, the agent will delay communication.

Figure 5 illustrates this with a simple example. If agent 1 collects A at site 1 then agent 2 should go to site 3, otherwise agent 2 should go to site 4. Similarly with agent 2 collecting B at site 2. Like the previous example, two A's or two B's have a reward of 10, and each C adds a reward of 1. α_1 and α_2 are the two communication points. The *Basic* approach will always communicate at both α_1 and α_2 for low communication cost (See Figure 6). When the cost increases to 0.5, the agents will only communicate when they have valuable information. Agent 1 will initiate communication 50% of the time at α_1 and agent 2 will 50% of the time at α_2, for a total expected communication of $0.5 + 0.5 = 1.0$. The *Delay* policy, on the other hand, recognizes that waiting a step is beneficial and will only communicate at α_2, which reduces the communication without decreasing the expected reward, yielding a higher expected value.

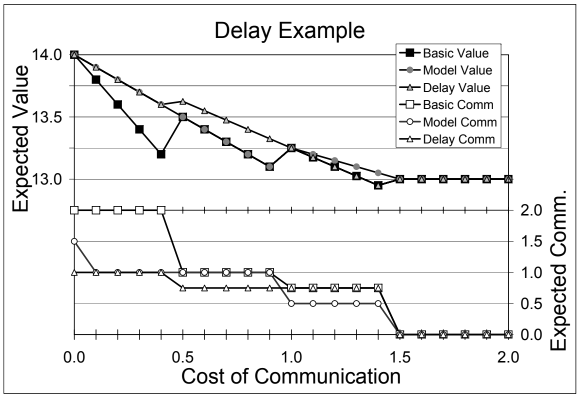

Figure 6. The expected value and expected amount of communication as a function of cost.

When the cost goes above 1, the *Model* approach realizes that it is more efficient to have only one agent initiate communication when it has valuable information. This illustrates that the *Model* and *Delay* approaches address different sources of error and neither dominates the other.

A second source of error in the assumption of no future communication is built in to the policies generated by the CSA. These policies may avoid situations which are valuable only when close coordination is possible. The optimal solution can exploit the possibility of future communication, while the domain-level policies generated here always assume no future communication. When the cost of communication is high enough, this solution is optimal. It is our belief that as the cost decreases, the solutions generated by this approach decline in quality compared to optimal. This source of error can also be partially compensated for by extending the 1-step delay to consider h-steps into the future.

5. Model-Lookahead Approach

This section demonstrates how the *Model* approach of 4.1 and the *Delay* approach of 4.2 can be merged together and extended to consider further into the future. The basic idea is an algorithm that makes optimal communication decisions within a horizon h given fixed domain-level policies based on zero communication.

To start we introduce two new value functions. $V^h(s_i, s_j)$ is the expected value of not communicating in the current step, following an optimal communication policy for the next h steps, and then not communicating again after h steps. $V^{*h}(s_i, s_j) - \mathcal{C}$ is similar but starts with an immediate communication. When the horizon is 0 these value

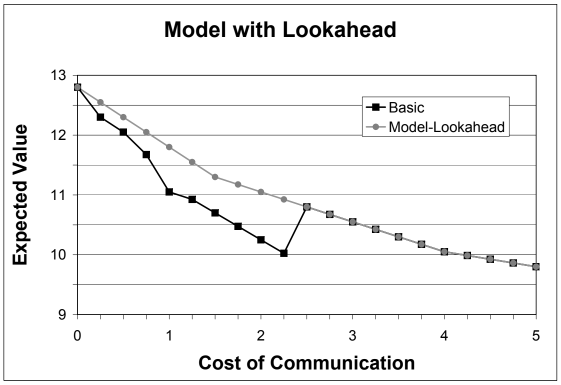

Figure 7. Performance of the *Model-Lookahead* **Approach with horizon 2.**

functions are equivalent to the single-step value functions from Definition 2, $V^0(\cdot) = V(\cdot)$, $V^{*0}(\cdot) = V^*(\cdot)$.

$$V^h(s_i, s_j) = \tag{2}$$

$$\sum_{s'_i, s'_j \in \mathrm{Comm}} P(s'_i|s_i, \pi_i^0) P(s'_j|s_j, \pi_j^0) \left[\mathcal{R}(s'_i, s'_j) + V^{*h-1}(s'_i, s'_j) - \mathcal{C}\right]$$
$$+ \sum_{s'_i, s'_j \in \neg\mathrm{Comm}} P(s'_i|s_i, \pi_i^0) P(s'_j|s_j, \pi_j^0) \left[\mathcal{R}(s'_i, s'_j) + V^{h-1}(s'_i, s'_j)\right]$$

where $\mathcal{R}$ is the sum of the reward functions, $\mathcal{R}(s'_i, s'_j) = R_i(s'_i) + R_j(s'_j) + R(s'_i, s'_j)$. Comm is the set of states in which communication will take place. How it is computed is not clear until we transform the equation:

$$V^h(s_i, s_j) = V(s_i, s_j) \tag{3}$$

$$+ \sum_{s'_i, s'_j \in \mathrm{Comm}} P(s'_i|s_i, \pi_i) P(s'_j|s_j, \pi_j) \left[V^{*h-1}(s'_i, s'_j) - \mathcal{C} - V^{h-1}(s'_i, s'_j)\right]$$
$$+ \sum_{s'_i, s'_j} P(s'_i|s_i, \pi_i) P(s'_j|s_j, \pi_j) \left[V^{h-1}(s'_i, s'_j) - V(s'_i, s'_j)\right]$$

The agents must find the set of communication states for the next step that maximizes $V^h(s_i, s_j)$. The next step communication policy only affects the second line of Equation (3), which bears a remarkable similarity to Equation (1), except that this is a recursive function. The same table algorithm can be applied to generate optimal communication policies over the horizon.

Figure 7 illustrates the performance of this approach on a larger problem. The two agents each had a local decision problem with 6 steps and more than 10,000 states. The *Model-Lookahead* approach performs significantly better than the original *Basic* approach and demonstrates a smooth reduction of the expected value as the cost for communication increases.

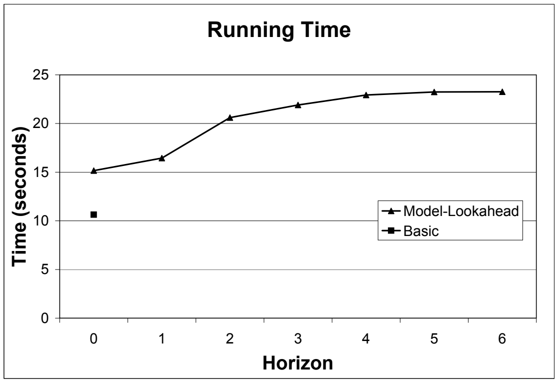

Figure 8. Comparison of the time to compute the policy for the *Basic* **approach versus the** *Model-Lookahead* **approach of various depths.**

Figure 8 shows the running time of *Model-Lookahead* compared to *Basic*. The *Basic* approach took about 11 seconds to generate the entire policy while *Model-Lookahead* took 50% longer with a horizon of 0 due to the added cost of finding the optimal communication policies of the tables. The worst case complexity of *Model-Lookahead* is exponential in the size of the horizon, but due to caching and the structure of the problem, in practice this is not always the case. In this example, the running time started out with an exponential curve but that changed as the horizon approached the number of steps in the problem.

This approach does have its limitations. Even when the horizon is equal to the number of steps in the decision problem, the policy generated is not the optimal joint policy. This is because the domain-level actions taken by the agents are generated assuming no future communication. This is effectively a horizon of 0 for choosing domain-level actions. Future work will include extending this algorithm to a larger domain-level action horizon.

6. Conclusion

This paper addresses the problem of choosing when to communicate in a multi-agent system. We formulate a condition for communication based on the value of information. The standard assumption used to efficiently generate communication policies is that communication is only possible at the present time. This is based on the myopic assumption from information value theory.

We show how to generate optimal joint policies under the myopic communication assumption. We also examine the implications of the assumption and show that it can lead to poor agent behavior. We identify two sources of error and provide modifications to the original algorithm to address these problems. Together, these modifications result in an improved algorithm for generating a decentralized joint policy. Moreover, the computational overhead of our modifications is small compared to the original algorithm.

While the sources of error that we identify and the general approach to addressing them are common to many multi-agent systems, the equations and specific algorithms we present do rely on certain structure being present in the problem. The key structure in the model that reduces the complexity to NP-complete is the synchronizing communication protocol. Without this, the agents must remember the entire history of communication to make correct decisions, which results in an exponential increase in the size of the policies and a doubly-exponential increase in the solution time.

There are two components that together allow the use of synchronizing communication as an exact model. First is the fixed cost of communication. If the agents can send partial state information at a reduced cost then the optimal solution may include communication that does not synchronize the agents' view of the world. Second is the transition and observation independence between the domain-level actions. If the agents are able to take domain-level actions that affect the observations or transitions of another agent then the agents have a form of implicit communication and must memorize the history to make correct decisions.

If a problem does not have synchronizing communication it can be added and the algorithm presented here can be used as an approximation. We also hope that identifying the sources of error common to many myopic approaches and the general approach we took to address them will help others design better communication algorithms.

References

[1] R. Becker, S. Zilberstein, V. Lesser, and C. V. Goldman. Solving transition independent decentralized mdps. *Journal of Artificial Intelligence Research*, 22:423–455, 2004.

[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, November 2002.

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

[4] M. Ghavamzadeh and S. Mahadevan. Learning to communicate and act in cooperative multiagent systems using hierarchical reinforcement learning. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 1114–1121, New York, July 2004. ACM Press.

[5] P. J. Gmytrasiewicz and E. H. Durfee. Rational communication in multi-agent environments. *Autonomous Agents and Multi Agent Systems Journal*, 4(3):233–272, 2001.

[6] C. V. Goldman and S. Zilberstein. Optimizing information exchange in cooperative multi-agent systems. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 137–144, Melbourne, Australia, July 2003. ACM Press.

[7] C. V. Goldman and S. Zilberstein. Decentralized control of cooperative systems: Categorization and complexity analysis. *Journal of Artificial Intelligence Research*, 22:143–174, 2004.

[8] E. J. Horvitz. Reasoning under varying and uncertain resource constraints. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 111–116, Minneapolis, MN, August 1988. Morgan Kaufmann.

[9] R. A. Howard. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, SSC–2(1):22–26, 1966.

[10] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, second edition, 1988.

[11] D. V. Pynadath and M. Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of Artificial Intelligence Research*, 16:389–423, 2002.

[12] J. Shen, V. Lesser, and N. Carver. Minimizing communication cost in a distributed Bayesian network using a decentralized MDP. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 678–685, Melbourne, Australia, July 2003. ACM Press.

[13] M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:83–124, 1997.

[14] P. Xuan and V. Lesser. Multi-agent policies: From centralized ones to decentralized ones. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 1098–1105. ACM Press, 2002.