Learning to Communicate in Decentralized Systems

M. Allen¹  C.V. Goldman²  S. Zilberstein¹

¹Department of Computer Science
University of Massachusetts, Amherst

²Caesarea Rothschild Institute
University of Haifa, Israel

Workshop on Multiagent Learning
Twentieth National Conference on Artificial Intelligence
(AAAI-05)
Outline

Framing the Problem
- Multiagent Communication
- Decentralized MDPs with Communication
- Communication and Translation

Solving the Problem: Approaches and Results
- The Bayesian Approach
- Reduction to Multi-Agent MDPs
- Solving Suitable Problems
- Empirical Results
Outline

Framing the Problem
  Multiagent Communication
  Decentralized MDPs with Communication
  Communication and Translation

Solving the Problem: Approaches and Results
  The Bayesian Approach
  Reduction to Multi-Agent MDPs
  Solving Suitable Problems
  Empirical Results
Outline

Framing the Problem
- Multiagent Communication
- Decentralized MDPs with Communication
- Communication and Translation

Solving the Problem: Approaches and Results
- The Bayesian Approach
- Reduction to Multi-Agent MDPs
- Solving Suitable Problems
- Empirical Results
Motivation

- Decentralized decision-making is a provably hard problem, but one that arises in many common contexts.
- Automated systems that can communicate between components can radically simplify such tasks.
- Systems may need to cooperate in situations where common communication is not initially available.
- Robust systems need to be able to recover from specification errors or unforeseen situations, by learning new understandings of what each other communicate.
An Overview of the Problem

Decision-making in decentralized and multiagent systems depends upon a number of factors:

- How system-states affect what agents observe.
- How agent’s actions affect system-states.
- The information that agents share by communicating.
- Agent’s beliefs about their environment.
- Agent’s beliefs about the meanings of communications.
Framing the Problem
Solving the Problem: Approaches and Results
Summary
Multiagent Communication
Decentralized MDPs with Communication
Communication and Translation

Policy of Action

Policy of Communication

Belief-state update

Learning to Communicate in Decentralized Systems
The Dec-MDP-Com Model

An $n$-agent decentralized MDP with direct communication,

$$M = \langle S, A, P, R, \Sigma, C_\Sigma, \Omega, O, T \rangle.$$ 

- $S$ is a finite set of states, with initial state $s^0$.
- $A = \{A_i \mid A_i$ is a finite set of actions $a_i$ for agent $\alpha_i\}$.
- $P$ is a transition function for state-pairs and joint actions.
- $R$ is a global reward function for state-action transitions.
- $\Sigma = \{\Sigma_i \mid \Sigma_i$ is a finite set of messages $\sigma_i$ for $\alpha_i\}$.
- $C_\Sigma$ gives the cost of each transmitted message.
- $\Omega = \{\Omega_i \mid \Omega_i$ is a finite set of observations $o_i$ for $\alpha_i\}$.
- $O$ is an observation function for agents, observations and state-action transitions.
- $T$ is the time-horizon (finite or infinite) of the problem.
An Important Constraint

- Dec-MDP-Coms (unlike Dec-POMDP-Coms) restrict the observation function $O$, so they are jointly fully-observable.
- Agents’ combined observations fix the global state.
- That is, if the observation-probability

$$O(o_1, \ldots, o_n \mid s, a_1, \ldots, a_n, s') \neq 0$$

then we can fix state $s'$ determinately, based upon the observations $(o_1, \ldots, o_n)$.
- Each agent alone does not observe the entire state; rather, the observations of all agents taken together determine it.
An Example: The Pumps Domain

- 2 agents, each controlling $n$ pumps and $m$ flow-valves.
- Each agent separately observes fluid entering along different inflow ducts, along with its own pumps and valves.
- **Task**: maximize flow out of the system, minimizing outflow ducts used.
- **Probabilistic effects**: pumps and valves susceptible to variations in throughput.
The Pumps Domain as Dec-MDP-Com

\[ M = \langle S, A, P, R, \Sigma, C_\Sigma, \Omega, O, T \rangle, \]

has elements as follows:

- **S**: the state-set is described by flow through the two inflow ducts, two sets of pumps, and two sets of valves.
- **A**: each agent chooses an action to control the pumps \( p_r^i \) (on, off, forward, back) or valves (open, shut).
- **P**: the transition function directs flow according to actions taken; pumps and valves may fail to respond probabilistically.
- **R**: the total reward is proportional to outflow relative to inflow, discounted by number of ducts used.
The Pumps Domain as Dec-MDP-Com (II)

- $\Sigma$: each agent possesses messages corresponding to each possible action, each pump/valve in the system, and observed units of inflow.
- $C_\Sigma$: the cost of each message is zero (0).
- $\Omega, O$: each agent observes just its own inflow duct, and all pumps and valves it controls.
- $T$: the problem has an infinite time-horizon.
Learning to Communicate

- Where agents understand one another, optimal linguistic action involves deciding what and when to communicate, based on cost-benefit analysis.
- Where agents do not fully understand one another, simple message-passing is not enough.
- Agents need to learn how to respond to messages.
- A limited, but interesting question of what messages mean.
Translations

We represent the degree to which agent $\alpha_i$ understands agent $\alpha_j$ by way of a probabilistic correspondence between messages sent by $\alpha_j$, and those that $\alpha_i$ might itself send.

**Definition (Translation)**

Let $\Sigma$ and $\Sigma'$ be sets of messages. A translation, $\tau$, between $\Sigma$ and $\Sigma'$ is a probability function over message-pairs: for any messages $\sigma, \sigma'$, $\tau(\sigma, \sigma')$ is the probability that $\sigma$ and $\sigma'$ mean the same.

$\tau^+_{\Sigma, \Sigma'}$ is the set of all translations between $\Sigma$ and $\Sigma'$. 
Belief-States

Agents need to consider multiple possible translations between messages, and possess beliefs regarding which translation might be correct.

**Definition (Belief-state)**
Let agents $\alpha_1$ and $\alpha_2$ utilize message-sets $\Sigma_1$ and $\Sigma_2$, respectively. A belief-state for agent $\alpha_i$ is a probability-function $\beta_i$ over the set of translations $\tau^+_{\Sigma_i, \Sigma_j}$ ($i \neq j$). That is, for any translation $\tau$ between $\Sigma_i$ and $\Sigma_j$, $\beta_i(\tau)$ is the probability that $\tau$ is correct.
Belief-States (II)

We can thus talk about an agent’s belief about meanings.

- Translations ($\tau$): distributions over message-pairs.
- Belief-states ($\beta_i$): distributions over translations.
- Given any pair of messages, agent $\alpha_i$ assigns that pair a likelihood, $\beta_i^+$, of having the same meaning:

$$\beta_i^+(\sigma_i, \sigma_j) = \sum_{\tau \in \tau_{\Sigma_i, \Sigma_j}^+} \beta_i(\tau) \cdot \tau(\sigma_i, \sigma_j).$$

- Learning to communicate is the process of updating belief-states about translations.
Outline

Framing the Problem
Multiagent Communication
Decentralized MDPs with Communication
Communication and Translation

Solving the Problem: Approaches and Results
The Bayesian Approach
Reduction to Multi-Agent MDPs
Solving Suitable Problems
Empirical Results
Bayesian Updates for Communication Learning

Agents base their translations upon the outcomes of actions taken in response to messages.

- Actions are chosen in response to local observations, messages received, and current belief-states.
- In the **predictive phase**, agents generate possible next belief-states, based upon their current beliefs, and the action they have chosen.
- The actions chosen by all agents together lead to state-transitions, and new observations.
- In a **retrospective phase**, following action, agents adjust belief-states in response to observed outcomes.
Special Properties of Dec-MDP-Coms

Under certain conditions, Dec-MDP-Coms can be reduced to somewhat simpler problems.

**Definition (Fully-describable)**
A Dec-MDP-Com is fully-describable if each agent possesses a language sufficient to communicate both: (a) any observation it makes, and (b) any action it takes.

**Definition (Freely-describable)**
A Dec-MDP-Com is freely-describable if the cost of communicating any message \( \sigma \) is 0.
Reduction to Multi-Agent MDPs

Claim (Reduction)
A Dec-MDP-Com is equivalent to a Multi-agent MDP if (a) it is fully- and freely-describable; and (b) all agents share a common language.

▶ Such a Dec-MDP-Com is reducible to an MDP in which each agent can fully observe the entire state-space.
▶ Simpler, with polynomial-time solution algorithms.
▶ Can be solved off-line.
▶ Still the problem of learning the common language.
Suitable Problem Instances

In order for communication-learning to be possible, agents need to be able to update belief-states meaningfully.

In a **suitable** Dec-MDP-Com, we have:

- Free and full communication.
- Agents communicate observations and intended actions.
- Each translator can choose a most likely interpretation of any received message, such that:
  - If incorrect, then after taking action and observing the outcome, the probability assigned to correct interpretation is strictly greater than one originally chosen.
- An example: meanings observed after some time delay.
Suitable Problem Instances

In order for communication-learning to be possible, agents need to be able to update belief-states meaningfully.

In a **suitable** Dec-MDP-Com, we have:

- Free and full communication.
- Agents communicate **observations** and intended **actions**.
- Each translator can choose a **most likely interpretation** of any received message, such that:
  - If **incorrect**, then after taking action and observing the outcome, the probability assigned to **correct** interpretation is **strictly greater** than one originally chosen.
  - An example: meanings observed after some **time delay**.
Suitable Problem Instances

In order for communication-learning to be possible, agents need to be able to update belief-states meaningfully.

In a suitable Dec-MDP-Com, we have:

- Free and full communication.
- Agents communicate observations and intended actions.
- Each translator can choose a most likely interpretation of any received message, such that:
  - If incorrect, then after taking action and observing the outcome, the probability assigned to correct interpretation is strictly greater than one originally chosen.
- An example: meanings observed after some time delay.
The Elementary Action Protocol

Agents can learn to communicate in suitable problem instances by way of the following protocol.

1. All agents share their observations.
2. Each agent calculates most likely observation-sequence, based on observations, messages, and current translations, and thus a most likely state.
3. Proceeding in turn, each agent chooses an action by:
   3.1 Calculating the most likely action sub-sequence, consisting of actions that might be taken by prior agents in the order.
   3.2 Choosing an action that maximizes value.
   3.3 Communicating this action to the others.
4. Agents take action after all agents complete the prior step.
5. Belief-states are adjusted based on next state-transition.
The Elementary Action Protocol (II)

- Such a straightforward procedure is not necessarily optimal, nor even necessarily close to optimal.
- Computing optimal policies for legitimately decentralized problems is generally intractable.
- Agents who follow the protocol can eventually come to communicate clearly, and so act properly.

Claim (Convergence)

Given an infinite time-horizon, agents acting according to the elementary action protocol in a suitable Dec-MDP-Com will eventually converge upon a joint policy that is optimal for the states they encounter from then on.
The Elementary Action Protocol (II)

- Such a straightforward procedure is not necessarily optimal, nor even necessarily close to optimal.
- Computing optimal policies for legitimately decentralized problems is generally intractable.
- Agents who follow the protocol can eventually come to communicate clearly, and so act properly.

Claim (Convergence)

Given an infinite time-horizon, agents acting according to the elementary action protocol in a suitable Dec-MDP-Com will eventually converge upon a joint policy that is optimal for the states they encounter from then on.
Empirical Results

We tested our methods on a variety of instances of the pump-world problem domain already described.

- Varied numbers of pumps and valves, and thus size of associated vocabularies.
- Agents proceeded until each had full confidence in translation of other agent.
- Results showed convergence to shared language proceeds quite steadily, with time an apparently polynomial function of problem/language complexity.
- Rate of reward-accumulation grows very quickly to near maximum, suggesting possibility of approximation.
Language and Total Reward Over Time

![Graph showing Language and Total Reward Over Time](image)

- % of Reward Earned
- % of Language Learned

Pct. Learning Progress vs. Time Steps in Learning Process

- % of Language Learned
- % of Reward Earned

(28/33) Allen, Goldman, Zilberstein
Learning to Communicate in Decentralized Systems
Rate of Reward Accumulation Over Time

![Graph showing the rate of reward accumulation over time.](#)
Learning to communicate allows complex decentralized problems to be solved more efficiently.

This learning problem is itself highly complicated.

We are interested in further exploration of:

- Properties of decentralized problems making learning easier (or even possible).
- Effective techniques for updating beliefs about communication.
- Approximate techniques and their relation to communication.

For Further Reading II

Goldman, C. V. and Zilberstein, S.
Optimizing information exchange in cooperative multi-agent systems.

Goldman, C. V., and Zilberstein, S.
Decentralized control of cooperative systems: Categorization and complexity analysis.
Goldman, C. V.; Allen, M.; and Zilberstein, S.
Decentralized language learning through acting.