

# Population and Agent Based Models for Language Convergence \*

**Kiran Lakkaraju** †

Dept. of Computer Science,  
University of Illinois, Urbana-Champaign,  
Urbana IL, 61801  
klakkara@uiuc.edu

**Les Gasser**

Graduate School of Library and Information Science, and  
Dept. of Computer Science,  
University of Illinois, Urbana-Champaign,  
Urbana IL, 61801,  
gasser@uiuc.edu

## Introduction

Adaptive agents cooperating in rich, open environments will need shared ontologies and linguistic conventions to communicate critical information. In the face of open systems and environmental change, pre-defined ontologies are sub-optimal. In open systems, new agents or web services can require new conventions not captured in existing conceptualizations. A more adaptive and more theoretically interesting approach is to have agents negotiate repertoires of categories, meanings, and linguistic forms among themselves. Many related issues arise in the general arena of “emergent semantics”; the case of language serves as an excellent “model organism” for studying the issues, and results generalize nicely to other domains including “collaborative tagging” and bioinformatics.

One central problem in emergent, adaptive language is getting a population of agents to converge on shared meanings and forms—this is called the “Language Convergence” problem. The problem can be framed as follows. Suppose we have a population of  $N$  agents, each of which is using their own language. The population arrives at a converged state when all the agents are speaking the same language. When this state is achieved, the agents can communicate with each other.

There are many different ideas on what a language is, but for our purposes we will use a simple definition. We assume that the lexicon (the set of possible words) is fixed, and that a language is a mapping from meanings (objects in the world to words). This is a common abstraction of language. When the mapping is represented as a matrix, it is often called an *association matrix*

## Two Approaches to the Language Convergence Problem

Two approaches have been studied to solve this problem. The population-based approach studies the dynamics regulating the change in the number of agents using a particular

language. The Language Dynamical Equation (LDE) (Komarova, Niyogi, & Nowak 2001) is an example of this type of approach. In this model the dynamics of the population language densities — the number of people using a particular language — is studied. The LDE relates the change in language density of a language  $i$  ( $x_i$ ) to the fitness ( $f_i$ ) of the language.

This model has been well-studied, and results are known for a restricted version of the LDE. (Mitchener 2003).

The population-based approach abstracts away from individual agents and studies the problem from a collective perspective; only the dynamics of groups are taken into account. This abstraction makes the problem easier to study, but ignores several important factors.

In particular, work based on the LDE typically assumes agents can interact with each other uniformly. This may be unrealistic, as many social networks have been demonstrated to be “small-world” networks (Watts & Strogatz 1998). In addition, LDE models assume that agents are changing their languages via a mutating reproduction processes (captured as imperfect parent-child learning). In many cases we do not have this kind of language change. The LDE does not allow us to model an individual agent changing its language; it is a probabilistic model that describes how population language densities change across multiple generations depending on general distributions of learning fidelity in the population.

Parts of Multi-Agent Systems research have addressed some of the problems above. Specifically, MAS research has attempted to scrutinize the actions of each agent in detail. An example of this type of approach is found in “Language Games”. Assume a population of agents, each with its own form-meaning association matrix, randomly moving in an environment filled with objects. When two agents meet, they play a language game: A teacher points to an object in the environment and the learner changes its association matrix to match that of the teacher. (Steels 1998) shows that in situations similar to this convergence can be reached, but provides no proofs or tractability results. Others have modeled communication decisions using Dec-POMDPs, showing in detail how specific knowledge of individual agent decisions can lead to robust collective understandings (Goldman & Zilberstein 2004), but at a tremendous cost: Dec-POMDPs have been shown to be NEXP-Hard (Bernstein *et al.* 2002)

\*Supplementary Information:

<http://big.cs.uiuc.edu/klakkara/aaai-sa-extended.pdf>

†Student and corresponding author

Copyright © 2006, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

## Mapping the language convergence problem to a Dec-POMDP

With the increase in detail about agents internal decision making and control comes a decrease in the tractability of finding an optimal solution. We have mapped the language convergence problem to an instance of the Dec-POMDP problem, allowing each agent to reason about possible language changes in the population given the current population's linguistic states. This is a "jointly fully observable" situation, meaning that though the complete linguistic state of the population is not known by any individual agent, the union of all agent's knowledge together can define the global state. Drawing on Bernstein et al.'s result above and our Dec-POMDP modelization of the language convergence problem, we can show that this level of detail makes it NEXP-Hard to find the optimal sequence of actions for each agent that will result in convergence in the fastest time.

A Dec-POMDP is a POMDP with decentralized control. There are  $n$  agents that independently observe the state vector. Each agent executes an action; the set of actions executed by the agents determines the next state.

We mapped the language convergence problem to a Dec-POMDP by creating a language space. Suppose there are a finite set of languages, denoted by  $L$ . Then the language space is the space of all possible combinations of languages that the population of agents can have:  $L^n$ . Each agent has a language and each agent can execute an action to change its language. We set a positive reward for all states in  $L^n$  where the population has converged. All other states generate negative rewards.

The actions that an agent can execute involve changing its language. This could be a "language game" type of interaction as in (Steels 1998). There are other types of language change actions though. In (Cucker, Smale, & Zhou 2004) each agent can sample word-meaning mappings from a subset of the other agents in the population. The agent will then change its language to the one closest to matching the sampled set of sentences. The crucial element to these actions is that they depend upon the state of the other agents in the population. This means that the Dec-POMDP will not have *Transition Independence* (Goldman & Zilberstein 2004).

A policy for an agent is a mapping from observations to actions. An optimal policy is one which will generate the greatest reward. An optimal policy corresponds to a population choosing actions that change their language so that they can converge in the quickest number of steps possible. Finding an optimal policy for a set of agents is NEXP-Complete. There are different cases that lead to complexity ranging from P-Complete to NEXP-Complete.

The difficulty in solving a Dec-POMDP stems from the uncertainty an agent has about its state and the state of the other agents. Situations in which these uncertainties are resolved make finding the policy computationally less expensive. We have found that the complexity can range from P-Complete to NEXP-Complete by mapping various situations with differing levels of uncertainty to instances of Dec-POMDPs.

For instance, in a situation where there is only a single language to converge upon, finding an optimal policy is only P-Complete. In this case, each agent does not need to know the state of the other agents - it can assume that the other agents will at some point arrive at the same language that it is heading towards.

On the other hand, in the NEXP-Complete case, there are multiple languages that the population can converge upon. In addition each agent cannot determine what languages the other agents have. In this case finding a solution is NEXP-Complete.

## A Population and Agent Based Model

Because of results like this, oftentimes simulation is the only way to explore these complex situations effectively, and parameters must be fine-tuned empirically to find workable values for specific domains. LDE models give up agent-level decision detail and exploit probabilistic abstractions to gain tractability and generality.

We are investigating an approach that incorporates the best of both worlds, with the generalizability, theoretical power, and tractability of LDE-like models, but also with the fine-grained control that MAS models can give us. We approach this problem by trying to add elements of agents to the LDE. To do this, we are adding a Q-function to the LDE model that returns the probability of an agent learning language  $k$  when it is using language  $j$ , and being taught language  $i$ . In addition, we are converting the generational, probabilistic LDE model to a model of language change based on specific individual agents changing their languages via learning, without generational changes. In the aggregate, individual learning still modifies language density distributions, but it makes the mechanisms clearer.

## References

- Bernstein, D. S.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of markov decision processes. *Math. Oper. Res.* 27(4):819–840.
- Cucker, F.; Smale, S.; and Zhou, D.-X. 2004. Modeling language evolution. *Foundations of Computational Mathematics* 4(3):315–343.
- Goldman, C., and Zilberstein, S. 2004. Decentralized control of cooperative systems: Categorization and complexity analysis. *Journal of Artificial Intelligence Research* 22:143–174.
- Komarova, N. L.; Niyogi, P.; and Nowak, M. A. 2001. The evolutionary dynamics of grammar acquisition. *Journal of Theoretical Biology* 209(1):43–59.
- Mitchener, W. G. 2003. Bifurcation analysis of the fully symmetric language dynamical equation. *Journal of Mathematical Biology* 46(3):265–285.
- Steels, L. 1998. The origins of ontologies and communication conventions in multi-agent systems. *Autonomous Agents and Multi-Agent Systems* 1(2):169–194.
- Watts, D. J., and Strogatz, S. H. 1998. Collective dynamics of 'small-world' networks. *Nature* 393(6684):440–442.