

# Modeling Uncertain Domains with Polyagents

H. Van Dyke Parunak, Sven Brueckner

Altarum Institute  
3520 Green Court  
Suite 300

Ann Arbor, MI 48105 USA  
+1 734 302 4600

{van.parunak, sven.brueckner}@altarum.org

## ABSTRACT

Agents are a powerful tool for systems modeling. Instantiating each domain entity with an agent captures many aspects of system dynamics and interactions that other modeling techniques do not. However, an entity's agent can execute only one trajectory per run, and so does not capture the alternative trajectories accessible in the evolution of any realistic system. Averaging over multiple runs does not capture the range of individual interactions. We address these problems with a new modeling construct, the *polyagent*, which represents each entity with a single persistent *avatar* supported by a swarm of transient *ghosts*. Each ghost interacts with the ghosts of other avatars through digital pheromone fields, capturing a wide range of alternative trajectories in a single run of the system that can proceed faster than real time.

## Categories and Subject Descriptors

I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence—*multiagent systems*.

## General Terms

Algorithms, Measurement, Experimentation.

## Keywords

Multiple futures, uncertainty, BDI, swarm intelligence, digital pheromones, dynamics.

## 1. INTRODUCTION

The fundamental entity in an agent-based model is the agent, corresponding to a discrete entity in the domain being modeled, and the fundamental operator is the interaction among agent behaviors. The fundamental entity in an equation-based model [9] is some system observable, and the fundamental operator is the evolution of that observable. Agent-based models often map more naturally to the structure of a problem than do equation-based models, have representations that are more naturally constructed and explored, and provide more realistic results [8], but have a shortcoming. The observables in an equation-based model are often averages across many agents, and so implicitly capture the range of agent variation (although at an aggregate level). The

software agent representing an entity in an agent-based model can execute only one trajectory in each run of the system, and so does not capture the alternative trajectories accessible to the entity in the evolution of any realistic system. Good modeling practice calls for averaging over multiple runs, but this approach still does not capture the range of individual interactions involved.

A new construct, the *polyagent*, represents each entity with a single persistent *avatar* and a swarm of transient *ghosts*. Each ghost interacts with ghosts of other avatars through digital pheromone fields, capturing a wide range of alternative trajectories in a single run that can proceed faster than real time for many reasonable domains. We have used this approach in a number of systems that we have constructed over the years. The purpose of this paper is to articulate the polyagent as a explicit modeling construct and provide some guidance concerning its use.

Section 2 reviews the challenge of adequate sampling in multi-agent modeling of complex domains. Section 3 proposes a new modeling construct, the polyagent, as an answer to this challenge. Section 4 discusses systems we have constructed that can be understood as instances of polyagents. Section 5 concludes.

## 2. THE CHALLENGE OF MODELING MULTI-AGENT INTERACTIONS

Imagine  $n + 1$  entities in discrete time. At each step, each entity interacts with one of the other  $n$  entities. Thus at time  $t$  its interaction history  $h(t) \in n^t$ . Its behavior is a function of  $h(t)$ . This toy model generalizes many interesting domains, including predator-prey systems, combat models, innovation and diffusion of ideas, and disease propagation.

An equation-based model typically reasons at the level of aggregate or average observables across the agent population (e.g., in the predator-prey example, predator population, prey population, average predator energy level, or average prey energy level, all as functions of time). No attempt is made to model the trajectory of an individual entity.

An agent-based model must explicitly describe the trajectory of each agent. In our general model, during a run of length  $\tau$ , each entity will experience one out of  $n^\tau$  possible histories. The population of  $n + 1$  entities will sample  $n + 1$  of these possible histories. The length of a run is often orders of magnitude larger than the number of modeled entities ( $\tau \gg n$ ).

Good practice recommends that such models be run many times, with different random seeds, to explore the range of possible outcomes. Each such run only samples one set of possible interactions. For large populations and scenarios that permit multiple agent interactions, the number of runs needed to sample the pos-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.  
Copyright 2006 ACM 1-59593-303-4/06/0005 ...\$5.00.

sible alternative interactions thoroughly can quickly become prohibitive, and the portion of the space of possible entity histories actually sampled by a single run is vanishingly small. We would need a number of runs on the order of  $\tau$  to generate a meaningful sample. Polyagents can capture the outcome of multiple possible interactions in a single run.

### 3. THE POLYAGENT CONSTRUCT

A polyagent represents each domain entity with a single *avatar* that manages its correspondence with the domain, and a swarm of *ghosts* that explore its alternative behaviors.

The *avatar* persists as long as its entity is active, and maintains state information reflecting its entity's state. Its computational mechanisms may range from simple stigmergic coordination to sophisticated BDI reasoning. Each avatar generates a stream of *ghosts*. Ghosts have limited lifetime, dying off after a fixed period of time or after some defined event to make room for more ghosts. The avatar controls the rate of generation of its ghosts, but typically has several ghosts concurrently active.

Ghosts explore alternative possible behaviors for the avatar that generated them. They are computationally very simple, and interact with one another through a digital pheromone field, a vector of scalar values that is a function of both location and time. That is, each ghost chooses its actions stochastically based on a weighted function of the strengths of the various pheromones in its immediate vicinity, and deposits its own pheromone to record its presence. A ghost's "program" consists of the vector of weights defining its sensitivity to various pheromones.

Multiple ghosts multiply the number of interactions that a single run of the system can explore. Instead of one trajectory for each avatar, we now have one trajectory for each ghost. If each avatar has  $k$  concurrent ghosts, we explore  $k$  trajectories concurrently. But the multiplication is in fact greater than this.

The environment maintains the digital pheromone field, supporting three functions [1]: It *aggregates* deposits from individual agents, fusing information across multiple agents and through time. Aggregation of pheromones enables each ghost to interact with multiple other ghosts at the same time through their pheromone field, which summarizes their individual behaviors. It *evaporates* pheromones over time, an innovative alternative to traditional truth maintenance in artificial intelligence. Traditionally, knowledge bases remember everything they are told unless they have a reason to forget something. Detecting inconsistencies is NP-complete. Pheromones constantly forget everything they learn, unless it is continually reinforced, so inconsistencies automatically remove themselves. It *propagates* pheromones to nearby places, disseminating information for access by nearby agents.

Propagation enables each ghost to sense the presence of multiple other agents. If  $n$  avatars deposit pheromones, each ghost's actions are influenced by up to  $n$  other agents (depending on the propagation radius), so that we are exploring in effect  $nk$  interactions for each entity, or  $n^2k$  interactions overall. If individual ghosts deposit pheromones, the number of interactions being explored is even greater, on the order of  $k^2$ . The interactions are not played out in the detail they would be in a conventional multi-

agent model. But empirically, they are reflected with a fidelity that is adequate for the problems we have addressed.

The avatar can *activate* its ghosts when it wants to explore alternative possible futures, modulating their generation rate to determine the number of alternatives explored and initializing the ghosts' weight vectors to reflect the breadth of alternatives it wishes to explore. It can *evolve* its ghosts to learn the best parameters for a given situation, monitoring the performance of existing ghosts against some fitness parameter, and then breeding the most successful to determine the parameters of the next generation. It can *review* its ghosts' behavior to estimate the direction and likely variation in how its own future behavior.

The polyagent bears comparison with functionally decomposed multi-agent architectures, population-based search, fictitious play, and two previous uses of the term "polyagent" (**Error! Reference source not found.**).

Our polyagent is distinct from the common use of multiple agents to model different functions of a single domain entity. In a polyagent, each ghost has the same function: to explore one possible behavior of the domain entity. The plurality among ghosts provides, not functional decomposition, but a range of estimates of alternative likely behaviors.

Many forms of evolutionary computation [3] allow multiple representatives of a single entity to execute concurrently, to evaluate their fitness. In these systems, each agent samples only one possible series of interactions with other entities in the domain. Pheromone-based coordination in the polyagent construct permits each ghost to adjust its behavior based on many possible alternative behaviors of other entities in the domain.

Similarly, the multiple behaviors contemplated in fictitious play [5] take place against a static model of the rest of the world.

Like the polyagent, ant-colony optimization (ACO) [2] uses pheromones to integrate the experiences of parallel searchers. The novel contribution of the polyagent is the avatar as a single point of contact for the searchers representing a single domain entity.

"Polyagent" is a neologism to capture the concept of several software agents that collectively represent a domain entity and its alternative behaviors. The term is used in two other contexts that should not lead to any confusion. In medicine, "polyagent therapy" uses multiple treatment agents (notably, multiple drugs combined in chemotherapy). K. Kijima [4] uses the term in analyzing

| <b>Table 1: The Polyagent and Other Technologies</b> | Multiple agents / entity | Avatar/Ghost dualism | Parallel search of alternate behaviors | Parallel search of multiple interactions |
|------------------------------------------------------|--------------------------|----------------------|----------------------------------------|------------------------------------------|
| Polyagent                                            | X                        | X                    | X                                      | X                                        |
| Functional multiple agents                           | X                        |                      |                                        |                                          |
| Evolution                                            | X                        |                      | X                                      |                                          |
| Fictitious play                                      | X                        |                      | X                                      |                                          |
| ACO                                                  | X                        |                      | X                                      | X                                        |
| Kijima's polyagents                                  |                          |                      |                                        |                                          |
| Polyagent therapies                                  |                          |                      |                                        |                                          |

**Table 2: Applications of Polyagents**

| Domain                                         | Avatar    | Ghosts sample ...                                               |
|------------------------------------------------|-----------|-----------------------------------------------------------------|
| Factory Scheduling [1]                         | Workpiece | Alternative routings                                            |
| Path Planning for Robotic Vehicles [6]         | Vehicle   | Alternative paths through battlefield                           |
| Characterizing & Predicting Agent Behavior [7] | Warrior   | Alternative personalities and interactions with the environment |

multiple decision-makers. In Kijima's approach, the term "poly-agent" makes sense only as a description of a system, and does not describe a single agent. In our approach, it makes sense to talk about a single modeling construct as "a polyagent."

#### 4. EXAMPLES OF POLYAGENTS

We discovered the value of polyagents as a modeling construct by reflecting on several applications that we have constructed and observing their common features. The full paper discusses these applications, which are summarized in Table 2.

Our current use of polyagents involves a fair amount of art. There is much room for theoretical work to make the technique more rigorous. Perhaps the most challenging question has to do with the legitimacy of merging pheromones of multiple ghosts representing agent A into a single field that then guides the behavior of agent B. This process enables B to explore concurrently its possible interactions with multiple alternative realizations of A, but the formal legitimacy of this process is open to question.

Pheromone field strength depends on the frequency with which agents visit various locations. It may be viewed as a probability field describing (for instance) the likelihood of finding an agent of a given type at a given location. If those agents are ghosts representing alternative futures of an entity's trajectory, the probability field represents the likelihood of different future states. An analogous situation arises in quantum mechanics (Table 3), which we are exploring as an intellectual and formal model for engineering polyagent systems and interpreting their behavior.

#### 5. CONCLUSION

One strength of agent-based over equation-based modeling is that it captures the details of the trajectory experienced by each entity. In uncertain domains, this strength is also a weakness, because any single set of entity trajectories is only a sample from a space of possible trajectories that is much too large to explore thoroughly by repeated experiments. Polyagents can sample multiple interactions in a single simulation run. An avatar mediates between the entity being modeled and a swarm of ghosts that sam-

ple alternative possible trajectories. The avatar may employ sophisticated cognitive reasoning, but the ghosts interact entirely through digital pheromone fields that they deposit and sense in their shared environment. The avatar modulates the continual generation of ghosts, and interprets their aggregate behavior to provide an estimate of its entity's likely behavior.

We have applied polyagents to several problems. Empirically, the construct functions very well, but it opens a wide range of questions about the relation between multiple ghosts interacting with a pheromone field that represents multiple alternative realizations of other entities. The system exhibits several parallels with quantum physics that suggest the latter discipline may be a useful basis for a more formal model of polyagents.

The full version of this paper is available at <http://www.altarum.net/~vparunak/AAMAS06Polyagents.pdf>.

#### 6. ACKNOWLEDGEMENTS

This material is based in part upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract Nos. F3062-99-C-0202 and NBCHC040153. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the DARPA or the Department of Interior-National Business Center (DOI-NBC). Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

#### 7. REFERENCES

- [1] Brueckner, S. *Return from the Ant: Synthetic Ecosystems for Manufacturing Control*. Thesis at Humboldt University Berlin, Department of Computer Science, 2000.
- [2] Dorigo, M. and Stuetzle, T. *Ant Colony Optimization*. Cambridge, MA, MIT Press, 2004.
- [3] Jacob, C. *Illustrating Evolutionary Computation With Mathematica*. San Francisco, Morgan Kaufmann, 2001.
- [4] Kijima, K. Why Stratification of Networks Emerges in Innovative Society: Intelligent Poly-Agent Systems Approach. *Computational and Mathematical Organization Theory*, 7, 1 (June): 2001, 45-62.
- [5] Lambert, T. J., III, Epelman, M. A., and Smith, R. L. A Fictitious Play Approach to Large-Scale Optimization. *Operations Research*, 53, 3 (May-June): 2005.
- [6] Parunak, H. V. D., Brueckner, S., and Sauter, J. Digital Pheromones for Coordination of Unmanned Vehicles. In *Proceedings of Workshop on Environments for Multi-Agent Systems (E4MAS 2004)*, Springer, 2004, 246-263.
- [7] Parunak, H. V. D. and Brueckner, S. A. Extrapolation of the Opponent's Past Behaviors. In A. Kott and W. McEneaney, Editors, *Adversarial Reasoning: Computational Approaches to Reading the Opponent's Mind*, Chapman and Hall/CRC, 2006.
- [8] Parunak, H. V. D., Savit, R., and Riolo, R. L. Agent-Based Modeling vs. Equation-Based Modeling: A Case Study and Users' Guide. In *Proceedings of Multi-agent systems and Agent-based Simulation (MABS'98)*, Springer, 1998, 10-25.
- [9] Sterman, J. *Business Dynamics*. New York, NY, McGraw-Hill, 2000.

**Table 3: Parallels between Quantum Physics and Polyagents**

| Quantum Physics                                                                        | Polyagents                                                                                             |
|----------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| Duality between (single, localized) particle and (distributed) wave function           | Duality between (single, localized) avatar and (distributed) swarm of ghosts                           |
| Interactions among wave functions' amplitude fields model interactions among particles | Ghosts' pheromone fields can be interpreted as probability densities that model interactions of agents |
| Wave function captures a range of possible behaviors                                   | Swarm of ghosts captures a range of possible behaviors                                                 |
| Observation collapses the wave function to a single behavior                           | Avatar interprets the aggregate behavior of the ghosts and yields a single prediction of behavior      |