The Evolution of Communication Schemes Over Continuous Channels

Gregory M. Saunders
The Institute for the Learning Sciences
1890 Maple Street
Northwestern University
Evanston, IL 60201
saunders@ils.nwu.edu

Jordan B. Pollack
Computer Science Department
Volen Center for Complex Systems
Brandeis University
Waltham, MA 02254
pollack@cs.brandeis.edu

Abstract

Many problems impede the design of multi-agent systems, not the least of which is the passing of information between agents. While others hand implement communication routes and semantics, we explore a method by which communication can *evolve*. In the experiments described here, we model agents as connectionist networks. We supply each agent with a number of communications channels implemented by the addition of both input and output units for each channel. The output units initiate environmental signals whose amplitude decay over distance and are perturbed by environmental noise. An agent does not receive input from other individuals, rather the agent's input reflects the summation of all other agents' output signals along that channel. Because we use real-valued activations, the agents communicate using real-valued vectors. Under our evolutionary program, GNARL, the agents coevolve a communication scheme over continuous channels which conveys task-specific information.

1. INTRODUCTION

Animals, both real and artificial must constantly interact with others by competing for limited resources, by cooperating on a difficult task, or by communicating information about the environment. This paper focuses on communication; in particular, on the issue of how a set of agents can *evolve* a communication scheme to solve a given task without a priori native structure in place.

In speaking about communication schemes, we wish to avoid the term "language." A communication scheme describes the actual signals passed between agents. Language, a collection of sentences drawn from a finite vocabulary, denotes an *interpretation* of a communication scheme. In this sense, language is a subjective phenomena and is ascribed by an observer (Kolen and Pollack, 1995). For instance, the description "agent 0 passes a value of 0.341 to agent 2 down channel 1" reflects the implementation of interaction, while the alternative description "agent 0 just signalled the presence of food to agent 1" involves the interpretation of interaction. These two terms, communication scheme and language, reflect different aspects of the same phenomenon. Their use depends upon which aspect of interaction – implementational or interpretational – we wish to emphasize.

The role of communication in multi-agent systems remains one of the most important open issues in multi-agent system design (Brooks, 1991; Arkin and Hobbs, 1993). Evolution, or genetic search, lends insight into these issues in two ways. First, as a practical matter, evolution opens the door to task-specific languages. Communication within a group of agents, robotic or simulated, should possess some level of flexibility. We resist the urge to adopt a single, fixed scheme for all tasks or to implement a new scheme for every task merely because we can easily understand these approaches. We have turned to evolution as our designer because it allows the possibility for communication schemes to emerge from the communicative needs of the agents actually solving a given problem with little or no regard for explanatory clarity.

The second motivation for studying the evolution of communication focuses on the means rather than the end. Understanding how groups of agents evolve a common communication scheme – knowing which aspects of internal state they choose to communicate, or how they represent information as messages, for instance – should provide useful insights into language development. Furthermore, such understanding would have practical consequences. If we find that our agents always adopt a similar communication scheme, then we should examine that language in detail. Perhaps it provides increased efficiency for the task, or is particularly robust. On the other hand, such regularities could emerge from learning bias in the evolutionary algorithm.

2. RELATED WORK

Several other researchers have studied the evolution of communication schemes, but their work all shares an emphasis on *discrete* communication. Yanco and Stein (1993) investigate a simple "follow-the-leader" task in which one agent, the leader, receives a command which must be followed by a group of agents. The leader chooses one of n symbols to represent the command, broadcasts the symbol to the other agents, and the subordinates respond. A reinforcement algorithm governs both the encodings of the leader and responses of the subordinates; over time, a consensus emerges between the two.

Werner and Dyer (1992) describe a more complex environment in which simulated animals must communicate to find mates. Females, while stationary, can sense potential mates within a limited range and "call out" to them by emitting a signal. Males, wandering around the environment, lack the capacity to produce signals or see the females directly, but they can sense the females' signals and respond by moving toward them. Using a neural network representation for agents and a genetic algorithm for search, Werner and Dyer show that the sexes can agree on a common language.[1]

MacLennan (1992) adopts a higher-level view of language by defining an abstract task in which a group of agents must learn to communicate. Each agent possesses local information in terms of one of n symbols; it chooses a second symbol (from a set of n) to convey that information, and other agents must respond appropriately. Using finite state machines to represent agents and a genetic algorithm, MacLennan shows how the group of agents evolve a common symbol-symbol mapping.

Collins and Jefferson (1991, 1992) study AntFarm, a simulated ant colony in which agents must learn to communicate the presence of food. At each time step, an agent drops between 0 and 64 units of pheromone, which then diffuses throughout the environment as a signal to other ants. Although they have yet to evolve cooperative foraging, the work sheds some light on representational issues, in particular, on the use of neural networks as an agent representation.

1. Werner and Dyer (1993) propose a very interesting model "BioLand" which supports the evolution of communication as well, but the results focus on herding behavior rather than the evolved communication scheme, and it is unclear how the signals generated by the agents affect their behavior.

Ackley and Littman(1994) is the closest in spirit to our work, though significantly more complex in its construction, and focusing mainly on issues of distributed evolutionary computation. The agents in their model operated on tracks and used discrete bit communication in 6 channels.

As stated earlier, all of this work focuses on discrete communication signals, with ensuing finite-sized[2] languages (2-20 for Yanco and Stein; 4-8 for Werner and Dyer; 8 for MacLennan; 65 for Collins and Jefferson and 6 for Ackley & Littman). Implicitly, all of these studies assume that each agent possesses a perceptual system capable of discriminating external events into discrete categories and that the "true" behavior control lies hidden behind such systems. Furthermore, some studies make an architectural distinction between the agent sending the message and the recipient (Yanco and Stein, 1993; Werner and Dyer, 1992; and to some extend MacLennan, 1992, in the sense that at any given time, there is a privileged agent attempting to convey its local information to the others).

3. COMMUNICATION WITH CONTINUOUS SYMBOLS

Our approach to understanding multi-agent communication differs from the work described above. Based upon our work in understanding where complexity arises in observations of dynamical systems, we believe that granting a discrete symbol system privileged position as a substrate for evolution is a confusion of levels (Saunders, Kolen, and Pollack, 1994). Consequently, rather than assume the transmission of discrete signals between agents, we provide our agents with continuous channels capable of supporting a variety of communication schemes. Furthermore, we make no architectural distinctions between transmitter and receiver.

This section describes our main experiments which are fully described in Saunders(1994). First we briefly describe GNARL, the algorithm we use to evolve our agents. Then we introduce an extension of the Tracker task (Jefferson et al., 1992), which will serve as a substrate for our experiments. Next, we describe the method of communication our agents employ. Finally, we describe our experimental results.

3.1 GNARL

GNARL (Saunders, Angeline, and Pollack, 1994) is an algorithm based on evolutionary programming (Fogel, 1992) that induces recurrent neural networks. It pro-

2. The size of a language is the number of distinct signals an agent may produce.

(a)

(b)

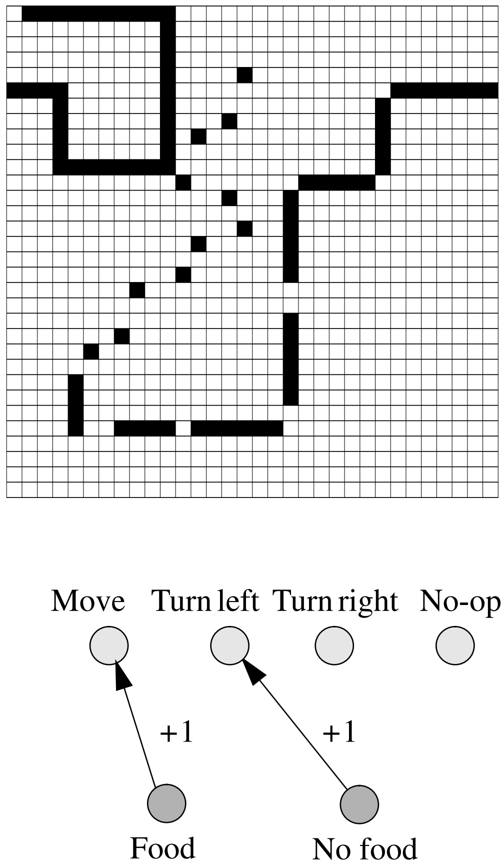

Figure 1: The Tracker task. (a) The trail is connected initially, but becomes progressively more difficult to follow. The underlying 2-d grid is toroidal; (b) The semantics of the I/O units for the ant network. The first input node denotes the presence of food in the square directly in front of the ant; the second denotes the absence of food in this same square. No-op, from Jefferson et al., allows the network to stay in one position while activation flows through recurrent links.

vides a mechanism for the simultaneous acquisition of network structure and weight values. GNARL employs a population of networks, replacing half each generation, and uses a fitness function's unsupervised feedback to modulate the amount of mutation applied to individual networks.

GNARL has been applied to several different problems (Angeline, Saunders, and Pollack, 1993). In particular, we have applied GNARL to the Tracker task (Jefferson et al., 1992) in which a simulated ant must learn to follow a broken trail of food (Figure 1a). Each ant receives two inputs: one indicating the presence of food in the square directly before the agent; and another detecting the absence of food in that same square. Jefferson, et al., allowed four primitive actions: move-forward (and implicitly eat food if present), turn left, turn right, and no-op (Figure 1b). Under these conditions GNARL evolved several different networks

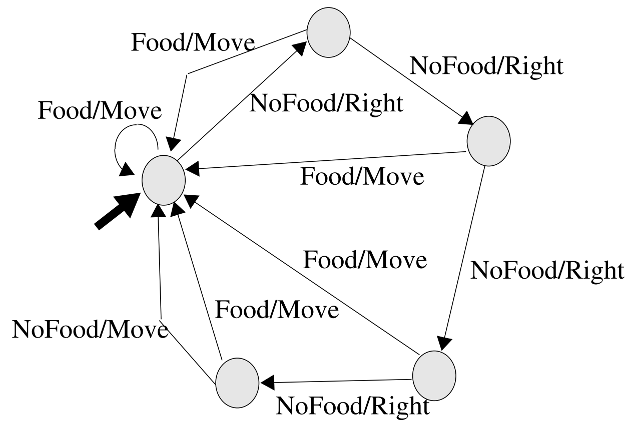

Figure 2: FSA hand-crafted for the Tracker task, from Jefferson, et al., 1992. The large arrow indicates the initial state. This simple system implements the strategy "move forward if there is food in front of you, otherwise turn right four times, looking for food. If food is found while turning, pursue it, otherwise, move forward one step and repeat."

as a solution, one of which closely approximates the finite-state automaton shown in Figure 2.[3]

3.2 The Tracker Task, Revisited

To study the evolution of communication in groups of agents, we extend the Tracker task in three ways:

- increasing the number of agents
- increasing the size of the grid to accommodate these agents
- moving all the food to a small area in the center of the environment

We assume that these modifications will shift the emphasis of the task from evolution of local internal state to evolution of distributed external state, i.e., communication. We concentrate the food within one area so that when an agent finds it and communicates, some food remains by the time other agents arrive. The size of the environment and the amount of food it contains far exceed the capabilities of a single ant: in the limited time available an ant can neither search the entire space nor consume all the food therein. Thus the task design ensures that the only method of complete success necessarily involves communication among the agents.

3.3 An Architecture for Communication

When faced with a task requiring communication, the architecture of Figure 1b will certainly fail; namely, be-

---

3. Note however that the network's behavior is not precisely captured by the FSA. Kolen (1994a, 1994b) shows that, in general, FSAs approximate networks only poorly. Another network induced by GNARL makes this point empirically. (See Saunders, Angeline, and Pollack, 1994).

cause it in no way supports communication. To remedy this shortcoming, we begin by adding *n* additional input and output units to the network of Figure 1b, representing *n* channels of communication. (These architectural changes, along with others not yet described, are shown in Figure 3.)

Output signals propagate throughout the environment, decaying in inverse proportion to squared distance.[4] Perception of these signals is governed by Equation 1. The input signal to agent *a* along the i^{th} channel, $s_{IN}(a, i)$, is a summation of the signals of all other agents along this channel. *A* is the set of agents, $s_{out}(b, i)$ is the i^{th} output signal of agent *b*. The noise in the channel, $U[-u_i, u_i]$ is a uniform random number with range specific to the channel, and σ is a linear threshold function, which bounds the signals in all channels to a user-specified range $[s_{min}, s_{max}]$. In the experiments below, $s_{min} = 0$ and $s_{max} = 40$.

$$s_{IN}(a, i) = \sum_{\begin{bmatrix} b \in A \\ b \neq a \end{bmatrix}} \frac{\sigma(s_{OUT}(b, i) + U[-u_i, u_i])}{distance^2(a, b)}$$

Eqn 1

We have already demonstrated that when hidden nodes are added to the base architecture of Figure 1b, the resulting network can display complex behavior despite the simplicity of its move/turn outputs (Saunders, Angeline, and Pollack, 1994). In this study, however, we wish to maintain a clear separation between complexity arising from communication, and complexity arising from clever activation of the output nodes. We accomplish this in two steps. First, we condense the "move," "turn," and "no-op" outputs of the Tracker task into a single output unit: "Follow FSA." Second, we add *n* additional output units, representing the agents actions relative to the *n* communication channels. We maintain, from the original study, an implicit winner-take-all network on the (non-signal) outputs: when the "Follow FSA" node receives highest activation, the agent follows the primitive food-collection strategy of the FSA in Figure 2; when the i^{th} "Follow gradient" node receives highest activation, the agent follows the gradient of communication channel *i*. Figure 3 shows the final architecture. All activations are continuous; only the hidden activation is squashed (with the standard sigmoid function).

4. We assume that the signals propagate much faster than the agents react (as would a sound wave), so that effectively, at each discrete time step, an agent's output signals establish a wave front whose strength decays over distance.

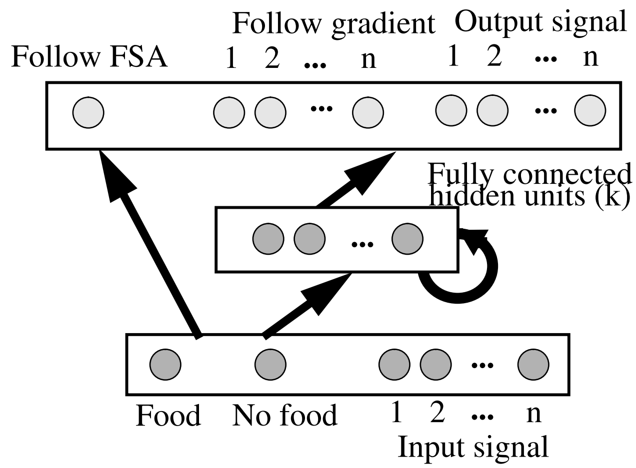

Figure 3: The semantics of the I/O units for evolving communication. The "food/nofood" inputs are from the Tracker task; the "Follow FSA" node represents one particular strategy found by GNARL. The additional nodes, described in the text, give the agent the ability to perceive, generate, and follow signals.

These modifications, though not essential to our results, greatly facilitate their analysis. The food collection strategy of the FSA is indeed quite simple; if activated repeatedly on a grid containing no food, the agent traverses its environment, turning in circles, but never veering from a straight line. Thus if we observe an agent moving non-linearly in the absence of food, we can assert with confidence that the agent is following a communication signal. Furthermore, because of the implicit winner-take-all network, we can easily observe *which* communication signal the agent is pursuing by simply comparing activations across the output nodes.

For the studies reported in this paper, all agents in an environment are homogeneous in that they share not only the architecture of Figure 3, but also common weights. As shown below, however, their behaviors will be quite different depending upon each agent's perspective of its world.

4. RESULTS

With this experimental setup, our thesis can be restated more precisely as follows. Multi-agent systems may evolve task-specific communication schemes. In particular, given the modified Tracker task (Section 3.2), a set of agents instantiated as recurrent neural networks (Figure 3), and a method of signal propagation (Section 3.3), then an evolutionary algorithm (GNARL, Section 3.1) is capable of evolving a communication scheme which allows the agents to perform their task.

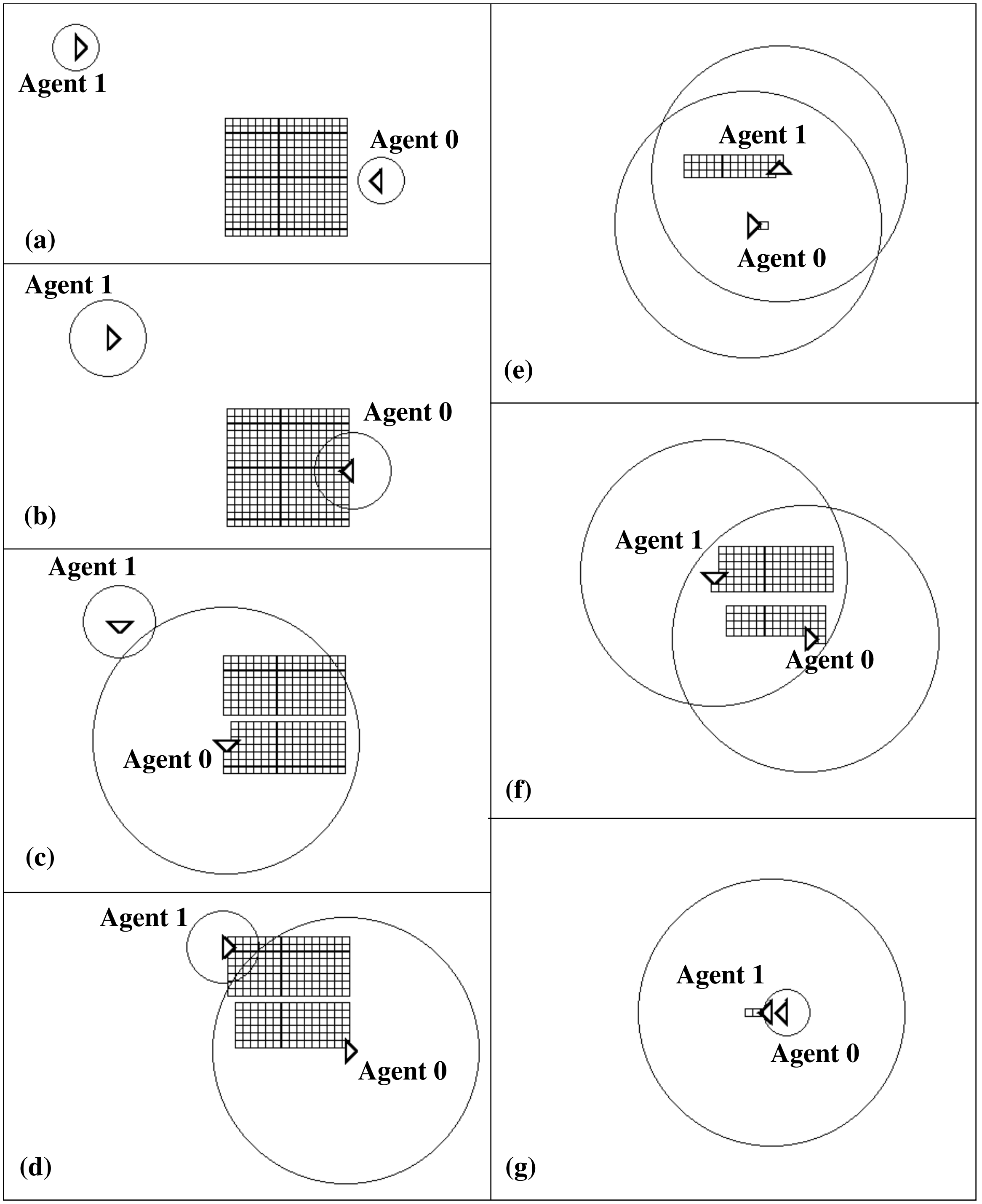

Figure 4: Scenes of evolved communication, 2 agents, 1 communication channel, no noise. The radius of the circles corresponds to the strength of communication. (a) Initial positions: neither agent can sense food; (b) One agent just reaches food, time is t=20; (c) Recruitment – first agent attracting the second, t=40; (d) Second agent just at food, t=60; (e) Agents eating, t=100; (f) Agents eating, t=14; (g) Recruitment – second agent attracting first, t=180.

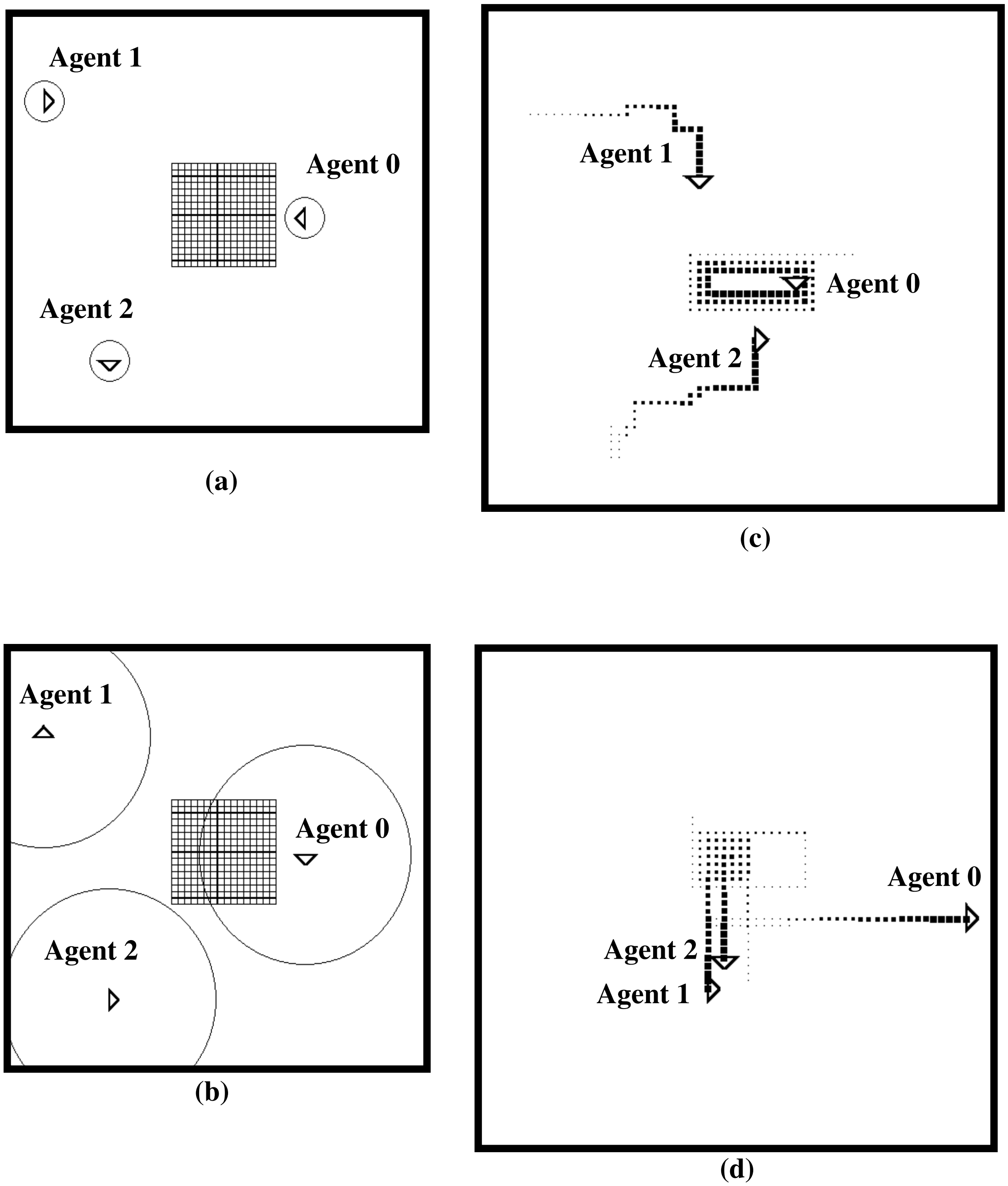

Figure 5: Scenes of evolved communication, 3 agents, 2 communication channels, one noisy. (a) Initial condition, circles denote signal 0; (b) After one *time step, signal 0 has grown to maximum value. It oscillates between the two values when no food is present. (c) Recruitment, time 0 to 150. Dots indicate path of agents (food and signals have been removed for clarity); (d) Agent paths, time 150 to 300.*

We begin testing this hypothesis with a very simple case: 2 agents, each with one hidden unit, capable of passing one real number between each other, with no noise (u_0 = 0). We measure fitness by simply observing the total amount of food eaten by the group. Figure 4a shows the initial environment. Without communication, each agent would follow the FSA, and agent 1 would move in a straight line, finding no food.

With communication, however, the story is quite different. In 800 generations, GNARL discovered a pair of agents (from a population of 50 pairs) which had learned to communicate the presence of food. Figure 4b shows the case just as agent 0 reaches the food; and Figure 4c shows recruitment: agent 0's strong signal, due to the food, attracts agent 1.[5] Figures 4e and f show both agents are emitting high signals while eating, and finally in Figure 4g, recruitment occurs again, this time in reverse.

We chose this case as a demonstration for several reasons. First, snapshots easily capture the evolved communication scheme: larger circles imply a higher signal. Second, the evolved language is fairly intuitive: each agent "yells" when it finds food by increasing the strength of its output signal; upon "hearing" such a signal, the second agent follows it to the source of food. We have also observed several different implementations of the same behavior, another common one being "Yell constantly when you're searching for food, but then grow quiet when eating." In this second case, agents learn to respond to silence.

We now focus in detail on a third, more complex communication scheme. For this experiment, we used the same food distribution, increased the number of agents to three, and retained a a single hidden unit for each agent. To investigate how the agents would respond to noise, we gave them two communication channels, the first clear (u_0=0), the second noisy (u_1=10). Figure 5a shows the initial environment. The circles reflect the strength of signal 0. We do not include signal 1, in the graphics, because it was not used by the agents (more on this below). After one time step, the signals along channel 0 have grown to their size in Figure 5b. In the absence of food, signals in this channel oscillate between these two extreme values. Figure 5c shows recruitment by agent 0; Figure 5d shows that recruitment is not permanent: when the food has been consumed, agent 0 strikes out on its own.

5. The circles denote not signal range, but the radius at which signal strength is one. (Signal strength is the summand in Equation 1.)

Figures 6-8 show how behavior is accomplished. Figure 6 gives the profile of agent 0 over the run. Note how its output signal 0 oscillates in the absence of food. Figure 7 shows the profile of agent 1 throughout the run. The lack of oscillation in agent 0's output is enough to turn agent 1 towards the food. (The 5 spikes in the behavioral profile indicate "Follow signal 0" behavior.)

Agent 2, however, is slightly different (Figure 8). Note the oscillation in its behavior, as it alternates between following the gradient of signal 0 and following the FSA. At first glance, this seems incorrect, because the inputs to agents 1 and 2 look identical, and their architectures are identical, but their output behaviors are very different. The problem might simply be one of perceptual scale (i.e. we can't see any differences). Figure 9 zooms in on the first 50 time steps of the signal 0 input to agents 1 and 2, and shows that there is a slight variance in magnitude of signals, and a phase reversal. Further sensitivity testing, by artificially varying the input signals and observing the resulting agent behavior, showed that the difference in behavior between agent 1 and 2 was caused by the phase difference, not the magnitude difference.

5. ANALYSIS

Detecting the presence of communication is more difficult than it sounds. Communication can occur across long and short distances of both space and time. Random noise can corrupt or masquerade as communication. To operationalize the effects of communication, we first adopted the following definition: *task-specific communication occurs between agents if performance drops when the communications channel is blocked*. In our experimental milieu, we blocked the agents' signals by shunting the channel with various constant values. In all cases, removal of channel 0 drastically reduced fitness, yet the removal of channel 1 failed to hamper the search behavior of the agents, confirming that the agents had learned to rely on the clear channel and ignored the noisy channel.

This definition of task-specific communication, however, makes two assumptions which limit its strength. First, it assumes that interagent communication is goal-directed; communication is the *means* to increasing the performance of a given task. Observing the behavior of the agents does not tell us why one agent squawks over a channel or why another agent reacts to the ruckus. Second, this definition assumes that an agent ignores channels providing it with irrelevant information, i.e., noise. Without dissecting the agent, one cannot tell if noise is a necessary environmental regularity contributing to the normal behavior of an agent. If this is true, blocking

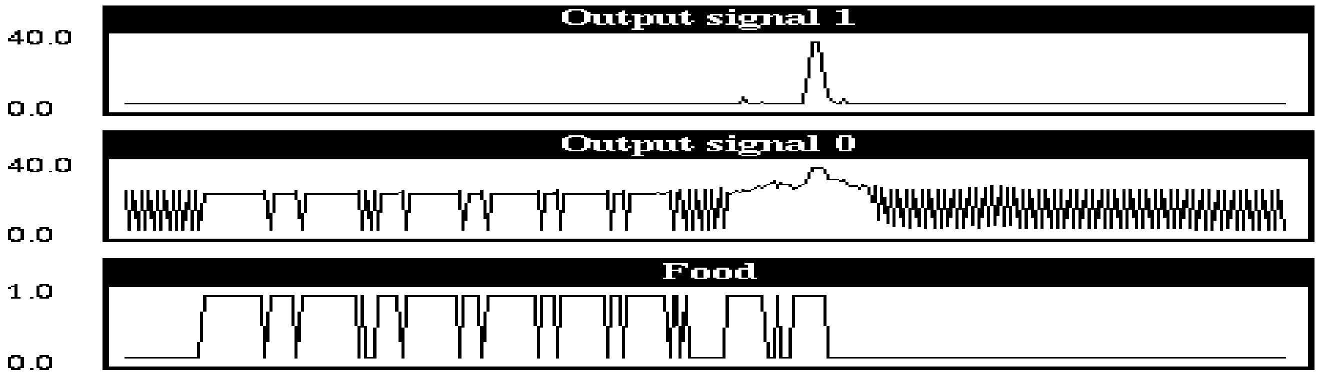

Figure 6: Profile of agent 0, for 300 time steps. The lowest graph is the food input: when food is detected, the value spikes to one; otherwise it is zero. This agent has learned to correlate oscillation of its output signal 0 with the presence of food.

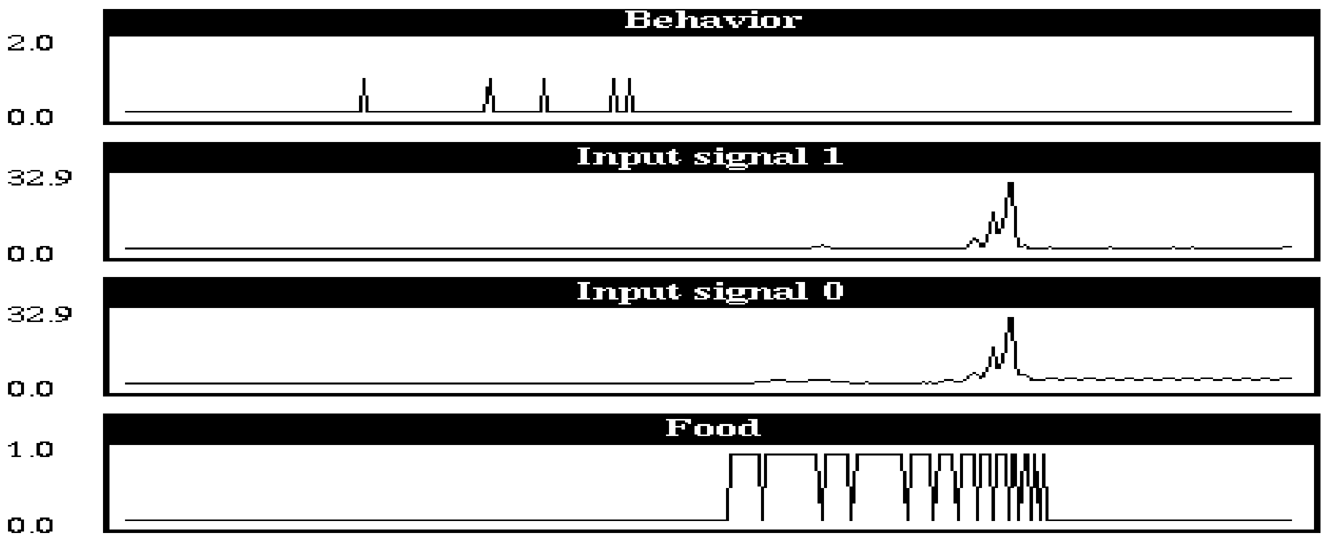

Figure 7: Profile of agent 1. The five spikes in behavior indicate points where the agent follows signal 0, as can be seen in Figure 5c. Because the agent perceives no food during this time, the resulting behavior occurs due to the agents input signals.

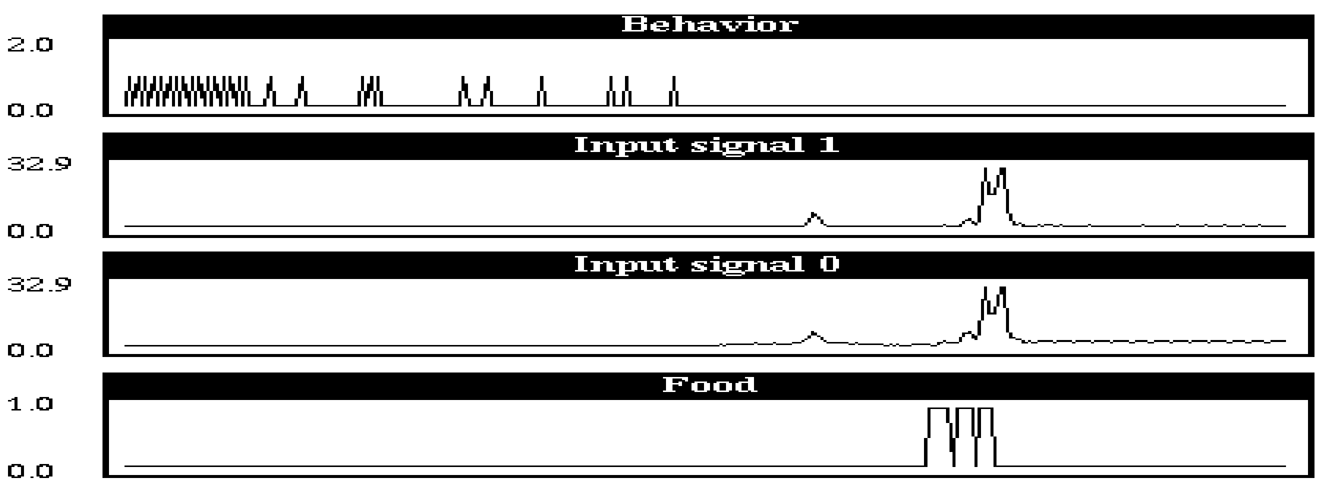

Figure 8: Profile of agent 2. Although its initial inputs (food & signals) look identical to that of agent 1, this agent's initial behavior oscillates between "Follow food" and "Follow signal." The difference is resolved in Figure 9.

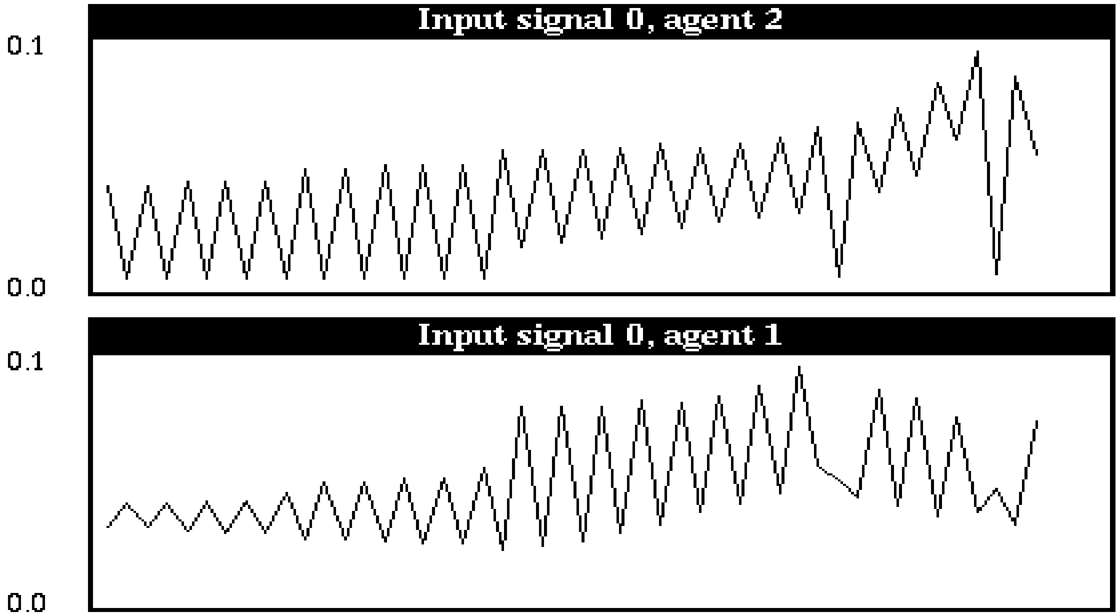

Figure 9: Close up view of the first input signal of agents 1 and 2, epochs 0 to 50, to see why their initial behaviors differ. Agent 1's input begins oscillating between .03 and .04. Agent 2's input begins oscillating between .06 and 0. Upon investigation, we discovered that it is not the magnitude, but the difference in phase which is responsible.

such a channel would more resemble severing an appendage rather than masking a sensor. In short, an evolved or programmed solution to a multi-agent problem is an *instance*, and drawing conclusions about the *class* from a single instance is risky business.

The switch from discrete to continuous signals brings into question the traditional notion of communication in this context. Recall the set of experiments involving three agents. An interesting communication scheme emerged which employed both constant and oscillatory signals. While one could claim that the agents learned to discriminate between oscillatory and constant signals or discern phase differences, we believe another mechanism is at work. Rather than recognize environmental patterns, the agent allows the input sequences to modulate its behavior-producing mechanisms. Kolen (1994b) used this approach to explain the behavior of recurrent neural networks. From this perspective, we view the agent as a state transform system consisting of a set of functions mapping internal state to internal state. Input, by this approach, selects the current transform from this set. At no time is the input stream partitioned, normalized, or recognized, it simply modulates the behavior of the network.

We began with very few assumptions about the nature of communication, essentially stripping away the information-theory veneer that has made previous systems easy to understand. First we replaced the engineer with evolutionary search. Second, we eliminated discrete events and allowed the agents to modify channels with continuous values. These assumptions did not prevent solutions to the modified Tracker problem, in fact some novel approaches were discovered. Identifying the contribution of communication to solving the task proved to be very difficult. Despite these difficulties with understanding how the agents operated, we were able to evolve agents which demonstrated such task-specific behaviors as recruitment. Our hope is that this work opens the door to the study of evolving continuous communication schemes.

Acknowledgments

This work was performed when both authors were with the Lab for AI Research at Ohio State University, along with Peter Angeline, John Kolen and Edward Large, all of whom greatly influenced the work. Partially supported by ONR grant N00014-92-J1195 to the second author.

References

Ackley, D. H and M. L. Littman (1994) Altruism in the evolution of communication. Proceedings of ALIFE IV. Cambridge: MIT Press 40-49.

Angeline, P., Saunders, G., Pollack, J. (1994). An evolutionary algorithm that constructs recurrent neural networks. *IEEE Transactions on Neural Networks,* 5(1):54-65.

Arkin, R. C. and Hobbs, J. D. (1993). Dimensions of communication and social organization in multi-agent robotic systems. In Meyer, J. A., Roitblat, H. L., and Wilson, S. W., editors, *From Animals to Animats 2: Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, pages 486–493, Cambridge. MIT Press.

Brooks, R. A. (1991). Challenges for complete creature architectures. In Meyer, J. A. and Wilson, S. W., editors, *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pages 434–443. MIT Press.

Collins, R. J. and Jefferson, D. R. (1992). AntFarm: Towards simulated evolution. In Langton, C. G., Taylor, C., Farmer, J. D., and Rasmussen, S., editors, *Artificial Life II*, pages 579–601. Addison-Wesley.

Fogel, D. B. (1992). *Evolving Artificial Intelligence*. Ph.D. thesis, University of California, San Diego.

Jefferson, D., Collins, R., Cooper, C., Dyer, M., Flowers, M., Korf, R., Taylor, C., and Wang, A. (1992). Evolution as a theme in artificial life: The Genesys/Tracker system. In Langton, C. G., Taylor, C., Farmer, J. D., and Rasmussen, S., editors, *Artificial Life II: Proceedings of the Workshop on Artificial Life*, pages 549–577. Addison-Wesley.

Kolen, J. F. (1994a). Fool's Gold: Extracting finite state machines from recurrent neural networks. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing 6*. Morgan-Kaufmann.

Kolen, J. F. (1994b). Recurrent Networks: State Machines or Iterated Function Systems. In M. C. Mozer, P. Smolensky, D. S. Touretzky, J. L. Elman, and A. S. Weigend, editors, *Proceedings of the 1993 Connectionist Models Summer School.* pages 203-210. Hillsdale, NJ: Erlbaum Associates.

Kolen, J. F., and Pollack, J. B. (1995). The observers' paradox: apparent computational complexity in physical systems. *Journal of Experimental and Theoretical Artificial Intelligence*, 7, 253-277.

MacLennan, B. (1992). Synthetic ecology: An approach to the study of communication. In Langton, C. G., Taylor, C., Farmer, J. D., and Rasmussen, S., editors, *Artificial Life II*, pages 631–658. Addison-Wesley.

McDermott, D. (1981). Artificial intelligence meets natural stupidity. In Haugeland, J., editor, *Mind Design: Philosophy, Psychology, Artificial Intelligence*, chapter 5, pages 143-160. MIT Press, Cambridge, MA, reproduced from *SIGART Newsletter*, No. 57, April 1976.

Saunders, G. M., Angeline, P. J., and Pollack, J. B. (1994). Structural and behavioral evolution of recurrent networks. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing 6*. Morgan-Kaufmann.

Saunders, G. M., Kolen, J. F., and Pollack, J. B. (1994). The importance of leaky levels for behavior-based AI. *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behavior.275-281*

Saunders, G. M. (1994) The Evolution of Communication in Adaptive Agents. Ph.D Dissertation Columbus: Computer & Information Science Department, Ohio State University. Available in ftp://nervous.cis.ohio-state.edu/pub/saunders/Dissertation/

Werner, G. M. and Dyer, M. G. (1992). Evolution of communication in artificial organisms. In Langton, C. G., Taylor, C., Farmer, J. D., and Rasmussen, S., editors, *Artificial Life II*, pages 659–687. Addison-Wesley.

Werner, G. M. and Dyer, M. G. (1993). Evolution of herding behavior in artificial animals. In Meyer, J. A., Roitblat, H. L., and Wilson, S. W., editors, *From Animals to Animats 2: Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, pages 393–399, Cambridge. MIT Press.

Yanco, H. and Stein, L. A. (1993). An adaptive communication protocol for cooperating mobile robots. In Meyer, J. A., Roitblat, H. L., and Wilson, S. W., editors, *From Animals to Animats 2: Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, pages 478–485, Cambridge. MIT Press.