Adaptive Behavior

The Origin of Epistemic Structures and Proto-Representations

Sanjay Chandrasekharan, Terrence C. Stewart
Institute of Cognitive Science, Carleton University, Ottawa, Canada

Organisms across species use the strategy of generating structures in their environment to lower cognitive complexity. Examples include pheromones, markers, color codes, etc. We provide a model of how such structures originate, and present a simulation where organisms with only reactive behavior learn, within their lifetime, to add such structures to their world to lower cognitive load. This implementation is then extended to show that the same underlying process could generate internal traces of the world (memories) in an internal environment. This model provides a novel account of the origin of internal representations. Further, as both external and internal traces are generated using the same mechanism, the model shows how an extended mind could be implemented. Also, as the stored internal traces develop entirely out of actions, these action components could be activated implicitly. This feature explains the origin of enactable and action-oriented mental content, suggested by recent experiments.

Keywords distributed cognition · epistemic structure · extended mind · representation · simulation theory · situated cognition

1 Introduction

Many organisms add stable structures to their environments to reduce cognitive complexity (minimize search, inference, memory load, etc.), for themselves, for others, or both. Wood mice (*Apodemus sylvaticus*) distribute small objects, such as leaves or twigs, as points of reference while foraging. Such way-marking is exhibited even under laboratory conditions, using plastic disks, and has been shown to diminish the likelihood of losing interesting locations during foraging (Stopka & MacDonald, 2003). Red foxes (*Vulpes vulpes*) use urine to mark food caches they have emptied. This marking acts as a memory aid and helps them avoid unnecessary search (Henry, 1977, reported in Stopka & MacDonald, 2003). The male bower bird builds colorful bowers (nest-like structures), which are used by females to make mating decisions (Zahavi & Zahavi, 1997). Ants drop pheromones to trace a path to a food source. Many mammals mark their territories (Bradbury & Vehrencamp, 1998). Bacterial colonies use a strategy called quorum sensing to know that they have reached critical mass (to attack, to emit light, etc.). This strategy involves individual bacteria secreting molecules known as auto-inducers into the environment. The auto-inducers accumulate in the environment, and when they reach a threshold, the colony moves into action (Silberman, 2003).

It is interesting to observe that such lowering of cognitive load by "doping" the world is commonly

Correspondence to: Sanjay Chandrasekharan, Interactive and Intelligent Computing Division, Georgia Institute of Technology, 85 5th Street NW, Atlanta, GA 30332-0760, USA or A-207, Tenth & Home, 251 10th Street NW, Atlanta, GA 30318, USA.
E-mail: sanjayan@cc.gatech.edu, *Tel.*: +1 404 206 3210

(2007), Vol 15(3): 329–353.
DOI: 10.1177/1059712307076256

reported in animals with smaller brains (such as insects and rodents). In general, animals with large brains (such as horses, elephants, monkeys, etc.) are not known to exploit this strategy, perhaps because larger brains constitute an increased reliance on internal traces of the world. However, humans are an exception to this possible trend of increasing reliance on internal traces. We exploit both internal and external traces simultaneously. The list of epistemic structures used by humans is almost endless: markers, color codes, page numbers, credit ratings, stains, traces, badges, shelf-talkers, speed bugs, road signs, post-it notes, etc.

The pervasiveness of such restructuring of the world across species indicates that adding structures to the world for "cognitive congeniality" is a basic adaptive strategy (Kirsh, 1996). From here onwards, we will call such stable organism-generated external structures that lower cognitive load *epistemic structures* (ESs). The term is derived from the distinction between epistemic and pragmatic action developed by Kirsh and Maglio (1994).

A significant chunk of the cognitive science literature on ESs is from the field of distributed cognition (Hutchins, 1995a, 1995b; Kirsh, 1995, 1996). Kirsh explores the structural and computational properties of such structures, how they function, and how organisms interact with such structures. Such run-time interaction with external structures has been used to argue the case for situated and distributed cognition (i.e. the use of the environment as a cognitive resource by the organism). In the extreme, this dependence has been used to argue against the existence of representation-based (i.e. symbolic) cognition (Brooks, 1991).

We are interested in how organisms generate such structures, which is the other half of the ES problem. Generation is about internal mechanisms that enable organisms to generate structures in the environment and use them. Besides this focus on internal mechanisms that could drive the generation of external structures, in this article we present three novel approaches, as follows:

1. Traditionally, research on cognitive modifications to the environment has ignored animal cases, focusing exclusively on human generation of ESs (Kirsh, 1995). Animals do make modifications to their cognitive environments, and these have been examined extensively under the rubric of signaling. However, the research focus in signaling is not cognitive complexity, but evolutionary models and game theory models (Bradbury & Vehrencamp, 1998). Similarly, recent work on niche construction (Laland, Odling-Smee, & Feldmann, 2000; Odling-Smee, Laland, & Feldmann, 2003) examines the construction of epistemic niches, but does not consider the cognitive mechanisms that lead up to the generation of such niches. In contrast, our analysis is based on the cognitive advantage provided by these structures, and we consider the human and animal cases to be of a kind. We seek to develop an integrated and evolutionarily plausible cognitive model of how ESs arise, where the underlying mechanism is similar in both human and non-human cases. In this view, the distinction between generation by humans and generation by non-human organisms is one of complexity, and not of mechanism (for details, see Chandrasekharan, 2005).
2. The exploitation of external structures has been used to argue that the mind extends into the environment (Clark & Chalmers, 1998). In a similar vein, but more conservatively, Hutchins (1995a, 1995b) has argued that the study of external structures can provide insight into the development and nature of internal representations, and cognition itself. In the second half of the article, we develop this claim beyond the descriptive level of interaction used by distributed cognition, and present an implemented model of how internal traces of the world could originate in reactive agents (agents who can only sense and act, they do no internal processing) within lifetime, using the same underlying process that allow organisms to generate external structures to reduce cognitive complexity. This model integrates generation of external and internal structures under a common mechanism, and thus provides a clearer picture of how cognition could extend out into the world. As this implementation is based on reactive agents interacting with an environment, and they develop the ability to store useful internal traces of the environment, this model also integrates the situated cognition position with the symbolic cognition position, by showing how useful internal traces of the world could arise out of situated activity. The model also presents a number of interesting characteristics of such internally stored traces of the world.
3. The agents in our model develop internal traces of the world entirely out of actions, and any represen-

tational content the traces possess consists of action information. Such action-oriented content offers the possibility of enaction or simulation. This view of the origin of internal content from actions provides support and evolutionary plausibility for the Simulation theory of cognition (a capital S is used here to avoid confusion with implemented simulations), which argues that cognition involves a form of "virtual enaction" (Metzinger & Gallese, 2003; Svenson & Ziemke, 2004). Further, our implementation offers a rudimentary model of the character of such simulatable internal traces of the world and how they could originate out of actions, thus integrating Simulation models with situated cognition models.

The article is organized as follows. In Section 2 we consider the generation of ESs. We provide a model and implementation of how such external structures are generated by non-human organisms. In Section 3 we extend this model to the generation of internal traces of the world. We develop an account and implementation of how organisms could generate such internal structures to lower cognitive load. In Section 4 we address the theoretical implications of this second (extended) model for two wider issues in cognition (representation, and the Simulation/enaction model of the mind).

To make the article accessible to a wide audience, we have simplified the description of our implementations in the main text. Most of the implementation details are provided in the endnotes. For those interested in further details, the code for the two implementations (in Python) is publicly available.[1]

2 Origin of Epistemic Structures

ESs can be classified into three types, based on whom they are generated for (examples of each in brackets):

1. structures generated for oneself (cache marking, bookmarks);
2. structures generated for oneself and others (pheromones, color codes);
3. structures generated exclusively for others (warning smells, badges).

There are other ways to classify ESs (e.g. by function – structures for mating, foraging, etc.), but the above classification is more suited to the objective of this article, which is to understand the mechanisms that lead up to the generation of such structures. Besides capturing the entire space of ESs generated, the above classification also provides a good framework to develop progressive models of ES generation – moving from structures generated for oneself to structures generated exclusively for others.

A central feature of such structures is their task-specificity (more broadly, function/goal-orientedness). To illustrate this concept, consider the following example. Think of a major soccer match in a large city, and thousands of fans arriving in the city to watch (the example is based loosely on the Paris World Cup). The organizers put up large soccer balls on the streets and junctions leading up to the venue. Fans would then simply follow the balls to the game venue. Obviously, the ball reduces the fans' cognitive load, but how? To see how, we have to examine the condition where big soccer balls do not exist to guide the fans.

Imagine a soccer fan walking from her hotel to the game venue. She makes iterated queries to the world to find out her world state (e.g. what street is this, and in which direction am I going?), and then performs some internal processing on the information gained through the queries. After every few set of iterated queries and internal processing, she updates her world state (I'm at point X) and internal state (now searching for point Y), and this process continues until she reaches her destination.

What changes when the ball is put up? The existence of the big soccer ball cuts out the iterated queries and internal processing. These are replaced by a single query for the ball, and its confirmation. The agent just queries for the ball, and once a confirmation of its presence comes in, she updates her internal state to look for the next ball. The ball allows the agent to perform in a reactive, or almost-reactive mode (i.e. move from perception to action directly). The key advantage is that almost no (or significantly less) inference or search is required, compared with the case where the ball does not exist.

This happens because the ball is a task-specific structure; it exists to direct soccer fans to the game venue. Other structures, such as street names and landmarks in a city, are function-neutral or task-neutral structures. The fans have to access these task-neutral structures and synthesize them to obtain the task-specific output they want. Once the widely visible ball, a task-specific structure, exists in the world, they can use this structure directly, and cut out all the synthesizing. (How the soccer fans manage to discover the ball's task-specificity is a separate and relevant issue; see Chandrasekharan, 2005 for an account.) In graph theoretic models (see Kirsh, 1996) such task-specific structures work by collapsing longer paths in a task-environment. Task-specificity is a common property of all ESs found in nature, including pheromones and markers.

2.1 Tiredness Model of Epistemic Structure Generation

How are such task-specific structures that lower cognitive complexity generated? In this article we consider the case of non-human organisms such as ants, wood mice and red foxes (for an account of the human case, see Chandrasekharan, 2005). We first describe our model in high-level terms, and then develop the computational model.

We make the following two assumptions.

1. Organisms sometimes generate random structures in the environment (pheromones, urine, leaf piles) as part of their everyday activity.
2. Organisms can track their physical or cognitive effort (i.e. they become tired), and they have a bias to reduce physical or cognitive effort. We use the terms *cognitive load* and *energy load* interchangeably to indicate this effort.[2,3]

Now, some of the randomly generated structures are encountered while executing tasks such as foraging and cache retrieval. In some random cases, actions executed during these encounters make the task easier for the organisms (following pheromones reduces travel time, avoiding urine makes cache retrieval faster, avoiding leaf piles reduces foraging effort). That is, these random structures shorten paths in the task environment in some random cases. Given the postulated bias to avoid tiredness,[2] these paths get preference, and they are reinforced. As more structure generation leads to more of these paths, structure generation behavior is also reinforced.

This high-level model gives us the outline for building a computational model, where artificial agents display the ability to learn to systematically generate such cognitively[3] congenial structures in their environment.

2.2 Implementation

To test and investigate the above model of ES generation, a multi-agent simulation was implemented. Multi-agent simulations typically consist of a number of agents (usually reactive agents) that have the ability to move around in an environment. The agents can sense some events and objects in the environment, and execute some actions that change the state of the environment. Such simulations are an effective way of understanding complex and dynamic agent–environment relationships (Seth, 2002), and have been used extensively to study a diverse range of phenomena, including honey-bee nest architectures (Camazine, 1991), ant foraging (Bonabeau, Dorigo, & Theraulaz, 1999), evolution of language (Kirby, 2002), human mate-choice (Todd & Miller, 1999), and the development of markets (Tesfatsion, 2002).

The task we have chosen is analogous to foraging behavior (i.e. navigating from a home location to a target location and back again). Our environment consisted of a 30 × 30 toroidal (doughnut-shaped) grid-world, with one 3 × 3 square patch representing the agent's home, and another representing the target. This target can be thought of as a food source, to fit with our analogy to foraging behavior.

2.2.1 Agent Actions

At any given time, an agent can perform one of five possible actions. The first and most basic of these is moving randomly. This consists of going straight forward, or turning to the left or right by 45° and then going forward. The agent does not pick which of these three possibilities occurs (there is a one-third chance of each).

In deciding the actions available to the agent, we needed to postulate some basic facilities within each agent. For our case, we felt it was reasonable to assume that the agents could distinguish between their home and their target, as we were interested in the origins of structure-generation behavior and not landmark-identification behavior. However, this ability to distinguish target and home was provided in a limited fashion, using two more actions. These were exactly like the first action, but instead of moving randomly, the agent could move towards whichever square was sensed to be the most home-like (or the most target-like). Initially, the only things in the environment that are home-like or target-like are the home and the target themselves.

One way to think about these actions is to consider the pheromone-following ability of ants. Common models of ant foraging (e.g. Bonabeau et al., 1999) postulate the automatic release of two pheromones: a *home pheromone* and a *food pheromone*. The ants go towards the home pheromone when they are searching for their home, and they go towards the food pheromone when foraging for food. This exactly matches these two actions in our agents. The home pheromone would be an example of a home-like structure in the ant environment.

The fourth and fifth possible actions provide for the ability to generate these home-like and target-like structures. In the standard ant models, this could be thought of as the releasing of pheromones. However, our simulation has a very key distinction: the ability to modify the environment is something the agents can do instead of moving around. That is, this generation process requires time and effort. They are similar to actions which inadvertently modify a creature's environment in some way. Examples include standing in one spot and perspiring, or urinating, or rubbing up against a tree. These are all actions that require effort and modify the environment, but they do not provide any immediate reward for the agent. Kirsh (1996) calls these "task-external actions".

Note that this implementation does not presume any sort of long-term planning on the part of the agents. We simply specified a collection of actions available to them, and they choose these actions in a purely reactive manner (i.e. based entirely on their current sensory state). They do not initially have any association between the action of making home-like structures and the action of moving towards home-like things. Any such association must be learned (either via evolution, or via some other learning rule). It is also worth noting that our actions are considered at a slightly higher level than is common in agent models. Our agents are not reacting by "turning left" or "going forward"; they are reacting by "following target-like things" or "moving randomly".

Our agents are not designed to form structures automatically as they wander around (as is the case in standard ant models). In our simulation, a creature must expend extra effort to generate these structures in the world. An agent that does this will be efficient only if the effort spent in generating structures is more than compensated for by the effort saved by having them in the world. Moreover, the agents' world is dynamic and the structures do not persist forever. The home-likeness or target-likeness of the grid squares decrease exponentially over time. These structures also spread out over time. A home-like square will make its neighboring squares slightly more home-like. This can be considered similar to ant pheromones dispersing and evaporating, or leaf/twig piles being knocked over and blown around by wind or other passing creatures.

2.2.2 Agent Sensing Our agents had four sensors, two external and two internal, to detect their current situation. The two external sensors sense how home-like and how target-like the current location is (digitized to four different levels). One internal sensor indicates whether the agent has been to the target yet (yes or no), and the other indicates how long it has been since the agent generated a structure in its environment (up to a maximum of five time units). This is all that the agents can use to determine which action to perform. This configuration gives each agent 192 ($4 \times 4 \times 6 \times 2$) possible different sensory states.

2.3 Learning Rules

Given a purely reactive agent, we needed some way of determining which action the agent will perform in each of these 192 states. We investigated two different methods for matching sensory states to actions: a genetic algorithm and Q-learning.

2.3.1 Stage A: Genetic Algorithm Before determining whether the agents could learn to drop "pheromones" to decrease their tiredness within their lifetimes, we first decided to check that it was possible to learn this task across generations (i.e. on an evolutionary time scale). For this, we used a genetic algorithm to evolve foraging behavior in the agents.[4] A genetic algorithm is a general-purpose, but usually very slow, method of finding good solutions to a problem. In this case, no learning at all would occur during the lifetime of one agent; each agent would be locked into a particular sense–response pattern. The agents would thus always perform the same task for a particular state. For example, the agents might be defined to always drop one type of pheromone whenever they are on a very home-like but not target-like square, if they are searching for food and if it has been three time steps since they dropped any pheromone.

The agents start with completely random settings for what to do in each sensory state. This results in agents requiring long periods of time to travel to the target and back. We defined the fitness function of the genetic algorithm to be the inverse of this time measure, which is interpreted as an indication of tiredness. To improve their behavior, the genetic algorithm makes slight modifications (random "mutations") to the set of rules. These mutations change the behavior in unpredictable ways. The changed agents are then simulated to discover how well they do. Over time, the agents evolve to become better and better at their foraging task (i.e. they reduce their tiredness).

In the simulation, 10 agents foraged at the same time. Initially, the agents behaved randomly. Starting at the home, they would wander about and might, by chance, find the target and then, if they were very lucky, their home. Most agents did not find the target and make it back within the 1,000 time steps. On average, we found that each agent was completing 0.07 foraging trips every 100 time steps. After a few hundred generations, the agents were completing an average of 1.9 trips in that same period. This result confirmed that the agents were able to systematically make use of their ability to sense and generate structures in the world, on an evolutionary time scale. Furthermore, this systematic adding of structures to the world provided a very large tiredness advantage over completely random behavior. It also showed that we had not chosen an impossible task for the agents to learn.

However, the heart of our investigation was to determine whether a simple, general learning algorithm would allow our agents to discover and make use of the strategy of systematically adding structures to the world within their lifetimes. Our analysis indicated that the delayed-reinforcement learning rule known as Q-learning (Watkins, 1989) would be the simplest method that was likely to perform this task.

2.3.2 Stage B: Q-Learning The Q-learning algorithm (Watkins, 1989) is a probabilistic learning rule that maps states in the world [s] to possible actions [a], using feedback from rewards and punishments. That is, it learns what actions in any given situation are likely to lead to the maximum long-term reward (or minimum long-term punishment). Given our assumption that the only feedback is tiredness, we give our agents a punishment (a reward of –1) whenever they perform an action, and a reward of 0 whenever they complete a trip. Thus, to minimize their long-term punishment, they would need to travel from their home to the target and back as quickly as possible. This objective function of minimizing punishment can be considered equivalent (monotonically) to the fitness function for the genetic algorithm version, but applied to individual agents. Ideally, the agents would learn that generating ESs can make their trips faster (by allowing them to find the target and the home more easily).

The method utilized by the Q-learning algorithm to achieve this result is structurally simple, but complex in practice. The idea is to estimate future rewards based on past experience. As an initial (approximate) example, consider an agent in state S_1. It may have learned from previous situations that performing action A_1 in state S_1 tends to lead to a reward of R_1. It may also have learned that performing action A_2 in state S_1 leads to reward R_2. The system could then compare R_1 and R_2 to choose which action it should perform. This is the basic idea behind Q-learning, with the vital exception that instead of R_1 and R_2, it uses Q_1 and Q_2, which are the predicted long-term rewards, not the simple one-moment-from-now rewards.

In other words, the agent chooses an action based on this Q value, which is an approximated projection of future reward, based on previous values from experience. Importantly, this projection is not calculated by explicitly running possible action chains, for every possible sequence of actions into the future and compiling their rewards. The projection is calculated using a function (the Q function) learned in real time, derived from previously executed actions, where every action in the world is considered a test action. Once derived, the use of this Q function can be considered as implicitly running possible future actions, across time. This is because every use of Q involves an implicit projection into the future. (For details of this projection, see Section 4.2.1, which provides a more graphical description of Q-learning; see also Stewart & Chandrasekharan, 2005).

Using the Q-learning algorithm, we ran 10 agents for 1,000 time steps.[5] To indicate tiredness, we gave

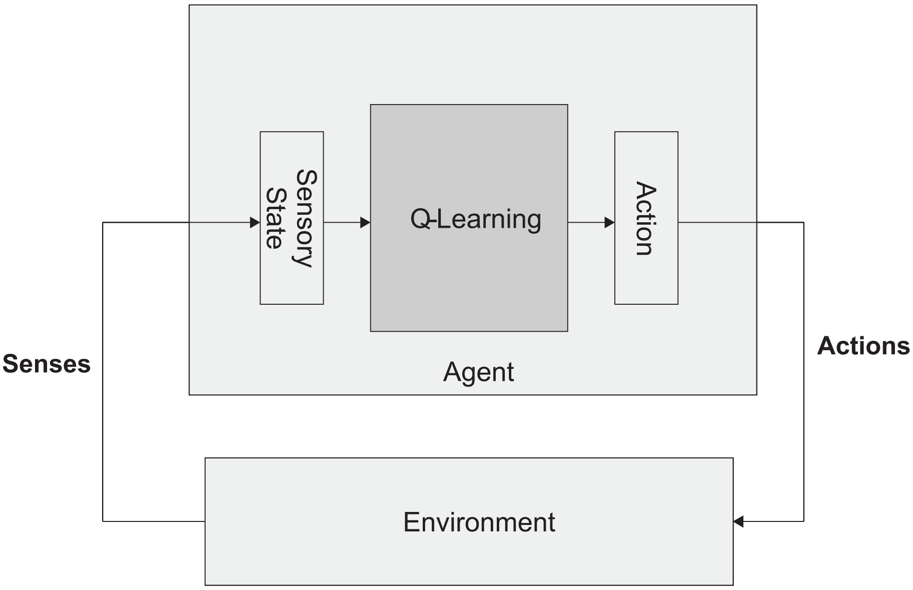

Figure 1 The architecture of the Q-learning model for external structures.

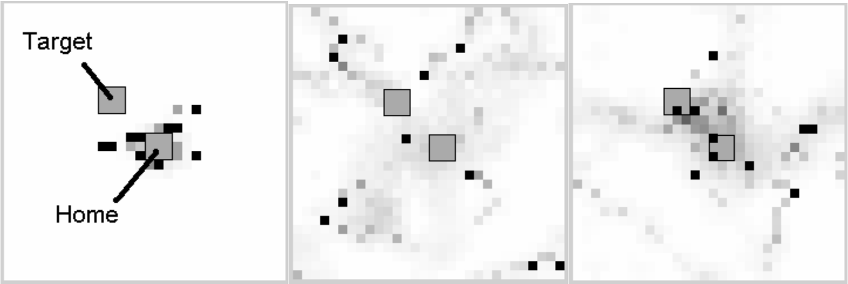

Figure 2 The computer model at 10, 100, and 300 time steps. Black dots are the agents. The shading is darker the more home-like or target-like a particular square is.

them a reinforcement value of –1 while foraging (indicating a constant punishment for expending any effort). When they returned home after finding the target, they were given a reinforcement of 0, and they were then sent back out again for another trip. Each agent independently used the Q-learning algorithm, and there was no communication among the agents. Figure 1 presents an outline of the learning model's architecture.

Result. Figure 2 shows the model at different stages of learning. The dark line in Figure 3 shows the results averaged over 100 separate trials. We can clearly see that the agents are improving over time (i.e. they are making more trips, which means they spend less time performing the foraging task).

2.3.3 Confirming the Role of Epistemic Structures Although we have observed improvement over time, we still need to show that it is the agents' ability to systematically add structures to the world that is causing this effect. To prove this, we re-ran the experiment, this time removing the agents' ability to generate structures in the world. No other changes were made.

Result. We found that when the agents were unable to generate structures in the world, Q-learning did not provide as much improvement. This result is shown by the lighter line in Figure 3. There is still a small improvement given by Q-learning, but the significant improvement seen in the previous experiment is a result of the agents' ability to modify their environ-

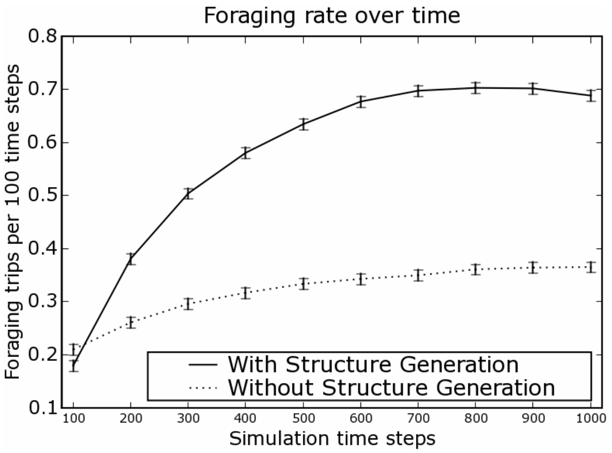

Figure 3 The effect of ES generation. The figure is an average over 1,000 runs of the simulation.

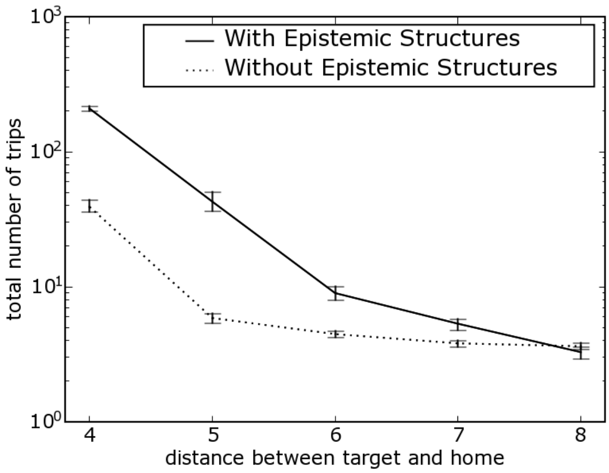

Figure 4 The effect of varying the distance between target and home. Allowing the creation of external ESs consistently improves performance. Note that for far distances, ESs may not provide an improvement for this task.

ment. Q-learning also did not provide significant improvement if the agents were only able to generate one type of structure, or if any of the agent's sensors were removed.

We can also see from Figure 3 that having these extra actions leads to the agents performing slightly worse initially. However, the advantage of being able to form ESs improves the agents' performance quickly. By the end of the simulation, agents require only around 150 time steps to make a complete trip (a foraging rate of 0.66 trips in 100 time steps). This is twice as quick as agents without the structure-forming ability.

An analysis of the actions of the agents showed that they spent 58% of their time generating structures. This is striking, as time spent generating these structures means less time for wandering, trying to find the target or home. Table 1 gives a breakdown of how time was allocated to different actions. It shows that ES generation allowed the agents to complete their foraging task in 150 time steps (down from 300 time steps) even though over half of those 150 time steps were spent standing still. This happens because the Q-learning algorithm learns that the long-term punishment (tiredness) resulting from generating these structures is lower than the tiredness resulting from not generating these structures. There is clearly a very large efficiency advantage in generating and using these markers in the world, and the Q-learning algorithm is able to discover this without explicit long-term planning.

We also investigated the effect of moving the target closer and further from the home. As can be seen in Figure 4, there is a significant improvement, as long as the target is no more than seven steps away. This

Table 1 Time spent performing various actions (ES generation).

Action	With structure generation	Without structure generation
Move randomly	10%	32%
Toward home-like	19%	36%
Toward target-like	13%	32%
Make home-like	35%	–
Make target-like	23%	–

range limitation is potentially because of the instability of the ESs, caused by their evaporation and diffusion rate. Unless otherwise noted, the distance from home to target for the remainder of this article will be set to six.

There are many reinforcement learning algorithms available other than Q-learning, and any of them could be used in this type of model. All these algorithms learn in a similar way, but with different details. So the resulting high-level behavior may be different. Our ongoing research explores the capabilities of these various methods.

2.4 Discussion

The Q-learning system is a concrete proof-of-concept implementation of our model: a simple learning mechanism that allows agents with purely reactive behavior to systematically add structures to the world to lower search. In related work, a similar implementation using simulated Khepera robots was recently reported (Ziemke, Bergfeldt, Buason, Susi, & Svensson, 2004), but the agents in this study evolved the ES strategy; they did not learn it within their lifetime. The generation of environmental structure over evolutionary time has also been explored by Buason, Bergfeldt, and Ziemke (2005), and Sipper (2001). Other related work based on foraging models includes van Dartel (2005) and van Dartel, Postma, van den Herik, and de Croon (2004). In a different vein, Galantucci (2005) used a virtual reality game to show how signing systems emerge in human participants.

The tiredness-based learning model implemented in this simulation can explain the generation of task-specific structure in Cases 1 and 2 (structures for oneself, structures for oneself and others). Case 2 (structures generated for oneself and others) is explained by appealing to the similarity of systems; if a structure provides congeniality for me, it will provide congeniality for other systems like me. The agents in our simulation formed structures that were useful for everyone, even though they were just concerned about reducing their own tiredness. This was possible only because the agents were similar to one another. This is comparable to how paths are formed in fields: one person cuts across the field to reduce his physical effort; others, sharing the same system and wanting to reduce their effort, find the same route optimal. As more people follow the route, a stable path is formed. The evolution of such Case 2 structures have been explored by work in stigmergy (Susi & Ziemke, 2001).

The model, as it stands, cannot explain the generation of Case 3 structures such as the male bower bird's bower (a mating signal that helps female birds make better mating decisions), as the bowers do not seem to provide any tiredness benefit for the generator (for some possible ways of modeling Case 3 structures, see Chandrasekharan, 2005, 2006).

It is worth noting that our model presents a novel heuristic to interpret foraging behavior, as it illustrates a mechanism that could lead to the evolution of ESs within the lifetime of an individual. Existing models of foraging behavior similar to ours are those in Bonabeau et al. (1999), which use the home pheromone and the food pheromone. This is in contrast to such models as that of Nakamura and Kurumatani (1996), where a land-based and an airborne pheromone are used, or models of the *Cataglyphis* species of ant, which uses a complex landmark-navigation scheme that allows it to return directly to the nest (Miller & Wehner, 1988). All of these models assume that pheromones are continually being released while the ant forages, and that there is no learning occurring during the foraging behavior. Our Q-learning model does not make either of these assumptions.

We were unable to find references indicating that real ants (or other creatures) might, in fact, learn to use pheromones (or other ESs, but see the pigeon example below) within their lifetime, or any research that indicates that the effort required to produce these pheromones might interfere with foraging behavior. Our simulation thus provides a very novel result, as current biological models assume (based on experimental evidence) that such ES structure-generation behavior is mostly innate, and is based on evolutionary learning. Interestingly, recent research shows that homing pigeons learn within their lifetimes to use human-generated environment structure in a similar fashion to reduce cognitive load. They follow highways and railways systematically to reach their destination, even following exits (Guilford, Roberts, & Biro, 2004). A similar landmark-based navigation system has also been reported in bees (Gould, 1990). Such use of existing environmental structures could be seen as a special case of our ES model, where encountering some existing environmental structures lowers cognitive load, and paths with such structures get preference.

More generally, our model illustrates a learning mechanism that could underlie niche construction (Laland et al., 2000; Odling-Smee et al., 2003). This is because the reinforcement factor we appeal to (tiredness, or energy efficiency) is general and organism-independent, and it could also drive the construction of physically congenial structures (such as beaver dams and spider webs). Sterelny (2006) considers the link between cognitive load and epistemic construction, but argues that the use and generation of ESs are themselves high cognitive load activities. Our model argues in the other direction, showing that ESs not only lower cognitive load, but that this lowering could itself drive the generation of further ESs. Sterelny (2005) also makes a distinction between mere effects and construction, and argues for splitting the notion of niche construction into different categories. We agree with this, although the within-lifetime model of learning we present here raises significant challenges to Sterelny's argument based on evolutionary dynamics. However, we also think the lumping of physical and cognition-oriented niche construction activities by Odling-Smee et al. is justified to some extent. This is based on two factors: (i) our model presents an organism-independent mechanism that could generate structures that provide physical congeniality as well; (ii) we agree with Godfrey-Smith's position (Godfrey-Smith, 1994) that cognition shades into other adaptive techniques. Although we support the lumping approach to a degree, we believe further analysis of each constructed structure would require a splitting as advocated by Sterelny (2005).

An interesting aspect of the within-lifetime learning model is that it scales well to human situations (see Chandrasekharan, 2005), and could be used to explain the developmental origins of structure generation behavior in humans. It can also be extended to account for the generation of internal structures that lower cognitive load. We present this model in Section 3.

3 Generation of Internal Structures

In Section 2 we established that a simple, plausible, and efficient learning mechanism could enable reactive agents to add ESs to their environment within their lifetime. The agents could perform actions that changed aspects of their environment in ways that could be sensed by that agent later, and the effects of reinforcement caused some structures to be systematically created in certain situations. We have established that this works with a simple foraging task and the basic Q-learning algorithm.

This within-lifetime learning model raises an interesting question: can similar within lifetime learning lead to the generation of novel structures in the agent's mind, rather than in the agent's environment? This seems to be both a natural extension of our work on external structures, and, more importantly, a novel way to model the origin of internal representations in rudimentary agents within their lifetime. If an agent can learn this strategy of generating internal structures to lower tiredness, then it can choose to remember particular things in particular ways to benefit it in the long term, just as our earlier experiments showed that it was possible to choose to drop pheromones in useful ways.

This requires developing a task and a set of actions that change internal states, so that an agent using the same Q-learning approach can learn to remember some states of the world and use this information to better execute a task. For consistency, we chose the same foraging-style task used in the previous experiment. To do this, an agent needs two things: a way of remembering where it has come from (or, equivalently, where it is going next), and a way of knowing how to get there. In our previous experiment, we gave the agent the first capability (the internal state sensors, indicating whether they were looking for the target or looking for home), and it had to learn the second capability. We now changed the task so that it had the second capability, and then had to learn the first. That is, the agent did not have the knowledge of whether it is supposed to be looking for its home or looking for the target. It must learn to keep track of that information on its own, via actions that change internal states.

3.1 A Model of Internal Structures

In theory, such a memory could be implemented using a single value that the agent could learn to set to a zero whenever it had found its home, and set to a one when it had found the target. Whenever it was wandering around in between, it could look at the value stored to let it know which way it was going. However, we discovered that this sort of memory is too fragile: the agent could make one random mistake (changing the value at the wrong time, for example), and the system would become useless. This brittleness in learning arises because such a system does not have two key features of the successful external actions/sensors in the previous experiment.

1. Context-specific generation of structure. When an agent is dropping pheromones, it does not have the option of dropping pheromones anywhere in its universe. It can only drop them (and sense them) where it is. The action is thus not drop-pheromones, but rather drop-pheromones-at-my-location. Similarly, the sensors sense pheromone-at-my-location.
2. Gradual generation of structure. Dropping pheromones makes a small change to the pheromone level at a particular location in space. This allows for the smoothing out of errors, and allows the learning process to converge to a solution, instead of learning from discrete bits of information.

Note that these are not canonical features of internally stored structures in classical models of internal structures (Fodor & Pylyshyn, 1988). In contrast, connectionist models do argue for internal structures with these features (Smolenksy, 1989). A mechanism supporting these two features would provide an internal equivalent to the pheromone dropping, spreading, and sensing mechanisms in our ES experiment. Such a mechanism needs to have the following three capabilities.

1. It needs to be able to store data associated with a particular context. That is, we need to be able to give it a particular sensory state and a particular piece of data (say a 1 or a 0), and it should be able to remember this pairing. This is functionally similar to dropping pheromones of different types at a particular point in the world; the data being remembered can be thought of as being at a particular point in the creature's memory.
2. It needs to be able to recall data when in a particular context. That is, when the agent is in a particular sensory state, it should have a sensor that indicates what value was stored in the past in this state. This is functionally similar to the sensor that indicates the level of pheromones at a particular point in the world; the value being given by this internal sensor is, in some sense, the value being stored at a particular location in the agent's memory.
3. The information needs to spread and change gradually. That is, data stored in one context should be available in similar contexts, and any new data being stored should cause only small changes. This is functionally similar to the spreading of pheromones in space, and the fact that dropping new pheromones makes only a small change to the amount of pheromones at that location.

The first two criteria are features of episodic memory (Tulving, 1983), which has recently been shown to exist in some birds and rats (Griffiths, Dickinson, & Clayton, 1999; Kart-Teke, De Souza Silva, Huston, & Dere, 2006). The third is consistent with a capacity for generalization (the ability to respond appropriately to novel situations by adapting experience from similar situations), but also emphasizes the need for gradual adjustment of these representations. However, the justification for these criteria is based on our results with external structures, not existing representational theories.

Internal structures with these three properties would allow a creature in state X to perform an internal action that associates a particular number with state X. Then, in the future, when it is in state X (or in another similar state), it will be able to remember that number. Furthermore, if it later chooses to associate a different number with state X, its memory will change gradually.

A well-studied mechanism that has exactly these characteristics is the standard feedforward neural network trained by back-propagation of error. Such a network was chosen as our internal memory mechanism (or "memory medium"). Just as our first simulation had agents with mechanisms for dropping and sensing pheromones, in our new experiment we gave the agent a mechanism for storing data into this sort of network, and a mechanism for sensing the current output of the network. This network thus plays the same role as the external environment in the first experiment, becoming an internal environment (Dennett, 1975; Hills, 2006).

Note that our reason for using this neural network is quite different from the traditional reasons for using a neural network (such as graceful degradation and neural plausibility). We are using a neural network because it is a system which is similar to the gradually changing world the agent lives in. An interesting speculation here would be that the neural-network-like structure of the brain evolved to approximate the agent's life world. This would be in line with the idea that "the function of cognition is to enable the agent to deal with environmental complexity" (the environmental complexity thesis; Godfrey-Smith, 1994; for a refinement of the thesis more in line with the work reported here, see Seth, 2002), Note also that in our model the neural network is not being used to represent the world. It is being used to allow the agent to represent those useful parts of the world that it cannot directly sense.

3.2 Defining the Experiment

Given the above characterization of an internal memory store (or internal environment), we can now precisely define the new version of the foraging problem. We again have a simple grid-world, with a home and a target. As the structures that agents would generate are internal and not shared by others in an external world, we have just one agent in the simulation. At any moment, the agent can perform one of the following actions:

1. move randomly;
2. move towards the target;
3. move towards the home;
4. train the internal environment to associate 1 with the current sensory state;
5. train the internal environment to associate 0 with the current sensory state.

As discussed previously, we are giving the agent the ability to simply move directly towards the target, as we are focusing on its ability to learn to remember which way it is currently supposed to be going.

The agent makes its decision on which action to perform based on its sensory information. Here, we have three sensors, which define the agent's current sensory state:

1. home detector (1 if the agent is at its home, 0 otherwise);
2. target detector (1 if the agent is at the target, 0 otherwise);
3. current memory (the output of the neural network for the current sensory state, i.e. the data currently being remembered).

The agent then uses Q-learning to learn to perform different actions based on its current sensory state. Whenever the Q-learning system chooses actions 4 or 5, the system uses back-propagation learning to train the internal neural network to associate a value (0 or 1) with the current sensory state.[6] Figure 5 provides an outline of the learning system's architecture.

It should be noted that there is a subtle recursion occurring in this model. One of the components of the agent's sensory state is the output of the neural network (the internal environment), but that output is itself dependent on the current sensory state. This means that what the agent remembers is dependent on what it is currently remembering. This architecture is similar to that used by Tani and Nolfi (1999). This memory system can be seen as a sort of internal environment, as it is functionally connected to the rest of the agent in exactly the same manner as the actual external environment. (For an interesting extension of this internal environment idea and foraging into more complex domains of cognition, particularly psychiatric disorders, see Hills, 2006.)

As before, the agent's only reward is based on the total amount of effort required to complete a trip to the target and back home. The agent's performance can then be compared with that of the same agent without the ability to perform these internal actions. This is the same approach taken in the previous experiment.

3.3 Results

Figure 6 compares the foraging performance of the agent with the ability to generate internal structures to that of an agent without this internal mechanism. As in the previous experiment, we can see that having the ability to generate internal structures results in behavior that is initially worse, but that then improves to be consistently better than the agent without this ability. These data are an average of 3,000 runs at each setting.

The agent spends only 22% of its time generating internal structures, compared with 58% in the external case. Comparing the two cases is not entirely justified, as the internal traces and external structures serve sig-

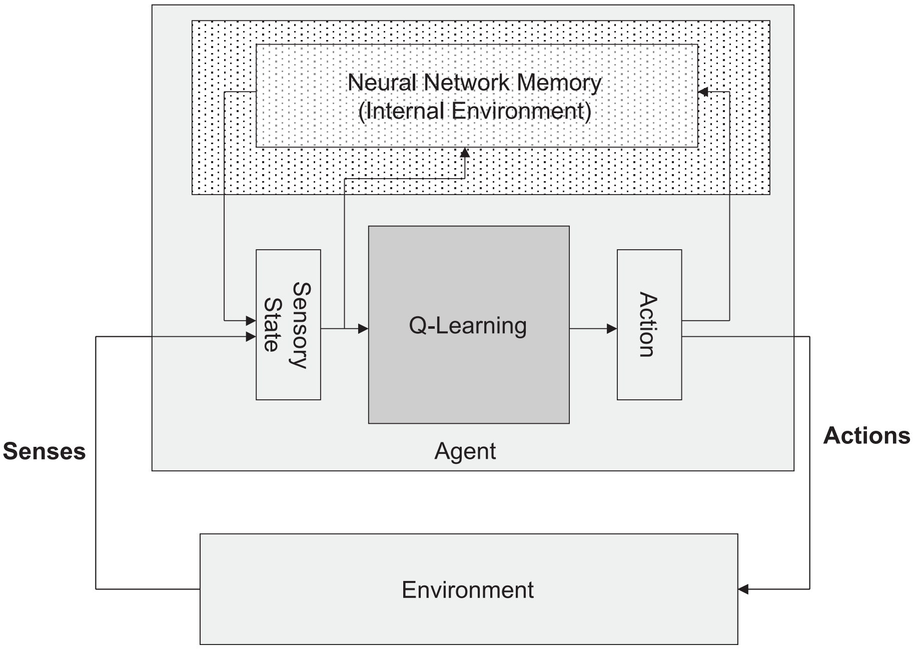

Figure 5 The architecture of the learning system for internal structures. Note that the only difference between this and the previous learning system is the neural network memory in the agent (upper segment).

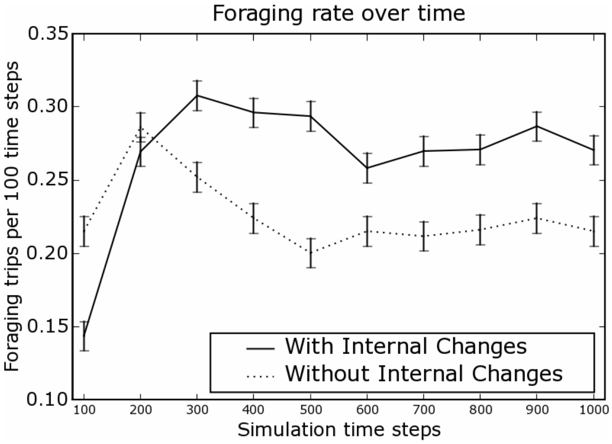

Figure 6 The foraging performance of the agent, with and without internal structures. This is an average over 1,000 runs of the simulation.

nificantly different purposes in the foraging task: the first orients the agent; the second marks the route. However, it is worth considering that internal structures may be more efficient and stable than external structures, as they are less costly and are under the complete control of the agent. This could be one reason for the "representational turn", the wider use of the internal trace strategy in nature than the external one.

Table 2 Time spent performing various actions (internal trace generation).

Action	With trace generation	Without trace generation
Move randomly	12.8%	30.4%
Go to home	37.2%	38.8%
Go to Target	27.5%	30.8%
Remember 1	11.0%	–
Remember 0	11.5%	–

To assess the robustness of our findings, we performed a series of simulations while adjusting the model in various ways. Our results (see Figure 7) show that it is, in fact, a robust phenomenon. The only situations where internal traces do not improve performance are if the neural network has too few hidden nodes (less than three), the neural network's learning rate is around 0.01 (a very low value for such systems), or the Q-learning learning rate is below 0.1 (also a low value for such systems). Furthermore, the results are robust for varying distances between the

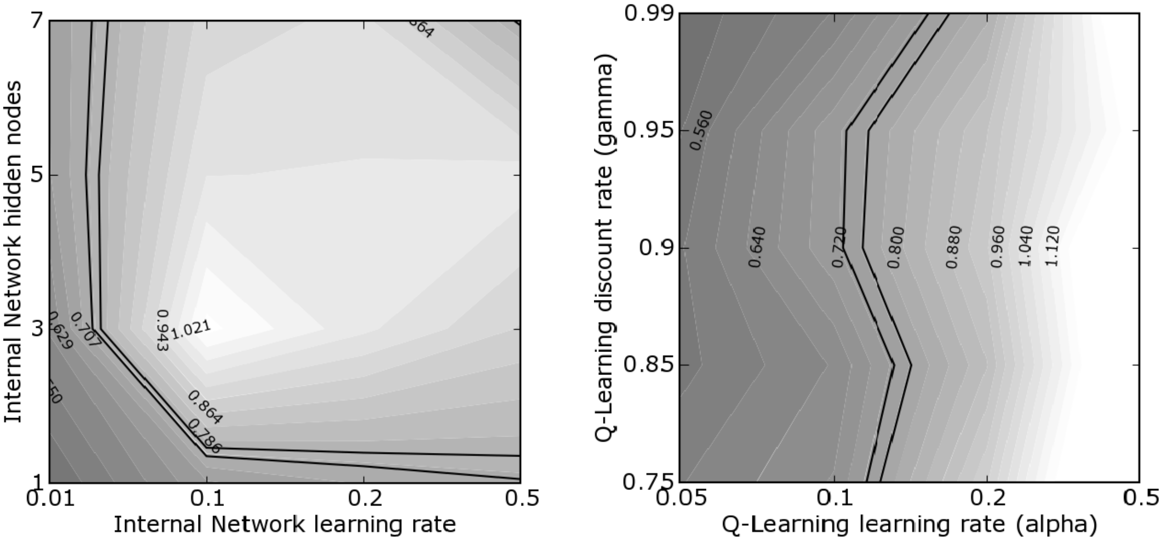

Figure 7 Performance (number of successful trips) of the internal ES model for varying parameter settings. The black lines indicate the performance without structure generation. All points drawn in a lighter shade indicate parameter settings for which structure generation results in more trips to the target and back. These results are representative for other combinations of parameter settings.

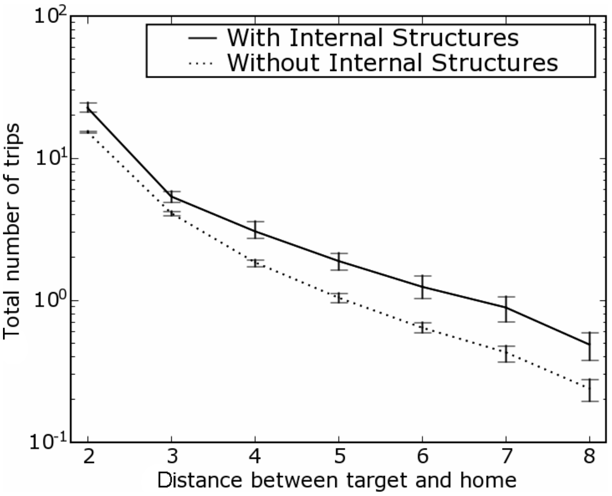

Figure 8 The effect of varying the distance between target and home for the internal ES model. Allowing the creation of internal ESs consistently improves performance. Note that if the target and home are right next to each other (distance of 1), structure generation actually slightly decreases performance.

home and target (see Figure 8), although we did find that if the home and target are right next to each other (a distance of 1), then the ability to form internal ESs actually decreases performance slightly. This indicates that the system may not be well suited to highly simplistic environments, but it degrades gracefully without severely impacting performance.

The results above show that the agents benefited from the ability to remember particular values in particular contexts. In other words, the simple reinforcement-learning approach that worked for learning to generate external ESs is also able to systematically generate and make use of internal structures. Importantly, it seems to work only when the internal structures generated are context-specific and gradually distributed.

4 Theoretical Implications

The above two simulations present an integrated proof-of-concept model of how both external structures and internal memory structures come to be used as task-specific structures, and how such structures could systematically be generated within lifetime, based just on the feedback of cognitive load via a steady tiredness punishment. The two models have wide theoretical implications, but we focus on the model of internal structures, and its implications for the following two areas of cognition:

- representation;
- the Simulation model of the mind.

4.1 Internal Traces as Proto-Representations

The defining feature of mental representations is that they stand in for other things; they are "about" other things (Dennett & Haugeland, 1987). Clark and Grush (1999) provide three criteria for a "minimal robust representationalism": (i) representations would be inner states whose adaptive functional role is to stand in for extra-neural states; (ii) the states with representational roles should be precisely identifiable; (iii) the representations should enhance real-time action. ESs, as we have defined them, do not meet the first of these criteria, as they are not inner states that stand in for extra-neural states. They are outer states, and could be viewed as similar to environmental states directly used by organisms.

However, based on the criteria of Clark and Grush, the stored internal traces of the world in our second simulation could be considered *proto-representations*, because they are inner states, they are precisely identifiable (to some extent; see Section 4.1.1), and they enhance real-time action. Once the structures stabilize, they stand in for something specific in the world, namely the home location and target location, and this is their adaptive function. The traces are about something in the world, and they are useful because of this aboutness. The agents store and use the traces to exploit the aboutness, as this feature helps them choose the best action in a context. The representational character of these structures is broadly in line with the functional theory of representation developed by Millikan (1993), with cognitive load reduction acting as the "proper function" in our model. Interestingly, the inner traces are also what Millikan (1996) terms pushmi–pullyu representations, acting simultaneously as both a directive for action and describing a state of the world.

However, these internal traces are not full-bodied representations, the stronger version of representation outlined by Clark and Grush, because our agents do not use the internal traces as surrogates to model the world when the actual structures do not exist in the world (as in the case of being able to mentally rotate an object when the object is not in the visual field; see Clark & Grush, 1999; Beer, 2003), although they support a weak form of surrogate modeling (see section on simulation). This is one reason why we consider our internal traces proto-representations. The second requirement laid out by Clark & Grush (1999) for an internal structure to be a full-bodied representation, is that the structures should be fully decoupled from ongoing environmental input. Our internal structures do not meet this criterion, but we consider this requirement to be needlessly strong, as this would mean the structures are also decoupled from the learning process, which may not be desirable (see also Section 4.2).

Another reason why we consider the stored internal structures as proto-representations is our agents' "selective representation" of the world (Mandik & Clark, 2002), where an organism is considered to perceive and cognize a "relevant-to-my-lifestyle world, as opposed to a world-with all-its-perceptual-properties". In this view, the mental representations of organisms are highly constrained by the biological niches within which the organisms evolved.

Our model assumes and builds in some basic internal structures such as those involved in sensing, acting and learning. We consider this justified because, strictly speaking, there are no reactive agents in the world; all agents have some basic internal structures (and therefore an *Umwelt*; for a careful discussion of this point, see Seth, 2002). We show that given this basic ability for sensing, acting and learning, agents could develop a secondary form of representation, a structure that stands in for something in the world. Moreover, our model explains what such "primitive" representations are: they are the internal traces of the world that allow the agent to shorten paths in a task environment. Roughly, they are computation-reducing structures (and equivalently, energy-saving structures). Metaphorically, they are internal stepping stones that allow organisms to efficiently negotiate the ocean of stimuli they encounter. By extension, aboutness, or the standing-in property of internal traces, is an energy-saving mechanism in our model. This view is very different from traditional conceptions of representation and aboutness (also see Section 4.1.1).

The model provides a unified account of the generation of external as well as internal structures, as the internal structures are stored using the same process as the external structures, and the structure of the internal traces is similar to the structure of the external traces. Given this same underlying mechanism, the agent can transform the world or itself, depending on task and resource conditions. The two manipulations (internal and external) are equivalent at the mechanism level. Internal changes lead to internal representations, and external changes lead to the world being used directly.

This integrated generation mechanism illustrates a way of storing task-specific traces inside and outside in an opportunistic fashion. The organism could exploit both together, thus extending cognition out into the world (Clark & Chalmers, 1998). In this proposal, the notions of storing representations and using the world directly (the symbolic and situated views) are not at odds with each other; they are just two ways of solving the adaptation problem.

According to the *extended mind* thesis, the structures organisms generate in the world are extensions of the cognitive system, in the sense that the organism would not be able to function if the extensions did not exist (as in the case of the patient with Alzheimer's disease losing his notebook and his notes in the world, or a blind person losing his cane). Our model does not support, but also does not rule out, this strong version of the extended mind thesis, as the agents in our model could function to some extent even without the ES, although the higher energy load involved in such cases lowered their performance. In the limit, this higher search cost may cripple the agents, in which case the ESs could be treated as extensions of the cognitive system. For the moment, the model only supports a weaker version of the extended mind thesis, where the external structures are resources for the agent. The extension-resource distinction is not a hard and fast one in our view; it is dependent on the cognitive load involved in the task.

4.1.1 Nature of Proto-Representations Our model of how such structures could be generated provides some insights into the character of proto-representations. The classical notion of internal traces of the world assumes the storing of static structures, with a one-to-one relation with structures in the world (Fodor & Pylyshyn, 1988). This is the traditional symbolic representation view. It is difficult for reactive agents to learn to store and make use of such static structures, as such structures do not support incremental and contextual learning, which is the type of learning our model uses and is commonly postulated in low-level organisms.

If we assume that the storing of internal traces of the world by lower-level organisms originated to support tasks, and this storing behavior was learned by organisms, the notion of storing static traces with one-to-one relations with entities in the world would need to be revised. This leads us to propose what we call the *distributed origin* thesis of representation: to learn the internal trace strategy (i.e. to begin to represent), the traces need to have a process structure, where elements are initially randomly stored in an internal network (which acts as an equivalent of the external environment), and the agents sense these internally stored elements and act. Through an incremental process based on feedback of cognitive load, these elements then gradually become systematically stored and acquire a representational nature. Such an internal representation is not a single well-defined structure that reflects the world mirror-like, but a systematic coagulation of contexts and associated actions, spread over a network. The structure itself is just a common thread of elements running through contexts and associated actions that lower cognitive load. It is this common status that leads to the thread becoming stabilized as an internal trace. Metaphorically, such an internal representation resembles the core of an active bee swarm, rather than static symbolic entities such as words or pictures.

This action-driven model of representation, and the recursive and dynamic relation between the internal traces of the world and the agent's actions-in-the-world, make our implementation more than a standard neural network model. A central difference is that in our model a reference relation develops between the neural network and elements in the agent's environment. This reference relation is usually hard-coded or assumed in standard neural network models. Also, although we use a neural network, it is used to create an internal equivalent of the agent's task environment, and the agent learns to store task-specific internal traces of some aspects of that environment. As far as we are aware, there are no neural network models that show the origin of such task-specific internal traces. Most neural network models assume that a correspondence relation, albeit a distributed one, exists between the network and the world.

We would also like to emphasize here that the internal traces developed by our agent are radically different from the categorical structures evolved in recent work using evolutionary learning (see Beer, 2003; Steels & Belpaeme, 2005). The central differences are: (i) our agent executes a task similar to a real-life task; (ii) it learns to store task-specific internal traces of the world; (iii) it learns to do this within its lifetime.

Our agent exists in a representation-hungry task environment (i.e. one that requires some form of representation to do the task well; see Clark & Toribio, 1994). The agent learns to represent an aspect of this environment, because such representations lower cognitive load and help the agent to execute the task with less effort. So the internal traces our agent develops are task-specific traces of the world, not categories. Our model considers internal traces as useful and action-driven structures, and we show that they arise because of these features. In contrast, evolutionary models of category learning presume categories that mirror the world, and it is not clear why they arise. A related difference is that storing traces requires effort in our model, and the agent chooses to store traces because the effort involved in storing a trace is compensated for by the advantage such storing provides. Category learning models assume that learning categories are useful, and therefore category learning is not something the agent chooses to do out of many possible actions.

In our model, the internal and external structures serve the same purpose for the organism, they arise out of the same learning mechanism, and they have a similar distributed and dynamic structure. This raises the question, why not call the external structures *representations*? This is a very interesting question, particularly given the complex, non-one-to-one, notion of standing in we argued for in Section 4.1 (which, incidentally, dilutes the first criterion of Clark & Grush of inner states pointing to extra-neural states). It is true that our notion of standing in makes the external structures in our model (collectively) eligible candidates for the *representation* label, as they could be (collectively) taken to stand in for "home" or "target". However, they still lack the two crucial properties required for a structure to be a full-bodied representation, as argued by Clark and Grush (1999): (i) the ESs cannot be fully decoupled from ongoing environmental input; (ii) as the structures are not within the skin of the agent, they are not transportable and utilizable in other circumstances. Thus, they do not support surrogate modeling of other environmental situations, a possibility provided for by the internal structures (see Section 4.2). So, while ESs could be called minimal representations based on our revised notion of standing in, they are still a lesser form of representation compared with the inner ones. However, note that more complex external representations such as visualizations and physical models of chemical structures would meet the two stronger criteria, and would qualify for the label *representation*. This is broadly in line with the common use of the term.

4.2 Internal Traces and the Simulation/ Enaction Model

In this section we describe the highly debated Simulation/enaction model of cognition, and how our model of internal structures supports one form of simulation/ enaction and explains its origins. To avoid confusion with the simulation we implemented, we use Simulation with a capital S when discussing this proposed cognitive mechanism.

In general, Simulation models of cognition propose that neural structures responsible for action and/ or perception are recruited in the performance of cognitive tasks (such as language processing or observing another agent execute an action). Such a recruitment process is indicated by experiments (for reviews, see Svenson & Ziemke, 2004; Brass & Heyes, 2005). This evidence is used to make the argument that different aspects of cognition involve a virtual running of actions (Metzinger & Gallese, 2003). Of particular significance is the claim that such Simulation/enaction grounds symbols and other representations (i.e. provides their content; see, for instance, Barsalou, 1999, 2003). The Simulation view is a rapidly developing theoretical framework in cognitive science (Clark & Grush, 1999; Hesslow, 2002; Grush, 2004), and is used to explain cognitive processes ranging from perception to language, reasoning and theory of mind phenomena (Metzinger & Gallese, 2003; Svenson & Ziemke, 2004).

The cognitive mechanism of Simulation is considered to involve "re-enactments of states in modality-specific systems" (Barsalou, Simmons, Barbey, & Wilson, 2003), as against non-Simulation models, which involve "redescriptions of states in amodal representational languages" (Barsalou et al., 2003). The central distinction is between re-enactment of actions and redescription using symbols. Simulation is considered to involve enactment or "acting out" an experience or action to cognize a state, while non-Simulation is considered to involve (just) retrieval and manipulation of descriptive symbols, as in performing logic or arithmetic. This dichotomy presents two ends of a continuum of processes, as there could be in-between processes that involve both.

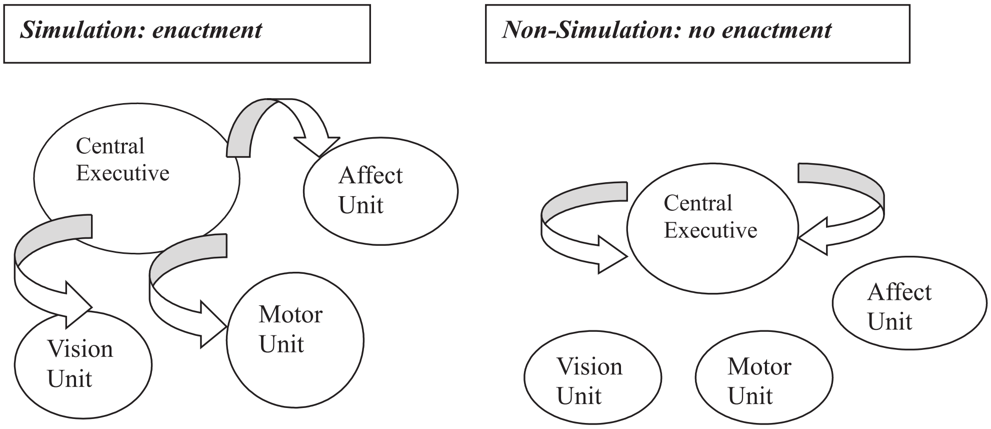

Figure 9 In the Simulation mechanism (left), the central executive is considered to pass processing of cognitive tasks onto the different component neural units, including the motor one, resulting in a process that is almost equivalent to the embodied agent acting in the world. In non-Simulation processing, the central executive is considered to process stored representations of the world by itself, with minimal or no input from the component neural units. This results in a disembodied process that is detached from the world. These two ways of processing (modal and amodal) need not be mutually exclusive and could be considered two ends of a continuum.

A crude example to illustrate the two processes would be two ways of remembering an accident. In the first case of remembering, the event is acted out in the mind, and results in bodily states associated with the event, such as shaking and crying. The other way to remember the event would be as images, without any acting out of the event, and therefore without the associated body states. Because the former involves acting out, it leads to changes in the perception and action modules of the brain associated with the actual experience of the event, so it is modality-specific (modal approach, in Barsalou's terminology). The latter does not involve acting out of the memory, just a retrieval (and/or manipulation) of stored images. This mechanism thus represents an amodal approach, in Barsalou's terminology. Figure 9 captures the distinction using traditional modules used in cognitive psychology.

There are two major types of Simulation/enaction identified in the literature. The two are closely connected, but we treat them as separate for the purposes of our discussion. The first involves enacting or running actions virtually while performing cognitive tasks such as processing verbs. That is, brain areas that are involved when actually performing the action associated with the verb (say, *chewing*) are implicitly activated while processing a representation associated with that action (the verb *chew*). Such implicit activation of action areas also occurs when an agent observes another agent performing an action (Brass & Heyes, 2005). We call this type of virtual enaction Simulation-R, for Simulation linked to representations.

Evidence in support of this Simulation mechanism comes from recent work in neuroscience, which shows that action areas are activated while observing (i.e. representing) an action, and also during linguistic processing. Gallese, Ferrari, Kohler, and Fogassi (2002) report that when we observe goal-related behaviors executed by others (with effectors as different as the mouth, the hand, or the foot), the same cortical sectors are activated as when we perform the same actions. Whenever we look at someone performing an action, in addition to the activation of various visual areas, there is a concurrent activation of the motor circuits that are recruited when we ourselves perform that action. We do not overtly reproduce the observed action, but our motor system acts as if we were executing the same action we are observing. This effect exists in monkeys as well, and has been replicated across a series of studies (see particularly the work on mirror neurons and canonical neurons; Hurley & Chater, 2005 provide a good review).

A similar process of Simulation of actions linked to representations has recently been demonstrated in language understanding. Bergen, Chang, and Narayan, (2004) report an imaging study where subjects performed a lexical decision task with verbs referring to actions involving the mouth (e.g. *chew*), leg (e.g. *kick*) or hand (e.g. *grab*), and areas of motor cortex responsible for mouth/leg/hand motion displayed more activation, respectively. It has also been shown that passive listening to sentences describing mouth/leg/hand motions activates different parts of pre-motor cortex.

The second type of Simulation discussed in the literature is more complex, and involves predicting another agent's behavior by virtually enacting the other agent's system states using one's own system as a proxy. This notion of Simulation is mostly found in the theory of mind literature (see Nichols, Stich, Leslie, & Klein, 1996). A closely related notion of Simulation is the virtual enaction of another agent's actions across time using one's own system, and then mutating these actions to generate alternatives to reality. This notion of Simulation is found in counterfactual thinking literature (see Kahneman & Tversky, 1982). Although not discussed much in the literature, such Simulation of system states could also be used to test alternatives to one's own current state (for instance, what would be my state at time T if I perform action X?). We call this type of virtual enaction to predict future system states (one's own or others') Simulation-S, for Simulation of system states.

There is only indicative evidence that this type of system-level enaction can allow agents to judge other agents', or one's own, system states. For such judgments, there should exist a system-level equivalence between an action and its Simulation. Svenson and Ziemke (2004) review three sources of evidence supporting such a system-level equivalence: mental chronometry, autonomic responses and neuroimaging experiments. Mental chronometry experiments show that the time to mentally execute actions closely corresponds to the time it takes to actually perform them. Autonomous response experiments show that responses beyond voluntary control (such as heart and respiratory rates) are activated by motor imagery, to an extent proportional to that of actually performing the action. Neuroimaging experiments show that similar brain areas are activated during action and motor imagery of the same action. Beside this evidence that supports action–Simulation equivalence at the system level, there are a whole host of theoretical arguments that support simulating of other agents' system states to predict their behavior (see Nichols et al., 1996 for a review). The literature on motor imitation also indicates that action observation could lead to a judgment of another agent's system state (Brass & Heyes, 2005; Hurley & Chater, 2005).

While these two Simulation mechanisms (Simulation-R, Simulation-S) are used to explain a range of cognitive phenomena, two aspects of the Simulation idea remain unclear.

1. What internal mechanisms lead to the origin of such enactable internal structures?
2. What is the nature of internal structures that support such virtual enaction?

We argue below that our model of stored internal traces provides tentative answers to both these questions, and thus provides an evolutionary basis to the Simulation model.

4.2.1 Simulatable Content We begin our discussion with Simulation-R, the idea that while processing a given internal representation (such as the verb *chew*), brain systems associated with performing actions related to that representation (such as the action of chewing) are also activated.

Our agent in the second experiment develops internal traces of the world using a feedback system based on actions and the cognitive load associated with actions. Values are initially randomly stored in an internal neural network environment, and then they are learned to be systematically stored, based on the feedback of cognitive load. As observed in Section 4.1.1, the systematically stored values are tightly coupled to actions; they are nothing but a thread that links actions that lower cognitive load. The systematically stored elements thus contain action information. That is, if zero is stored at target always, this means storing zero, sensing zero, and executing the action associated with zero lowered cognitive load. The sensing of the zero is thus not just a sensing of the zero, but a sensing of the actions and cognitive load associated with zero. These action components are implicitly activated when zero is sensed.

At a high level, it could be argued that such a proto-representation emerging out of actions encapsulates information about actions and cognitive load, because there is nothing else contributing to such an internal trace. This means such a representation supports the Simulation of actions related to the representation, because the representation is a task-specific structure that emerges out of actions and feedback based on actions. This possibility for Simulation does not exist if representations are considered as static structures learned in a mirror-like fashion, with no link to actions or system states such as cognitive load. For instance, if the verb *chew* is just captured and stored (in either word-like or image-like fashion), it is hard to see how (and why) processing *chew* leads to Simulation of chewing. The same applies to a stand-alone neural network that learns to categorize inputs about an action as *chew*. In contrast, task-specific internal structures are stored by our agents while acting in an environment. The structures arise out of actions and system states, and they therefore naturally support virtual enaction of those actions.

The above is a high-level view. To obtain a more detailed sense of the link between Simulation and the internal trace generation process, we have to examine the nature of the Q-learning algorithm. One way to think of Q-learning is to think of "pretend play" by novice chess players, where they try out potential moves. The organism tests the environment with different potential actions to see what reward that particular environment provides for that particular action. However, for both the novice chess player and the Q-learning system, it is not the immediate reward for that action which is important. Instead, it is the long-term reward (Q) that is important. Using the Q function is equivalent to a novice chess player who tries out the move of taking a knight with his queen, and then looks at the new board position and has the feeling of "that looks dangerous, I'd better not do that". Importantly, this Q function is constantly being updated by the results of all of an agent's actions in the world. Indeed, Q-learning systems tend to try out exploratory actions to gather information about what rewards will be in unknown situations.[7]

Furthermore, like the novice chess player, the Q-learning algorithm only enacts actions one step ahead, but through the use of the Q function, its evaluation of how good that step is includes the whole future set of actions, because the Q function approximates the possible outcome of an entire range of state-action combinations. Instead of developing an estimate of rewards for a single action, the Q function can be thought of as perturbing the agent-environment system, and then developing an estimate of the reward structure of these perturbations as they propagate. This means it can look ahead (i.e. test run) only one step, but the output of that test run provides an estimate of how the system as a whole would perform many steps into the future, and the reward structure after that time. Once the Q function is developed, the agent still technically looks ahead only one step, but it can be considered to implicitly run many states ahead.

It can be seen from the above description that the Q-learning algorithm is performing a rudimentary form of Simulation-S; it is evaluating possible alternative system states by enacting them using the agent's own system. While learning, the agent is simulating itself and its own interactions with the world. In our second experiment, this means the agent is able to simulate itself, its interactions with the world, and its own modifications of its own memory. It is this Simulation-S that allows the system to learn to generate internal structures.

This means we assume Simulation-S, and a simple version of it is built into our model. The proto-representations are a product of this basic Simulation-S process. However, this process illustrates something important: not only do the proto-representations in our model implicitly contain action information (as they arise out of actions that lower cognitive load), but the Q-learning system also virtually enacts these actions to judge cognitive load. This is because the internal structures are test run for their reward structure. So the proto-representations in our model not only support enaction linked to representations, but also provide a working model of this enaction process.

However, note that the above explicit enaction process is executed primarily when the system learns, and not when the proto-representations are used by the agent (to choose between target and home). Any enaction that occurs post-learning is only implicit (i.e. only in the sense that the proto-representations are tightly coupled to actions and action-related information, and these actions and information are activated when the representations are sensed). This implicit enaction may be similar to Simulation, in which case the Simulation mechanism arises out of learning. A similar idea is proposed by Hurley and Chater (2005) and the associative sequence learning model of imitation (see Brass & Heyes, 2005; Heyes, in Hurley & Chater, 2005).

Given this relation between learning and simulation, one possible way of interpreting the role of Simulation-R in grounding content could be as follows.

When Simulation-R (say enacting chewing while processing the word *chew*) occurs in a system, it is not grounding the content of *chew* directly. The enaction is part of the learning process (i.e. the process that led to the storing and use of the word *chew*), which is what grounds the content of the word, by activating the contexts, actions and environmental conditions linked to that trace. In this interpretation, the learning process is always running in the background, as agents in dynamic environments cannot afford to stop learning. The Simulation mechanism is a way of linking to this learning process, a type of "running the learning in miniature", as done by our agents when they access the internal traces.

The constant learning process is particularly true in our model, where the proto-representations, actions and cognitive load are tightly coupled. For instance, some changes in the environment could make the stored structures useless for our agent, and the agent will then have to reconfigure the link between actions and the stored structures in the internal network. If the learning system is activated when the trace is accessed, it will allow the agent to use the most current relation(s) between the trace, the world and the agent's system. In this view, if Simulation plays any role in the grounding of representations, this role comes from the learning mechanism, which links internally stored structures with (i) entities in the world, (ii) environmental conditions, and (iii) the agent's task and biological needs. Such a (useful) tight link between learning, use of internal structure, and simulation argues against the full decoupling required by Clark and Grush (1999) for an internal structure to be a full-blooded representation.

A related implication of this model of enaction of traces is that Simulation-S is more basic than Simulation-R, as the former type of Simulation leads to internal structures that support the latter type of Simulation. This could mean that at least part of the enaction that occurs during Simulation-R is related to system states that led to the learning of that representation. In this interpretation, if Simulation is considered to ground representations (i.e. provide their content), at least part of that content relates to system states, particularly cognitive load. (This ties in very well with our claim in Section 4.1 that aboutness of internal traces is partly a mechanism to reduce cognitive/energy load.) The task-specific nature of internal traces plays a central role in this view, as the Simulation capability of traces arises out of action-driven learning linked to tasks. This means task-specificity and action-driven learning of internal traces would need to be central components of any project that seeks to use Simulation to ground representational content.

5 Conclusion

We have presented two proof-of-concept models which show that the generation of both external and internal structures that lower cognitive load could be learned within lifetime, using the same learning mechanism, based just on feedback of cognitive load. These models facilitate an integration of the symbolic and situated views of cognition, and develop a possible mechanism supporting the extended mind thesis. They also present an implementation of the origin of enactable content, and raise questions about the viability of the one-to-one model of internal representation.

Notes

[1] The source code for both the simulations (written in Python) can be downloaded from http://www.carleton.ca/ics/ccmlab/epistemic.html.

[2] The term "tiredness" in the high-level model indicates the "felt" quality of the feedback in organisms, which allows tracking of the cost using affect (i.e. without using a separate computational module that tracks cost).

[3] The distinction between physical and cognitive congeniality is quite thin ("shades off"; see Godfrey-Smith, 1994) at the level of lower-level organisms. Avoiding cognitive effort usually means avoidance of search; at best this can be viewed as indirect physical congeniality. Avoiding physical effort is more direct, as in the case of pulling a grain from the side, instead of the front.

[4] The genetic algorithm used involved a look-up table genome, indicating which action to perform for each of the 192 possible sensory states. Mutations consisted of randomly changing exactly one item in the genome, and crossover was uniform. Extrema selection (Stewart, 2001) with a threshold of 90% was used to increase evolution speed along neutral networks. Population size was 50 and the system ran for 300 generations.

[5] Q-learning (for both the external and internal cases) was performed using a standard look-up table memory. The exploration rate (ε, the chance of performing an action at random instead of the system's best guess) was 0.1, the

learning rate (α, the amount by which to change the internal look-up table values) was 0.2, and the discounting rate (λ, the geometric reduction of importance of future rewards) was 0.95. Q-learning maintains a table of estimated reward values for taking each possible action in each possible sensory state, and is represented by $Q_{S,A}$. These values are all initially zero. As experience in the world occurs, and rewards/punishments are received, the values are changed. To update, we first calculate $r + \lambda \bullet \max Q_{S2}$ (where r is the received reward, λ is the fixed discounting rate, and $\max Q_{S2}$ is the largest value found looking at all the actions that could be taken from the sensory state the agent finds itself in after performing the action.) This value is then combined with the old prediction ($Q_{S1,A1}$) using the learning rate (α), resulting in the following formula:

$$Q_{S1,A1} \leftarrow (1 - \alpha)\, Q_{S1,A1} + \alpha(\mathrm{r} + \lambda \bullet \max Q_{S2}).$$

To choose an action to perform, the system simply takes the current state and looks at each possible action that could be performed. The action with the highest Q value is chosen 1 – ε of the time. In the remaining ε times, a random action is chosen.

6 The neural network was a feedforward multilayer perceptron, trained using back-propagation of error (Rumelhart, Hinton, & Williams, 1986). It had three input nodes (for the three values of the sensory state, scaled to be between –1 and 1), one output node, and three hidden nodes. The activation function for all nodes was the hyperbolic tangent, and the learning rate (α) was 0.2. To handle the feedback between the output value and the input state, the network was run 100 times.

7 Interestingly, such tests are known to exist in the animal world. Curio (1976) reports that most animals that predate on herds make a “test attack” to identify animals whose ability to run away is insufficient to protect them. In such cases, the actions in the world are not “real”, but “tests”, or “simulated” actions. Also, the organism uses itself and the environment as a “test bed” or “simulation environment” to judge the quality of its own actions.

Acknowledgments

We thank Dr. Andrew Brook for clarifying and sharpening the philosophical ideas presented here. Thanks are also due to Babak Esfandiari, Robert West, David Kirsh, Narayanan Srinivasan, Thomas Hills and Jennifer Schellinck for critical review and feedback. We are grateful for the clear pointers and suggestions from the three reviewers and the action editor, which have contributed significantly to the paper.

References

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences*, *22*, 577–609.

Barsalou, L. W. (2003). Situated simulation in the human conceptual system. *Language and Cognitive Processes*, *18*, 513–562.

Barsalou, L. W., Simmons, W. K., Barbey, A. K., & Wilson, C. D. (2003). Grounding conceptual knowledge in modality-specific systems. *Trends in Cognitive Sciences*, *7*, 84–91.

Beer, R. D. (2003). The dynamics of active categorical perception in an evolved model agent. *Adaptive Behavior*, *11*, 209–243.

Bergen, B., Chang, N., & Narayan, S. (2004). Simulated action in an embodied construction grammar. In K. D. Forbus, D. Gentner, & T. Regier (Eds.), In *Proceedings of the 26th Annual Meeting of the Cognitive Science Society*, Chicago, IL. Hillsdale, NJ: Lawrence Erlbaum.

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence: from natural to artificial systems*. New York: Oxford University Press.

Boyd, R., & Richerson, P. J. (1996). Why culture is common but cultural revolution is rare. *Proceedings of the British Academy*, *88*, 73–93.

Bradbury, J. W., & Vehrencamp, S. L. (1998). *Principles of animal communication.* Sunderland, MA: Sinauer Associates.

Brass, M., & Heyes, C. (2005). Imitation: Is cognitive neuroscience solving the correspondence problem? *Trends in Cognitive Sciences*, *9*, 489–495.

Brooks, R. (1991). Intelligence without representation. *Artificial Intelligence*, *47*, 139–159.

Buason, G., Bergfeldt, N., & Ziemke, T. (2005). Brains, bodies and beyond: Competitive co-evolution of robot controllers, morphologies and environments. *Genetic Programming and Evolvable Machines*, *6*, 25–51.

Camazine, S. (1991). Self-organizing pattern formation on the combs of honey bee colonies. *Behavioral Ecology and Sociobiology*, *28*, 61–76.

Chandrasekharan, S. (2005). *Epistemic structure: An inquiry into how agents change the world for cognitive congeniality.* Ph.D. dissertation, Carleton University, Ottawa, Canada. Retrieved 5 December 2006 from http://www.carleton.ca/iis/TechReports/files/2005-02.pdf

Chandrasekharan, S. (2006). Money as epistemic structure. *Behavioural and Brain Sciences*, *29*, 183–184.

Clark, A., & Chalmers, D. (1998). The extended mind. *Analysis, 58*, 7–19.

Clark, A., & Grush, R. (1999). Towards a cognitive robotics. *Adaptive Behavior*, *7,* 5–16.

Clark, A., & Toribio, J. (1994). Doing without representing? *Synthese*, *101*, 401–431.

Curio, E. (1976). *The ethology of predation*. New York: Springer-Verlag.

Dennett, D. (1975). Why the law of effect will not go away. *Journal of the Theory of Social Behaviour*, *5*, 179–187.

Dennett, D., & Haugeland, J. (1987). Intentionality. In R. L. Gregory (Ed.), *The Oxford companion to the mind*. Oxford: Oxford University Press.

Fodor, J., & Pylyshyn, Z. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, *28*, 3–71.

Galantucci, B. (2005). An experimental study of the emergence of human communication systems. *Cognitive Science*, *29*, 737–767.

Gallese, V., Ferrari, P. F., Kohler, E., & Fogassi, L. (2002). The eyes, the hand and the mind: Behavioral and neurophysiological aspects of social cognition. In M. Bekoff, C. Allen, & M. Burghardt (Eds.), *The cognitive animal (*pp. 451–462). Cambridge, MA: MIT Press.

Godfrey-Smith, P. G. (1994). *Complexity and the function of mind in nature*. Cambridge, MA: MIT Press.

Gould, J. L. (1990). Honey bee cognition. *Cognition*, *37*, 83–103.

Griffiths, D. P., Dickinson, A., & Clayton, N. S. (1999). Declarative and episodic memory: What can animals remember about their past? *Trends in Cognitive Science*, *3*, 74–80.

Grush, R. (2004). The emulation theory of representation: Motor control, imagery, and perception. *Behavioral and Brain Sciences, 27*, 377–442.

Guilford, T., Roberts, S., & Biro, D. (2004). Positional entropy during pigeon homing II: Navigational interpretation of Bayesian latent state models. *Journal of Theoretical Biology*, *227*, 25–38.

Henry, J. D. (1977). The use of urine marking in the scavenging behaviour of the red fox (*Vulpes vulpes*). *Behaviour*, *62*, 82–105.

Hesslow, G. (2002). Conscious thought as simulation of behaviour and perception. *Trends in Cognitive Sciences*, *6*, 242–247.

Hills, T. (2006). Animal foraging and the evolution of goal-directed cognition. *Cognitive Science, 30*, 3–41.

Hurley, S., & Chater, N. (2005). *Perspectives on imitation: From neuroscience to social science* (Vols 1–2). Cambridge, MA: MIT Press.

Hutchins, E. (1995a). *Cognition in the wild*. Cambridge, MA: MIT Press.

Hutchins, E. (1995b). How a cockpit remembers its speeds. *Cognitive Science*, *19*, 265–288.

Kahneman, D., & Tversky, A. (1982). The simulation heuristic. In D. Kahneman, P. Slovic, & A. Tversky (Eds.), *Judgment under uncertainty: Heuristics and biases* (pp. 201–208). New York: Cambridge University Press.

Kart-Teke, E., De Souza Silva, M. A., Huston, J. P., & Dere, E. (2006). Wistar rats show episodic-like memory for unique experiences. *Neuorbiology of Learning and Memory*, *85*, 173–182.

Kirby, S. (2002). Natural language from artificial life. *Artificial Life*, *8*, 185–215.

Kirsh, D. (1995). The intelligent use of space. *Artificial Intelligence*, *73,* 31–68.

Kirsh, D. (1996). Adapting the environment instead of oneself. *Adaptive Behavior*, *4,* 415–452.

Kirsh, D., & Maglio, P. (1994). On distinguishing epistemic from pragmatic action. *Cognitive Science*, *18*, 513–549.

Laland, K. N., Odling-Smee, F. J., & Feldmann, M. W. (2000). Niche construction, biological evolution and cultural change. *Behavioral and Brain Sciences*, *23*, 131–175.

Mandik, P., & Clark, A. (2002). Selective representing and world making. *Minds and Machines*, *12*, 383–395.

Metzinger, T., & Gallese, V. (2003). The emergence of a shared action ontology: Building blocks for a theory. *Consciousness and Cognition*, *12*, 549–571.

Miller, M., & Wehner, R. (1988). Path integration in desert ants, *Cataglyphis fortis*. *Proceedings of the National Academy of Sciences, USA*, *85*, 5287–5290.

Millikan, R. G. (1993). *White Queen psychology and other essays for Alice*. Cambridge, MA: MIT Press.

Millikan, R. G. (1996). Pushmi–pullyu representations. In L. May & M. Friedman (Eds.), *Mind and morals* (pp. 145–161). Cambridge, MA: MIT Press.

Nakamura, M., & Kurumatani, K. (1996). Formation mechanism of pheromone pattern and control of foraging behavior in an ant colony model. In C. G. Langton and K. Shimohara, (Eds.)*, Artificial life V: Proceedings of the 5th International Workshop on the Synthesis and Simulation of Living Systems*. Nara, Japan. Cambridge, MA: MIT Press.

Nichols, S., Stich, S., Leslie, A., & Klein, D. (1996). Varieties of off-line simulation. In P. Carruthers & P. Smith (Eds.), *Theories of theories of mind* (pp. 39–74). Cambridge, UK: Cambridge University Press.

Odling-Smee, F. J., Laland, K. N., & Feldman, M. W. (2003). *Niche construction: The neglected process in evolution*. Princeton, NJ: Princeton University Press.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*, 533–536.

Seth, A. K. (2002). Agent-based modelling and the environmental complexity thesis. In B. Hallam, D. Floreano, J. Hallam, G. Heyes, & J. A. Meyer (Eds.), *From animals to animats 7: Proceedings of the 7th International Conference on the Simulation of Adaptive Behavior* (pp. 13–24). Edinburgh, UK. Cambridge, MA: MIT Press.

Silberman, S. (2003). The bacteria whisperer. *Wired*, (April), 104–108.

Sipper, M. (2001). On the origin of environments by means of natural selection. *AI Magazine*, *22* (4), 133–140.

Smolensky, P. (1989). Connectionist modeling: Neural computation/mental connections. In L. Nadel, L. A. Cooper, P. Culicover, & R. M. Harnish (Eds.), *Neural connections, mental computation* (pp. 49–67). Cambridge, MA: MIT Press.

Steels, L., & Belpaeme, T. (2005). Coordinating perceptually grounded categories through language: A case study for colour. *Behavioral and Brain Sciences, 28*, 469–529.

Sterelny, K. (2005). Made by each other: Organisms and their environment. *Biology and Philosophy*, *20*, 21–36.

Sterelny, K. (2006). Cognitive load and human decision, or, three ways of rolling the rock uphill. In S. Stich, S. Laurence, & P. Carruthers (Eds.), *The innate mind: Culture and cognition*. Cambridge, UK: Cambridge University Press.

Stewart, T. C. (2001). Extrema selection: Accelerated evolution on neutral networks. In *Proceedings of the IEEE Congress on Evolutionary Computation 2001.* Seoul, Korea. New York: IEEE Press.

Stewart, T., & Chandrasekharan, S. (2005). *Two cognitive descriptions of Q-learning*. Carleton University Cognitive Science Technical Report. Retrieved 5 December 2006 from http://www.carleton.ca/iis/TechReports/files/2005-03.pdf.

Stopka, P., & Macdonald, D. W. (2003). Way-marking behavior: An aid to spatial navigation in the wood mouse (*Apodemus sylvaticus*). *BMC Ecology*, *3* (3).

Susi, T., & Ziemke, T. (2001). Social cognition, artifacts, and stigmergy: A comparative analysis of theoretical frameworks for the understanding of artifact-mediated collaborative activity. *Cognitive Systems Research*, *2*, 273–290.

Svenson, H., & Ziemke, T. (2004). Making sense of embodiment: Simulation theories and the sharing of neural circuitry between sensorimotor and cognitive processes. In K. D. Forbus, D. Gentner, & T. Regier (Eds.), *Proceedings of the 26th Annual Meeting of the Cognitive Science Society.* Chicago, IL. Hillsdale, NJ: Lawrence Erlbaum.

Tani, J., & Nolfi, S. (1999). Learning to perceive the world as articulated: An approach for hierarchical learning in sensory-motor systems. *Neural Networks*, *12*, 1131–1141.

Tesfatsion, L. (2002). Agent-based computational economics: Growing economies from the bottom up. *Artificial Life*, *8*, 55–82.

Todd, P. M., & Miller, G.F. (1999). From pride and prejudice to persuasion: Realistic heuristics for mate search. In G. Gigerenzer, P. M. Todd, & the ABC Research Group, *Simple heuristics that make us smart* (pp. 287–308). New York: Oxford University Press.

Tulving, E. (1983). *Elements of episodic memory.* Oxford: Clarendon Press.

van Dartel, M. F. (2005). *Situated rrepresentation*. Doctoral dissertation, Maastricht University, the Netherlands.

van Dartel, M. F., Postma, E. O., van den Herik, H. J., & de Croon, G. (2004). Macroscopic analysis of robot foraging behaviour. *Connection Science*, *16*, 169–181.

Watkins, C. (1989). *Learning from delayed rewards*. Doctoral dissertation, Department of Psychology, University of Cambridge, Cambridge, UK.

Zahavi, A., & Zahavi, A. (1997). *The handicap principle: A missing piece of Darwin's puzzle*. Oxford: Oxford University Press.

Ziemke, T., Bergfeldt, N., Buason, G., Susi, T., & Svensson, H. (2004). Evolving cognitive scaffolding and environment adaptation: A new research direction for evolutionary robotics. *Connection Science*, *16,* 339–350.

About the Authors

Sanjay Chandrasekharan holds an undergraduate degree in physics and a master's degree in communication from the University of Kerala, India. A chance encounter with *The Mind's I* introduced him to cognitive science. This led to a second master's degree, in linguistics (Jawaharlal Nehru University, New Delhi), and a Ph.D. in cognitive science (Carleton University, Ottawa, Canada). As a graduate student, he worked as a predoctoral fellow with the Adaptive Behaviour and Cognition Group (Max Planck Institute for Human Development, Berlin) and was a visiting researcher at the Institute of Information Technology, National Research Council, Canada. After graduation, he joined India's first cognitive science program (Center for Behavioral and Cognitive Sciences, University of Allahabad) as a faculty member. He is currently a visiting research scientist at the Interactive and Intelligent Computing Division, Georgia Institute of Technology, USA. His research focuses mainly on the internal mechanisms underlying situated and distributed cognition, particularly epistemic structures and epistemic actions. He is also interested in simulation/enaction theory, imitation, social cognition, technological applications of embodied cognition, and behavioral economics.

Terrence C. Stewart is completing his Ph.D. at the Institute of Cognitive Science at Carleton University, Canada. He received his M.Phil. in computer science and artificial intelligence at the University of Sussex, UK, and his B.A.Sc. in systems design engineering at the University of Waterloo, Canada. His work involves whole-agent modeling of both high- and low-level cognition, comparisons between modeling frameworks, philosophy of modeling, and making computational modeling accessible for a broader range of researchers. *E-mail*: terry@ccmlab.ca