# Experiments on the internalisation of language

Carlos Herrera Pérez
University of Ulster

euCognition Network Action 083-1
www.euCognition.org

## 1. Introduction

The internalisation of language is a thesis embraced by authors such as Vygotsky and Clark that considers public language to be an external cognitive resource that may be internalized. A characteristic phenomenon is described by Clark

“When the child, confronted by a tricky challenge, is ‘talked through’ the problem by a more experienced agent, the child can often succeed at tasks which would otherwise prove impossible (think of learning to tie your shoelaces). Later on, when the adult is absent, the child can conduct a similar dialogue, but this time with herself. But even in this latter case, it is argued, the speech (be it vocal or `internalized') functions so as to guide behavior, to focus attention, and to guard against common errors. In such cases, the role of language is to guide and shape our own behavior -- it is a tool for structuring and controlling action and not merely a medium of information transfer between agents.” (Clark, 1998)

The following experiments aim to explore how the process of internalisation might get underway. Using these minimal simulations, we hope to gain a better conceptual grasp on how it might be possible for language to take up a cognitive role. We conceive this process as involving the immersion of a developing agent in a language mediated leaning task. The internalisation of language can thus be explained in the context of substituting external instructions with auto-generated ones for self-guidance.

## 2. Experimental background

Evolution and learning are major design techniques for highly distributed artificial systems, and have been widely applied in the fields of neural networks and neuro-robotics. Artificial neural networks are often defined by a network architecture, weighted connections between neurons and neuron activation. A neural network may use different rules, such as the Hebbian rule or back-propagation, to alter the weights between different neurons, depending on the current activation of the network. This method has proven extremely successful in adjusting the network functionality to different patterns of stimulation, and, in the case of robots and embodied networks, interaction with the environment.

Artificial evolution of neural networks, on the other hand, often assumes that weights between neurons are fixed during lifetime, and selects the weights that produce best overall results, given some criteria, through a selection mechanism. Artificial evolution, nevertheless, is not restricted to the selection of weights. It could be applied to the selection of any relevant parameter, for instance the network architecture, embodiment in the case of robots, or learning rules in the case of plastic neurocontrollers.

It would be a mistake to conflate artificial evolution and learning with the phenomena as present in biological systems. Yamauchi and Beer (1994), for example, have shown how evolved non-plastic controllers can appear to perform reinforcement learning. Neural plasticity, on the other hand, can be used to shape the overall features of an agent, based on patterns of sensory-motor coordination with no fitness evaluation or genetic mutation (as in e.g. Morse, submitted). Even though some important features are present, the natural phenomena are still vaguely understood. One of the differences is that, in natural agents, evolution and learning are not two competing adaptive mechanisms, but necessarily co-occurring phenomena. The distinction between mechanisms responsible for generating behaviour and those responsible for learning “… is difficult to defend biologically, because many of the same biochemical processes are in involved in both processes” (Yamauchi and Beer 1994, p.243).

The relationship between evolution and learning in neuro-robotics has been investigated by several researchers. For instance, (Nolfi, Elman, & Parisi, 1994) demonstrates how a population selected on one task may increase their performance when, at an individual level, an individual learns a task differently to the one it is selected for. The experimental setup in this paper follows up the one described in Floreano and Mondada 1996. Rather than evolving the weights of a neural network, the authors evolve a genotype that encodes how the network (with a fixed architecture) should modify its weights during lifetime. Concretely, they allow evolution to mutate the learning properties of each individual synapse, as what type of Hebbian learning rule it uses and what is the learning rate.

## 3. Robot and architecture

The following experiments were carried out in a modified version of Evorobot, developed by Stefano Nolfi. Evorobot is a Khepera simulator that incorporates a genetic algorithm and neural networks .

The agent's controller has the following structure. The network receives sensory inputs from infrared (8 sensors) and light (front and rear) sensors. The network has a number of internal neurons (8 or 12), and two motor outputs (left and right motors). The network also receives three extra inputs (intended to represent linguistic instructions from an external agent, in this case the experimenter), and produces three extra outputs, which can be used to replace the external instructions. The linguistic inputs can take the values of either 0 or 1.

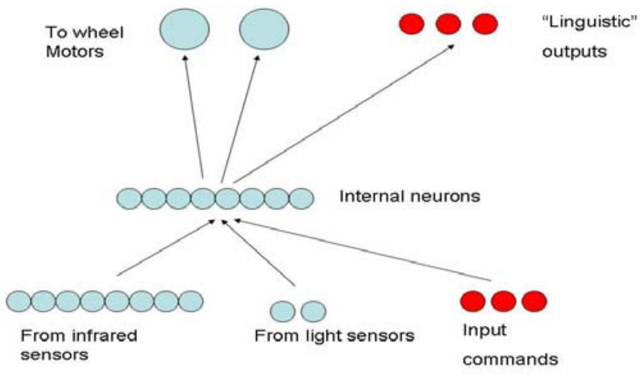



**Figure 1. Controller architecture**

The method employed in the following experiments replicates the one reported in (Floreano and Mondada 1996), where the authors investigate how neural mechanisms underlying ontogenetic learning are themselves developed and shaped by the evolutionary process. Here we turn this method to a novel domain.

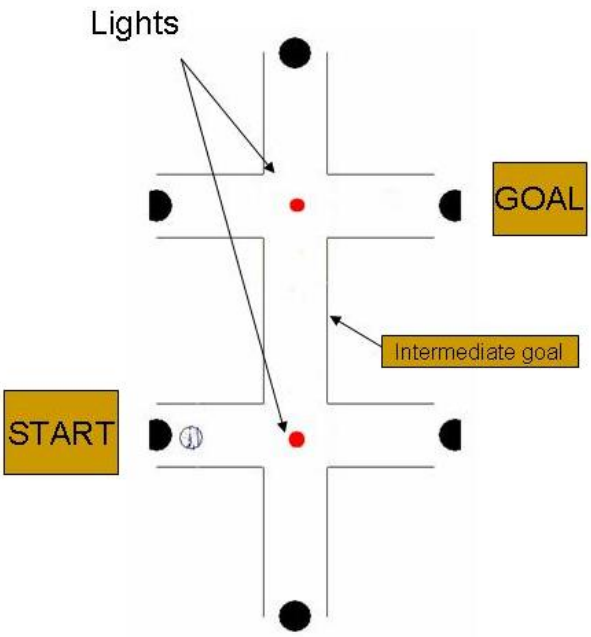
A simple genetic algorithm is used to generate new genotypes. At the beginning of each epoch a chromosome is decoded into the corresponding neural controller, and its performance evaluated. Each of the genotypes contains enough information to generate a controller, given the architecture represented in Figure 1. Where in many experiments this is achieved by encoding in the chromosome the weight that defines each synapse, this method the genotype encodes the neural architecture and learning rules of each of the 144 synapses.

The four allowed learning rules were: pure Hebbian, Postsynaptic, Presynaptic and Covariance(based on Willshaw & Dayan, 1990). The learning rate could take four different values {0.0, 0.3, 0.7, 1.0}. The two other properties are whether each synapse is excitatory or inhibitory and whether it drives or modulates the postsynaptic neuron., and the weights of the network synapses are initialised to small random values. A chromosome would therefore contain 6 bits per synapse (2 for rule, 2 for rate, and 2 more for the other 2 properties)

Given the initial random weight, each synapse changes its weight according to the conditions specified in the chromosome (with the exception that weights are constrained to a maximum of 1). More details on the method can be found in (Floreano & Mondada, 1996)

### 3.2 Environment and task

The environment is a maze consisting of two parallel corridors crossed by a transversal perpendicular corridor, as represented in the Fig 2. Corridors converge to two central areas illuminated by light bulbs, and, in the simulated environment, corridors are blocked with obstacles. At the beginning of its lifetime the robot is, placed at the west end of the bottom corridor, near the point marked as START, facing east. The task of the robot is to navigate towards the point marked as GOAL. Although different routes may be possible time constraints tend to impose a direct route: the task is therefore to navigate the corridor towards the first central area, turn left towards the second central area, to then turn right towards the east end of the top corridor.



During a first stage, in addition to the inputs from the sensors embodied in the robot, the robot receives three inputs which are intended to represent three commands, go to light, turn left and turn right. In other words, when we want the robot to perform one of these actions (which depends on the location of the robot), we activate one of the three linguistic input nodes (*input commands* in fig 1).[1] Given the inputs from sensory stimulation (infrared and light sensors) as well as the command inputs, the robot is let free to navigate the maze for a determinate amount of time.

**Figure 2. Maze which the robot needs to navigate.**

We call this phase *the scaffolded phase* because during its course external instructions are provided, and the network is trained on the task while being given external instructions. At the start of the scaffolded phase the network is initialised to small random weights, which are then updated at every time step following the evolved learning rules and rates. If, when the goal has been reached, the robot has successfully completed the scaffolded phase the weights are "frozen" to their current values and are not allowed to change anymore. Instructions also cease, and as an alternative to the command inputs, the state of the linguistic outputs (normalised to 0 or 1) take their place in the network structure. During *the autonomous phase* the navigation task is carried out in exactly the same manner as in the scaffolded phase but now without the benefit of external instructions to guide action.

During the scaffolded phase, individuals are rewarded for their ability to perform the first turn (1 point when the intermediate goal is reached), as well as reaching the GOAL state (3 points). This way, we encourage an incremental evolution of abilities. During the autonomous phase, rewards are increased to 5 and 6 points respectively. The fitness function is given in the following table

| | Scaffolded phase | Autonomous phase |
|---|---|---|
| Intermediate goal | 1 | 5 |
| GOAL | 2 | 6 |

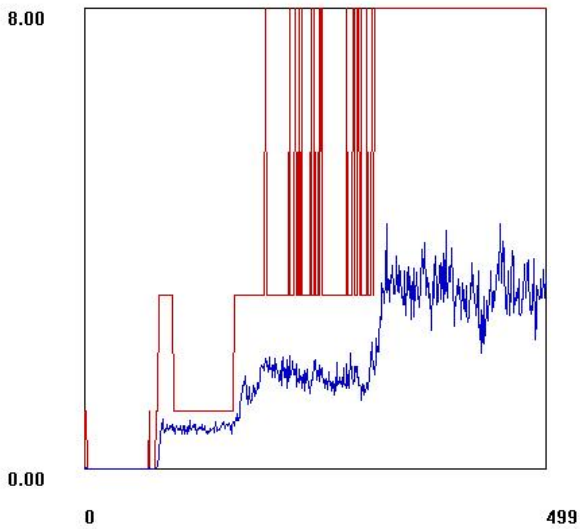



**Figure 3. Left, fitness assigned to each individual given its achivements. Right, evolution of fitness (maximum and average score per generation) over 500 generations in a typical evolutionary run.**

The rationale behind this fitness function is to allow the tracking of what tasks have been achieved, and to encourage agents to perform well on the autonomous phase. After several hundred generations, evolution consistently found genotypes that allowed for the learning task to be completed, as well as the first task of the autonomous phase, scoring a total of 8. In several occasions we found robots that reached the goal during the autonomous phase, scoring 14 points. These solutions, nevertheless, were not robust, as they did not allow replication, possibly because they could not overcome the randomisation of initial weights. Nevertheless, the experimental results show some interesting points that can inform current debates about how language takes up its cognitive role. In what follows we show graphical

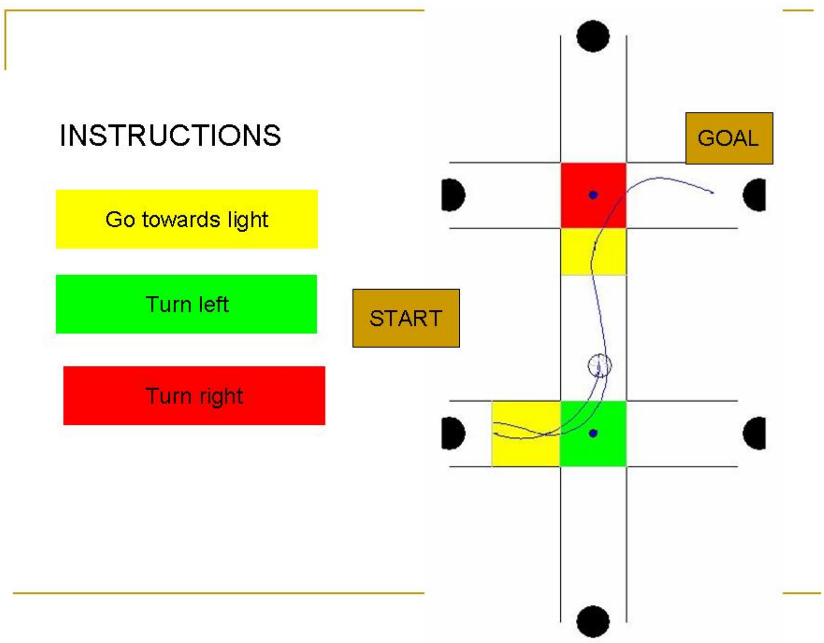



**Figure 4 – The task with instructions**

[1] As received by the robot, these signa… meaning rests on the potential groundi… ground such symbols insofar as instructions will be given when required in skilled interaction.

depictions of the robot in several trials as these help illuminate one possibility for this discussion.

## 4. Results

### 4.1 Scaffolded phase: grounding instructions in embodied interaction

Our analysis of the evolved behaviours starts by investigating what agents do during the scaffolded phase. As we can see from the fitness function (Figure 3), after about 200 generations, individuals consistently complete the task during the scaffolded phase. As we expected, the availability of external signals at the right time allows the agent to evolve a controller that takes the appropriate turns, reaching the goal in good time. Since the sensory inputs are practically the same at both crossroads (except differences due to the position of the robot), the turns are effected in response to the external instructions.

Figure 5 shows a breakdown of the behaviour at the two crossroads. At the left the simulated Khepera begins the scaffolded phase. Soon the robot receives a signal in the input node labelled *centre,* which is given in the scaffolded phase whenever we intend the robot to navigate to the light. As the Khepera enters the area in the centre of the maze, it receives a new instruction fed into the *left* node that can be glossed as turn left. The Khepera responds to this instruction (and the other data about the situation in which it is embedded) and turns left (Figure 5, left).

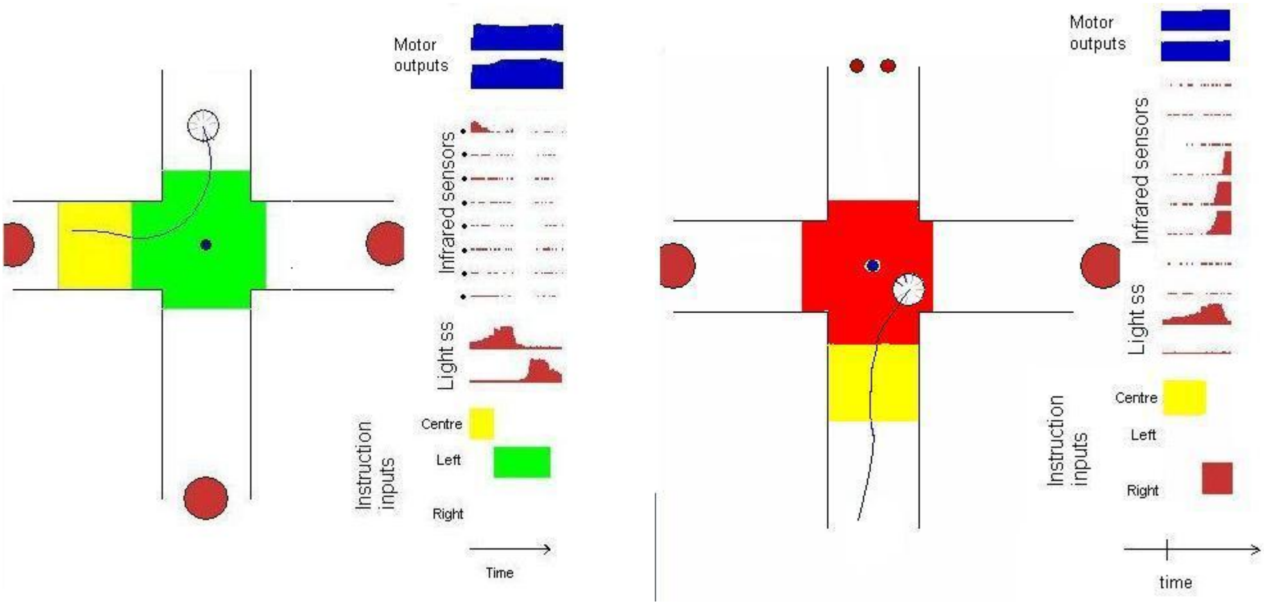


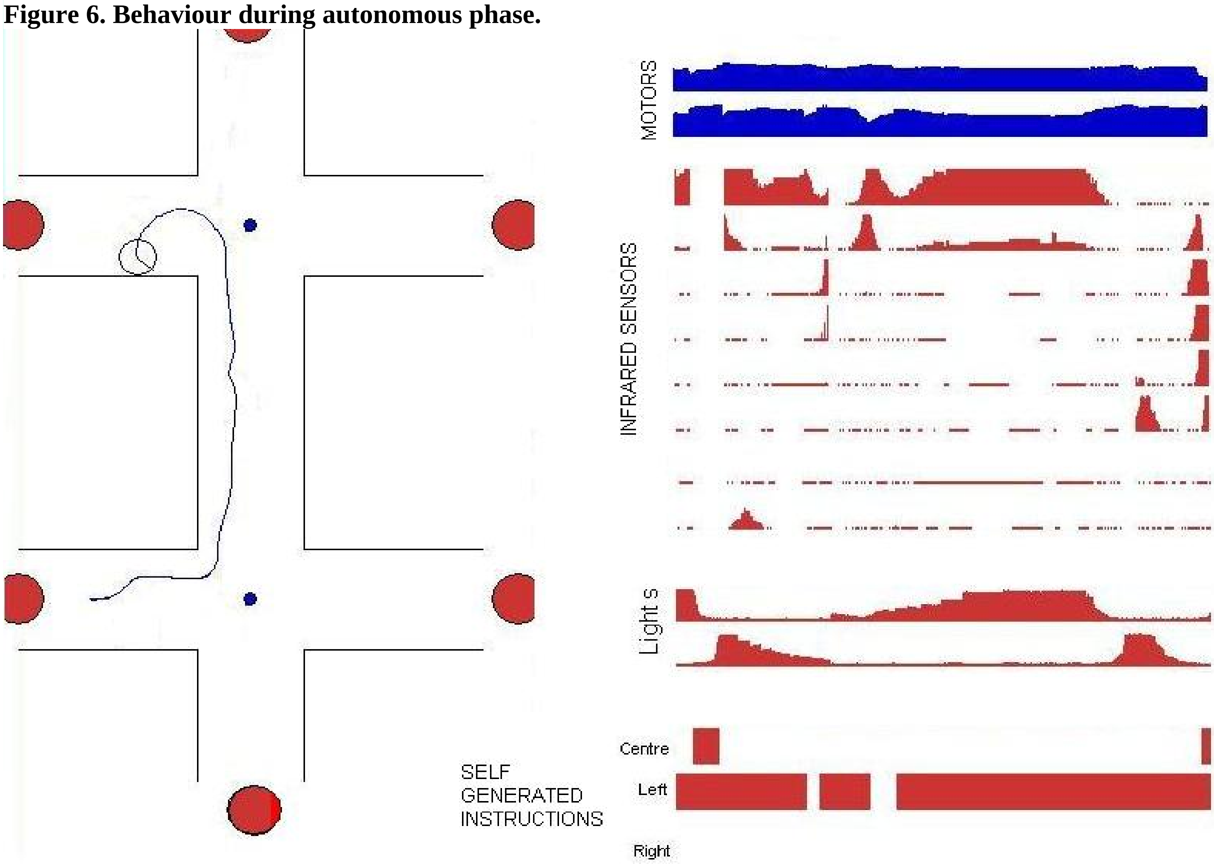

**Figure 5. Trajectory of the robot and activation of inputs / outputs during the two turns.**

Next (Figure 5, right) the robot faces the second crossroads, and gets an input once more signalling the robot to move towards the light. The robot moves forward and, as it approaches again the centre area, receives an instruction in the *right* node (the last externally generated signal it received), and it successfully responds to this by turning to the right.

### 4.2 Autonomous phase: Re-using commands to scaffold one's own behaviour

As we mentioned above, the main differences between the scaffolded phase and the autonomous phase are that during the latter weights are fixed to the last value of the scaffolded phase and that instructions are not given thereafter, instead replaced by an internally generated input stream. During the scaffolded phase, the weights of the neural network (initialised to random numbers between 0 and 1), are changed given the learning type and rate. Once the agent has reached the goal, there are no more external instructions or Hebbian learning. The weights are fixed but the linguistic outputs are now connected to the linguistic inputs. Agents now have the possibility of re-using these channels to structure and control their own ongoing activity. Any further 'instructions' they receive are internally generated out of the agents own ongoing dynamics.

**Figure 6. Behaviour during autonomous phase.**




The question that the experiments target is whether the internal signals produced by the robot will assist the robot in performing the right actions, or, in other words, whether the robot will talk to itself and find use for the commands that were externally given when each action was required in the scaffolding phase. It is not obvious that the robot should be able to do this as it has not had to learn to produce commands in the scaffolded phase, only respond to them. A further question is whether this re-use of instructions will allow the robot to robustly sequence its actions to complete the autonomous navigation task: turn left at the first crossroad, then right at the second one.

In figure 6 we can note that the robot produces two signals. Most of the time, it produces the turn left signal, while only at certain times it produces the go_to_light signal. This production allows the robot to perform the first turn to the left, achieving the intermediary goal. This answers the first question, whether the robot actions will be able to reproduce self-generated signals to control and perturb its own behaviour and indeed the capacity to produce one or two ‘utterances’ for self-control is a robust finding over many experimental runs. It also demonstrates a basic proof of concept in the realm of autonomous agents: the possibility that externally generated commands used to structure behaviour can be appropriated and turned to the autonomous self-structuring: the process that Vygotsky referred to as internalisation.

Regarding the second question, unfortunately, the controllers could not consistently solve the whole navigation task to reach the goal. The robots continue delivering the same instruction (go_left) at the next stage of its task, which causes the robot to crash into the wall to its left. This is not to say however a different learning regime could not manage to produce agents that learn to complete the whole navigation task, and this is a task for further research.

## References

Clark, A. (1998). Magic Words: How Language Augments Human Computation. In P. Carruthers & J. Boucher (Eds.), *Language and Thought. Interdisciplary Themes* (pp. 162 - 183). Oxford: Oxford University Press.

Clark, A. (2002). Anchors not inner codes, coordination not translation. *Behavioral and Brain Sciences, 25*.

Clark, A. (2003). *Natural Born Cyborgs: Minds, Technologies and the Future of Human Intelligence.* New York: Oxford University Press.

Clark, A. (2004). Is language special? Some remarks on control, coding, and co-ordination. *Language Sciences, 26*(6), 717-726.

Clark, A. (2006). Language, embodiment, and the cognitive niche. *Trends in Cognitive Sciences, 10*(8), 370-374.

Clark, A., & Chalmers, D. (1998). The Extended Mind. *Analysis, 58*, 10-23.

Clark, A., & Grush, R. (1999). Towards a Cognitive Robotics. *Adaptive Behavior, 7*(1), 5-16.

Clowes, R. W. (2007a). A Self-Regulation Model of Inner Speech and its Role in the Organisation of Human Conscious Experience. *Journal of Consciousness Studies, 14*(7), 59-71.

Clowes, R. W. (2007b). Semiotic symbols and the missing theory of thinking. *Interaction Studies, 8*(1), 105-124.

Deacon, T. W. (1997). *The Symbolic Species: The Co-Evolution of Language and the human brain*: The Penguin Press, Penguin Book Ltd.

Dennett, D. C. (1991). *Consciousness Explained*: Penguin Books.

Di Paolo, E. A., Noble, J., & Bullock, S. (2002). *Simulation models as opaque thought experiments.* Paper presented at the Artificial Life VII: Proceedings of the Seventh International Conference on Artificial Life.

Floreano, D., & Mondada, F. (1996). Evolution of plastic neurocontrollers for situated agents. *From Animals to Animats, 4*, 401-410.

Fodor, J. (1975). *The Language of Thought*. Cambridge, MA: MIT Press.

Fodor, J. (1983). *The Modularity of Mind*. Cambridge, MA: MIT Press.

Grush, R. (2004). The emulation theory of representation Motor control, imagery, and perception. *Behavioral and Brain Sciences, 27*(3).

Hermer-Vazquez, L., Spelke, E., & Katnelson, A. (1999). Sources of flexibilty in human cognition: Dual task studies of space and language. *Cognitive Psychology, 39*, 3-36.

Hesslow, G., & Jirenhed, D.-A. (2007). The inner world of a simple robot. *Journal of Consciousness Studies*.

Holland, O., & Goodman, R. (2003). Robots With Internal Models A Route to Machine Consciousness? *Journal of Consciousness Studies, 10*(4), 77-109.

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

Lindblom, J., & Ziemke, T. (2003). Social Situatedness of Natural and Artificial Intelligence: Vygotsky and Beyond. *Adaptive Behavior, 11*(2), 79-96.

Mirolli, M., & Parisi, D. (in press). Towards a Cognitive Robotics: The Role
of Language as a Cognitive Tool. *New Ideas in Psychology*.

Morse, A. (submitted). Dynamic Liquid Association: Complex Learning Without Implausible Guidance. *The Neural Networks Journal*.

Nolfi, S., Elman, J. L., & Parisi, D. (1994). Learning and Evolution in Neural Networks. Adaptive Behavior, Vol. 3, 1, pp.5-28. *Adaptive Behavior, 3*(1), 5-28.

Rumelhart, D. E., Smolensky, P., McClelland, D., & Hinton, G. E. (1986). Schemata and Sequential Thought Processes in PDP. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition 2* (pp. 7-57): Cambridge MA: MIT Press.

Sinha, C. (2006). Epigenetics, Semiotics, and the Mysteries of the Organism. *Biological Theory, 1*(2), 112-115.

Thompson, R. K. R., Oden, D. L., & Boysen, S. T. (1997). Language-naive chimpanzees (Pan troglodytes) judge relations between relations in a conceptual matching-to-sample task, J. *Exp. Psychol. Anim. Behav. Process, 23*, 31-43.

Thompson, R. K. R., Oden, D. L., & Boyson, S. T. (1998). Relations-Between-Relations: Language-Naive Chimpanzees (*Pan Troglodytes*) judge relations-between-relations in an abstract meaning task.

Wertsch, J. V., & Addison Stone, C. (1985). The concept of internalization in Vygotsky's account of the genesis of higher mental functions. In J. V. Wertsch (Ed.), *Culture, communication, and cognition* (pp. (pp. 162-179)). Cambridge, England: Cambridge University Press.

Wheeler, M. (2004). Is language the ultimate artefact? *Language Sciences, 26*(6), 688-710.

Willshaw, D. J., & Dayan, P. (1990). Optimal plasticity in matrix memories: What goes up must come down. *Neural Computation, 2*(85-93).

Yamauchi, B., & Beer, R. (1994). Sequential Behavior and Learning in Evolved Dynamical Neural Networks. *Adaptive Behavior, 2*(3), 219-246.

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, 4*(19), 339-350.

Ziemke, T., Jirenhed, D. A., & Hesslow, G. (2005). Internal Simulation of Perception: A Minimal Neuro-Robotic Model. *Neurocomputing, 28*, 85-104.