Categorisation through internal simulation of perception and behaviour

Michel van Dartel Eric Postma Jaap van den Herik

IKAT, Universiteit Maastricht, Maastricht, The Netherlands[1]

Abstract

The 'simulation hypothesis' is an intriguing explanation for cognition, and holds that 'thinking consists of simulated interaction with the environment' ([4], p.242). However, the neuroscientific proof for a simulation mechanism in the brain is indirect. In this paper we present a minimal-model approach to investigate the 'simulation hypothesis'. Our minimal model is called ACP* and is an extension of the Active Categorical Perception model (ACP) presented in [8]. In ACP*, robots have a neurocontroller with an output-input feedback mechanism that allows them to simulate perception and behaviour internally. Our experiments focus on the performance of robots with three different types of neurocontroller (two feedforward and one recurrent type of neurocontroller). Their performance is compared over three experimental conditions in which the output-input feedback mechanism is functional for variable durations. The results show that feedforward-neurocontrolled robots benefit from output-input feedback, while recurrent-neurocontrolled robots do not. Based on these results, two closely related conclusions are drawn: (1) the 'simulation hypothesis' may be too specific, and (2) predicting future perception may depend on neural recurrency (i.e., internal feedback) in general, rather than on the ability to simulate perception by feeding back actions.

1 Introduction

The 'simulation hypothesis' of cognition holds that, in humans, 'thinking consists of simulated interaction with the environment' ([4], p.242). Such simulation may occur by imagining behaviour and predicting the perceptual changes the behaviour would cause. The simulation hypothesis is widely supported by findings in neuroscience that show *motor* areas in the brain to exhibit similar patterns of activation during imagined and actual behaviour (see, e.g., [5]). In addition, the activation patterns in *sensory* areas are similar for imagined and actual perception [6]. These neuroscientific results suggest a relation between the motor and sensory areas of the brain and thinking (i.e., imagining). However, it still remains to be established that a 'simulation' mechanism forms the actual foundation for imagining and thinking.

We claim that, if the 'simulation hypothesis' is correct, having the ability to simulate perception and behaviour should be advantageous for a systems' performance on a cognitive task as compared to the same system lacking this ability. Therefore, we formulate

[1] {mf.vandartel, postma, herik}@cs.unimaas.nl

the following research question: *Do cognitive systems benefit from the ability to simulate perception and behaviour internally?*

According to Beer [1], debates on cognition should be grounded in concrete examples, and the best examples to start with are minimal models (see, e.g., [1]). We employ an extended version of the minimal model of Active Categorical Perception (ACP), introduced in [8], to explore whether situated agents benefit from the ability to simulate perception and behaviour internally.

ACP, and its extended version ACP*, are described in section 2. Section 3 presents the experiments with ACP*. The results of these experiments are provided in section 4. In section 5 these results are discussed and our conclusions are given.

2 The model

In the Active Categorical Perception model (ACP), described in subsection 2.1, robots are optimised to categorise two classes of falling objects by actively catching or avoiding them. In ACP*, described in subsection 2.2, ACP is extended with three mechanisms.

2.1 ACP

ACP consists of the following five elements: the environment, an object, a robot, the categorisation task, and the evolutionary algorithm that optimises performance on the task.

The environment is defined as a two-dimensional grid G_t with positions (x,y), with $1 \leq x \leq x_{max}$ and $0 \leq y \leq y_{max}$, and t the current discrete time step ($0 \leq t \leq t_{max}$, with $t_{max} = y_{max} - 1$). The objects and robots are allowed to move through the left and right boundaries of the environment, and to re-appear at the opposite side of the environment.

An object, represented in G_t by a sequence of ones, can start at any horizontal position in the top row of G_t, and always moves one row down and two columns rightward or leftward at each time step. An object always either moves rightward or leftward after initialisation in the top row ($y = y_{max}$) at time step $t = 0$. Only one object can be present in G_t at a time step. Two classes of objects are defined: small objects (2 grid cells wide) and large objects (4 grid cells wide).

A robot always starts in the middle of the environment at time step $t = 0$, it is restricted to the bottom row of grid G_t ($y = 0$) at all time steps t, and it can move to any horizontal position at any time step. A robot consists of a neurocontroller, an array of 4 sensors, and a motor system. The neurocontroller is either a perceptron (P), a multi-layered perceptron (MLP), or a recurrent neural network (RNN) (cf. [8]). The RNN is a simple Elman network with recurrent connections on the hidden nodes only. The sensors of a robot occupy neighbouring grid cells in the bottom row of grid G_t, $G_t(x,0)$ for all t. The activation of a sensor at position x at time step t is represented by $I(x,t)$ and is defined as:

$$I(x,t) = \sum_{y=1}^{y_{max}} G_t(x,y) = G_t(x, y_{max} - t) \tag{1}$$

A robot's sensory state does not contain any information regarding its own position or its distance to an object. The motor system rounds the output of the neurocontroller to the

nearest integer value, expressed as st, which defines the number of grid cells moved to the left (negative output) or right (positive output). If $st = 0$, the robot does not move. For a sensor positioned at x, the new position after movement is defined by

$$(x + st) \; mod \; x_{max}. \tag{2}$$

Movement of a robot leads to a new position of the sensor array and, consequently, to a new sensory state determined by the new position of the robot and equation 1. Figure 1 illustrates the movement of a robot and an object in the environment (grid G_t) over four consecutive simulation time steps (denoted by $t = 4$ to $t = 7$). The large object (represented by 4 black grid cells) moves leftward. The four circles in the bottom row of each grid represent the sensors of the robot; they are activated (grey circles) by the presence of an object in the same column. In the figure, the robot moves four grid cells leftward in each time step. The movement of the robot depends on the activation of the sensors and on the structure of the robot.

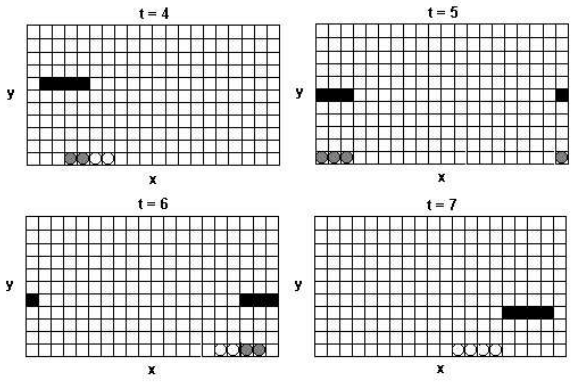
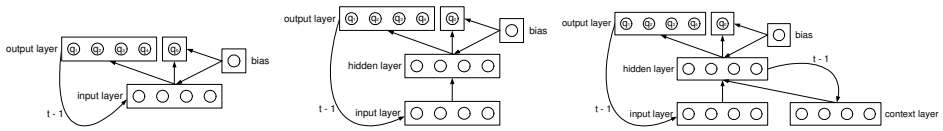


Figure 1: Movement of a robot and object over four consecutive simulation time steps (from $t = 4$ to $t = 7$). Black grid cells represent the object, and circles represent the active (grey) and inactive (white) sensors of the robot (taken from [8]).

The categorisation task that a robot in ACP has to perform is to categorise the two classes of objects (small and large) correctly. Robots are optimised to avoid large objects and to catch small objects, thus exhibiting its ability to categorise. The behaviour of a robot is evaluated when the object reaches row $G_t(x,1)$ at $t = t_{max}$. An object is caught by the robot *iff* $|cr - co| \leq 4.5$ (modulo the boundaries), with cr representing the centre of the robot and co the centre of the object; an object is avoided *iff* $|cr - co| > 4.5$ (modulo the boundaries).

The evolutionary algorithm determines the weight values for all randomly initialised connections in a robot's neurocontroller using the standard evolutionary techniques of reproduction, crossover, and mutation [2]. The evolutionary algorithm is described in detail in [8]. A robot's fitness F, i.e., the success of a tested robot, is calculated as

$$F = (CC + CA) - (FC + FA) \tag{3}$$



(a) P (b) MLP (c) RNN

Figure 2: Topology of two types of extended neurocontrollers (MLP and RNN). The additional output nodes are labelled q1 to q4.

with *CC* the sum of correctly caught objects, *CA* the sum of correctly avoided objects, *FC* the sum of caught objects that should have been avoided, and *FA* the sum of avoided objects that should have been caught. Robots are tested on 80 trials, all possible starting positions times the number of object classes times the number of directions (left or right) in which objects can move ($x_{max} \cdot 2 \cdot 2$). The performance of a robot is expressed by its success rate ($\in [0,1]$), which is calculated by $(F+80)/(2 \cdot 80)$.

2.2 ACP*

To be able to address our research question, ACP is extended with three mechanisms to form ACP*: (1) an output-input feedback mechanism, to enable robots to simulate perception and behaviour internally (subsection 2.2.1), (2) an occlusion mechanism, to occlude robots from environmental input for a variable period of time (subsection 2.2.2), and (3) a normalisation mechanism, to normalise the input received by the robot (subsection 2.2.3).

2.2.1 Output-input feedback mechanism

To enable robots to simulate perception and behaviour internally in ACP*, all three types of ACP neurocontroller (P, MLP, and RNN) are extended with an output-input feedback mechanism that consists of an additional output node for each input node. The output nodes are assigned indices that range from zero to the number of input nodes. The original ACP output node is assigned the index zero, the additional output nodes the indices 1 to the number of input nodes. The patterns that these extra output nodes produce constitute internal input patterns. At time step t, the input nodes are excited by the superposition of the external input ('the environmental input') and the internal input produced at time step $t-1$. To implement the output-input feedback mechanism, equation 1 is replaced by

$$I(x,t) = G_t(x, y_{max} - t) + \tanh(O(q, t-1)) \tag{4}$$

with $O(q,t-1)$ the q-th element of the output pattern (i.e., the internal input pattern) at time step $t-1$, where q is defined as $q = x - s + 1$ with s the x position of the first sensor. Figure 2 shows how the output-input feedback mechanism is embedded in the topology of the three types of neurocontrollers (P, MLP, and RNN).

Each of the additional output nodes' activation is initialised by assigning a value of 0 at $t = 0$, as was done for all other nodes' activation of neurocontrollers in ACP*. By optimising the weights between the output nodes with $q > 0$ and the input nodes, the internal

input patterns constitute a recurrency much like the recurrent connections on the hidden nodes of the RNN do. However, recurrent connections from output to input nodes allow robots to predict future external inputs, i.e., simulate perception and behaviour internally, while recurrent connections on hidden nodes do not.

2.2.2 Occlusion mechanism

To test the ability of robots to simulate perception and behaviour internally with ACP*, we extend the model with an occlusion mechanism. The mechanism occludes the environmental input for a predefined period of time ot. We assume that the occlusion mechanism encourages robots to simulate the future sensor states (i.e., environmental input). Occlusion always occurs at the last ot consecutive time steps between $t = 0$ and $t = t_{max}$. The occlusion mechanism is implemented by replacing equation 4 by equation 5 for $t > y_{max} - ot$.

$$I(x,t) = \tanh(O(q,t-1)) \tag{5}$$

2.2.3 Normalisation mechanism

Whereas no normalisation of input occurred in ACP, in ACP* activation of the input nodes is normalised by

$$I_{norm}(x,t) = I(x,t) * (2/\sum_{x=1}^{x_{max}} I(x,t)) \tag{6}$$

If $\sum_{x=1}^{x_{max}} I(x,t) = 0$, then equation 6 is ignored and the activation of all input nodes is set to 0.5. The normalisation mechanism keeps the summed normalised input constant (cf. [3]) at a value of 2 irrespective of the source of input (internal, external, or internal and external), which enhances the biological plausibility of the model [7].

3 Experiments

For the experiments described below, robots are equipped with four sensors that directly pass their binary activation on to the input nodes. The boundaries of grid G_t are set to $x_{max} = 20$ and $y_{max} = 10$. For all experiments the evolutionary algorithm is applied for 20,000 generations with a population size of 100 robots (cf. [8]). Through optimisation of a neurocontroller's weights the evolutionary algorithm can optimise the input-output mapping, including the output-input feedback mechanism encoded in the weights between the hidden nodes and the output nodes with $q > 0$. These weights are initialised by assignment of random values (cf. [8]), as was done with all other weights of neurocontrollers in ACP*.

Experiments with ACP* are conducted for all three types of neurocontrollers and for three conditions in which output-input feedback was: (1) not present (no feedback), (2) only present during occlusion (feedback during occlusion), or (3) present at all time steps (feedback at all times).

In the first condition (no feedback) equation 1 is used for all time steps t to update the input nodes. In the second condition (feedback during occlusion) equation 1 is replaced by equation 4 for time steps $t > t_{max} - ot$ to update the input nodes (hence, at $t = 0$ robots

are only initialised in G_t and no input is received). In the third condition (feedback at all times) equation 1 is replaced by equation 4 for all time steps t to update the input nodes.

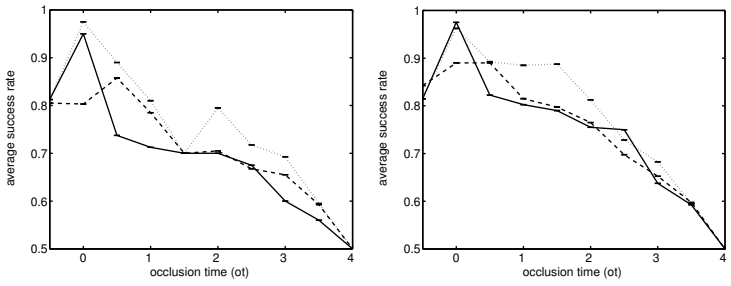
All nine experimental conditions, i.e., the three types of neurocontroller and the three output-input feedback conditions, are tested for all possible values of ot, $0 \leq ot \leq t_{max}$. Each experiment is repeated 5 times, over which the success rates of the best-performing robots are averaged. The results of the experiments are presented in section 4.

4 Results

To compare the three feedback conditions, we plotted the average success rates of robots in each condition over the different occlusion times for each type of neurocontroller (see figures 3(a), 3(b), and 3(c)). Standard deviations (*sd*) were computed for all data points, but appear as single lines in figures 3(a), 3(b), and 3(c), because $sd < 0.001$ for all data points.

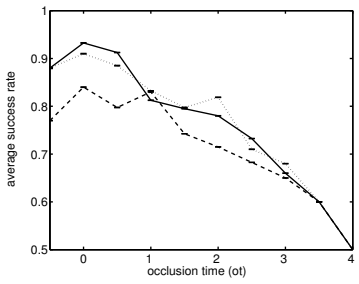
Figure 3(a) shows the average success rate of P-controlled robots for the three feedback conditions. For most occlusion times, feedback during occlusion leads to the best-performing robots on the active categorical perception task, and feedback at all times leads to the second-best-performing robots. Figure 3(b) shows the average success rate of MLP-controlled robots for the three feedback conditions. Again, feedback during occlusion leads to the best-performing robots on the active categorical perception task, and feedback at all times leads to the second-best-performing robots for most occlusion times. Figure 3(c) shows the same results for the RNN-controlled robots. For most occlusion times, no feedback and feedback during occlusion are preferred over feedback at all times. For RNN-controlled robots there seems to be no general preference for feedback during occlusion over no feedback. Four relations between figures 3(a), 3(b), and 3(c) can be observed. First, when $ot = 0$, robots with feedback during occlusion perform equally well compared to robots that do not use the output-input feedback mechanism at all. The reason is that the feedback is not operative for this value of ot. Second, when $ot = 9$ all robots perform at chance level, i.e., an average success rate of 0.5. For this value of ot, the occlusion time extends over the entire simulation episode, i.e., no external input is received. Third, the main variations in average success rates of the different robots occur for intermediate occlusion times. Fourth, the average success rate does not vary smoothly with the occlusion time for any of the robots. For instance, there is an increase in performance for most robots when $ot = 1$ compared to when $ot = 0$, while, moreover, an increase in ot results in a decrease in the average success rate. This can be attributed to how the problem space relates to variables in ACP*, such as occlusion time and neural structure of a robot. Although it would be interesting to reveal the exact relation between the problem space and the variables in ACP*, such an analysis is beyond the scope of our current research goal.

The main result of the simulations is that for both the P and MLP-controlled robots there is a general preference, although small, for feedback (either during occlusion or at all times) above no feedback at all (see figures 3(a) and 3(b)). This general preference is not present in the average success rates of RNN-controlled agents (see figure 3(c)). It implies that feedforward-controlled robots benefit from output-input feedback to cope with occlusion from external input, while recurrent-controlled robots do not. Apparently,



(a) P

(b) MLP



(c) RNN

Figure 3: Average success rate of robots with three different types of neurocontrollers as a function of occlusion time *ot* for (a) P, (b) MLP, and (c) RNN. The three conditions are (1) no feedback (solid lines), (2) feedback during occlusion (dotted lines), and (3) feedback at all times (dashed lines).

the advantage of internal simulation of perception and behaviour is restricted to robots with feedforward neurocontrollers.

5 Discussion and conclusions

Our results show that, for this simple task, feedforward-controlled robots benefit from the ability to simulate perception and behaviour internally. This benefit was also pointed out by Ziemke, Jirenhed and Hesslow [9], who trained robots to follow 'blindly' corridors without collision, using predictions of sensory input instead of real sensory input. Ziemke et al. found robots with recurrent neurocontrollers to perform worse than robots controlled by feedforward neurocontrollers. However, they did not compare these results with the performance of robots without the ability to simulate perception and behaviour. Therefore, they failed to note that there is no benefit from internal simulation of perception and behaviour for recurrent-controlled robots.

RNN neurocontrollers without an output-input feedback mechanism are still able to generate internal input, because they have recurrent connections on their hidden units. However, the feedback produced by the output-input feedback mechanism differs from RNN-type feedback in that the latter maps hidden activity onto the hidden layer, rather than mapping output activity onto the input layer. Apparently, this difference is immaterial to the success rate while coping with occlusion in the categorisation task.

Our results agree with those of Ziemke et al., but show in addition that the benefit from internal simulation of perception and behaviour is restricted to robots with feedforward neurocontrollers. Our results suggest that any recurrency, be it by means of an internal input or by means of RNN-type feedback, suffices to deal with occluded external input. On the basis of this finding, we draw two closely related conclusions: (1) the 'simulation hypothesis' may be too specific, and (2) predicting future perception may depend on neural recurrency in general, rather than on the ability to simulate perception by feeding back actions. Further studies are needed to elucidate the types of recurrencies required to deal with events or objects that are temporarily out of view.

References

[1] R. D. Beer. The dynamics of active categorical perception in an evolved model agent. *Adaptive Behavior*, 11(4):209–243, 2003.

[2] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Reading: Addison-Wesley Publishing Company, 1986.

[3] S. Grossberg, editor. *The Adaptive Brain*, volume 1--2. Elsevier Science, Amsterdam, 1987.

[4] G. Hesslow. Conscious thought as simulation of behaviour and perception. *Trends in Cognitive Sciences*, 6:242–247, 2002.

[5] M. Jeannerod. The representing brain. Neural correlates of motor intention and imagery. *Behavioral and Brain Sciences*, 17:187–245, 1994.

[6] S. M. Kosslyn, G. Ganis, and W. L. Thompson. Neural foundations of imagery. *Nature Reviews Neuroscience*, 2:635–642, 2001.

[7] N. Rochester, J. H. Holland, L. H. Haibt, and W. L. Duda. Tests on a cell assembly theory of the action of the brain, using a large digital computer. *Transactions on Information Theory*, 2:80–93, 1956.

[8] M. F. van Dartel, I. G. Sprinkhuizen-Kuyper, E. O. Postma, and H. J. van den Herik. Reactive agents and perceptual ambiguity. submitted.

[9] T. Ziemke, D.-A. Jirenhed, and G. Hesslow. Toward internal simulation of perception in mobile robots. submitted.