

Reactive agents and perceptual ambiguity

Michel van Darter and Eric Postma

IKAT, Universiteit Maastricht¹

Abstract

Situated and embodied reactive agents solve relatively complex tasks by coordinating action and perception. Reactive agents are generally believed to be incapable of dealing with identical perceptual states that require different responses (i.e., perceptual ambiguity). In contrast to reactive agents, non-reactive agents can deal with perceptual ambiguity by storing and integrating sensory information over time. This paper investigates how and to what extent reactive and non-reactive agents cope with perceptual ambiguity. An active categorical perception model is introduced in which agents with different neuro-controllers categorise objects by coordinating action and perception. The agent's neuro-controllers are evolutionary optimised for the task. Our results show that reactive agents can cope with perceptual ambiguity despite their incapability to store perceptual information over time. An analysis of behaviour reveals that while non-reactive agents use internal memory to represent past sensory input, reactive agents use the environment as an external memory.

1 Introduction

Perceptual ambiguity occurs when identical (or similar) perceptual states require different responses. In situated models, the problem of perceptual ambiguity can be solved by storing information over time [1, 2, 4]. Reactive agents, i.e., agents reacting directly and exclusively to the current sensory input, are generally regarded as unable to cope with perceptual ambiguity [1, 2]. Beer [2] claims that only non-reactive agents with internal (recursive) dynamics can solve perceptual ambiguity (in the form of object persistence problems) by organising 'their behaviour according to sensory stimuli that are no longer present' (p. 424). However, numerous animals behave reactively [8, 6, 14] and are still able to cope with perceptual ambiguity in their natural environments. The research question addressed in this paper reads: How and to what extent can reactive and non-reactive agents cope with perceptual ambiguity? To answer this question, we use a model of active

¹P.O.Box 616, 6200 MD Maastricht, The Netherlands. E-mail: {mf.vandarter, postma}@cs.unimaas.nl

categorical perception in which neuro-controlled agents are optimised on a categorization task using an evolutionary algorithm. In the active categorical perception model, agents are optimised to categorise two classes of falling objects by either catching or avoiding them. In the model, perceptual ambiguity corresponds to a pattern of sensory stimulation that is not unique to an object class. The level of perceptual ambiguity is controlled by the sensor configuration of the agent. The performance and behavioural strategy of agents with reactive neuro-controllers is compared with those of agents with non-reactive neuro-controllers.

The outline of this paper is as follows. In section 2, the active categorical perception model is presented. Section 3 describes the conducted experiments, reports on the results, and analyses the behaviour of agents with optimised neuro-controllers. In section 4, results and behavioural analysis are discussed in terms of internal and external memory. Finally, in section 5, we provide our conclusion.

2 The active categorical perception model

An agent model of active categorical perception is used to investigate how and to what extent agents cope with perceptual ambiguity. In the model, agents are optimised to catch and avoid falling objects. The model is outlined in terms of environment (section 2.1), agent (section 2.2), categorisation task (section 2.3), and evolutionary algorithm (section 2.5).

2.1 The environment

A two-dimensional grid G of size $x_{max} \times y_{max}$ defines the environment in which the agent acts. For all experiments we set $x_{max} = 20$ and $y_{max} = 10$. The objects and agents are allowed to move through the left and right boundaries of the environment, defined by $x = 0$ and $x = x_{max}$, to re-appear at the opposite side of the environment. Simulations are performed in discrete time steps. Objects are always falling from $y = y_{max}$ at $t = 0$. An object is represented by a sequence of ones: $G_t((x + j) \bmod x_{max}, y) = 1$ for $j \in (0, 1, \dots, j_{max})$, with $j_{max} + 1$ the width of the object. Two classes of objects are defined: small objects ($j_{max} = 1$) and large objects ($j_{max} = 3$). The initial horizontal position of an object x is selected from $x \in \{0, 1, \dots, x_{max}\}$. The dynamics of falling objects is represented as follows. At each new time step $t + 1$ grid G is initialized with zeros to clear the preceding position of the object. The new position of the object is then ‘written’ in G as follows:

$$G_{t+1}(x, y) = G_t((x + 2d) \bmod x_{max}, y - 1), \quad (1)$$

with $d \in \{-1, +1\}$ a direction parameter that is fixed during a trial. The object defined by the sequence of ones in G moves leftward for $d = -1$ and rightward for $d = +1$. Figure 1 illustrates 4 consecutive time-steps in the active categorization model with a large object (black grid cells) falling leftward. The four circles in the bottom row of each grid represent the sensors of the agent. In the figure, the agent moves 4 grid cells leftwards in each time-step.

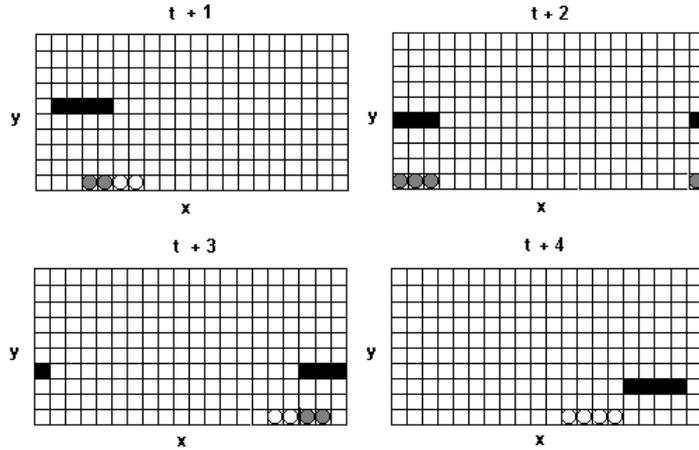


Figure 1: An agent with inactive sensors (white circles) and activated sensors (grey circles) as a result of detecting a large object (black grid cells) in the environment (grid G) over 4 time steps.

2.2 The agent

The agent consists of an artificial brain (section 2.2.1) that receives environmental input from an array of sensors (section 2.2.2). A motor system (section 2.2.3) moves the agent according to the output of the artificial brain, resulting in either avoidance or catch behaviour (section 2.2.4).

2.2.1 Brain

Three different neural networks were used as control structure for the agent: (1) a perceptron, (2) a multilayer perceptron, and (3) a recurrent neural network. They are briefly described below.

The perceptron (P) consists of an input layer of four nodes, one for each sensor. The input nodes are all connected to a single output node providing the motor output for the agent. The multilayer perceptron (MLP) has a similar structure, but is extended with a hidden layer of four nodes that is fully connected to both the output and input layer. In the recurrent neural network (RNN) four context nodes are added to the MLP. The context nodes are fully connected to the hidden layer. The context nodes always contain exact copies of the activation of the hidden nodes at time $t - 1$, which is used as additional input to the hidden layer at time t . Such recurrent connections enable an agent to retain past sensory information over time. This capacity enables only RNN-controlled agents to behave non-reactively. In all networks a bias node with a constant output of 1 is connected to the hidden nodes, if present, and the output node. The output of the hidden nodes is determined by

their *net* input and the *tanh* sigmoid activation function. The connective weights of the neural networks are optimised by the evolutionary algorithm described in section 2.5.

2.2.2 Sensors and blind spot

The sensor array of the agent consists of two types of sensors: functional sensors and blind sensors. Each agent has s functional sensors ($s = 4$, in all our experiments), each one connected to an input node. A variable number of blind sensors b ($b \in \{0, 1, 2, 3\}$), i.e., sensors that are not connected to the input layer, are inserted into the sensor array to control the level of perceptual ambiguity (see section 2.4). The blind sensors are always positioned in the middle of the sensor array. The sensors occupy neighbouring grid cells and are constrained to the bottom row of the grid G , $G(x, 0)$. The activation of the sensor at position x at time t is represented by $I(x, t)$ and is defined as:

$$I(x, t) = \sum_{y=0}^{y_{max}} G(x, y). \quad (2)$$

Figure 2 illustrates an agent with four functional sensors and two blind sensors. The two leftmost sensors are active ($I(x, t) = I(x + 1, t) = 1$), and are ‘sensing’ the object. We note that the agent’s sensory input does not contain any information regarding its own position or its distance to the object.

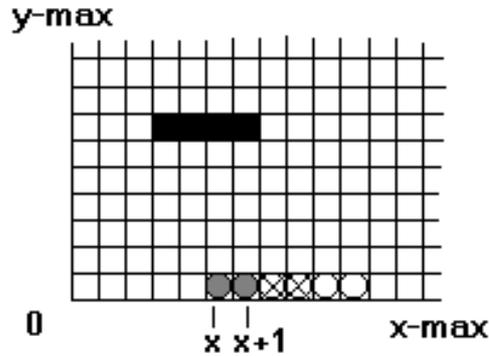


Figure 2: An agent with 2 blind sensors (crossed circles) sensing a large object (black cells) with two of its functional sensors (grey circles).

2.2.3 Motor

In order to catch and avoid objects, agents can move left, right, or stand still in the the bottom row of G . After movement, the new position of a sensor originally

at position x equals:

$$(x + a) \bmod x_{max}. \quad (3)$$

Movement of the agent leads to a new position of the sensor array. The new position is determined by the sensor array's old position plus a , representing the neuro-controller's output rounded to the nearest integer.

2.2.4 Catch and avoidance behaviour

Performance of agents is evaluated on the basis of their catch and avoidance behaviour when the object reaches the bottom row of grid G at $t = 10$. An object is caught by the agent *iff* $|ca - co| \leq 3$, with ca the centre of the agent and co the centre of the object, and avoided otherwise.

2.3 The categorization task

The agent's task is to categorise the two classes of objects (small and large). Agents are optimised to avoid large and to catch small objects. By correctly avoiding large and catching small objects, an agent can show it is capable of successful categorization. The challenge for an agent is to perform the categorization task in the information-poor environment described in 2.1, which provides no other cues than the sensor activation caused by the objects to categorise. Only nine or fewer (depending on the number of blind sensors) different patterns of sensory activation can occur in the agent's sensor array during the task. This makes studying the sensory-motor mapping of optimised agents' neuro-controllers feasible, and enables determination of the perceptual ambiguity in the model.

2.4 Perceptual ambiguity

Perceptual ambiguity is defined as the proportion of input patterns that are ambiguous to the agent. Input patterns are ambiguous when the presence of objects from different classes can give rise to that particular input pattern. Hence, no categorization is possible solely on the basis of a single ambiguous input pattern. The number of blind sensors inserted in the middle of the sensor array can manipulate the perceptual ambiguity in the model. When more blind sensors are added, less unique and more ambiguous input patterns occur, i.e., it becomes more difficult for the agent to categorise objects. Table 1 shows the percentages of unique and ambiguous sensor patterns as a result of the number of blind sensors an agent has. The percentages listed in table 1 were obtained by recording the occurrences of each sensor pattern while exhaustively generating trials with different starting positions of objects from both classes.

2.5 The evolutionary algorithm

An evolutionary algorithm determines the connective weight values for all connections in the agent's neuro-controller over N generations. After randomly initializing the weights of the neuro-controllers, the complete generation of n individuals

number of blind sensors	% of unique sensor patterns	% of ambiguous sensor patterns
$b = 0$	44	56
$b = 1$	44	56
$b = 2$	37	63
$b = 3$	0	100

Table 1: Percentages of unique and ambiguous sensor patterns in relation to the number of blind sensors with which the agent is equipped.

(i.e., agents) is tested on the active categorical perception task described above. An agent’s fitness F , i.e., the success of an agent on the test trials, is calculated as:

$$F = (\overline{CC} + \overline{CA}) - (\overline{FC} + \overline{FA}) \quad (4)$$

with \overline{CC} the average number of correctly caught objects, \overline{CA} the average number of correctly avoided objects, \overline{FC} the average number of caught objects that should have been avoided, and \overline{FA} the average number of avoided objects that should have been caught. Agents are tested on 80 trials, all possible starting positions times the number of objects and drop-directions ($x_{max} \cdot 2 \cdot 2$).

The evolutionary algorithm works as follows. Parents are selected by taking the 5 best ranking agents and adding 30 winning agents from tournaments (of size 3) held among randomly selected agents. 64 new agents are created by one-point crossover on basis of random selection among the parents, and 100 random mutations are performed over the complete new generation. Elitism is introduced to the algorithm by adding the best agent of the last generation to the new generation. During the evolutionary process, symmetry is maintained between the weights to reduce the search space. As a consequence the behaviour of the agent is mirror-symmetric. The performance of an agent is expressed in its success rate, which is the average proportion of fitness over 5 evolutionary runs.

3 Experiments

The active categorical perception model was used to conduct 12 different experiments. Agents with each combination of neuro-controller and number of blind sensors were optimised using the evolutionary algorithm described in 2.5, with $N=2000$ and $n=100$. Comparing the results of the 12 experiments provides insight in the difference in capacity to cope with perceptual ambiguity, determined by b , between reactive (P and MLP-controlled) agents, and reactive (RNN-controlled) agents. Results of the experiments are discussed below in section 3.1. After evolution the behaviour of optimised agents was plotted. An analysis of behaviour is conducted in section 3.2, to provide insight in the difference in strategy of coping with perceptual ambiguity between reactive and non-reactive agents.

3.1 Results

Figure 3.1 displays the results of the 12 experiments. The success rate of the best performing optimised agent in each experiment is averaged over 5 runs of an experiment. All neuro-controllers appear capable of performing categorical

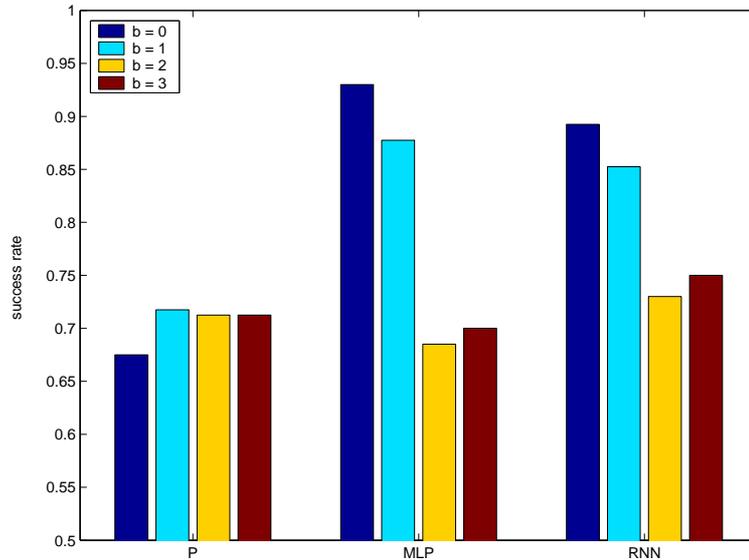


Figure 3: Average success rate of the best performing agents on the active categorical perception task. Each bar represents averaged performance of agents with a different combination of number of blind sensors (b) and neuro-controller (P, MLP, RNN).

perception above chance level (i.e., success rate > 0.50), independent of the number of blind sensors an agent has, i.e., independent of the perceptual ambiguity in the model. Surprisingly, for reactive P-controlled agents, those with one blind sensor ($b = 1$) outperform agents without blind sensors ($b = 0$). It should be noted that for both cases, the percentage of ambiguous patterns is identical (i.e., 44%; see table 1). Further increasing the number of blind sensors ($b = 2$ and $b = 3$) results in only a small decrement in performance, despite the increase in the percentage of ambiguous patterns (63% and 100%, respectively; see table 1). The reactive MLP-controlled agents outperform the P-controlled ones for $b = 0$ and $b = 1$. For a larger number of blind sensors, the MLP-controlled agents perform slightly worse than the P-controlled ones. For the non-reactive RNN-controlled agents a similar pattern of results is observed, although the performance for larger b 's appears to be slightly better than for the P-controlled agents. Changes in performance accompanying unaltered perceptual ambiguity in the model (from $b = 0$ to $b = 1$) can be attributed to two factors. First, to the change in sensor configuration as a result of altering the number of blind sensors. Second, to the difference in

the spatial extent or scope of the sensors. Presumably, both factors affect the performances in our study.

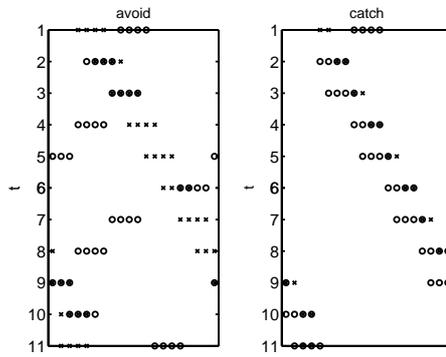
3.2 Analysis

For all neuro-controllers without blind sensors, the typical behaviour of optimised agents on successful trials is plotted in figure 4. The graph in figure 4(a), shows an agent with a P neuro-controller avoiding (left) and catching (right) an object. This example shows that the agent is capable of successful categorization (i.e., avoiding large objects and catching small objects) by reacting consistently to sensor input. This reactivity causes the agent-environment interaction to go into an attractor state, resulting in a consistent relative position to an object over the remainder of the trial [9, 10], eventually leading to successful catch or avoid behaviour. Such is the case, for example, from the second time-step onward in figure 4(a) (right). Figure 4(b) shows the reactive behaviour of a MLP-controlled agent. Here, an attractor occurs from the 7th time-step onward in the catch-trial (left). In the sample behaviour of the non-reactive (RNN-controlled) agent shown in figure 4(c), an important difference with 4(a) and 4(b) must be noted. As expected, in this case, the input-output mapping is not consistent. Similar input, for example at $t = 5$ and $t = 7$, does not cause the same reaction. The agent has been optimised to give different responses to similar input in different temporal contexts, i.e., exploit its ability to retain sensor information. Figures 4(a) and 4(b) illustrate how reactive agents cope with perceptual ambiguity despite their incapability to store sensor information. Although reactive agents are forced to react consistently and exclusively to current sensor patterns, their local actions do result in a different perceptual and 'physical' situation, i.e., a new perceptual input and a changed relative position between agent and object. Ultimately, at the time of evaluation, this sensory-motor loop results in the relative distance between agent and object required for successful behaviour.

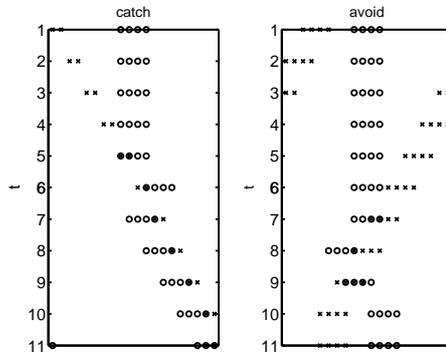
4 Discussion

Our findings contradict Beer's claims that reactive agents (i.e., agents without internal dynamics) 'cannot organise their behaviour according to sensory stimuli that are no longer present' and that 'what is needed is for the circuits controlling these agents to have internal dynamics' [2] (p.224). Both reactive controllers (P and MLP) perform categorization above chance-level, even when all sensory patterns are ambiguous, i.e., when $b = 3$. The better performance of MLP-controlled agents over the P-controlled agents indicates the importance of a non-linear perception-action transformation in reactive agents coping with perceptual ambiguity. At the lowest levels of perceptual ambiguity ($b = 0$ and $b = 1$), an MLP-controlled reactive agent even outperforms agents with internal dynamics.

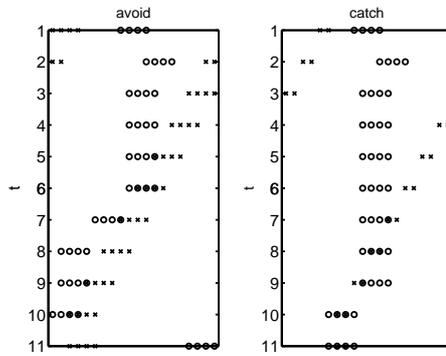
Important to note is that evolutionary selection occurs on the basis of complete sequences of sensory-motor behaviour, rather than on single perception-action steps. Consequently, our results can be explained by the fact that perceptual ambiguity is defined locally in time, while behavioural success (catching or avoiding) is



(a) P-controlled reactive agent without blind sensors



(b) MLP-controlled reactive agent without blind sensors



(c) RNN-controlled non-reactive agent without blind sensors

Figure 4: Examples of catch and avoid behaviour of agents with three different neuro-controllers (P, MLP, and RNN) and no blind sensors. The relative horizontal position between an object's elements (crosses) and an agent's functional sensors (circles) is depicted over a test trial.

defined globally. A similar conclusion was drawn by Nolfi [7], who found reactive robots to cope with perceptual ambiguity in a task in which all sensory states were ambiguous. He concluded that they did so by exploiting the 'emergent behaviours resulting from a sequence of sensory-motor loops and the interaction between robot and the environment' (p.119). In Nolfi's experiment [7], reactive agents were evolutionary optimised to find a goal position in a static environment. The results of our experiments extend Nolfi's results to a categorical perception task in a dynamic environment. Categorization is often argued to be fundamental to cognition [3, 5, 13].

In our experiments, the environment acts as an external memory for the reactive agents. Past actions are (to a certain extent) reflected in the current sensor state. Hence, in this task an internal memory is not needed to deal with perceptual ambiguity. Natural cognitive systems have also been shown to use the environment as an external memory [11, 12].

5 Conclusion

Our results illustrate how and to what extent situated agents cope with perceptual ambiguity. Reactive agents cope with perceptual ambiguity by using the environment as an external memory, whereas non-reactive agents may use their internal memory as well. The extent to which both types of agents deal with perceptual ambiguity is reflected in the graph shown in figure 3. All agents perform above chance level. For high levels of perceptual ambiguity, the differences in performance for reactive and non-reactive agents are small. We conclude that reactive agents can deal with perceptual ambiguity by exploiting the environment as an external memory store.

References

- [1] B. Bakker and G. van der Voort van der Kleij. Trading off perception with internal state: Reinforcement learning and analysis of q-elman networks in a markovian task. *Proceedings of the International Joint Conference on Neural Networks*, III:213–218, 2000.
- [2] R.D. Beer. Toward the evolution of dynamical neural networks for minimally cognitive behavior. *From Animals to Animats 4. Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, 4:421–429, 1996.
- [3] R.D. Beer. The dynamics of active categorical perception in an evolved model agent. *BBS preprint*, 2001.
- [4] A. Clark. *Being there: Putting brain, body, and world together again*. Cambridge MA: MIT Press, 1997.
- [5] S. Harnad (ed.). *Categorical perception: The groundwork of cognition*. New York NY: Cambridge University Press, 1987.

- [6] K. Lorenz. *Foundations of Ethology*. New York NY: Springer-Verlag., 1973.
- [7] S. Nolfi. Power and limits of reactive agents. *Neurocomputing*, 49:119–145, 2002.
- [8] S. Nolfi and D. Marocco. Active perception: A sensorimotor account of object categorization. *From Animals to Animats 7. Proceedings of the VII International Conference on Simulation of Adaptive Behavior*, 2001.
- [9] S. Nolfi and D. Marocco. Evolving robots able to integrate sensory-motor information over time. *Theory in Biosciences*, 120(3-4):287–310, 2001.
- [10] S. Nolfi and D. Parisi. Exploiting the power of sensory-motor coordination. *Advances in artificial life: Proceedings of the Fifth European Conference on Artificial Life*, pages 73–182, 1999.
- [11] J.K. O’Regan. Solving the ‘real’ mysteries of visual perception: The world as an outside memory. *Canadian Journal of Psychology*, 46:461–488, 1992.
- [12] J.K. O’Regan and A. No. A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(5):883–917, 2001.
- [13] A. Tijsseling. *Connectionist Models of Categorization: A Dynamical Approach to Cognition*. PhD thesis, University of South-Hampton, 1998.
- [14] N. Tinbergen. *The Study of Instinct*. New York NY: Oxford University Press, 1951.