Cognitive paradigms: which one is the best?☆

Action editor: Valerie Hardcastle

Carlos Gershenson*

Centrum Leo Apostel, Vrije Universiteit Brussel, Krijgskundestraat 33, Brussels 1160, Belgium

Received 15 July 2003; accepted 1 October 2003

Abstract

I discuss the suitability of different paradigms for studying cognition. I use a virtual laboratory that implements five different representative models for controlling animats: a rule-based system, a behaviour-based system, a concept-based system, a neural network, and a Braitenberg architecture. Through different experiments, I compare the performance of the models and conclude that there is no “best” model, since different models are better for different things in different contexts. Using the results as an empirical philosophical aid, I note that there is no “best” approach for studying cognition, since different paradigms have all advantages and disadvantages, since they study different aspects of cognition from different contexts. This has implications for current debates on “proper” approaches for cognition: all approaches are a bit proper, but none will be “proper enough”. I draw remarks on the notion of cognition abstracting from all the approaches used to study it, and propose a simple classification for different types of cognition.

© 2003 Elsevier B.V. All rights reserved.

Keywords: Cognition; Modelling; Virtual laboratory; Rule-based; Behaviour-based; Concept-based; Neural networks; Braitenberg architectures

1. Introduction

Cognition comes from the Latin cognoscere, which means “get to know”. We can say that cognition consists in the acquisition of knowledge. We can say that a system is cognitive if it knows something. Humans are cognitive systems because they know how to communicate, build houses, etc. Animals are cognitive systems because they know how to survive. Autonomous robots are cognitive systems if they know how to navigate. However, does a tree know when spring comes because it blossoms? It has always been controversial to discuss when a system is cognitive and when it is not.

In classical cognitive science and artificial intelligence (e.g. Newell & Simon, 1972; Newell, 1990; Shortliffe, 1976; Fodor, 1976; Pylyshyn, 1984; Lenat & Feigenbaum, 1992), people described cognitive systems as symbol systems (Newell, 1980). However, it seemed to become a consensus in the

☆ This article is dedicated to the memory of Javier Fernández Pacheco, from whom I learned so much.

* Tel.: +32-2-644-26-77; fax: +32-2-644-07-44.
E-mail address: cgershen@vub.ac.be (C. Gershenson).
community that if a system did not use symbols or rules, it would not be cognitive. From this perspective, animals are not cognitive systems because they do not use or have symbols. Nevertheless, if we dissect a human brain, we will not find any symbol either. Opposing the symbolic paradigm, the connectionist approach was developed (Rumelhart, McClelland, & the PDP Research Group, 1986; McClelland, Rumelhart, & the PDP Research Group, 1986), assuming that cognition emerges from the interaction of many simple processing units or neurons. To my knowledge, there has been no claim that “therefore a cognitive system should be able to perform parallel distributed processes, otherwise it is not cognitive”. Still, there has been an intense discussion on which paradigm is the “proper” one for studying cognition (Smolensky, 1988; Fodor & Pylyshyn, 1988). The behaviour-based paradigm (Brooks, 1986, 1991; Maes, 1994) was developed also opposing the symbolic views, and not entirely different from the connectionist. There have been also other approaches to study cognition (e.g. Maturana & Varela, 1987; Beer, 2000; Gärdenfors, 2000).

The main goal of this work is to show that there is no single “proper” paradigm for studying cognition, but different paradigms that study cognition from different <perspectives/contexts> and with different goals. Moreover, I argue that in theory any cognitive system can be modelled to an arbitrary degree of precision by most of the accepted paradigms or theories, but none can do this completely (precisely because they are models). I believe that we will have a less-incomplete understanding of cognition if we use all the paradigms available rather than trying to explain each and every aspect of cognition from a single perspective.

This view is currently shared by many researchers, but to my knowledge, there has been no empirical study to backup these claims. For achieving this, I implemented different models from different paradigms in virtual animats, to compare their cognitive abilities. The models I use are not very complex, not to at all to be compared with humans, but they are useful for understanding the generic processes that conform a cognitive system. After doing several comparative experiments, I can suggest, using the simulation results as a philosophical aid, that there is no “best” paradigm, and each has advantages and disadvantages.

In the following section, I present the virtual laboratory developed to compare the implementations in animats of models coming from five different perspectives: rule-based systems, behaviour-based systems, concept-based systems, neural networks, and Braitenberg architectures. In Section 3, I present experiments to compare the performance of the animats in different scenarios. With my results, I argue in Section 4 that each model is more appropriate for modelling different aspects of cognition, and that there is no “best” model. I also discuss issues about models, and from my results I try to reach a broader notion of cognition merging all the paradigms reviewed. I also propose a simple classification of different types of cognition.

2. A virtual laboratory

Following the ideas presented in Gershenson, González, and Negrete (2000), I developed a virtual laboratory for testing the performance of animats controlled by mechanisms proposed from different perspectives in a simple virtual environment. Programmed in Java with the aid of Java3D libraries, this software is available to the public, source code and documentation included, at http://homepages.vub.ac.be/~cgershen/cogs/keb.

In this virtual laboratory, the user can create different phenomena, such as rocks (grey cubes), food sources (green spheres), rain (blue semi-transparent cylinders), lightnings (black cylinders), and spots of different colours (circles): randomly or in specific positions. These also can be generated randomly during the simulation at a selected frequency. Lightnings turn into rain after ten time steps, and rain turns into food after 50 time steps.

All the animats have an energy level, which decreases when their hunger or thirst are high, and

1 Since symbols are manipulated by rules, I will use rule-based system as a synonym of symbol system.

2 Tyrrell (1993) did an interesting comparison, but of different action selection mechanisms and all from a behaviour-based perspective.
is increased when these are low (energy, thirst, hunger ∈ [0, . . . , 1]). An animat dies if its energy is exhausted. Eating food decreases their hunger. They can decrease their thirst by drinking under rains. Hunger and thirst are increased if they attempt to drink or eat “incorrect” stimuli. They lose energy if they touch lightnings or rocks. Basically, an animat to survive just needs to eat when hungry, drink when thirsty, and avoid lightnings and rocks. We can say that they are cognitive systems if they are successful, because they would know how to survive. The animats can leave a coloured trail to observe their trajectories. We can appreciate screenshots of the virtual environment Figs. 1 and 2. A detailed technical description of the implementation of the virtual laboratory can be found in Gershenson (2002d).

There are many models that would solve the problem of surviving in such an environment, but I decided to implement representative models of different paradigms to observe their differences and similitudes. These models are as follows: a rule-based system typical of traditional knowledge-based and expert systems (e.g. Newell & Simon, 1972; Maes’s (1990, 1991) action selection mechanism, an already classical behaviour-based system (Brooks, 1986; Maes, 1994); my original architecture of recursive concepts as an example of the novel concept-based approach (Gärdenfors, 2000); a simple feed-forward artificial neural network (for

2.1. Rule-based animats

“For anything to happen in a machine some process must know enough to make it happen” – Allen Newell

Rule-based animats have perceptual and motor functions that ignore the problems of implementing perception and motion in physical agents, focussing only on the control mechanism. They are inspired in classical knowledge-based systems (Newell & Simon, 1972; Newell, 1990), which use logic rules manipulating symbols to control a system. Table 1 shows the rules that control the animats in their environment.

If rule-based animats perceive a rock near them, they avoid it. If they are thirsty and perceive rain, then if they can reach it drink, otherwise approach to it, and so on. They approach lightnings when thirsty and rains when hungry because “they know” that these will turn into rain and food, respectively.

The perceptual system detects phenomena in all directions at a distance lesser than the animat’s radius of perception, detects phenomena near when they are at a distance lesser than the radius of the animat’s body, and detects phenomena at range when the animat touches them. The motor system approaches phenomena in a straight line,
explores with random movements, and avoids obstacles semi-randomly turning about $\pm 90^\circ$.

Knowledge-based systems have several limitations (e.g. see Maes, 1994), and are not optimal for implementing different types of system, but they are suitable for this task. We could see them as models of cognizers, even humans, which in such conditions would deliberately take those decisions. Classical cognitive science argues that humans are cognitive systems because they use rules and reasoning as the ones these animals could model (e.g. Newell, 1990).

### 2.2. Behaviour-based animats

Maturana and Varela give the following definition: “behaviour is a description an observer makes of the changes in a system with respect to an environment with which the system interacts” (Maturana & Varela, 1987, p. 163). We should just remember that behaviours are defined by an observer.

Behaviour-based systems (Brooks, 1986; Maes, 1994) have been inspired in ethology for modelling adaptive behaviour and building adaptive autonomous agents. In problem domains where the system needs to be adaptive, they have several advantages over knowledge-based systems (Maes, 1994). Yet when it comes to modelling the type of cognition that knowledge-based systems model, they have not produced any better results (Kirsch, 1991; Gershenson, 2002b).

I implemented Maes’s (1990, 1991) action selection mechanism (ASM) for controlling the behaviour-based animats. It consists of a network of behaviours. Each behaviour has an activation level, a threshold, and a set of conditions to be “executable”. An executable behaviour whose activation level surpasses the threshold becomes active. The creatures controlled by Maes’ ASM also have motivations such as hunger or safety, which also contribute to the activation of behaviours. The behaviours are connected through “predecessor”, “successor”, and “conflicter” links. There is a predecessor link between A and B (B precedes A) if B makes certain conditions of A come true. For example, “eat” has “approach food” as a predecessor. There is a matching successor link in the opposite direction for every predecessor link. There is a conflicter link from A to B if B makes a condition of A undone. Behaviours activate and inhibit each other, so after some time the “best” behaviour becomes selected. For details of this ASM, the reader is referred to Maes’s (1990, 1991).

As with the rule-based animats, here I also simplify perceptual and motor systems (using the same systems already described), since they take the form of procedures such as “food perceived” or “avoid obstacle”. These systems have to deal with the problem of distinguishing food from non-food, or how to move in order not to crash. Fig. 3 shows the behaviour network of the animats.

The external conditions of the behaviours are obvious: rock or lightning near for “avoid”, food perceived for “approach food”, food at range for “eat”, none for “explore”, etc. The internal motivations are safety (constant) for “avoid”, hunger for the ones related with food, thirst for the ones related with thirst, none for “approach lightning”, hunger, thirst and curiosity (constant) for “explore”, and boredom (constant) for “none”.

Curiously, the animats, even when they make discriminations between the highest motivations

<table>
<thead>
<tr>
<th>Table 1: Control of rule-based animats</th>
</tr>
</thead>
<tbody>
<tr>
<td>if (rockNear OR lightningNear) then {</td>
</tr>
<tr>
<td>avoid</td>
</tr>
<tr>
<td>} else if (((thirst) = 0.1) AND (rainPerceived)) then {</td>
</tr>
<tr>
<td>if (rainAtRange) then {</td>
</tr>
<tr>
<td>drink</td>
</tr>
<tr>
<td>} else {</td>
</tr>
<tr>
<td>approachRain</td>
</tr>
<tr>
<td>} else if (((hunger) = 0.1) AND (foodPerceived)) then {</td>
</tr>
<tr>
<td>if (foodAtRange) then {</td>
</tr>
<tr>
<td>eat</td>
</tr>
<tr>
<td>} else {</td>
</tr>
<tr>
<td>approachFood</td>
</tr>
<tr>
<td>} else if (((thirst) = 0.1) AND (lightningPerceived)) then {</td>
</tr>
<tr>
<td>approachLightning</td>
</tr>
<tr>
<td>} else if (((hunger) = 0.1) AND (rainPerceived)) then {</td>
</tr>
<tr>
<td>approachRain</td>
</tr>
<tr>
<td>} else if (((thirst) = 0.1) OR (hunger) = 0.1)) then {</td>
</tr>
<tr>
<td>explore</td>
</tr>
<tr>
<td>} else {</td>
</tr>
<tr>
<td>noAction</td>
</tr>
<tr>
<td>}</td>
</tr>
</tbody>
</table>

(approach food if more hungry than thirsty even if rain is closer), behave in a reactive way (eat when they are not hungry). This is because of the activation of behaviours by the successor links. Therefore, if the behaviour “none” is active for a long time, it will increase the value of “explore”, and this “approach rain” if this one is present, because thirst is not a direct condition. Of course, I could fix Maes’ ASM adding the motivations as a condition for a behaviour to become executable, or explore exhaustively the parameter space of the mechanism. This last option is not an easy one, since it takes some time to adjust all the parameters by trial-and-error. This has led people to propose evolutionary techniques for tuning this type of ASMs (Singleton, 2002).

2.3. Concept-based animats

Concepts can be considered as discrete categorizations (Gärdenfors, 2000), as opposed to sensation and perception that can be considered as continuous categorizations. Concepts can be seen as a discretization of the perceptual space.

For the concept-based animats I used with minor improvements my original, previously developed, architecture (Gershenson, unpublished), named KEBA, where “concepts” are recursively generated from regularities of the sensors and of other concepts, and linked to actions by simple reinforcement learning. The activity of the concepts decides which action will be executed.

Actually, KEBA is not at all an optimal architecture in pragmatic terms, since it takes lots of effort in solving a problem that other architectures do quite easily. However, it was created with an explanatory role, trying to understand how concepts, and logic, could be developed from the interactions of an agent with its environment. One thing I have learned is the importance of the gradual development (Piaget, 1968; Balkenius, Zlatev, Brezeal, Dautenhahn, & Kozima, 2001) for acquiring proper performance in complex environments. The animats have a difficult time if they are set right into a random complex environment. Nevertheless, if we set them in simple environments, so that they can learn different tasks at different times, gradually, they perform better.

If we would be interested only in performance, we could define the concepts and manually link them to actions. However, that was not the intention, since it would easily work, but not better than simpler models in any interesting way. For a detailed description of KEBA, please refer to Gershenson (unpublished, 2002d).

2.4. Neural network animats

Artificial neural networks (for an introduction, see Arbib, 1995) are mathematical models, which consist of interconnected “units” or “neurons” which have activation values. These values change accordingly to the state of the inputs to the network and to the connections between neurons. Neurons multiply their inputs by given weights (which can be adjusted by learning algorithms) and combine them accordingly to a specific function. They were not used at first for controlling autonomous agents, but they have been useful for this purpose as well (e.g. Beer, 1996; Slocum, Downey, & Beer, 2000).

The neural animats have pairs of sensors for detecting food, rain, and rocks. Each sensor is on one side of the animat, so for example the left sensor for rain becomes active when there is rain on the left of the animat, and so on. For implementation purposes, we measure a ponderation ratio of the size of the phenomenon over its distance, so that for example larger objects at the same distance, or the closest of several objects of
the same size, will cause a higher activity in the sensor. This would simulate the fact that for example if a lamp is very bright but farther than a close dim one, a photosensor in a real robot could react more to the far bright light. Each sensor perceives 180°, so that each pair of sensors is mutually exclusive and combining them the animats perceive 360°. They also have two motors (left and right), which simulate the movement of wheels. This setup is traditional of simulated and robotic experiments with kephera-like robots. In Fig. 4 we can appreciate a diagram of the structure of these animats.

These animats use a three-layered feed-forward neural network for controlling their motors. The inputs of the network are the sensors and the internal motivations hunger and thirst. These activate eight input neurons, which combine their values for the inputs of six hidden neurons. Two output neurons determine the speed of the motors, which have small random noise, to make the simulation a bit less unrealistic (Jakobi, Husbands, & Harvey, 1995). The neurons simply multiply their inputs by fixed weights and sum them. Then, the activation of the neuron will be this sum if it is higher than a threshold, and zero otherwise. We can see a diagram of this neural network in Fig. 5.

Note that there is no learning in this specific neural network, which is one advantage of these models. Even so, they perform appropriately in their simple environment. Learning can allow animats to adapt to specific conditions of their environment (Gershenson & González, 2000).

If the animat is hungry or thirsty and touching a food source or rain, it eats or drinks, respectively. The consummatory “behaviours” (eat and drink) are independent of the motor activity. This causes that the animats cannot “stop and eat”, but only “eat on the go”.

One advantage of solving the problem of survival at this level is that we are not predetermining behaviours as “approach food” or “avoid” to be selected (Seth, 1998), and the sensors are quite simple. Nevertheless, these animats also survive in their environment. One small issue is that since they move in zigzag because of their wheels, they take more time to reach, let us say a food source, than an animat with an implementation of “approach food” (following a straight line).

2.5. “Vehicle” animats

“Vehicles” are architectures introduced by Braitenberg (1984). They are autonomous agents
with direct links from sensors to motors. One example is a two-wheeled robot with two light sensors. The left sensor is connected to the right motor and the right sensor is connected to the left motor. This causes that if there is more light on the left of the vehicle, the right motor turns faster. Then the right sensor will be closer to the light, causing the left motor to go faster. The behaviour this simple setup produces can be described as phototactic: the vehicle will approach the light. On the other hand, if we connect the left sensor to the left motor and the right sensor to the right motor, then the vehicle will flee from light sources. Many interesting behaviours can be produced with several sensors and different connections. Braitenberg vehicles have been extended, among others, by Lambrinos and Scheier (1995) and by Seth (1998). The type of cognition exhibited by Braitenberg architectures can be compared with the one exhibited by Grey Walter’s tortoises (Walter, 1950, 1951).

My Braitenberg-style animats were handcrafted (i.e. not evolved). They have a setup similar to the neural network animats, shown in Fig. 4: six sensors and two motors. There are direct links from sensors to motors, which have some small random noise: food and rain sensors are connected to the opposite motors (left to right, right to left), and rock sensors are connected to their corresponding motors (left to left, right to right). The speed of the motors is multiplied by the energy of the animats (slower when energy is low), and the links of food and rain sensors are multiplied with the values of hunger and thirst, respectively (faster when motivations are high). As with neural network animats, Braitenberg animats have independent consummatory behaviours: they eat if they are hungry while touching food and drink if thirsty while touching rain. Fig. 6 shows the connection diagram of the Braitenberg animats.

With this simple setup, Braitenberg animats are able to survive successfully in their simple environment. The main idea is that they approach to food proportionally to their hunger, to rain proportionally to their thirst, but also flee from rocks when they are close enough, providing efficient obstacle avoidance. Because every sensor contributes to the motor speed, both motors compete in such a way to produce a robust and decisive behaviour. Also, animats have noise in their motors proportional to their motivations, so when they are hungry or thirsty they move randomly at a speed proportional to their internal need, providing a kind of random exploration. This noise is also present when they are “approaching food” or any other behavioural description an observer could make, but this does not affect their final performance, since Braitenberg architectures are quite robust to noise.

3. Experiments

I have realized series of illustrative experiments to compare the different properties of the architectures that control each type of animat. The reader is invited to download my virtual laboratory and test the experiments I show here, and to devise her/his own.

I do not suggest that my simulations and experiments prove anything, but I think of them as

---

3 This is the reason for the lack of statistical rigour, which in any case would be very difficult to obtain due to the complexity of the system. The experiments are illustrative.
an aid to present and produce ideas. I agree with the possible critic who would say that I am solving a toy problem, but most of the models, simulations or robots, do not go much farther (see Hallam, Floreano, Hallam, Hayes, & Meyer (2002) for recent work). The only way they go farther is by adding more behaviours, stimuli, motivations, etc., but the degree of simplification is as terrible as the one I have here.

I could also be criticized by some arguing that my experiments are not “really cognitive”, because the animats do not have any kind of problem solving or symbol manipulation. This is not correct. Every animat needs to solve the problem of surviving in its environment (problems are determined by an observer). And rule-based animats can be said to be using the symbols received by their perceptual system. Another critic can be that this problem of survival is very simple and not “representation-hungry” (Clark & Toribio, 1995).

As we will see in Section 3.4, these problems are possible to solve as well with architectures from different paradigms, but my ideas can be seen more clearly in this simple setup rather than in problem domains in which different architectures from different paradigms are specialized because of computational efficiency.

In this section, I present experiments for observing the performance of the animats in shared scarce and abundant environments. I also test the survival of the animats in individual environments of fixed resources. In these experiments I also use a “control” animat with random action selection. Then I test the ability of the animats to discriminate between the values of different stimuli, as an example of a simple relational task. Finally, I test how different animats can cope with conflicting goals.

3.1. Survival in a scarce environment

For this experiment, I set one animat of each type, and an animat with random action selection, with no initial internal needs and highest energy value. I set the ratio of random generation of phenomena to very low, making the rain and food scarce. Fig. 7 shows a typical history of internal states of the animats for this scenario.

We can see that some animats, like the ANN, do not even have the opportunity of tasting food or rain. The survival depends very much on luck: if a proper stimulus is generated near them, and if other animats are not around so that they can satisfy themselves. Yet the chances of obtaining necessary resources are too low, and therefore, eventually, all animats die. In this presentation, the behaviour-based animat survived for a longer time. This and the rule-based animat tend to survive more than the others in this setup, because they can move faster than the “wheeled” animats with their direct behaviours. The concept-based animats do not even have time to learn what is edible and what is not. The random animats really do not have a chance in such a scarce environment. Also, my implementation of the behaviour-based animats resulted in reactive behaviour: the animats eat even when they are not hungry. This “bully” strategy gives the other animats a harder time than if they would all eat only when they would be hungry. I should note that this is not a property of the creatures in which the Maes’ ASM was originally implemented (Maes, 1991).

3.2. Survival in an abundant environment

With the same initial conditions, I set up a medium random generation ratio for the phenomena, resulting in an abundant environment. We can observe the dynamics of the internal states for a typical run in Fig. 8.

Even the random animat could drink a bit. That it randomly drank when it was under the rain was just a coincidence. However, this allows it to survive a bit more. The KEBA animat is able to learn that food is edible, but the environment is too complex, and it sadly died learning how to avoid obstacles. Still, we can see that the other animats perform quite well, and we can see that they return to their initial internal states several times. Not that they will never die (if the environment is filled with rocks, they will die), but we can see that they have a good performance. The Braitenberg animat has more trouble finding phenomena before the others, because it moves slower since it models wheels. Still, the neural network animats also, and it performs better. Of course there is a chance el-
lement, but Braitenberg animats cannot keep approaching a food source or a rain source as persistently as the neural network animats.

3.3. Individual survival in environment of fixed resources

For this experiment, I set each animat individually in an environment with five randomly generated rocks and five randomly generated food sources. There is no random generation of phenomena. Fig. 9 shows the history of the internal states for a typical run.

The random animat is even able to eat some food by chance. All other animats extinguished the food sources before dying. The KEBA animat is able to learn to eat food, but it does this reactively, so as the Maes animat, it extinguished the food sources faster than the other animats, thus dying faster. The KEBA animat survives less time than the Maes animat because the first one needs to learn what not to eat or drink, whereas this is “innate” in the second. In this setup the Braitenberg animat performed better than the neural network, precisely because it

---

4 For a deeper study on the degree of reactivity and motivation in virtual creatures, see González, Negrete, Barreiro, and Gershenson (2000) and Gershenson and González (2000).
cannot keep approaching to food sources as persistently as the neural network does, and this makes it not to exhaust the resources so fast. The rule-based animat performed better than the wheeled animats. This is because the rules determine that it should eat only when the internal needs are higher than 0.1, and the wheeled animats eat “on the go” if their internal needs are higher than zero. I made another trial patching the Braitenberg animat to eat only when its hunger is greater than 0.1, and it survived for more than fifty thousand time steps, at the same level of the rule-based animat. However, this patch causes some inconveniences: for example, if the Braitenberg animat has less than 0.1 hunger, it will not eat, but it will approach food persistently in a not so smart way. These inconveniences could be patched with a threshold.

3.4. Discrimination of stimulus value

It could be argued that all the previous experiments were easy to solve because they had very strong statistical correlations (Thornton, 2000), what Clark and Thornton (1997) have called “type one” problems. Even if they are very simple, the animats should be able to solve simple “type two” problems, where the task is more relational than statistical, and many algorithms fail to reach a solution.

One very simple relational problem is to observe if the animats are able to distinguish between food sources “larger than” others, when they are at the same distance of the animats, independently of their order, position, or absolute size. An example of this setup can be appreciated in Fig. 10(a).
Fig. 9. Survival of animats in a limited resources scenario.

Fig. 10. Discrimination of the value of different stimuli. (a) Initial state. (b) Task solved by an ANN animat.
All the animats are able to pass this test without problems. The only thing is that it does not relate much to their “cognitive” architectures, but more with the fact that they are situated in their environment. Actually, the problem is solved by interactions of the motor and perceptual systems with the environment.

Rule-based, behaviour-based, and concept-based animats solve this problem because their behaviour “approach food” simply approaches the closest food they perceive. The distance to a phenomenon is measured from the border of the animat to the border of the phenomenon. Since larger foods have larger radiiues, the measured distance to the animats will be lesser than the one of small foods, even if their centres are at the same distances from the animats. Therefore, because of this relationship with the environment, the animats will always distinguish the larger food, even when this was never programmed directly.

With the neural and Braitenberg animats the task is solved by their sensors, which basically become more active to larger stimuli, so they move towards them without hesitation. An example of the locomotion pattern of a neural animat solving this task can be appreciated in Fig. 10(b).

The fact that these tasks are solved not by the “cognitive” architectures we are comparing but by the interaction of the perceptual and motor systems with the environment, reminds us of how important are embodiment and situatedness when modelling cognition (Varela, Thompson, & Rosch, 1991; Clark, 1997; Riegler, 2002).

3.5. Buridan’s animat

The popular allegory of Buridan’s ass (actually in the original work of Jean Buridian it was a dog, but anyway) tells the following: the animal has to choose between two equal amounts of food. The conclusion was that it should choose at random. Otherwise, it would starve to death. In the virtual environment, we can set up a Buridan’s animat with the same amount of thirst and the same amount of hunger, and at the same but opposite distances from food and water. This problem can be seen as how to solve conflicting goals and/or motivations.

In order to test how the different animats performed in such a situation, I set them with maximum hunger and thirst, at the same distance from food and rain, but with an obstacle on the way there. Fig. 11 shows the initial setup and the trails that each animat followed, together with their internal variables.

We can observe that the rule-based animats solve this situation without difficulties, giving a preference to rain over food (this is convenient since rains last only fifty time steps). Well, I designed the rules so that the animat would solve this problem efficiently. My Maes animat unfortunately turned out to be Buridan’s animat... it would die in the indecision if the rain were there forever. Also, the high internal needs surpassed the avoidance of the obstacle. Once the rain is transformed, the animat goes for food. The problem here was that the behaviour nodes for water and food give the same importance to thirst and hunger. I could patch this situation giving more weight to thirst, or by introducing conflictier links between these consummatory behaviour nodes. I should note again that this is a problem particular of my implementation, since the original Maes’ creatures did not suffer from this deficiency. The KEBA animat, after several repetitions of the same environment, is able to learn most of the concepts required for a successful run: the only thing is that it bumped into the obstacle and lost a bit of energy, but apart from that its behavioural sequence is similar to the one of the rule-based animat. The neural animat has no problems with the indecisions, and successfully drinks and eats “on the run”, without switching. This is not so of the Braitenberg animat, which switches between phenomena, and is unable to satisfy its thirst completely. The neural animats turned out to be more persistent than the Braitenberg ones, but these issues could be tuned up. Actually, the wheeled robots do not give preference of rain over food, but they give preference to the phenomena on their left side. This is because the right motor is computed first, making the animats closer to the phenomena on the left. Again, the wheeled robots seem slower because of their locomotion patterns. In real robots this can easily be overcome with faster motors.
One surprising thing was that the wheeled robots could decide where to go without indecisions, even when this was not contemplated when they were designed.

4. Discussion

I take my virtual laboratory and the experiments performed in it as a philosophical aid for discussing the suitability and theoretical equivalence of models. I do not take the simulations as a proof of my ideas, but as “opaque thought experiments” (Di Paolo, Noble, & Bullock, 2000).

With the ideas generated so far, I try to broaden the notion of cognition in order to propose one valid in as many contexts as possible. Finally I propose a simple classification of different types of cognition.

Fig. 11. Buridan’s animat experiment. (a) Initial setup of all animats. (b) Rule-based animat. (c) Behaviour-based animat. (d) Concept-based animat. (e) Neural net animat. (f) Braitenberg animat.
4.1. About the architectures

From the few experiments I presented, we can see that there is no general “best” architecture for the simple task of surviving in the presented virtual environment. We can say that each animat is better in different situations. Still, it seems that this is more a consequence of the particular implementation than of the paradigm on which it stands, because the models can be adjusted and refined to any desired degree of detail. In order to judge which architecture is better, we need to refer to a particular context. Their performance cannot be generally measured, but only relatively to specific tasks. We cannot say that one animat is cognitive because it uses rules and another is not because it has only direct connections. The cognition of a system is independent of its implementation. Thus, we have to observe their cognition basing ourselves on their performance.

Some models were very easy to implement in software code (rule-based, Braitenberg), others not so much (KEBA, Maes), but occasionally it is a different story if we want to implement an architecture in a real robot. Moreover, if a model works in a simulation and/or robot, it does not mean that animals function in the same way. Some models are very robust (Braitenberg), others would break up quite easily (rule-based). Some models are quite good if we have just practical purposes (rule-based), or if we want things only to work, but this also depends on the experience of the engineer. Yet if we are interested in using them as explanatory models, the simplicity of their implementation might be secondary (KEBA). Also, if we would like to increment the systems, for example to include more environmental stimuli and internal variables, some would need to be redesigned (rule-based), others could be easily extended (Maes, KEBA). Also some models would have more ease in adapting to changes of their environment (KEBA, neural network) than others (rule-based, Braitenberg), but this does not mean that we cannot adjust different architectures in order to obtain the desired behaviour. Furthermore, we should note that in spite of historical preferences, any system from the paradigms we compared can be developed with a bottom-up or a top-down design.

The animats could be criticized by saying that they are specific to their environment, and that if we take them out of their toy world, they would not exhibit cognition at all. Well, if we change the environment of any cognitive system too much, it will break (in the Ashby’s (1947) sense) inevitably (raise the temperature few thousand degrees, remove oxygen, or just leave a man in the middle of the ocean, and tell him “adapt! Don’t you know how?”). This might be a matter of adaptability rather than about cognition (but of course adaptability is tightly related to cognition). Yes, they are specific to their environment, but any cognitive system is.

There are dozens of other models that could be implemented in order to try to solve this or other problem. For example, we have developed a complex behaviour production system that exhibits many properties of animal behaviour desired in such a model (González, 2000; Gershenson, 2001), but it is not as easy to implement as any of the models I implemented here. We could use a huge set of nice differential equations, or GasNets (Husbands, Smith, Jakobi, & O’Shea, 1998), or random boolean networks (Kauffman, 1969; Gershenson, 2002c), or an extension of the quantum formalism (Aerts, 2002), just to name some alternatives. Moreover, we can say that different models, architectures, and paradigms, can be studying different aspects of cognition (Gershenson, 2002b). Will we find ever a “best” model? Well, as I said, it depends on our purposes and our context.

We can say that different cognitive paradigms describe different aspects of cognition, and that each paradigm is better at describing the aspect for which it was developed. Rules and symbols are very suitable for “higher level”, abstract, logical aspects of cognition, but they are not very adaptable. Architectures such as neural networks and behaviour-based systems tell us many things about how the aspects of cognition studied by classical AI could have evolved. Nevertheless, they are not very efficient for developing systems that should exhibit those “higher” aspects. Nevertheless, those systems can be very adaptive and robust. They are
also very useful for practical purposes, when we do not have a clear idea of what should be the final behaviour of the system we develop. They are useful for explanatory purposes as well, when we do not want to make too many assumptions. The conceptual paradigm (Gärdenfors, 2000) can be helpful for bridging the previous paradigms. For practical purposes, many people take the good things of all paradigms, developing hybrid architectures. I believe that this should also be done in cognitive science, where different paradigms can be complementary. This will be more fruitful than continuing to argue on which paradigm is “better” for describing cognition, when they are describing different things.

4.2. Equivalence of different models

“If you are perverse enough, you can describe anything as anything else” – Inman Harvey

Different models of the same phenomenon have this phenomenon in common, but this simple fact does not make the models equivalent. They would be equivalent if they could produce the same <results|predictions|behaviour> in the same situations. They would be equivalent to a certain degree if they only produce some of this similar performance. What I said for relationships between models can also be said for the relationships between model and modelled. Clearly, we will not reach complete equivalence, because the differences between models or between model and modelled would have to disappear (the best model of a cat is the same cat (Rosenblueth & Wiener, 1945), the best model of a model is the same model). So actually we can only speak about equivalence to a certain degree.

But we can already say that all the models presented here are equivalent to a certain degree, because most often the animals eat when they are hungry, drink when they are thirsty and so on. The question is, how high can this degree of equivalence go?

Let us begin with a mental exercise. Since the models were programmed in a computer, in theory they can be mapped into a universal turing machine (UTM) (Turing, 1936), well, basically because they are computable, and Turing used his machine precisely to define computability. Therefore there is a function that would map the models to a UTM. Could we find an inverse function, such that, any model already in the UTM, could be mapped back to any other model? The idea is tempting, but it is not as easy as that, because basically the model to be mapped to would need to have the capabilities of the UTM (which in theory it has because it can be implemented in a UTM, but we would like the capabilities as a property of the model, not of its implementation).

Let us see an example. Neural networks are mathematical models. Russell and Whitehead (1910–13) showed that all mathematics can be described in terms of logic. Therefore, we can describe any neural network in terms of logic rules. Of course implementing them that way is not practical. But in this context we can say that neural networks could be seen as rule-based systems. However, also neural networks can be designed to produce rules (e.g. Balkenius & Gärdenfors, 1991; Gärdenfors, 1994). Actually the neural network animat can be said to be modelling the rules of the rule-based animat, but also vice versa! And, at least for my particular setup, we could say this for all the animats. We can see any animat as a model of any other. If we want it to mimic the behaviour of other model more tightly, we just need to adjust the implementation, but in theory, there is no task that an architecture can do

5 More precisely, we also need a clock for modelling any dynamical system, since original Turing Machines do not contemplate the temporal aspect of computation. I thank Inman Harvey for pointing this out to me.

6 A general way of deciding if all the models are equivalent would be the following: If a model can simulate a UTM, then it could simulate through this any other model. It has been shown that there are several models which can simulate a UTM (e.g. Berlekamp, Conway, & Guy, 1982; Wolfram, 2002), and we could think of even more, such as neural networks. But even when we could think that we could implement a UTM with any of the architectures I used, or even with my animats, would this have any explanatory or practical relevance? Well, it would show that models with this property would not only be equivalent, but we could model a model with another model and vice versa, just as I will show shortly with rule-based systems and neural networks.
and another cannot do with proper extensions and patches. Then, we can say that in theory all the implemented architectures can be equivalent to any desired degree of detail.

Any implemented model could then be described in terms on another paradigm, and therefore there is nothing a paradigm can describe which another one cannot. Of course, there are different aspects of cognition modelled with less effort and in a more natural way from different perspectives. That is why we should not reject other paradigms, because we can see that different paradigms describe with more ease different aspects of cognition.

This is all nice in theory, but what about in practice? Well, it seems that implementing a medical expert system for an intensive care unit using only neural networks would be not so easy, nor it would be easy to perform character recognition implementing only rules. Because then, in theory, any paradigm could also simulate human cognition with any degree of detail, but none has even been near.\footnote{Actually, Turing computation should be called “computation in theory”, since there are Turing-computable functions which are not computable in practice (not enough time in the universe), and non-Turing-computable functions which are computable in practice, with the cheating aid of an “oracle” (to compute the halting function of a Turing machine). The human brain can very well be computable in theory, but is it in practice?}

It seems that we are falling into a problem if we do not stop and identify the differences between the models and the modelled.

4.3. About models

“Explanations are for ourselves, not for the explained”

All models, by definition, are simplifying: they are abstractions of the things they model. It depends on what we are interested in modelling and how we justify our simplifications that we can judge the suitability of a model.\footnote{Webb (2001) has discussed several dimensions in which we can make models of animal behaviour more or less close to the modelled: medium, generality, abstraction, level, relevance, structural accuracy, and behaviour match.}

But clearly there is no “best” model outside a context. Some could argue that better models are the ones in which prediction is both maximally accurate and minimally complex. But this is inside the context of information theory.

I argue that there is no way of determining which model is better or worse, good or bad (Gershenson, 2002a). But one thing we can do, is to distinguish from different degrees of incompleteness. All models are incomplete, but if a model contains several models, it will be less incomplete than those models. This would be valid only in the context of understanding, because in a pragmatic context we would just want a model to work with the less effort. But if we try to contain as many models as possible, one will encounter fewer contradictions inside the less-incomplete model.

Returning to cognitive models and architectures, we can say not only that there is no general good model or architecture. We will have a less incomplete understanding of cognition only if we study it from as many perspectives/contexts/paradigms as possible. Each model is abstracting different aspects of what cognition is: any cognitive behaviour can be described in terms of rules, parallel distributed processing, behaviours, mathematics, etc. And now I can say that all these models are equivalent in the degree that they model the same aspect of cognition. All my animals were modelling the same phenomenon: knowing how to survive in a simple environment. And certainly there are other aspects of cognition. Of course some models might be easier to apply. Some might be more illuminating than others, and so on. Nevertheless, they are just different ways of perspectives for describing the same thing.

And this does not mean that cognition is characterized by rules, behaviours, apples or pears. Things do not depend on the models we have of them.

So, what is cognition then?

4.4. About cognition

Cognition has been studied from a variety of contexts, such as philosophy, artificial intelligence, psychology, dynamical systems theory (Beer, 2000), etc. And it is because of this that in each context cognition is considered with different eyes,
and to be a different thing. So in different contexts we will be able to define cognition as the manipulation of symbolic representations (Newell, 1990), or as autopoiesis (Stewart, 1996; based on Maturana & Varela, 1980), or as the ability to solve a problem (Heylighen, 1990), or as the ability to adapt to changes in the environment, or as “the art of getting away with it”. 9 We can say that cognition can be a different thing in different contexts, but can we say what cognition is in general? No, but we can approach as much as we want to. The way of achieving this is to make our context as less-incomplete as possible, by containing as many contexts as possible. Therefore, we will not be able to dismiss a model just because it is of a certain paradigm, since all paradigms suffer from limit-edness.10 We can only learn from any model of cognition. We cannot say whether a model is right or wrong outside a context. Of course, less-incomplete models will be more robust and will be valid in more contexts. For example, we cannot judge internal representations in a neural context just because these are not described at that level.

I will try to reach a broader notion of cognition based upon the results and ideas exposed previously. I can make some general remarks:

- Systems can be judged to be cognitive only inside a specific context. For example, in a chess-playing context, a bee is not cognitive, but in a navigational context, it is. People agree in contexts, and these are contrasted with experience of a shared world, so we are not in danger of any radical relativism or wild subjectivism.

- Cognition is a description we give of systems, not an intrinsic constituent of them, i.e. systems do not have cognition as an element, we observe cognition from a specific context. The cognition of a system does not depend on its implementation.

- If a system performs a successful action, we can say that it knows what to do in that specific situation. This success is tied to a context and to an observer. Therefore, any system performing a successful action can be considered to be a cognitive system. This is a most general notion of cognition, and other types of cognition and definitions can be applied in different contexts with different purposes without contradicting this notion.

So, a tree knows when spring comes because it blossoms, in a specific context (not common in cognitive science, though). And a protein knows how to become phosphorylized. And a rock knows how to fall... if we find a context where this makes sense.

It might seem that we are falling a bit into a language game. Yes, but we are victims of the same language game when we speak about human cognition! We are the ones who judge that a tree may know when to blossom, and consider this as knowledge. But this is not different from the process we make when we judge the knowledge of a human. We can describe human problem solving in terms of behaviour and classical conditioning, but we can also describe biology in terms of epistemology.

I am not insinuating that atoms and humans have the same cognitive abilities, there is a considerable difference in complexity, but not in the “essential” nature of cognition (well, the ability to do things “properly” is not entirely essential, since we judge this properness). (But for example an oxygen atom knows how to bind itself to a couple of hydrogen atoms, and humans do not!)

We can measure this complexity, but we should note that this can only be relative to an abstraction level (Gershenson, 2002a). And there are many definitions and measures of complexity, so again there is no “most appropriate” measure outside a context. Moreover Kolen and Pollack (1995) have shown that complexity is dependent on the observer and how she measures a phenomenon.

So, what should cognitive science study? I would suggest that cognition at all levels, not only at the human, in order to have the broadest notion of cognition. People already speak about bacterial (Jonker, Snoep, Treur, Westerhoff, & Wijngaards, 2001), immunological (Hershberg & Efroni, 2001), plant, animal (Vauclair, 1996; Bekoff, Allen, & Burghardt, 2002), machine, social, and economical

---

9 This phrase is original of Arturo Frappé.
10 “All ideas are valid in the context they were created” (Gershenson, 2002a).
cognitions. But we could also speak about proteic, molecular, atomic, planetary, etc. cognitions. Of course all of this is our interpretation, but if we take “the real thing”, what cognition is, we humans are not different from any other system. What changes is just how we describe ourselves (and our complexity. This complexity allows us to identify new abstraction levels, and this is very important, but at the end we all are a bunch of molecules, a mass of quarks, and infinitude of nothings...). “How does the immune system know which antigens are foreign of the organism?” is not a question very different of “How people know when someone is lying?”. And research in complex systems (see Bar-Yam (1997) for an introduction) has shown that systems classically considered as cognitive can be modelled with the same models of systems that are classically not considered as cognitive, and vice versa.

That we are interpreting cognition does not mean that there is no objective notion of cognition. What it means is that it is everywhere, and therefore there is no general way (i.e. outside a specific context) to draw a borderline between “cognition” and “non-cognition”.

How useful is to describe the behaviour of a particle in terms of cognition, when physics already describes it with a different terminology? It is not about usefulness. We should just realize that cognition, in essence, is the same for all systems, since it depends on its description. What makes us different is the complexity degree and the names we use to describe our cognition.

4.5. Different types of cognition

We can quickly begin to identify different types of cognition, and this will relate the ideas just presented with previous approaches for studying cognition. This does not attempt to be a complete or final categorization, but it should be helpful for understanding the ideas just presented.

We can say that classical cognitive science studies human cognition. Several disciplines are involved in the study of human cognition, such as neuroscience, psychology, philosophy, artificial intelligence, etc. Human cognition can be seen as a subset of animal cognition, which has been studied by ethologists (e.g. McFarland, 1981) and behaviour-based roboticists (e.g. Brooks, 1986). But we can also consider the process of life as determined by cognition and vice versa, as the idea of autocoesis proposes (Maturana & Varela, 1980, 1987; Stewart, 1996), in which we would be speaking about cognition of living organisms. Here we would run into the debate of what is considered to be alive, but in any case we can say that biology and artificial life (Langton, 1989) have studied this type of cognition. Artificial cognition would be the one exhibited by systems built by us. These can be built as models of the cognition of the living, such as an expert system, an octapod robot, or my virtual animats. But we can also build artificial systems without inspiration from biology that can be considered as cognitive (the thermostat knows when it is too hot or too cold). Most of the previous types of cognition can be considered as adaptive cognition, since all living organisms also adapt to modest changes in their environment, but also many artificial and nonliving systems. Cybernetics (Wiener, 1948), and more recently certain branches of artificial intelligence and artificial life have studied adaptive systems (e.g. Holland, 1992). We can contain all the previous types of cognition under systemic cognition. Complex systems (Bar-Yam, 1997), and general systems theory (von Bertalanffy, 1968) can be said to have studied this type of cognition. I cannot think of a more general type of cognition because something needs to exhibit this cognition, and that something can always be seen as a system. We can see a graphical representation of these types of cognition in Fig. 12.

It is curious that cognitions considered as simpler contain the ones considered to be more complex. It seems that this is because when we speak for example about human cognition, we do not see humans as a system. And when we speak for example about cognition in living organisms, we do not think right away about human cognition. We should also note that all types of cognition can be studied at different levels and from different approaches.

This is only one way of categorizing different types of cognition, but there can be several others. One could be by measuring the statistical correlations between the “inputs” and the “outputs” of a
cognitive system (if we can identify them). If the outputs can be obtained by pure statistical correlations, then the cognition is simpler than if it requires complex transformation or re-representation of the inputs (Clark & Thornton, 1997; Thornton, 2000). The more transformation the inputs require, the higher and complex the cognition would be. So for example a rock would have low cognition, because if it is on the ground (input), it will stay there (output), and if it is on the air (input), it will fall (output). Now if we try to do the same predictions/descriptions with a cat, we will see that they have higher cognition. This categorization is also not universal, but it seems to be useful in several contexts, rather than in a single one. We could also identify different levels of cognition, similar to the observed levels of behaviour proposed previously (Gershenson, 2001, 2002b).

I believe that the proposed classification can be useful in cognitive science, because it can help people who study different aspects of cognition in clarifying which type of cognition which they are speaking about. Instead of arguing on which type of cognition is the “proper” one, we can refer to different types of cognition to understand each other better.

5. Conclusions

In classical cognitive science, it seems that there was the common belief that human cognition was a symbol system (Newell, 1990). I believe that the confusion was the following: human cognition can be modelled by symbol systems (at a certain level), but this does not mean that human cognition (absolutely) is a symbol system. But the same applies to all models. Human cognition (absolutely) is not a parallel distributed processor, nor any other model about which we can think. Things do not depend on the models we have of them. That some aspects of cognition (e.g. navigation) are implemented more easily under a certain paradigm, does not mean that natural cognitive systems do it the same way.

The implemented animats are cognitive at the same degree, because they model the same aspect of cognition roughly with the same success. Systems are not cognitive because they implement a specific architecture. Of course different architectures can be more parsimonious, others more explanatory, others easier to implement, etc.; but this is dependent of the context in which we are modelling.

Different cognitive models and paradigms can be said to be modelling different aspects of cognition. They are different metaphors, with different goals and from different contexts. Therefore, we will have a less-incomplete view of cognition if we take into account as many paradigms as possible.

There have been several proposed definitions of cognition, in different contexts. I proposed a notion that is applicable to all of these contexts and possibly more (although this makes it less practical).

A human doing the same things a simple robot does would be considered cognitive, just because her behaviour would be described with different terminology. But if the observed processes are the same, we believe that there is no intrinsic cognitive difference related to a specific task between two different systems (i.e. functional not material) if they solve the same task in the same context with the same success. This is very similar to Turing’s (1950) test for intelligence, only that limited to a context, rather than comparing machines with humans in general. This is why we say that cognition is observed. Just as a brain needs a body and an environment (Clark, 1997), a cognitive system also needs an observer.
Someone could say that "real" cognition is given when a system (such as a mature human) is able to explain and understand, and that we are the ones describing other systems, thus giving the cognition. I would agree, but even go further: we are the ones describing our own cognition, along with that of any cognitive system. Our cognition does not depend only on our nature, but also on how we <observe|describe> it.

In different contexts, different systems can be judged as cognitive or not, but how we judge them does not change the system, only our description. But contemplating as many systems as cognitive can only enhance our understanding of what cognition is.

Acknowledgements

The main part of this work was done at Sussex University. I thank Chris Thornton, Inman Harvey, Andy Clark, Emmet Spier, Ezequiel Di Paolo and an anonymous referee for useful comments and suggestions. This work was supported in part by the Consejo Nacional de Ciencia y Tecnología (CONACYT) of México.

References


Riegler, A. (2002). When is a cognitive system embodied? [Special issue on situated and embodied cognition],


