An Intermediate Form of Behavioral Control in ‘Reactive’ Robots

Sjoerd Lagarde [a] Ida Sprinkhuizen-Kuyper [ab] Guido de Croon [ac]

Pim Haselager [ab]

[a] *Department of Artificial Intelligence, Radboud University Nijmegen*
[b] *Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behaviour*
[c] *Micro Air Vehicle lab, Faculty of Aerospace Engineering, Technical University of Delft*

Abstract

The hypothesis of behavioral control as a form of traffic regulation is tested by means of simulation. Agents have to survive in an environment with a day/night rhythm by eating food and avoiding obstacles. This goal is achieved easily at day, but not at night due to interference of darkness with sensor readings. Two categories of agents are tested: reactive and control system agents. The control system is a neural network with inhibitory outputs to the agent’s behavioral layers. Therefore, the system is referred to as a traffic regulator. The neural network is optimized by an evolutionary algorithm. Results showed that control agents are able to adapt to the day/night rhythm by developing a day/night rhythm. These results are in favor of the traffic regulator hypothesis.

1 Introduction

Many attempts have been made to extend the well-known reactive robot architecture [1] with higher levels of control, involving explicit world modeling and planning, resulting in so-called hybrid architectures [6]. However, it could be argued that such hybrids combine the two extremes of the reactive-planning continuum and questions may be raised about their efficacy. While some studies focus on building more complex architectures by extending the employed neural networks (e.g. [2]), this research focuses on intermediate behavioral control by devising and testing this form of behavioral control in reactive robots. One of the simplest forms of such control would be of a purely inhibitory nature, basically just saying ‘no’ to the otherwise most active behavioral layer. The control structure has no direct connections to effectors, but can only operate through inhibiting behavioral layers. The metaphor of a ‘traffic regulator’ has been introduced [3, 10] to describe the functioning of this form of control: the control structure is just giving ‘no go’ signs based on cues from the body and environment, thereby providing space for other, perhaps more appropriate, types of behavior.

In this paper the idea of control as a from of traffic regulation is explored by means of a simulation in the context of an environment with a day and night rhythm. Agents are supposed to survive as long as possible in this environment. Four types of agents are used. The first type of agent to be tested is a reactive agent [1]. This agent responds directly to stimuli from the environment and has no higher control structures. The second type of agent is a reactive agent now with a higher control structure ‘on top’. This structure, much like the prefrontal cortex in humans [4], can influence the different behavioral responses by means of inhibition. An additional cost in terms of energy consumption is calculated for having this control structure. The third type of agent is the same as the second one, but now the additional energy consumption is dependent on how much the control structure is used. The last type of agent is a reactive agent with a built-in day and night rhythm. This agent is used to compare results of the other agents to.

By using the environment and agents described above, the following questions will be addressed:

1. Will reactive agents with an inhibitory control system be more effective than purely reactive agents in an environment displaying a day and night rhythm?

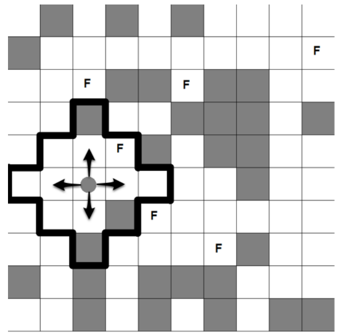

Figure 1: A typical environment. The letter 'F' indicates food, dark colored squares are obstacles. The agent is indicated with the gray circle. The thick black lining indicates the visual field of the agent. The arrows indicate the movements the agent is able to take. Note that the agent may also remain at its current position.

2. If so, is this effectiveness due to the development of an interaction between the control system agents and environments?

The remainder of this paper is structured as follows. First the simulation environment will be discussed, followed by the parameters used in the simulation. Next the results of the simulations are discussed together with their implications. This paper will end with a discussion and ideas for future research.

2 Method

In order to answer the research questions, a simulation is used. In this simulation different types of agents have to survive as long as possible in a given environment. The time the agents survive averaged over a number of different (i.e. randomly created) environments is the measure by which the agents are compared. Agents are able to survive in the environment by eating food and avoiding obstacles when searching for food. The aspects of the environment and the types of agents will be discussed in more detail below.

2.1 Environment

The world of the agents consists of a two-dimensional grid. A number of cells of the grid are randomly filled with food and obstacles. Cells can be occupied by one object at a time, meaning that each cell can be classified as food, obstacle or ground. An example of an environment can be seen in Figure 1. Obstacles are like quicksand: low to the ground and, with a lot of effort, one can go through these obstacles. The feeding places and obstacles reflect their own unique pattern of light, while ground reflects no light at all. Agents are able to classify each cell by looking at the reflections. Obstacles have a weaker reflection than food has: agents are able to detect food two cells away and obstacles one cell away from the current position. There also is a day and night rhythm in this world. At daylight this reflection is perfectly clear and so an agent does not make mistakes in classifying. At night, however, the reflections become fuzzy and agents will not be able to classify all the cells correctly. In the experiments a cell will be classified correctly at night with a chance of $\frac{2}{3}$. Since obstacles are low to the ground, these differences in reflection imply that food can be detected behind obstacles. The comparison with scent might be intuitive; one is able to smell a good meal being prepared, even when standing behind a closed door.

At the start of each simulation one agent is placed at a ground cell (i.e. an empty cell) in the environment. Agents are able to move to one of the adjacent cells of their current position. When an agent moves through one of the four boundaries of the environment, it reappears at the opposite side of the environment. Time passes in discrete steps and agents may act at each of these time steps.

Agents need food in order to survive. When the agent hits a feeding place, it will consume the food at that location. At this point, the food is no longer available at the current cell and a new food source will appear at a random location in the environment. When an agent hits an obstacle, some of its energy will be lost. In contrast to food, obstacles remain static during one run of the simulation.

Parameter	Value
Costs (sleeping/moving)	Reactive: 1/2; Reactive-DN: 2/3; Control: 2/3; Control-2: 1/2 (+extra costs)
Initial Energy Level	250
Hunger Threshold	240
Extreme Hunger Threshold	20
Environment Size	10x10 cells
Energy Increment by Food	10
Energy Decrement by Obstacle	15
Number of Feeding Places / Obstacles	10 / [0 . . . 80]
Maximum number of time steps	750
One full day cycle (day / night)	30 time steps (15 / 15)

Table 1: Parameter settings used in experiments

2.2 Agents

Agents have to survive as long as possible. Each agent has a certain amount of energy available. The agent dies when all of its energy is depleted, which means the agents has to search for food. However, moving around in the environment also costs some energy. Even worse, when the agent hits an obstacle, a lot of its energy is taken away at once. To make life not too hard, the agent is equipped with a light sensor that can be turned up to 360 degrees so it can detect obstacles and food (note that sensing does not cost any energy). It also has a motor system with which the agent can turn up to 360 degrees and move forward.

Four different agents are tested and these agents can be divided into two categories: reactive agents and control system agents. The main difference between the two categories is that control system agents have a higher control structure, like the prefrontal cortex in humans. This higher control structure, however, also uses some energy. This extra energy usage is not totally artificial. In humans, for example, the brain also uses a rather large portion of the energy available [9, 7]. Sections 2.2.1 to 2.2.4 give an overview of the different types of agents. Table 1 summarizes the parameter settings used in the experiments[1].

2.2.1 Reactive

The first type of agent is the reactive agent. This agent is designed according to the reactive paradigm. The design of the behavioral layers is displayed in the left-hand side of Figure 2 [1]. A brief explanation of each of the behavioral layers is given below.

- The Wander layer turns the agent to a random direction and then moves the agent in that direction.
- The Food Direction layer finds food in the surrounding of the agent based on the sensory input from the light sensor. The output of this layer is an excitatory link to the first layer.
- The Evaluate Hunger layer checks the energy level of the agent and, based on this reading, inhibits the Food Direction layer when the agent is not hungry.
- The Obstacle Avoidance layer takes its input from the light sensor and is then used to prevent the agent from running into an obstacle by changing directions when the agent is about to hit an obstacle.
- The Evaluate Extreme Hunger layer can inhibit the Obstacle Avoidance layer when the agent is almost starving. The agent may then still have a chance of finding food, although it has to move over an obstacle to reach it.

2.2.2 Reactive-DN

The Reactive-DN agent is also a reactive agent, but now with a built-in day and night rhythm. The reactive part is a copy of the one used in the Reactive agent. Having a day and night rhythm means that the agent knows when to go to sleep and when to wake up again. The agent therefore provides a good measure to compare the control system agents to. It is expected that if control system agents develop day and night rhythms, their results should be closer to the results obtained from the Reactive-DN agents than to the results

[1] See [5] for more informations on parameter settings and implementation details.

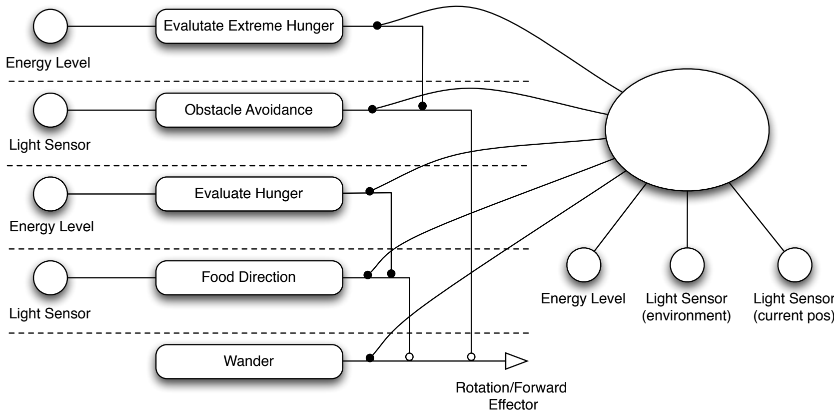

Figure 2: Behavioral layers and control structure of the Control System Agent. The control structure provides inhibitory links to the behavioral layers of the agent. The inputs of the control structure are two light sensors (one reading for the surrounding and one for the current location) and the energy level of the agent.

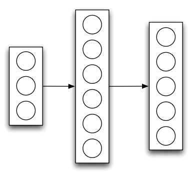

Figure 3: Topology of the multilayer perceptron. Three units take inputs (left layer) and propagate it to a hidden layer (middle layer, 6 units) which propagates it to the output layer (right layer, 5 units). The subsequent layers are fully connected, resulting in 48 connections.

of the Reactive agents. To make the results of the different categories of agents comparable, the costs of performing actions and going to sleep for this type of agent are equal to those of the control system agents.

2.2.3 Control

The Control agent is a reactive agent with a control system on top. This control system provides inhibitory links to the behavioral layers. With this control system, the agent can shut down behavioral layers based on the inputs it gets. The inputs of this control system are two light sensor readings (see below) and the energy level of the agent. Figure 2 illustrates the control system in combination with the behavioral layers.

The control system is a feedforward multilayer perceptron. The topology of the network is displayed in Figure 3. The different layers are represented by rectangles and the circles in these rectangles are the nodes. The arrows between layers indicate full connectivity between these layers. The network takes three inputs: the energy level of the agent, and two readings from the light sensor. The first reading of the light sensor is from the surrounding and the second reading is from the current position of the agent. The difference between the two readings is that it is assumed that an agent is able to correctly identify on what type of ground it stands (the current position), even when it is dark. The weights of this network are optimized by the evolutionary algorithm described in Section 2.3.

The network takes real-valued inputs. Biases are set to 0 and three different activation functions are used: the input layer uses the identity function, the hidden layer uses the hyperbolic tangent (tanh) and the output layer makes use of the logistic sigmoid function. The outputs lie in the interval $[0, 1]$ and this property is then used to interpret the outputs as probabilities. The higher the output, the higher the probability of deactivating the behavioral layer to which the output link is connected.

2.2.4 Control-2

The fourth and last type of agent to be discussed is the Control-2 agent. This type of agent is the same as the control system agent, with the difference that the additional cost of the higher control structure is not a fixed value, but is dependent on the output of the control system. Each active output link, or deactivated layer, adds some small value to the cost of having a higher control structure. If no links are active, there is also no extra cost involved. Each active output link adds an amount of 0.20 (one divided by the total number of output links) to the energy consumption. When all behavioral layers are inhibited and thus all output links are active, the extra energy consumption is 1. An interesting observation is that the Control-2 agent will already use more energy than a Reactive agent when inhibiting just one layer.

Please note that the Control-2 agent needs to inhibit at least two layers in order to sleep: the Wander layer and Food Direction layer. As can be seen in Figure 2, the inhibitory link that operates on the Wander layer is connected *before* the excitatory links from the higher level layers. This means that if the Wander layer is inhibited, it is still possible for the Food Direction layer to move the agent to a food source. Therefore, both layers need to be inhibited to put the Control-2 agent to sleep.

2.3 Evolution and Simulation

An evolutionary algorithm is used to optimize the weights of the neural network of the control system agents. A genome consists of 48 genes which represents the weights of all 48 connections of the neural network. The population size is fixed to 125 individuals. The genes are initialized with random *integer* values in the interval $[-300, 300]$. Integer values are used to reduce the search space. The double weighted values for the neural network are then calculated by dividing each gene by 100. Weights of the network thus lie in the interval $[-3.00, 3.00]$. Fitness is calculated as follows. The genome is first converted to a neural network which is then given to an agent. Next this agent goes through one simulation in a random environment. The number of time steps the agent survived in this environment is taken as the fitness. Since the number of time steps an agent may live is constrained, the fitness value is also bounded: $[0, 750]$. The evolutionary algorithm is implemented with the use of JGAP [8] in which the mutation probability was set to $\frac{1}{12}$ and one-point crossover is used for recombination[2]. After 400 generations the evolutionary algorithm is terminated and the best individual is selected as the one that will be used in the final simulation.

Results are gathered for each type of agent by averaging over 5000 simulations. The other settings of the simulation can be found in Table 1.

3 Results

This section first presents the results for agents in different environments (i.e. the number of obstacles is varied). Next the results are further analyzed by means of visual inspection of the simulations and data analysis.

3.1 Effectiveness of Control System

Figure 4 shows the results of the different types of agents. Agents are placed in environments with a fixed number of 10 feeding places. The number of obstacles is varied (horizontal axis) and the fitness of agents is measured in different environments (vertical axis). Environments with more feeding places tend to be too friendly; agents may constantly eat. Environments with few feeding places are hard to survive in. The number of obstacles shows this effect too, although in reverse order, as can be seen in Figure 4: environments with no obstacles are friendly, while environments with a lot of obstacles are hard to survive in.

Note that in environments with no obstacles sleeping behavior is suboptimal. Figure 4 shows that both Reactive-DN and Control perform worse than Reactive and Control-2 in environments with no obstacles. Visual inspection of the simulations shows that both Control and Control-2 agents apply the same strategy as the purely reactive agents: when there are no obstacles, there is no need to sleep. The differences in performance between Control and Control-2 can be attributed to the extra costs associated with Control agents for having a higher control structure.

Varying the number of obstacles provides a measure to indicate whether or not the control system works in different environments. It is interesting to compare the graph of the Reactive-DN agent with the graphs of Control and Control-2 agents to see how similar they are. When compared with the Control agent, it is found that Reactive-DN performs better than the Control agent in all cases. However, the performance of both agents is qualitatively similar. The line of the Control agent follows the same trend as the Reactive-DN

[2]The default configuration of JGAP was used. Details on this configuration can be found in [5].

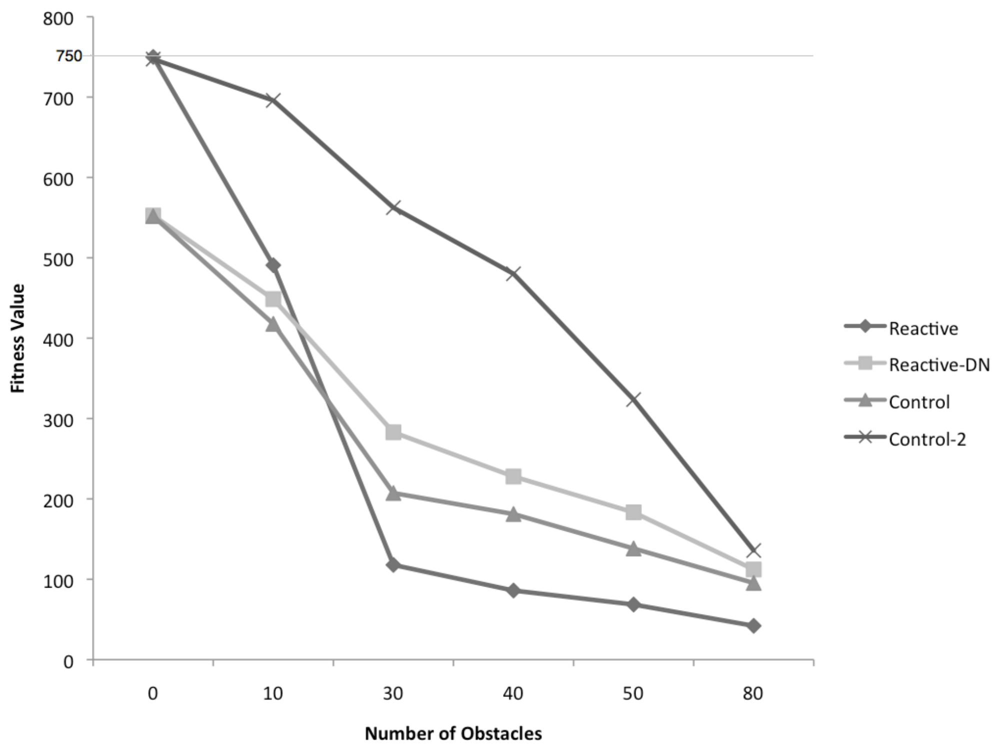

Figure 4: Visualization of the fitness (vertical axis) of different types of agents with a varying number of obstacles (the horizontal axis). The number of feeding places is held constant at the value of 10.

agent. The Control agent will most likely due to early stopping of evolution only approach the optimal behavior (i.e. the Reactive-DN behavior). The same holds for the rather large difference in performance around the points of 30 to 50 obstacles. This hypothesis was verified by increasing the number of generations to 1000 and doubling the population size.

When comparing the Reactive-DN agent to the Control-2 agent, two surprisingly different graphs are found. The Control-2 agent performs a lot better than the Reactive-DN agent in almost all cases. Apparently the Control-2 agent has learned a strategy to save energy. The cost in energy for a Control-2 agent to inhibit one behavioral layer is low. The results show that the agent has learned to only inhibit the necessary layers. For example, if sleeping would be the appropriate behavior given the inputs, only the Wander and Food Direction layers need to be inhibited. The other three layers are then inhibited implicitly (see Figure 2).

3.2 Circadian Rhythm

Since the results described above showed that control system agents perform better in most environments, an analysis can be made of the simulations to see whether or not this effectiveness is due to the development of a circadian rhythm. When agents develop a circadian rhythm they should be in a sleeping state at night, while searching for food at day. The analyses performed on the simulations are a visual inspection and a data analysis. Both are discussed below.

3.2.1 Visual Inspection

Ten environments of the simulation of control system agents are visualized and inspected (see Figure 1 for a visualization). Reactive agents and Reactive-DN agents are not inspected in detail since their behavior is already known. It is found that the control system agents develop a day and night rhythm. At day they are actively searching for food, but at night they mostly remain in their current position.

The Control-2 agents show, next to the day and night rhythm, some other, unexpected behavior as well. When the energy level of the Control-2 agent is high (about 100 units or more), the agent may behave as if it is night: it does not move around in the environment to look for food. Or to put it in a wake and sleep

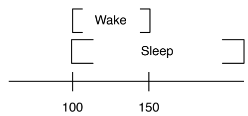

Figure 5: The ranges in which the Control-2 agent will take a siesta and wake up again. From circa 100 energy units and up the agent will take a siesta. Somewhere in the range of $[100, 150]$ energy units the agent will wake up again to search for food.

	Reactive	Reactive-DN	Control	Control-2
Day Actions	52.8%	52.2%	51.5%	49.9%
Night Actions	47.1%	1.4%	0.7%	0.5%
Day Sleep	0.1%	0.1%	0.6%	1.6%
Night Sleep	0.0%	46.3%	47.2%	48.0%

Table 2: Average number of steps in percentages.

context: the agent takes a siesta. As can be seen in Figure 5, the agent may take a siesta when the energy level is 100 units or higher. However, when the energy level is in the range of 100 to 150 units, the agent may also wake up again. This means that the agent will take a short siesta when the energy level is in the range of 100 to 150 units and a longer one when the energy level is higher than 150 units.

This siesta behavior is interesting, because, compared to the Control agent, the Control-2 agent shows more complex behavior than just a day and night rhythm. There is no difference in the design of the higher control structure, so the difference in behavior needs to be attributed to other factors. The defining characteristic of Control-2 agents is that the additional costs for using the higher control structures is based on how much this structure is used (i.e. how much of the behavioral layers it inhibits). It is highly likely that by inhibiting the Wander and Food Direction layers (see Figure 2), and thus effectively putting the agent to sleep, the agent uses less energy than when it moves around in the environment. However, at some point, looking for food will be necessary in order to survive and the agent wakes up again.

3.2.2 Data Analysis

The number of times agents sleep and move per phase of the day are counted and can be found in Table 2. These results indicate that control system agents are close to the Reactive-DN agents with respect to their sleeping behavior.

It is interesting to note that the unexpected behavior of Control-2 agents – taking a siesta – can also be seen in the data. The average time of sleeping during the day is still small, but, in comparison to the other types of agents, a lot higher. This suggests that sleeping at day when having enough energy is not just fluke behavior that only happens in a few runs, but that it is in fact systematically occurring in Control-2 agents.

Another surprising result is that the Control agent also seems to take a siesta once in a while. The percentage of sleeping at day is much lower than in Control-2 agents, but higher than in Reactive and Reactive-DN agents. The reason for this behavior is not as clear as with the Control-2 agents. Visual inspection of the simulations with the Control agent shows that agents with very low energy levels and not located near food sources, tend to sleep more at day. By doing so, the agent may live a few steps longer, since it is not using energy for movement. This behavior can be seen as suicide through inertia. Note that this happens only in a few cases and that no research was done to investigate this behavior further.

3.3 Traffic Regulator

The results described above show that a day and night rhythm is developed by agents with a control system and that this day and night rhythm is advantageous to the agents. It is also found that the number of obstacles can be set to a wide range of values without having a very large influence on performance of the control agents. This shows the robustness of the control system agents. One other observation is that the effectiveness of the control system agents is due to sleeping behavior. Agents are able to learn when it is most beneficial to them to go to sleep and when to wake up again. The question now is to what extent do these results support the traffic regulator hypothesis?

4 Discussion

In this paper the hypothesis of behavioral control as a form of traffic regulation was tested by placing agents with higher control structures in environments with a day and night rhythm. The agents had to survive as long as possible by eating food and avoiding obstacles. This goal was easy to achieve at day, but at night the agents were not able to clearly perceive the environment due to the darkness. Two categories of agents were tested: reactive agents and control system agents. The control system consisted of a neural network with inhibitory output links to the behavioral layers of the agent. The inputs of the network are the agent's bodily state and perception of the environment. Results showed that the control system agents were able to effectively adapt to their environment by developing a rhythm in which agents rested at night and searched for food at day. The overall performance of control system agents was better than that of reactive agents.

The higher control structures in the control structure agents can be seen as traffic regulators. The higher control structures do not generate behavior, but merely assist in the inhibition of appropriate layers based on cues from the agent's body and environment. For example, when it is dark the behavioral responses will be limited to sleeping. Inputs that otherwise would result in some action are now ignored (e.g. the food searching and eating). Another example is that when no food is available in the environment, there is no need to move around looking for food. Even when it is daylight and everything is perfectly visible, a search for food would result in nothing but an energy loss. The control structure learns, by trial and error, that inputs should be ignored and sleeping is the best behavior given the environment.

The results are now based on the costs structures as listed in Table 1. Future work might focus on finding more plausible cost structures to make results slightly more generalizable. One way of finding more plausible cost structures is to base the costs on neurophysiological data of the human brain metabolism (e.g. [7]). Another way would be to base cost structures on the processing time they require on a computer.

References

[1] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, 1986.

[2] P. Durr, C. Mattiussi, and D. Floreano. Genetic representation and evolvability of modular neural controllers. *IEEE Computational Intelligence Magazine*, 5(3):10–19, 2010.

[3] W. F. G. Haselager, J. van Dijk, and I Van Rooij. *A lazy brain? Embodied embedded cognition and cognitive neuroscience*, pages 273–290. Oxford: Elsevier, 2008.

[4] E. Koechlin, C. Ody, and F. Kouneiher. The architecture of cognitive control in the human prefrontal cortex. *Science*, 302:1181–1185, 2003.

[5] S Lagarde. Evolving a circadian rhythm. http://www.ru.nl/kunstmatigeintelligentie/onderzoek/weird/student_projects/vm_students_projects/finished_projects/, 2009. Bachelor's thesis.

[6] R. R. Murphy. *Introduction to AI Robotics*. Intelligent Robots and Autonomous Agents. The MIT Press, 2000.

[7] G. Oz, E. R. Seaquist, A. Kumar, A. B. Criego, L. E. Benedict, J. P. Rao, P. Henry, P. van de Moortele, and R. Gruetter. Human brain glycogen content and metabolism: implications on its role in brain energy metabolism. *American journal of physiology. Endocrinology and metabolism*, 293(5):E1378–E1384, 2007.

[8] N. Rotstan. JGAP: The Java Genetic Algorithm Package. http://jgap.sourceforge.net/, 2009. Version 3.4.3.

[9] R. G. Shulman, D. L. Rothman, K. L. Behar, and H. Fahmeed. Energetic basis of brain activity: implications for neuroimaging. *TRENDS in Neurosciences*, 27(9):489–495, August 2004.

[10] J. van Dijk, R. Kerkhofs, I. van Rooij, and W. F. G. Haselager. Can there be such a thing as embodied embedded cognitive neuroscience? *Theory & Psychology*, 18(3), 2008.