
On the development of spectral properties of visual agent receptors through evolution

Achim Liese

Institut für Informatik
Johannes Gutenberg-Universität
D-55099 Mainz, Germany
liesa000@mail.uni-mainz.de

Daniel Polani

Institut für Neuro- und Bioinformatik
Medizinische Universität
D-23569 Lübeck, Germany
polani@informatik.uni-mainz.de

Thomas Uthmann

Institut für Informatik
Johannes Gutenberg-Universität
D-55099 Mainz, Germany
uthmann@informatik.uni-mainz.de

Abstract

In this paper we study a model for the evolution of the spectral sensitivity of visual receptors for agents in a virtual environment. The model uses a Genetic Algorithm to evolve the spectral characteristics of the sensors along with the agent control mechanism by requiring the agents to find certain objects with a given emission spectrum. The studies extend previous results and show that the GA is able to find a balance between sensor costs and agent performance in such a way that the spectral sensor sensitivity reflects the emission spectrum of the target objects.

1 INTRODUCTION

Agents in a given environment usually live under the restriction of limited resources, e.g. food, space or the number of reproductive partners. An important factor within the struggle for survival is the access to information about certain aspects of the current state of the environment. Sensors have to be used to exploit this information resource. This particularly includes the ability to analyze sensor information by the agent's control unit.

In recent time there has been an increasing research effort to evolve artificial sensor channels in simulated and in real-world (hardware) environments. In [4] the construction of new sensors in real devices are considered and in [16] an example by constructing an artificial ear is provided. Lund et al. evolve ears for the Khepera robot [9] and Lee et al. simulate an agent acquiring distance information from any number of sensors [7]. There also exist several approaches coding some sensor parameters genetically [5, 11, 18] or switching off given sensors [2, 14]. Vaario et al. intro-

duce a complex model forming agents and sensors by production rules [20].

The evolution of visual sensors enjoy a particular interest since, in spite of the complexity they seem to require to be evolved, a variety of 40-60 independent lines of descent have been observed in nature [19]. [13] studied the morphology evolution for a simple eye model and were able to reconstruct typical morphological milestones found in nature.

Another important property of visual sensors in living beings is their limitation to a certain range of the electro-magnetic spectrum. In principle, the biological substrate would allow the realization of wider ranges of sensitivity [17]. It therefore seems likely that certain environmental conditions lead to the observed limitations of spectral sensitivity. E.g. it is more important to be able to see the light clearly that is emitted by food or enemies than light emitted by other objects. If the eyes are particularly sensitive in this spectral region, one expects this to increase the survival probability of the individual.

Our goal is to develop a simulation framework suitable to study this question. To model the evolution of sensors a fitness function can be used that directly quantifies the performance of a sensor. [13], for instance, used the visual acuity as fitness measure. However, it is not always clear which factors in the end turn out to be actually responsible for the survival of an individual. Chameleons, for instance, estimate the distance to a target insect by the accommodation of their eye lens [15]. Accommodation speed is therefore vital to the chameleon's strategy of survival.

This is but one example for the variety of sensor properties that may prove relevant for survival in nature. Therefore, in our simulation framework we try to avoid a bias towards a particular choice of sensor quality measure that would act as fitness function for the evolution process. Instead, our sensor selection

mechanism is based solely on the success of an agent equipped with this sensor. In our model, the quality of a sensor therefore is only determined in an indirect fashion, suffers a delay, is noisy and — this is important — also depends on the agent control unit which *co-evolves* with the sensors to find the best action sequence.

Our goal is it to identify mechanisms able to develop or adapt sensors in a suitable manner for tapping information present in the environment. The model will also take into account the costs incurred by the effort expended in opening up qualitatively new sensory channels. In the setting studied in this paper we introduce costs for sensor use to exert pressure on the agents to handle their resources economically, i.e. so that they only employ sensors important for survival. This paper provides an extension and interpretation of previous preliminary results [8]. In Sec. 2 the paper first gives a self-contained description of the simulation model. In Sec. 3 the experimental settings are provided. The results are presented and discussed in Sec. 4, and the paper is summarized in Sec. 5.

2 GENERAL MODEL PROPERTIES

The *agents* in our simulation move in a continuous 2-dimensional world. They are controlled by a simple linear neural network whose inputs are fed by the sensors and whose outputs feed into a motor activation (see Sec. 2.1 for more details). When the agents are created (e.g. by a Genetic Algorithm, Sec. 2.2), they start with a certain initial life energy which is used up during time. When the life energy drops below a certain threshold, the agent dies. Apart from agents there are other objects in the simulated world, called *lamps*. If an agent is close enough to a lamp, its life energy is replenished at a certain rate. The life energy balance of an agent and thus its life span depends on the ability of the agent to convert sensor readings to effective motor actions.

In the GA run we do not use any explicit fitness function. The selection mechanism is completely based on the performance of the agents, i.e. determined by their life span, since agents which live longer will have more opportunities to procreate. In particular, we do not select individuals on the basis of some measure applied directly to the sensors. All that is relevant to selection is the ability of the sensors, integrated with the controller, to obtain an energy balance sufficient to survive. In principle the agent solves a delayed reinforcement learning problem, namely maximizing its lifespan, to evaluate the quality of its sensors in conjunction with its controller. Thus, the neural network

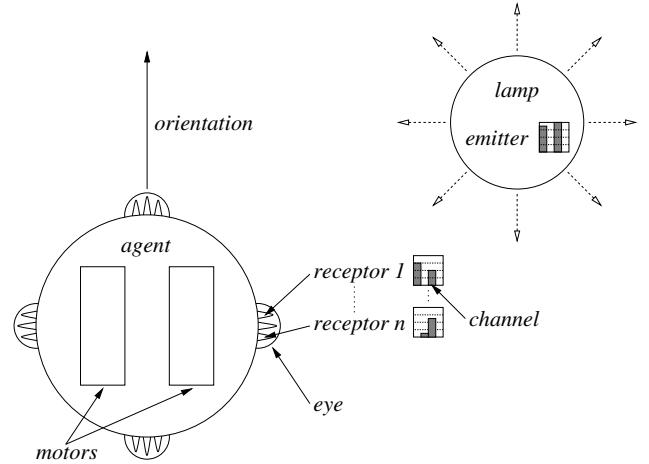


Figure 1: Description of the model

of the agent has to coevolve with its sensory equipment to be able to utilize the information conveyed by sensors and to adapt to possibly changing sensor configurations. This is a general key necessity for flexible models of sensor evolution study (e.g. [6, 10]).

In the following sections we will describe the experimental scenario in more detail. Section 2.1 presents the structure of the agents, their sensors and their control and Sec. 2.2 gives some details about the GA used to evolve the agents.

2.1 SCENARIO DETAILS

The simulation is performed using the XRAPTOR tool as environment [12]. As mentioned above, our scenario contains two kinds of objects, *agents* and *lamps*, which we will describe in the following.

The agents are modelled after vehicle 3c from [3]. They are driven by two motors, one to the left and one to the right of their orientation axis, respectively (see Fig. 1). The motor power can be set to a real-numbered value between -1 and 1 . When set to the value 1 , the respective motor is running forward with maximum speed, when set to -1 , it runs backward with maximum speed. If both motors are set to the same positive (negative) value, the vehicle will move forward (backward) in the direction of its orientation axis. If both motors are set to different values, the vehicle will, apart from the resulting net movement, rotate around its center. In particular, this means that the motor settings are directly transformed into a velocity value. In other words, the movement corresponds to an overdamped dynamics.

The lamps are further objects in the simulated world.

In their neighborhood the agent can replenish its life energy. An agent can move not only towards, but also into a lamp.

For the agents to be able to find their way to the lamps, we have to equip them with appropriate sensors, define how the lamps are detected by them and how the agent control may transform the corresponding information into an action.

The agent sensors we use in our simulations are an extension of the sensors used in [10]. The agents possess eyes at four positions of the agent periphery (Fig. 1). Each eye is oriented outwards with respect to the agent center. A *visible* object evokes a signal I_b in the eye which is larger for a closer object (and smaller for an object farther away). Also the eyes possess an orientation sensitivity, such that the signal evoked by a visible object is largest if the object is in direction of the eye orientation and becomes smaller if the object is farther away from the eye orientation axis.

Visibility of an object derives from the spectral compatibility of an observed object and the sensors it is detected with. Every object in the simulation emits a spectrum via an *emitter* which is characterized by an m -dimensional real-valued vector $v_{em} \in \mathbb{R}^m$. Every entry of this vector represents the intensity of the emitted light in a specific wavelength interval. Those entries are called *channels*.

The sensors contain a corresponding structure. An agent eye includes n *receptors*. Each receptor is defined by an m -dimensional vector v_{rec} where each entry represents the input sensitivity for the corresponding spectral channel. Each receptor returns a single real-valued scalar

$$I = f_{rec}(\langle f_{cha}(v_{rec}), v_{em} \rangle) \cdot I_b \quad (1)$$

as signal, where f_{rec} and f_{cha} are filters (see Sec. 3 for details), v_{em} denotes the emitter characteristics of the observed lamp, $I_b \in \mathbb{R}$ is a raw intensity factor depending on the distance of the lamp and its angle w.r.t. eye orientation and $\langle \cdot, \cdot \rangle$ denotes the scalar product. If there is more than one lamp inside the visibility range, I_b is obtained by an additive superposition of the raw intensities (see e.g. [10] for details).

The signals generated by the different receptors are directly fed into a linear “neural” network that in turn feeds the motors (Fig. 2). As the agents have to solve a delayed reinforcement learning problem, the weights of the network are not trained by any network learning rule, but instead adapted by the GA (Sec. 2.2).

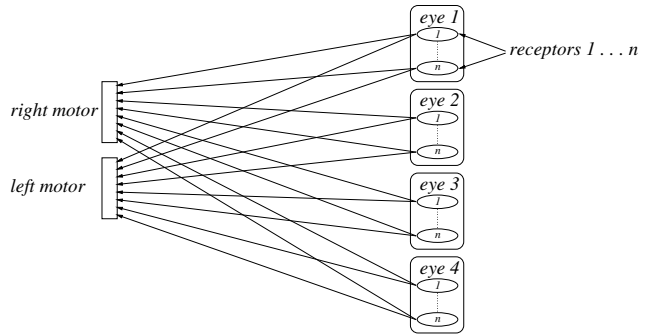


Figure 2: The linear neural network controlling the motors

2.2 THE GENETIC ALGORITHM

The GA does not use an explicit fitness function, instead its selection operator is determined by the agent energy balance, i.e. by the differential survival rates arising from energy considerations. The parameters evolved by the GA are the weights of the linear neural network and the receptor characteristics v_{rec} . Each chromosome encodes the properties of one agent. The real-valued weights and vector entries are encoded as binary substrings of the GA chromosome and do not change during its lifetime.

The initial population is generated randomly. A genetic cycle consists of the creation of a number of agents, until the population has size `pop_size`. The newly created agents are set to a random starting position inside a given square region of space. Then the simulation is started during which agents begin to die. A generation step takes place when the number of agents in the population drops below a given threshold. Then new agents are created by recombination of surviving agent chromosomes and by *immigration* (i.e. by creating new agents with random chromosomes) until the population has again `pop_size` individuals. To increase the selection pressure, the lamps are randomly repositioned after a certain time step t .

In addition to the life energy depletion we introduce an operating cost for every receptor activated by the GA. In particular, every receptor entry (i.e. channel sensitivity) has a cost component proportional to the sensitivity, and an extra cost bias for being non-zero. By this we intend to model the additional cost of implementing different qualitatively new sensor channels, while for channels already open the additional cost for an increase of sensitivity is not considered as high. These additional sensor costs influence the energy balance of an agent. GA evolution has the double task to create useful sensor configurations and neural network

weights capable of utilizing those sensors to be able to find the lamps quickly. We now wish to study under which conditions it is possible to evolve agents whose receptor characteristics reflects the lamp emitter characteristics.

3 EXPERIMENTS

In our runs we specialize the settings from Sec. 2.1 and Fig. 2. We use one *receptor* ($n = 1$) with 4 *channels* ($m = 4$). See Fig. 1 for an overview over the different elements of our model.

We used a GA with a rank-based selection and an immigration rate of 30%. The population size was 60 agents. Every time the number of living agents fell under 20 individuals, a new generation was created. The agents had an initial energy of 6,000,000 ($\pm 1\%$ random noise) and died when their energy dropped below 60,000. Their life energy depletion per time step was set to 7000. There were no extra costs for motor activation, but each activated channel had a base cost of 500 plus 500 times the input sensitivity. Figure 4 shows the internal representation of a chromosome for an agent used in our runs. Each value is represented in the chromosome as an 8 bit word which is converted to a decimal number between 0 and 1.

In the runs presented here we used 15 lamps that all had the same emitter and the agents did not emit any light. So their only task was to find the lamps as fast as possible while using as little energy as possible.

In order to obtain a broad statistical basis, we used different lamp emitter spectra and performed 3000 runs with different random seeds for each spectrum. In the first configuration the emitter of the lamps had only one activated channel ($v_{em} = (0, 1, 0, 0)$). In the second configuration two were activated ($v_{em} = (0.8, 0, 1, 0)$). A run ends after $t=31000$.

During the runs the development of the population is logged. The results will be shown in Figs. 5 and 8 and discussed in detail in Sec. 4.

In order to ensure that agents were capable of locating and moving towards lamps, the lamps were regularly repositioned. Consequently, agents with better matched sensors and control networks experienced an advantage because they required shorter time to find the lamps.

For $t \leq 3000$ the lamps were not repositioned. The agent population had thus some time to evolve a basic ability of finding lamps. After $t = 3000$ the lamps were repositioned every 1000 time steps, after $t = 10000$ repositioning took place every 500 time steps and af-

ter $t = 20000$ every 250 time steps. This strategy increased the selection pressure, i.e. it forced the agent population to find the optimal balance between cost and input sensitivity of their sensors necessary to find the lamps as quickly as possible by using as little energy as possible.

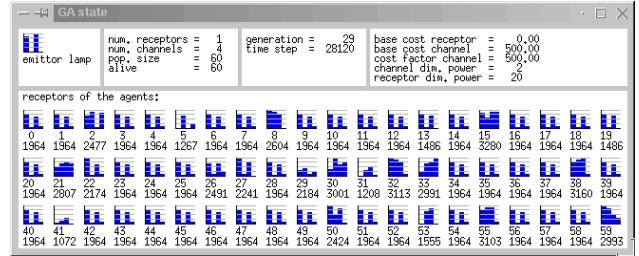


Figure 3: GA state window

Figure 3 shows a screenshot obtained in a typical run. The window displays the receptor characteristics and relevant parameters of the current agent population. In the following we give a detailed description of the parameters from left to right and from top to bottom.

emitter lamp: emitter characteristics v_{em}

num. receptors: number n of receptors of each agent

num. channels: number m of channels of each agent

pop. size: maximum population size

alive: current population size

generation: current generation

time step: current time step t

base cost receptor: base cost for each receptor that is activated, i.e. that has at least one channel with an input sensitivity larger than 0. This quantity is only relevant if more than one receptor is being used. This acts as fixed cost, as we use only one receptor in the current runs.

base cost channel: base cost for each channel that is activated, i.e. that has an input sensitivity larger than 0

cost factor channel: cost factor, by which the input sensitivity of each channel is multiplied to calculate the cost for this channel.

channel dim. power: in Eq. (1) f_{cha} takes each component of v_{rec} separately to the power given by **channel dim. power**. This is used to enhance the contrast (diminuate small values, intensify large ones) and as incentive to set the input sensitivity to a value as high as possible.

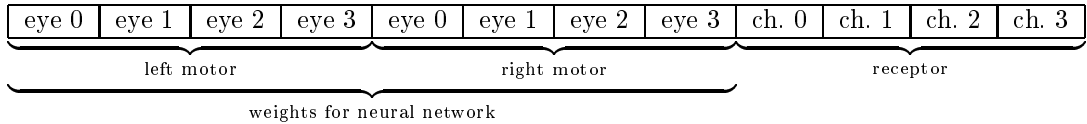


Figure 4: The internal representation of the chromosome. Note that we use only one receptor. Therefore there is only one connection from an eye to the corresponding motor and only one section specifying the $m = 4$ receptor channels.

receptor dim. power: in Eq. (1) f_{rec} takes the result of the scalar product to the power given by **receptor dim. power**.

While **channel dim. power** acts as individual, **receptor dim. power** serves as collective contrast enhancer.

receptors of the agents: this entry shows the receptor characteristics v_{rec} of the current agent population. A fully activated bar corresponds to an entry value of 1. Each agent’s index and sensor cost is given below the bar diagrams.

In the long block on the bottom of Fig. 3 the receptors and their costs for all agents of the population are shown. Below each bar graph in the first line the index of the agent, in the second line the value of the total cost for this receptor is given. It can be observed that in a significant part of the 60 agent population the receptor characteristics reflects the characteristics of the lamp emitter.

4 RESULTS

We emphasized in the introduction that the evolution of the agent population and in particular the sensor properties would not be driven by an explicit fitness function. In our runs only agent performance is relevant for selection. On the other hand, for an analysis and interpretation of the resulting sensors, it proves to be very useful to introduce an adequate measure.

The measure we use in our analysis gives a kind of correlation between v_{rec} and v_{em} , i.e. how well the agents’ spectral sensitivity matches the spectral emitter characteristics. We define a *specificity* S as measure for the “similarity” between a receptor and the emitter of the lamps for each agent, using the following formula:

$$S(v_{\text{rec}}, v_{\text{em}}) = \cos \angle(f_{\text{cha}}(v_{\text{rec}}), v_{\text{em}}) \quad (2)$$

where $\angle(.,.)$ denotes the angle between two vectors. The plots in Figs. 5 and 8 show the development of the population over time. The plots display the number

(*frequency*) of agents that die at a given time step t and have a given specificity S . This serves to trace the number of agents with a given S present in the population at a given time. To obtain this number we bin the time step/specificity (the t/S) plane into a 50×50 -grid. Note the different scales for the frequency axes in Figs. 5 and 8.

These plots show the increasing adaption of the agent population to the simulated environment. Fig. 5 for the lamp emitter with *one* activated channel is probably the easiest to interpret. It shows for all time steps t of the simulation an accumulation of agent death events at $S \approx 0$ and $S = 1$. These values correspond to channel vectors which are orthogonal or collinear, with respect to the lamp emitter characteristics. Each generation random *immigrant* agents are created with a probability of 30%. The peak at $S \approx 0$ is caused by these agents as a randomly generated receptor vector will yield a specificity close to zero for a given non-zero emitter vector (see also Fig. 6 and its description further below). At $S = 1$ one has a receptor vector that perfectly matches the lamp emitter characteristics. Since only one channel is active in the emitter, it is easy for the GA to find the optimal receptor setting which switches on the corresponding receptor channel and turns off the rest, saving sensor costs. At the beginning of the runs there are still some agent death events with S between 0 and 1 (somewhat difficult to visualize in the space available for Fig. 5), but during the run those values become increasingly scarce. Several changes in the lamp repositioning frequency take place during the runs. When the frequency doubles to a repositioning every 500 steps at $t = 10000$ the character of the $S = 1$ peak changes. The variance of the $S = 1$ peak which is caused by lucky survivors reduces (this can be also seen much more pronouncedly in high resolution plots not shown here because of lack of space), as with the higher lamp repositioning frequency the agents must be more efficient in finding lamps. At the next frequency doubling at $t = 20000$ the efficiency of the agents has to improve again, as the agents have only 250 time steps to reach the lamps. The jump in the $S = 1$ peak shows that an even larger

number of agents has now switched to match the emitter characteristics exactly. In other words, Fig. 5 shows that after some time and exerting appropriate selection pressure the agent population has learned the task and has learned to match the sensor to the lamp for this purpose.

As we activate 2 channels in the emitter, the situation is more difficult to interpret. As opposed to before, in Fig. 8 the $S = 0$ peak has vanished and instead one finds 3 peaks for $S \approx 0.62$, $S \approx 0.78$ and $S \approx 1$. Due to the different scaling as compared to Fig. 5 here also the distribution of values of S between the peaks and particularly between $S = 0$ and $S = 0.62$ can be seen clearly. Again, we see the different phases of evolution: the first phase, before $t = 10000$ with no clear stabilization of S ; the intermediate phase before $t = 20000$ where the peaks show some more stable behavior; and the phase where the agents have learned the task efficiently and the favored channel settings have stabilized.

The presence of several active channels generates a diversification of possible solutions for the sensor configurations. At first the cause for that particular peak structure was not clear to us. Particularly striking was the fact that the characteristic peaks were always found at the same value of S for all 3000 simulation runs. We therefore had to clarify whether it was a particularly favorable receptor configuration or caused by the particular structure of search space. To answer this question we calculated the a priori *state density* of S and compared it with the density obtained by the GA.

The state density is the probability for obtaining a particular value $s \in [0, 1]$ for S in an a priori unbiased population. The “raw” state density is defined by

$$\hat{d}_S(s) = \frac{|\{ (i, j, k, l) \in \mathbb{J}^4 \mid S(v_{\text{rec}}^{(i,j,k,l)}, v_{\text{em}}) = s \}|}{|\mathbb{J}^4|} \quad (3)$$

where $v_{\text{rec}}^{(i,j,k,l)} = (i/256, j/256, k/256, l/256)$, $\mathbb{J} = \{0, \dots, 255\}$ and $|\cdot|$ denotes the number of elements in a set. It follows that $\hat{d}_S(s) \neq 0$ only for discrete values of s . We then bin this raw density into a histogram of 10000 bins to obtain the smoothed state density $d_S(s)$. Figures 6 and 7 show these calculated densities for the different lamp emitter spectra we used in our runs.

To compare the state densities of the evolved populations with the calculated state densities of the unbiased populations, we used the data from our logfiles and created 2D-histograms (Figs. 9 and 10). For these histograms we used all events for $25000 \leq t \leq 31000$ to

improve the quality and enable a resolution of 10000 like for the calculated density $d_S(s)$. This choice proved to be suitable, because the characteristics of the data in this time interval changes only very slightly.

Fig. 6 shows the a priori state density for a single activated emitter channel. A massive peak at $S = 0$ is seen and a very small peak at $S = 1$. The structure of the state density is exactly reproduced by the GA runs (Fig. 9), but now the $S = 0$ peak is much smaller, while the $S = 1$ peak is much larger. This reflects clearly how the GA adaption supersedes the unbiased search space structure, since $S = 1$ represents the well-adapted individuals.

This is even more prominent in the case of two active emitter channels. The positions of the peaks of S at 0.62, 0.78 (Fig. 10) match exactly the peak values calculated $d_S(s)$ (Fig. 7) and though the frequency results from the GA runs are much noisier than in Fig. 9, it is clearly seen that in the GA runs there is a strong bias towards higher values of S : the slightly smaller peak at 0.78 becomes much stronger than the peak at 0.62. In addition, the GA runs also show a prominent peak at $S = 1$ as opposed to the slight peak just below $S = 1$ in the plot for $d_S(s)$. This shows two things: the peaks in the density for S observed in the GA runs result from the structure of the search space and the learning task biases the evolved receptor sensitivities towards higher values of S , and to a significant amount towards $S = 1$, i.e. towards a good match between emitter and receptor characteristics.

We performed also further experiments with three active emitter channels (not shown here because of space limitations). For this case, the state density structure becomes even more complex, but several of the observations can still be made. The peaks of d_S are still reflected as peaks in the S histogram of evolved agents. In addition, they display a tendency to stronger populate regions that have a higher specificity. This shows that even for three active channels the selection mechanism favors receptors that better match the emitter. However, some new structures emerge in the case of three active channels: the GA histogram shows further peaks that correspond to specificity values where d_S has an infinite slope. The reason for this effect is not yet understood.

5 SUMMARY

We presented a model for the evolution of the spectral sensitivity of visual agent receptors. An important aspect was to avoid the direct optimization of the spectral sensor properties via an explicit sensor fitness

function. Instead, agent populations have to evolve in a given scenario. In particular, this means that, along with an adaptation of the agent control networks, also an adaptation of the sensory equipment has to take place as integral part of the evolution process. I.e. no subjective direct fitness criteria imposed by the experimenter are used to evolve the sensors, but their usefulness in helping an agent to solve the given task.

The runs show that our scenario proved suitable to study specific aspects of sensor evolution. In particular, spectral receptor characteristics were successfully evolved to match the characteristics of the emitters of the lamps to a significant degree. These observations show that it is possible for an evolutionary model to attain a natural match between the characteristics of an environment and the characteristics of agent sensors without explicitly optimizing particular sensor properties.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for helpful hints.

REFERENCES

- [1] C. Adami, R. Belew, H. Kitano, and C. Taylor, editors. *Proc. of Artificial Life VI, Los Angeles, June 26-29*. MIT Press, June 1998.
- [2] K. Balakrishnan and V. Honavar. On sensor evolution in robotics. In J. Koza, D. Goldberg, D. Fogel, and R. Riolo, editors, *Proc. of the first Int. Conf. on Genetic Programming (GP-96)*, pages 455–460. MIT Press, 1996.
- [3] V. Braitenberg. *Vehicles: Experiments in Synthetic Psychology*. MIT Press, Cambridge, 1984.
- [4] P. Cariani. Some epistemological implications of devices which construct their own sensors and effectors. In F. J. Varela and P. Bourgine, editors, *Toward a Practice of Autonomous Systems: Proc. of the First European Conference on Artificial Life*, pages 484–493, 1992.
- [5] D. Cliff, I. Harvey, and P. Husbands. Explorations in evolutionary robotics. *Adaptive Behavior*, 2(1):73–110, 1993.
- [6] B. Hutt and D. Keating. Artificial Evolution of Visually Guided Foraging Behaviour. In Adami et al. [1], pages 393–397.
- [7] W.-P. Lee, J. Hallam, and H. H. Lund. A hybrid GP/GA approach for co-evolving controllers and robot bodies to achieve fitness-specified tasks. In *Proc. IEEE 3rd Int. Conf. on Evolutionary Computation*, NJ, 1996. IEEE Pr.
- [8] A. Liese, D. Polani, and T. Uthmann. Evolution of the spectral properties of a visual agent receptor. In A. Wu, editor, *Proc. of the 1999 GECCO Workshop Program, Workshop on Evolution of Sensors in Nature, Hardware, and Simulation*, 1999.
- [9] H. H. Lund, J. Hallam, and W.-P. Lee. Evolving robot morphology. In *Proc. IEEE 4th Int. Conf. on Evolutionary Computation*, NJ, 1997. IEEE Pr.
- [10] A. Mark, D. Polani, and T. Uthmann. A Framework for Sensor Evolution in a Population of Braitenberg Vehicle-like Agents. In Adami et al. [1], pages 428–432.
- [11] F. Menczer and R. K. Belew. Evolving sensors in environments of controlled complexity. In P. M. Rodney A. Brooks, editor, *Artificial Life IV: Proc. of the Fourth Int. Workshop on the Synthesis and Simulation of Living Systems*, pages 210–221. MIT Press, 1994.
- [12] P. Mössinger, D. Polani, R. Spalt, and T. Uthmann. A virtual testbed for analysis and design of sensorimotoric aspects of agent control. *Simulation Practice and Theory*, 5(7-8):671–687, 1997.
- [13] D.-E. Nilsson and S. Pelger. A pessimistic estimate of the time required for an eye to evolve. *Proc. Roy. Soc. Lond.*, 256:53–58, 1994.
- [14] S. Nolfi, O. Miglino, and D. Parisi. Phenotypic plasticity in evolving neural networks. In J.-D. Nicoud D. P. Gaussier, editor, *Proc. of the first Int. Conf. From Perception to Action*, Los Alamitos, CA, 1994. IEEE Press.
- [15] M. Ott and F. Schaeffel. A negatively powered lens in the chameleon. *Nature*, 373(6516):692–694, Feb 1995.
- [16] G. Pask. Physical analogues to the growth of a concept. In *Mechanisation of Thought Processes: Proc. of a Symposium held at the National Physical Laboratory (No. 10)*, volume II, pages 877–928, London, 1959. Her Majesty's Stationary Office.
- [17] S. A. Shaaban, M. A. Crognale, J. B. Calderone, J. Huang, G. H. Jacobs, and S. S. Deeb. Transgenic Mice Expressing a Functional Human Photopigment. *Investigative Ophthalmology & Visual Science*, 39(6):1036–1043, May 1998.
- [18] P. M. Todd and S. W. Wilson. Environment structure and adaptive behavior from the ground up. In *From Animal to Animals 2: Proc. of the Second Int. Conf. on Simulation of Adaptive Behavior*, pages 11–20. MIT Press, 1993.
- [19] L. v. Salvini-Plawen and E. Mayr. On the evolution of photoreceptors and eyes. In B. Wallacs, M. K. Hecht, W. C. Steere, editor, *Evolutionary Biology*, volume 10, pages 207–263. Plenum Press, New York, 1977.
- [20] J. Vaario, K. Hori, and S. Ohsuga. Toward evolutionary design of autonomous systems. *Int. Journal in Computer Simulation*, 5:187–206, 1995.

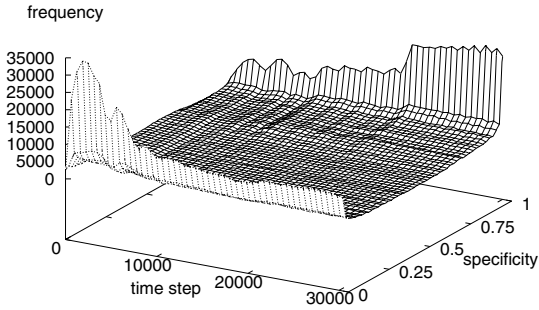


Figure 5: 3D-plot of the runs, lamp emitter with one active channel. Note the different frequency scales for Figs. 5 and 8.

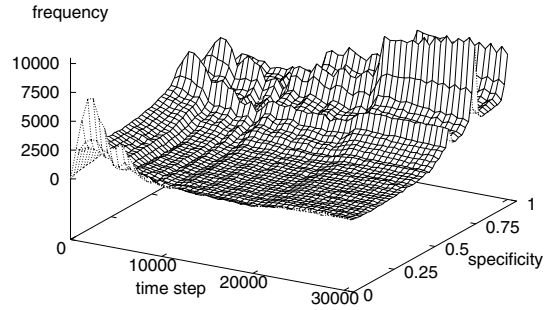


Figure 8: 3D-plot of the runs, lamp emitter with two active channels. Note the different frequency scales for Figs. 5 and 8.

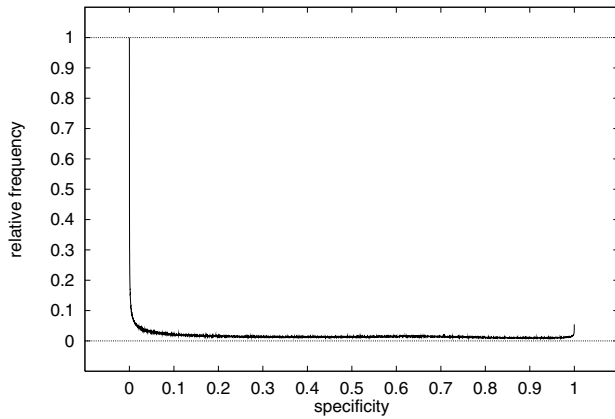


Figure 6: calculated a priori state density $d_S(s)$ for lamp emitter with one active channel

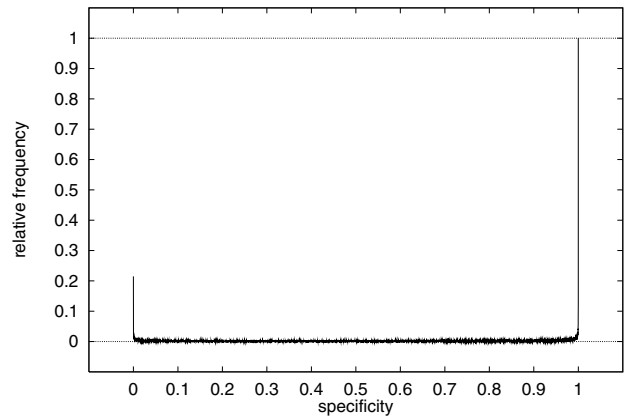


Figure 9: density histogram obtained from evolution runs for lamp emitter with one active channel

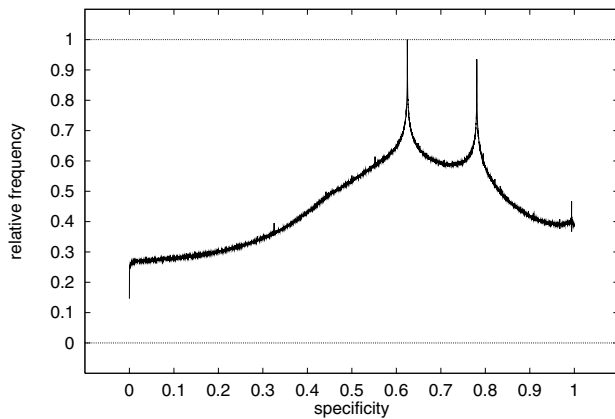


Figure 7: calculated a priori state density $d_S(s)$ for lamp emitter with two active channels

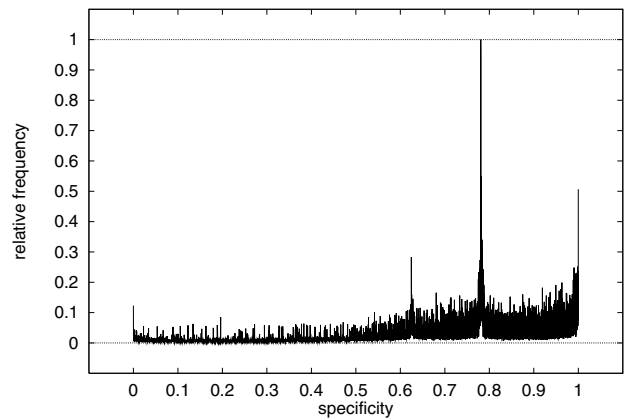


Figure 10: density histogram obtained from evolution runs for lamp emitter with two active channels