Proposal of an Intrinsically Motivated System for Exploration of Sensorimotor State Spaces

Matthias Kubisch **Manfred Hild** **Sebastian Höfer**
Neurorobotics Research Laboratory
Humboldt-Universität zu Berlin
Unter den Linden 6
10099 Berlin, Germany
{kubisch|hild|hoefer}@informatik.hu-berlin.de

Abstract

For a well-adapted behavior, an individual has to establish and refine its own body model continuously during life time. The best way to collect the necessary data for bootstrapping this model is active movement. At this, it is useful if actions are chosen in such a way that the gathered information matches well with the current stage of the learning system. In this paper, we investigate the method of self-exploration by intrinsic motivation, whereby the individual is driven to select appropriate actions to support its own learning progress. We implemented an unsupervised neural multi-expert architecture and tested the learning algorithm on an abstract artificial individual.

1. Introduction

The creation of artificial individuals may help us to understand the basic structure of autonomous learning. Therefore, it is useful to give an individual the possibility to establish its own body model. In order to continuously adapt the body model, suitable sensory input has to be provided at appropriate time. The selection of adequate motor actions has to be adjusted to the internal state of the learning system. This is necessary in order to avoid too simple or too complex sensory data, which in either case is suboptimal for the individual's learning progress.

(Schmidhuber, 2006, Oudeyer et al., 2007) suggests that the decreasing error of prediction of the internal state possibly functions as intrinsic motivation, and that this will help to build the body model in a more adaptive way. We supplemented the framework with different components, and for now, tested our implementation on simple abstract individuals. We can obtain similar results which raise hope, that the intrinsic motivation hypothesis is a fruitful approach.

The purpose of this work is to take a step forward in the understanding of basic learning algorithms, which are mainly shaped by the individual's properties. We try to reduce the problem as much as possible. Therefore, as a first step, the individual will have a restricted morphology and will be released from basic survival motivations, providing unlimited energy supply and preventing self-harm. The applied online learning algorithm operates in an overall unsupervised manner. It is capable of life-long learning and will bootstrap all information from scratch. Emerging structure and decreased complexity in behavior can be observed in comparison to plain random activities. This structure is only generated from inside, exclusively formed by the shape of the body and the architecture of the learning mechanism. There are no explicit objectives except intrinsic motivation. This is given in the form of reward, which is simply a function of the individual's learning progress. To put it crudely, the individual is only wanted to *have fun* during learning.

The outline of this paper is as follows: Firstly, we clarify the sensory basis for self-exploration experiments and explain in detail the neural multi-expert system, which mainly serves for state identification. Refining some computational aspects, in section 3 we pick up a definition of learning progress and intrinsic motivation. Section 4 shows a straight forward mechanism for action selection. Section 5 describes the experimental test case and the following one presents the derived results.

2. Competing Experts

Body and behavior of biological individuals are commonly well-adapted to environmental conditions. Until now, differentiating between body and environment is not mandatory when thinking of basic learning algorithms. Hereinafter, we will make no distinction between body and environment. Beyond that, all sensory data will be taken into account, as appropriate with preprocessing. Conventionally, these inputs may be raw sensory data or higher level percepts.

In order to decide which sensorimotor context the individual is situated in, we have to process the ongoing sensory data and generalize distinct *states*. This will also be useful to decide what *action* should be chosen next. One way to determine a certain state is to make a prediction of next sensory data, and compare the predicted values to the true ones. If the prediction error has been low then the predicting unit matches with the current sensorimotor state. Imagine a bunch of such prediction units, each qualified for another state. Such a single predictor is called an *expert*, if it has specialized to a certain sensorimotor context, and is able to predict the next sensory state with sufficient precision.

For a robust state distinction, all available sensors can *and should* be used as a rich sensory input

$$\mathbf{x}(t) = (x_1(t), x_2(t), \ldots, x_D(t))^T \qquad (1)$$

with $\mathbf{x}(t) \in S \subseteq \mathbb{R}^D$ at discrete time t and $D \in \mathbb{N}$ denoting the number of different sensors. The learning system should be allowed to sort out unnecessary information by itself. Classically, the prediction is either an estimate of the next sensory value composed from preceding ones or the particular part of the sensory state space is reconstructed from the knowledge of current and past motor activities. Therefore, the past motor commands have to be provided as additional sensory input, too. Usually, an expert will make use of both. In consequence, multiple experts will evolve in similar parts of the sensory state space, but have different expertise for particular motor activities.

In order to make good predictions, we have to find an adequate representation of time, so that we can use some kind of memory for a better distinction of different dynamic states. The easiest way to do so is *explicit time embedding*. Therefore, we handle time as additional spatial dimensions, by implementing a tapped delay line for every single sensory channel x_i. We expand the sensory state space by time, so that the experts could make use of the current sensory data, as well as their short term history. The time expanded sensory state space is given by

$$\tilde{\mathbf{x}}(t) = (\mathbf{x}(t), \mathbf{x}(t-1), \ldots, \mathbf{x}(t-K+1), 1)^T \qquad (2)$$

as a single column vector $\tilde{\mathbf{x}}(t) \in \mathbb{R}^{DK+1}$ of all available sensory data, including its $K \in \mathbb{N}$ time delayed previous values and the *bias*.

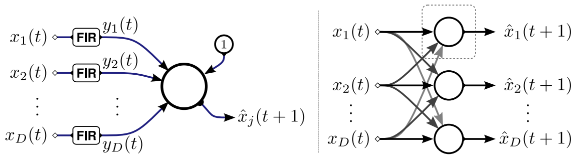

Figure 1: Structure of a simple prediction unit (expert) using FIR-type synapses and *hyperbolic tangent* as neural transfer function.

In its simplest form, the architecture used for prediction consists of one feed forward neuron for each prediction value. Therefore, the prediction is nothing but a non-linear weighted sum of all available input data. In this context it is helpful to think of a special type of synapse. Referring to filter design techniques, this will be labeled as a FIR-type synapse (Back and Tsoi, 1991), because the weighted sum of time delayed values is a finite impulse response filter. Remember the common filter structure

$$y_i(t) = \sum_{k=0}^{K-1} w_{ik} x_i(t-k) \qquad (3)$$

where $y_i(t)$ is the filter's output and $w_{ik} \in \mathbb{R}$ the coefficients (weights) of the linear filter. In our application, a delay line length of $K = 5$ is sufficient. This type of synapse sheds new light on how the expert actually uses every single input channel. From a filter design point of view, we are able to distinguish low-pass, band-pass or high-pass types of synapses. Concluding, the architecture of a single expert unit is given by

$$\hat{\mathbf{x}}(t+1) = \tanh\left(\mathbf{W}\tilde{\mathbf{x}}(t)\right) \qquad (4)$$

$$E(t+1) = \|\mathbf{x}(t+1) - \hat{\mathbf{x}}(t+1)\|^2 \qquad (5)$$

with the weight matrix $\mathbf{W} \in \mathbb{R}^{D\times(KD+1)}$ (see fig. 1). Every expert unit makes a prediction $\hat{\mathbf{x}}(t+1)$ which is compared with the next sensory value $\mathbf{x}(t+1)$ and the unit n with minimal prediction error $E_n(t)$ is assigned to be the winner. This process can be thought of as a state discretization. Hard competitive learning (*winner-takes-all*) discretizes the sensorimotor input space in $N \in \mathbb{N}$ distinct states, each detected and represented by a single expert unit. This is equivalent to the concept of *regions* as stated in (Oudeyer et al., 2007). Only the winning unit is allowed to adapt its weights. Due to the feed forward nature of the prediction network, this can be done easily with the online variant of the well-known backpropagation of error algorithm (Rumelhart et al., 1986, Schiffmann et al., 1993).

Growing Multi-Experts Bounded rationality plays an important role for the design of a competing expert architecture. For almost all robotic applications computing resources are limited to a certain extent. For the sake of life-long adaption, further expert units have to be included when the robot experiences new sensorimotor situations. Therefore, it is necessary to detect redundant or futile units and remove them to clear the way for new experts. Those will be inserted right in time when such situations appear.

To handle the insertion and deletion of expert units, we use a modified version of *growing neural gas with utility criterion* (Fritzke, 1997). In general, a neural gas follows the shape of the sensory input data by placing units on locations of high density. Those will be interconnected with edges in the manner of setting up a topology to make an approximation of the input space. In a certain time interval, new units will be inserted where the highest approximation error occurs. To be able to follow non-stationary distributions, there is a utility criterion for the purpose of detecting futile or ineffective units. If the input distribution changes gradually existing units adapt their weights, while rapid changes cause instantaneous deletion of edges and units.

For the application in our multi-expert architecture, some modifications have to be done to the GNG-U. Firstly, we replace the classical distance based activation unit by the neural expert unit. The prediction error of the expert is equivalent to the previous distance error. Hence, the best predicting expert n is assigned as the winning unit. Major modifications affect the time dependence of the insertion process. To be as reactive as possible, new units will only be inserted when the current winning expert is fully trained. We re-import the idea of an annealing learning rate, but implement one for every single expert unit. In doing so, we can implicitly detect novelty by testing, if the winning expert's learning rate has been fallen below a certain threshold ϵ_θ. That implies that a previously trained expert should be supported by another unit to keep the expert's specialization. The new unit will be inserted next to the current sensorimotor context by inheriting the weights of a fast adapting scout unit (Martius et al., 2008). Note that the learning rate will be annealed only by the amount of the weight change. In consequence, if the winning unit is already perfectly adapted nothing will be changed in terms of the network's topology. The update rule for the dynamic part of the learning rate is

$$\epsilon_n(t) = \epsilon_n(t-1) \cdot e^{-\kappa \, \|\Delta \mathbf{W}_n\|} \tag{6}$$

with $\Delta \mathbf{W}_n$ denoting the change of the weight matrix of unit n, weighted by $\kappa \in \mathbb{R}$, $\kappa > 0$. New experts were initialized with ϵ_0. Finally, the learning rate used for the adaption process is set to $\eta_n(t) = \epsilon_n(t) + \epsilon_R$ for the reason that slight adaption is always possible.

3. Learning Progress Definition

Since we released a primitive artificial individual from basic survival motivations, all that is left may be of an intrinsic nature. Intrinsic motivation has been discussed in detail in (Oudeyer and Kaplan, 2008). The notion is that learning progress itself functions as intrinsic motivation. Imagine an individual which receives reward when it has learned something. In order to obtain more reward, it has to repeat actions that lead to successful learning. Moreover, it is clear that such activities should only be executed when the individual observes the same sensorimotor context. Such a context will now be detected by an expert unit as presented in the previous section.

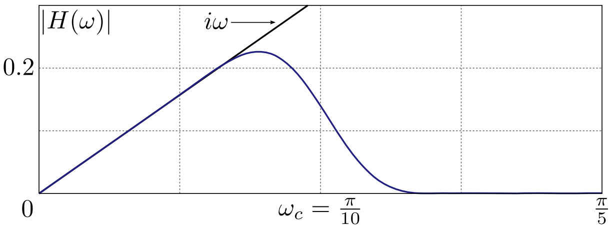

Figure 2: A low-pass differentiator.

The individual's learning progress, i. e. the reward can be derived from the prediction error of the learning experts. There is strong evidence that the prediction error constitutes a learning signal in biological individuals (Doya, 2002). In our approach, only the winning expert adapts its weights. Therefore, reward can be directly defined as

$$r(t) = -\frac{\mathrm{d}E_n(t)}{\mathrm{d}t} \tag{7}$$

with $E_n(t)$ being the prediction error of the winning expert.

A Low-pass Differentiator In order to use real world sensory data, one has to pay attention to the computation of the prediction error's derivative. Noise comes into the system through sensors and the prediction error is a function of sensory input. Hence, a discrete differentiation like $y(t) = (x(t) - x(t-1))\,\Delta t^{-1}$ will in most cases lead to unintended amplification of noise. For a precise and noise resistant differentiation, we implemented a simple but effective 21-tap FIR low-pass differentiation filter, following the method of (Hamming, 1989). An ideal differentiator with low-pass filter properties would have a transfer function like

$$H_{ideal}(e^{i\omega}) = \begin{cases} i\omega & |\omega| \leq \omega_c \\ 0 & \omega_c < |\omega| < \pi \end{cases} \tag{8}$$

with adjustable cutoff frequency $f_c = \omega_c/2\pi$ (here, i denotes the imaginary unit). Fourier series expansion of (8) leads to

$$c_k = -\frac{1}{\pi}\left(\frac{\sin(k\omega_c)}{k^2} - \frac{\omega_c \cos(k\omega_c)}{k}\right) \tag{9}$$

with $k \in \mathbb{Z}$, $k = -L \dots L$. The truncated filter coefficients are weighted by the *hamming window function*

$$w_k = 0.54 + 0.46\cos(\pi k/L) \tag{10}$$

and are manually adjusted by $b = 2.091$ to reduce the *Gibbs phenomenon*. The resulting filter is given by its coefficients

$$q_k = b\, w_k c_k \tag{11}$$

for $L = 10$, leading to the transfer function depicted in fig. 2. Note, the filter has a constant but not negligible *group delay* of L time steps. Therefore the reward signal is delayed. This has to be taken into account when using the reward for updating the corresponding probabilities of actions.

4. Action Selection

We use the standard approach of Q-learning in its most elementary form for online and on-policy learning, SARSA(λ) (Sutton and Barto, 1998, Rummery and Niranjan, 1994). The method leads to an action-value-matrix $\mathbf{Q} \in \mathbb{R}^{N\times M}$ which contains valuations for every state-action pair. The discrete set of N states is provided by the multi-expert architecture. The discrete set of $M \in \mathbb{N}$ actions is usually a sampled subset of the available motor space and will be converted into the real valued and time discrete motor vector $\mathbf{m}(t)$. For example, this can be easily implemented as a neural field of M interconnected neurons as stated in (Toussaint, 2006).

The winning expert n is allowed to choose the next action. A robust and popular mechanism for action selection is the ε-greedy method. Basically, the action with the maximum Q-value is chosen. However, with a small probability of ε we choose a random action. This consequently ignores the next best action while selecting a completely random one. A more sophisticated method is Boltzmann selection (Doya, 2002) where Q-values are converted into probabilities through the application of the softmax-function

$$P_{nk} = \frac{e^{\beta_n Q_{nk}}}{\sum_{l=0}^{M-1} e^{\beta_n Q_{nl}}} \tag{12}$$

for all $k = 1 \ldots M$. Now we choose the actions based on their probabilities. The inverse temperature $\beta_n \in (0, \infty)$ regulates the ratio of randomness and greediness in action selection. Note that the exponential function can easily break the bounds of common computer numbering formats. As a property of the softmax activation function, one can simply *normalize* the tempered Q-values $q_k = \beta_n Q_{nk}$ by

$$\tilde{q}_k = q_k - \max(\mathbf{q}) \tag{13}$$

where $\tilde{\mathbf{q}}$ are the new and secure Q-values. This simple transformation leads to exactly the same probabilities but does not accidentally produce an arithmetic overflow.

The Boltzmann selection method is quite ineffective with a constant value of β_n. Once maladjusted, this will inhibit the system in one way or another. A collateral learning rule like

$$\beta_n(t) = \beta_n(t-1)\cdot\left(\frac{3}{2} - vM\right) \tag{14}$$

adjusts β_n in the way, that the variance of the selection probabilities $v = \operatorname{Var}(P_n) \in \left[0, \frac{1}{M}\right]$ quickly approaches the constant value $\frac{1}{2M}$.

5. Experimental Test Case

Most of the robot platforms—simulated or real—are much too complex. In order to study basic learning algorithms, it is of interest to have a simple, completely foreseeable system, which is well-defined and easy to visualize.

Neuron in a Box To begin with, one or two degrees of freedom are sufficient. Therefore, our test system is a two dimensional non-linear iterated map in the form of a fully connected neural network given by

$$\mathbf{x}(t+1) = \tanh\left(\tilde{\mathbf{W}}\mathbf{x}(t) + \mathbf{b} + a\mathbf{m}(t)\right) \tag{15}$$

$$\tilde{\mathbf{W}} = \begin{pmatrix} 1.01 & 0.1 \\ 0.1 & 1.01 \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} 0.0398 \\ 0 \end{pmatrix} \quad a = 0.1 \tag{16}$$

where $\tilde{\mathbf{W}}$ and $\mathbf{b}$ defines the behavior of the body, a is the input strength for application of motor actions $\mathbf{m}(t)$ and $\mathbf{x}(t+1)$ is the resulting next time step sensory input for the learning system. In figure 3, the vector field is depicted with the colored areas denoting the basins which each lead to a stable fixed point. The system is slightly unsymmetrical and the second fixed point is located close to the separatrix.

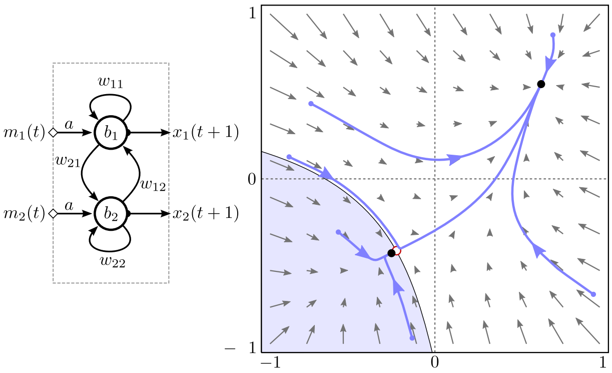

Figure 3: A neural vector field as a simplified and restricted body. Due to the application of the *hyperbolic tangent*, the range of sensory values is bounded to $(-1, 1)$ in each dimension. Dots denote stable fixed points.

Before we continue, figure 4 on the next page gives a brief overview of the entire algorithm. Concluding, the table lists the set of parameters used for the test case given above.

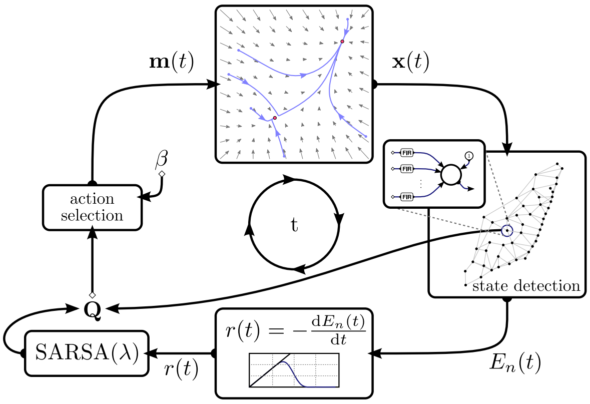

Figure 4: Overview of the entire learning architecture.

N_{max}	M	K	ϵ_0	ϵ_R	ϵ_θ	κ
50	30	5	0.1	0.01	0.01	0.5

Table 1: List of parameters used for the experiments.

6. Results

Starting with only two experts, the learning algorithm continuously acquires more units, rapidly growing into a multi-expert network. Hence, the discretization of the sensory state space gets more and more fine-grained. Figure 5 shows the configuration of the network in early and later stages. Note the way locations of experts match with the vector field of figure 3. Due to the reduced motor strength, some regions of the state space cannot be visited by the individual. If the predefined maximal amount of experts N_{max} is exhausted, the network will be pruned, and therefore remains under continuous development.

Figure 6 shows the area of the sensor space frequently used by the individual. It also offers the mean motor action performed, with respect to the individual's sensory state space. Meanwhile, the initially random action selection develops a certain structure by means of a non-uniform distribution. In comparison to purely random activity the intrinsically motivated individual rapidly develops preferences in action selection.

Figure 7 shows the frequencies of possible actions with respect to the selection method. The more sophisticated Boltzmann method reveals the highest variance in action selection. Due to the unsymmetrical nature of the body this result matches with the expectations.

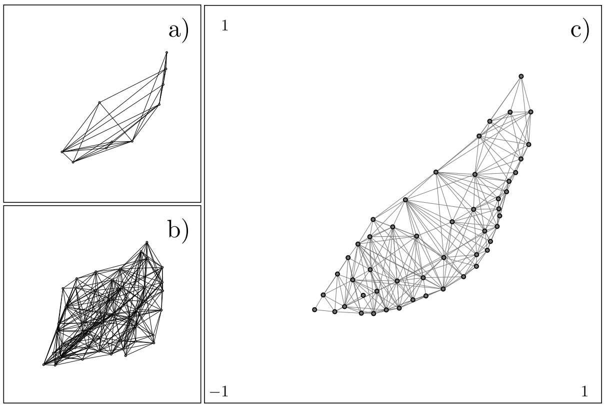

Figure 5: The configuration of the multi-expert network at an early *a)*, a later *b)*, and the final stage. Every node represents a single expert unit while edges constitute the neighbourhood relation and illustrates the expert's importance.

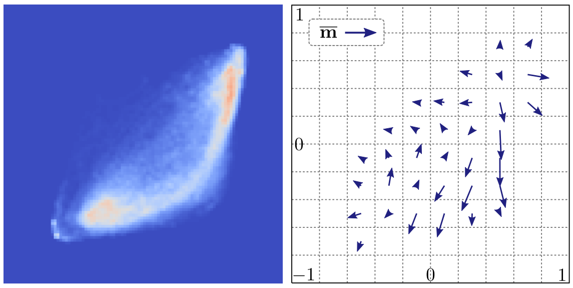

Figure 6: Area of frequently used sensor space (to the left), mean motor action exerted with respect to position in sensor space (to the right).

Observing the behavior of the individual exhibits a specific structure. As depicted in figure 8, there are phases of apparently random behavior alternating with phases of nearly deterministic behavior. Temporarily emerging structure in behavior can be recognized as sequences of successive states. Such a sequence grow out of the interplay of multiple expert units, each making its own short-term action selection. These sequences appear in different sizes and shapes, and apparently stabilize for a short period of time before they eventually disappear.

These behaviors partially reappear several times. Their shape is greatly influenced by the dynamics of the body. Varying the form of the vector field, considerably changes the shape of the sequences so that every body shape reveals its distinctive set of sequences. This implies that the algorithm makes extensive use of the underlying body dynamics.

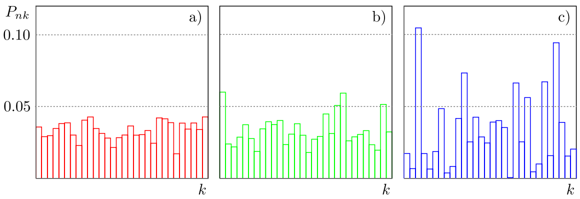

Figure 7: Histograms of selected actions during long-term runs of 2 hours length. In comparison: *a)* random selection, *b)* ε-greedy, and *c)* the Boltzmann method.

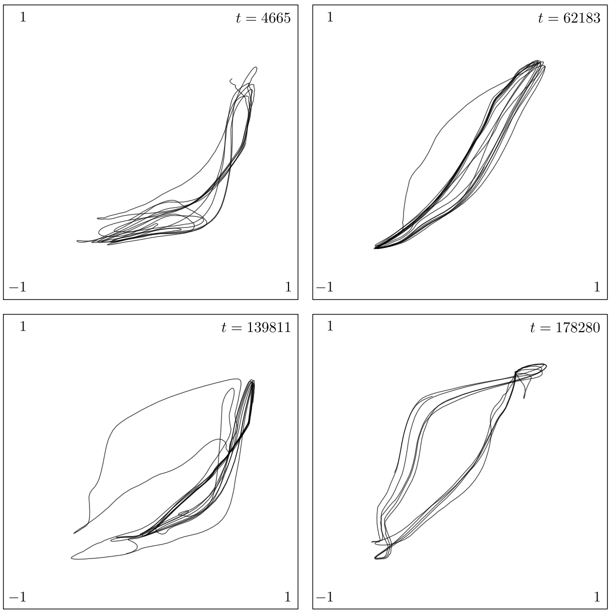

Figure 8: Snapshots of successive stages of the artificial system, each showing 2000 past time steps of the trajectory $\mathbf{x}(t)$.

7. Summary and Outlook

We proposed an algorithm based on the idea of intrinsic motivation for self-exploration of an artificial individual. The algorithm implements the *learning-progress as reward* hypothesis and has been applied to a simple abstract body. Emerging structure in behavior can be observed with respect to the shape of the body and the learning architecture.

Currently, we are investigating the application of the presented learning algorithm on a small sized robot, which is suitable for unsupervised self-exploration experiments due to its rich proprioceptive sensors and its capability for avoiding self-harm.

Acknowledgements

This work has been supported by the European research project ALEAR (FP7, ICT-214856).

References

Back, A. D. and Tsoi, A. C. (1991). FIR and IIR Synapses, A New Neural Network Architecture for Time Series Modelling. *Neural Computation*, 3(3):375 – 385.

Doya, K. (2002). Metalearning and Neuromodulation. *Neural Networks*, 15.

Fritzke, B. (1997). A Self-Organizing Network That Can Follow Non-Stationary Distributions. In *Proc. of ICANN-97, International Conference on Artificial Neural Networks*, pages 613 – 618. Springer.

Hamming, R. W. (1989). *Digital Filters.* Prentice Hall, 3rd edition. Paperback reprint: Courier Dover Publications, 1998.

Martius, G., Fiedler, K., and Herrmann, J. M. (2008). Structure from Behavior in Autonomous Agents. In *IEEE International Conference on Intelligent Robots and Systems*, pages 858 – 862.

Oudeyer, P.-Y. and Kaplan, F. (2008). How Can We Define Intrinsic Motivation? In *International Conference on Epigenetic Robotics.*

Oudeyer, P.-Y., Kaplan, F., and Hafner, V. V. (2007). Intrinsic Motivation Systems for Autonomous Mental Development. *IEEE Transactions on Evolutionary Computation*, 11.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning Internal Representations by Error Propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1: Foundations, pages 318 – 362. MIT Press.

Rummery, G. A. and Niranjan, M. (1994). On-Line Q-Learning Using Connectonist Systems. Technical report, CUED/F-INFENG/TR 166, Cambridge University Engineering Department, England.

Schiffmann, W., Joost, M., and Werner, R. (1993). Comparison of Optimized Backpropagation Algorithms. In *Proc. of ESANN'93, Brussels*, pages 97–104.

Schmidhuber, J. (2006). Developmental Robotics, Optimal Artificial Curiosity, Creativity, Music, and the Fine Arts. *Connection Science*, 18(2):173–187.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction.* MIT Press.

Toussaint, M. (2006). A Sensorimotor Map: Modulating Lateral Interactions for Anticipation and Planning. *Neural Computation*, 18:1132 – 1155.