

**Self-referential dynamical systems for the
self-organization of behavior in robotic
systems**

Naglaa Hamed

PhD Thesis
University of Leipzig 2007
Supervisor Prof. Dr. Ralf Der

Contents

Chapter 1. Introduction	6
1. The dynamical systems approach	6
2. Developmental robotics	7
3. Embodiment	8
4. Self-organisation	9
5. Motivation and aims of the thesis	9
Chapter 2. An elementary sensorimotor loop	11
1. Dynamics of the SM loop	11
2. Including a bias	13
3. Systems with delay	15
Chapter 3. The approach to self-referential systems	22
1. Dynamical system formulation	22
2. Adaptive systems	23
3. Self-referential systems	24
4. Explicit expressions	28
5. Formal considerations	28
6. Fixed point flow from self-referential dynamics	30
Chapter 4. One-dimensional system with learning	32
1. Parameter dynamics for the model system	32
2. Including the bias dynamics	33
Chapter 5. Two-dimensional systems	37
1. Dynamics with fixed parameters	37
2. Including learning I: The linear case	41
3. Including learning II: The nonlinear case	44
4. Including the bias dynamics	51
5. Correlated noise	55
6. Application - The barrel robot	59
Chapter 6. Three-dimensional systems	63
1. General observations	64
2. Regular oscillations	64
3. The frequency wandering effect	64
Chapter 7. Natural gradient parameter dynamics	67
1. The natural gradient method	67
2. Application: One neuron several channels	68
3. An algorithm for updating the inverse noise matrix.	71

Chapter 8. Summary and outlook	74
Bibliography	77

ACKNOWLEDGEMENT 1.

My thanks are first and foremost to my supervisor Prof. Dr. Ralf Der for proposing the topic of the thesis, his continuous interest, encouragement, and help in many questions. The work could not have been completed without the help by Georg Martius, Frank Hesse, Marcel Kretschmann and others from the robotics group or R. Der. I would like also to thank my parents and my husband for their encouragements.

CHAPTER 1

Introduction

Recent years have witnessed a tremendous increase in the computational power, the sensory and motoric equipment and materials for the construction of autonomous robots. Nevertheless, the behavioral competencies of current robots is lagging far behind that of humans or even animals. In fact, any ant is much better in surviving in a natural environment than the most advanced robot. This concerns questions of robustness, fault tolerance, flexibility, and autonomy in general. This has triggered a series of new developments trying to develop robots which are closer to living beings. In particular one has to reject at least partially the strong computer metaphor of artificial intelligence but instead look for other realizations of the behavior control. These new developments not only will give the possibility to draw from the wisdom of nature in the sense of finding new control paradigms but also will have an impact on the computational theory of cognitive processes in living beings. Moreover, as has been stressed in the context of embodiment, these approaches also give rise to a new approach to artificial intelligence and thus are an important contribution to the development of computer science, see [48].

Of the many approaches we will in the following mention the use of dynamical systems, the developmental and epigenetic robotics, the paradigm of embodiment and the theory of self-organization.

1. The dynamical systems approach

Dynamical systems are an important field of research in both mathematics, physics and also in the life sciences. Their importance for the robotics was noted first by the seminal paper of Randall Beer [2] where he described the robot in its environment as a dynamical system. This idea was developed further in later papers, see for instance the paper on dynamical approaches to cognitive science [3] and the related work in cognitive science based on the book by Port, van Gelder et al. [49]. A systematic study of the approach is given in [44]. Following these ideas, dynamical system terminology is applied in two ways, metaphorical and formal. In its metaphorical use, one essentially applies the dynamical systems vocabulary to describe what a robot is doing. For example, if an agent's behavior stabilizes in a particular environment and the agent starts going in circles or oscillates to the left and to the right in a corner, this is said to constitute a limit cycle. When using dynamical systems formally, we need to specify what the system we intend to model and then we have to establish the differential (or difference) equations. One approach would be to model the agent and the environment

separately and then to model the agent-environment interaction by making their state variables mutually dependent.

In a more practical perspective dynamical systems are used as generator of behaviors based on the fact that a dynamical system of a certain complexity offers a wide range of behaviors ranging from fixed point over limit cycle to chaotic behavior. Most importantly these behaviors can be modified by changing the parameters of the system. Many researchers therefore follow the idea to use a dynamical system as the "substrate" for the controller of the robot, see for instance [15], [54], [60], [21], and [4]. In particular many researchers use the dynamical systems with limit cycles since the latter can be used for the control of rhythmic movements like walking, see [25], [45], or drumming [30] to name only a few. Another approach is followed in [28] where a nonlinear $2 - d$ system is used for the navigation in a dynamical environment.

These approaches are very interesting, they require however that the parameters are tuned by hand which is a rather complex procedure. Artificial evolution is one way of solving this problem [20]. Further progress is achieved by using adaptive oscillator systems, for instance [39], [51], [5], [50], which are self learning in some sense so that they tune their parameters so that the correct frequencies emerge.

The dynamical systems to be analyzed in the present paper have some similarities with the latter approach. The main difference is in the fact that they are self-referential in the sense that there is no prespecified goal for the adaptation. Nevertheless these systems develop behaviors which are closely related to the special environmental conditions. As compared to the adaptive oscillator systems the difference is that our systems develop under certain conditions a self-induced sweep through the frequency space with a frequency locking if there is some resonance with the environment. This effect has been described for the first time in the present thesis, see Chapter 5.

More on the biological side, several authors have followed the idea of using the language of dynamical systems for the identification of behavior primitives in real biological systems, see [6] and [14] for some early papers. Subsequently the synthesis of behavioral primitives as a self-organization process utilizing a distributed representation scheme has been considered in [57] based on [56]. Using imitation learning of a four degree robot arm as an example the authors seek to integrate both fixed point and limit cycle attractors in the neural network through supervised learning of end-point and oscillatory behavior.

2. Developmental robotics

The problems with programming robots on a rule based paradigm has evoked the desire to have a true individual development of a robot. In 2001 several of the protagonists of this paradigm have presented in a paper in Science these ideas to a broader public [59]. A more recent review of the achievements in this field is given by the paper [36]. The aim of this approach is the robot which is "born" without (advanced) knowledge and behavioral competencies but acquires the latter by self-exploration and

interactions with the environment, and in a later phase also with a human trainer in a process of *life-long learning* in an on-line (and one-shot) scenario. A prominent example is the playground experiment, see [43], [41], and the book [42]. The embodiment plays a central role in these approaches, see [31] and the papers cited in [36].

A systematic approach is provided by the concept of an inner motivation, so that the agent is driven to actions with the aim of knowledge acquisition, see [52], [53]. Alternative approaches are based on information theoretical paradigms like the information self-structuring [37], the maximizations the information transfer in the sensorimotor loop [29]. These measures are used then in the artificial evolution, see for instance [40], in order to generate a selection pressure endowing the individuals with an artificial curiosity. Prominent examples of this approach are the emergence of a visuomotor coordination [37]. In a similar sense, the learning progress has been used as an internal motivation see [26] for details.

The approaches so far suffer from the disadvantage that they work in rather structured environments with few degrees of freedom. By way of example, in the playground experiment the state action space is restricted to only three bits of sensory information and a few behavioral primitives, see [43]. This also means that the embodiment plays only a minor role. The approach of R. Der et al. which forms the basis of the present thesis has been demonstrated to work with embodied agents with many independent degrees of freedom in unstructured, dynamic environments. In particular, it equips the agents with a playful explorative behavior which is vital for the developing robot in its early sensorimotor stage. The results of the present thesis can be understood as a theoretical basis for the developmental robotics in this early phase of development.

3. Embodiment

In recent years, both in robotics and in cognitive science, there has been an increasing awareness that the computer metaphor of artificial intelligence is not any longer applicable in the strong sense. The latter sees intelligence as an algorithm, the body simply as the device which executes the control actions prescribed by the algorithm (the brain). Instead the new approach takes full account of the embodiment of the agent. One of the shared ideas of this work is that a significant fraction of the behavior must be seen as emerging from the ongoing interaction between an agent and its environment, see [48], [47], [46] for an introduction into these ideas. In robotics one tries above all to reduce the control effort by exploiting the physical properties of the robot, a prominent example being the passive walker as introduced by McGeer [38], others are [22], [23], and many applications may be found in [24].

The approach considered in the present thesis is much involved with the phenomenon of embodiment since the self-referential dynamics manages to feel the physics of the body and in this way makes also use of the physic of the robot, see [9] for details. Therefore the present thesis may also be considered as a contribution to the systematic approach to embodied Artificial Intelligence.

4. Self-organisation

Self-organization in the sense used in physics means the spontaneous creation of patterns in space and/or time in dissipative systems consisting of many components (particles), see [13]. A systematic approach to the phenomenon of self-organization has been given by Hermann Haken by the paradigm of synergetics [16] which has also a standing in cognitive science, see [17], [58] for details. Central in this context is the notion of emergence meaning the spontaneous creation of structures or functions which are not directly explainable from the interactions between the constituents of the system. Typical examples are given by reaction diffusion systems where the diffusion is driving the system towards complete homogeneity whereas there is an autocatalytic chemical reaction providing a self-amplification process. Thus, small local fluctuations can be amplified and form the origins of the emerging structure.

These effects known from physics and synergetics find a direct application in swarm robotics because of the similarity with a system of many constituents in physics (reaction diffusion systems) or biology (insect colonies), examples may be found in [27], [35],[34], and [33]. The aim of the present thesis is a contribution to a general approach to the self-organization of behavior of a single robot. We understand by a general approach that there is a completely domain invariant objective function for the adaptation of the robot which produces seemingly domain related behaviors. There are only few approaches known from literature in the present sense, an example being the work by Kuniyoshi. [31].

The present thesis is based on such a paradigm as developed and tested in various examples in recent years, see Chapter 4 and 3, the publications [8], [12], [10], [11] and the video page <http://robot.informatik.uni-leipzig.de/research/videos> for details. The objective function is derived from the following principle. We consider robots controlled by a neural network with fast synaptic plasticity. The ideal behavior of the robot is qualified by (i) a maximum sensitivity to current sensor values. This induces a self-amplification of changes in the sensor values and thus is the source of activity; and by (ii) a maximum predictability of future sensor values. This keeps the behavior in “harmony” with the physics of the body and the environment. Formally this leads to a dynamical system approach based on a self-referentiality paradigm.

5. Motivation and aims of the thesis

Based on the aforementioned paradigm, in recent years a great number of different robots have been shown to develop modes of behavior of its own without a specific goal prescribed. The computer simulations and also the experiments with real robots (like Khepera and pioneer robots) have produced often very unexpected results like the jumping of a snake out of a narrow pit, the emergence of a rotational mode in an underactuated snake artefact, stable rolling modes in spherical robots and quite generally a more or less playful exploration behavior, see <http://robot.informatik.uni-leipzig.de/research/videos> . These behaviors are observed at the phenomena level so that they require a deeper understanding in order to further develop

the approach. This is one of the aims of the present paper. Moreover, at the formal level the approach is based on a self-referential dynamical system. This is a kind of dynamical systems which has a theoretical interest in itself since it produces a number of interesting phenomena which may be useful in the further development of the dynamical system approach to robotics, cognitive science, and artificial intelligence.

The thesis is organized as follows. In Chap. 2 we exemplify the dynamical system formulation of the robot in its environment by an elementary example which however already contains much of the peculiarities of the present approach. Then, in Chapters 4 and 3 we introduce the self-referential dynamical systems in both the simple toy system and in a more general context. The Chapters 5 and 6 contain the investigations of systems in two and three dimensions where the self-referential dynamics is already quite complex but still more or less understandable in theoretical terms. In Sec. 6 an application to an embodied agent is reported. In 7 we discuss the application of the method of natural gradient in order to get the robot more concentrated on the relevant sensor information and finally in Chap. 8 some final considerations and remarks on future work are found.

CHAPTER 2

An elementary sensorimotor loop

The aim of the present chapter is to introduce a very simple example of an elementary sensorimotor (SM) loop by which we can demonstrate large parts of our paradigm of formulating the robot in the environment as dynamical system. In particular we want to demonstrate the benefits of closed loop control in the form we use it in the present work.

1. Dynamics of the SM loop

We consider a robot moving with velocity along the x-axis of our coordinate system with true velocity q_t . The velocity is read by a wheel counter which outputs a measured velocity x_t so that

$$x_t = \alpha q_t + k_t$$

where k_t is the measurement (sensor) noise and α is a hardware constant. The controller of the robot outputs the target velocity y_t for the current time step which is given in terms of the sensor value x_t as

$$y_t = K(x_t) \tag{1.1}$$

In the most simple case we use

$$K(x) = g(cx) \tag{1.2}$$

where

$$g(z) = \tanh(z) \tag{1.3}$$

In general, the dynamics of the sensor values can be written as

$$x_{t+1} = ay_t + \xi_{t+1} \tag{1.4}$$

where a is the hardware constant which can be found easily by a supervised learning process and ξ_t now takes account of both measurement and actuation noise (target and true wheel velocity may well differ). Using 1.1, 1.2, in 1.4 leads to the following dynamics of the SM loop

$$x_{t+1} = ag(cx_t) + \xi_{t+1} \tag{1.5}$$

1.1. The feed-back strength. The SM loop is a feed-back loop, the (linear) feed back strength being given by

$$R = ca$$

The effect of R can be seen by the following argument. We consider the case of small z where the approximation $\tanh(z) = z - \frac{z^3}{3}$ can be used. Using the first term only, one gets (ignoring the noise for the moment) from 1.5

$$x_t = R^t x_0$$

therefore the robot slows down for $0 < R < 1$ (because $x_t \rightarrow 0$ as $t \rightarrow \infty$) and accelerates for $R > 1$ (x_t increases exponentially), but this velocity can not increase unlimitedly since $|\tanh(z)| < 1$. The asymptotic value of x in the nonlinear case is obtained as the solution of the fixed point equation

$$z = R \tanh(z)$$

1.2. Fixed point and its stability. The FP equation

$$z = R \tanh(z)$$

has always $z^* = 0$ as fixed point and two other fixed points $z^* = \pm q$ only if $R > 1$. The fixed point $z^* = 0$ is stable if $R < 1$ but it is unstable if $R > 1$. The other two fixed points are stable. Using the approximation $\tanh(z) = z - \frac{z^3}{3}$, the fixed points can be obtained explicitly as

$$z = \pm \sqrt{\frac{3(R-1)}{R}} \quad (1.6)$$

The properties of the fixed points are most easily obtained by a linear stability analysis. We put $z_t = z^* + u_t$ where u is small. The state dynamics $z_{t+1} = Rg(z_t)$ is linearized as

$$z^* + u_{t+1} = Rg(z^* + u_t)$$

and by using Taylor expansion one gets

$$u_{t+1} = L(z^*) u_t$$

where

$$L = Rg'(z^*)$$

One has only to show that $0 < L < 1$ in order to prove that $u_t \rightarrow 0$ as $t \rightarrow \infty$. Using the approximated eq. 1.6 we get

$$L = Rg'(z^*) = R\left(1 - \frac{3(R-1)}{R}\right) = 3 - 2R$$

so $L < 1$ if $1.5 > R > 1$, provided $\tanh(z) = z - \frac{z^3}{3}$ is valid, i.e. as long as R close to 1. However if $R \gg 1$ then $|z|$ is very large and one gets from the fixed point equation ($z = R \tanh(z)$) that $z^* \simeq R$, so $L = R(1 - \tanh^2(R))$ or approximately $L = 4R \exp(-2R)$, since at large R we may write approximately

$$\tanh(R) = \frac{e^R - e^{-R}}{e^R + e^{-R}} = \frac{1 - e^{-2R}}{1 + e^{-2R}} \approx 1 - 2e^{-2R}$$

Hence L goes exponentially to 0 as $R \rightarrow \infty$ so that the stability of the fixed point increases with increasing R .

These results have been derived in some detail since L is the central quantity of the time loop error, see 3.3. We have seen here that it is directly related to the stability of the fixed points of the state dynamics. In the higher dimensional cases L is the Jacobian matrix of the dynamical system which is well known to give direct expressions for analyzing the stability of the system.

2. Including a bias

When including a bias into the controller, i.e.

$$K(x) = g(cx + H)$$

the fixed point equation is

$$x = a \tanh(cx + H)$$

or

$$z = R \tanh(z) + H \quad (2.1)$$

Using the approximation of $\tanh(z)$, one gets from the above equation

$$z = R\left(z - \frac{z^3}{3}\right) + H$$

or

$$z^3 - \frac{3(R-1)}{R}z - \frac{3H}{R} = 0$$

The solutions of the above cubic equation according to Cardano's formula are

$$z_1 = u + v$$

$$z_2 = k_1u + k_2v$$

$$z_3 = k_2u + k_1v$$

where u , v , k_1 and k_2 are given by

$$u = \sqrt[3]{-q + \sqrt[2]{q^2 + p^3}}$$

$$v = \sqrt[3]{-q - \sqrt[2]{q^2 + p^3}}$$

where $p = \frac{(1-R)}{R}$, $q = \frac{-3H}{2R}$, $k_1 = -\frac{1}{2} + i\frac{\sqrt{3}}{2}$ and $k_2 = -\frac{1}{2} - i\frac{\sqrt{3}}{2}$. There are three cases depending on the value of $q^2 + p^3$:

- (1) If $q^2 + p^3 < 0$, then there are 3 real solutions which are different.
- (2) If $q^2 + p^3 = 0$, then there are 3 real solutions but two of them coincide.
- (3) If $q^2 + p^3 > 0$, there is only one real solution and 2 complex solutions.

With a given R value, the corresponding critical value of H_c (at which two of the real solution of the cubic equation coincide) can be found from the equation $q^2 + p^3 = 0$, therefore the critical value of H at which a fixed point disappears is

$$H_c = \frac{2}{3R} \sqrt{(R(R-1))} (R-1)$$

For instance with $R = 1.5$, the corresponding value $H_c \simeq 0.2$.

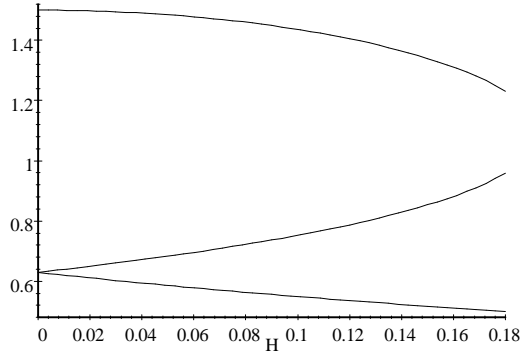


FIGURE 1. The value of the Jacobian L for $-H_c < H < H_c$.

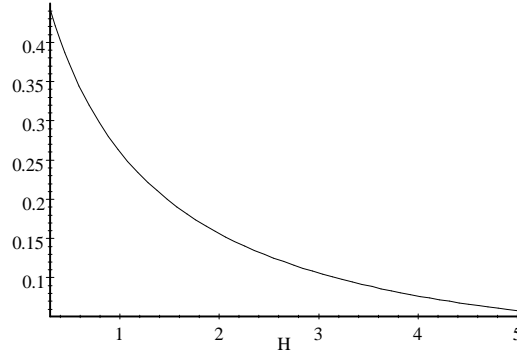


FIGURE 2. The value of the Jacobian L for values of H in the case that $H = H_c$ where there is only one stable fixed point $z > 0$.

2.1. Stability analysis of the fixed points. Depending on the value of H we have the following cases:

- (1) If $-H_c < H < H_c$, the stability of the fixed point is determined from the value of $L = R \tanh'(z^*)$. Plotting L as a function of H in this range, one sees from the following Fig. 1 that $0 < L < 1$ for the two fixed points z_1 and z_2 , therefore they are stable. However the value of L for z_3 is greater than 1, therefore it is unstable.
- (2) If $H = H_c$, then there will be two different real solutions namely $z_1 = 2u$ (this is positive as long as $H > 0$, otherwise it is negative) and the other two solutions are $z_{2,3} = -u$ (negative if $H > 0$ otherwise positive). For $R = 1.5$, $z_1 = 1.1547$ and $z_{2,3} = -.57735$. Calculating $L(z_1) = -0.5$, $L(z_{2,3}) = 1$, therefore z_1 is stable but $z_{2,3}$ is neither stable nor unstable.
- (3) If $H > H_c$ or $H < -H_c$. If $H > H_c$, therefore there is only one solution the stability of which can be seen from Figure 2.

2.2. The hysteresis effect. If the system starts with a very large negative value of H , after some time, regardless of the initial value of z , the system will be at the negative stable fixed point. Now let us continuously increase the value of the parameter H . Since the system was at the fixed point when we vary H , it will stay at the fixed point until $H = H_c > 0$. At this point the negative stable fixed point will become unstable, so the system will jump to the positive stable fixed point. Increasing H further, the system will stay at this positive stable one. If on the other hand H is decreased again, the system will stay at the positive fixed point until $H = -H_c$ is reached and it is only then that the dynamics jumps to the negative fixed point. This can be illustrated by Fig. 3

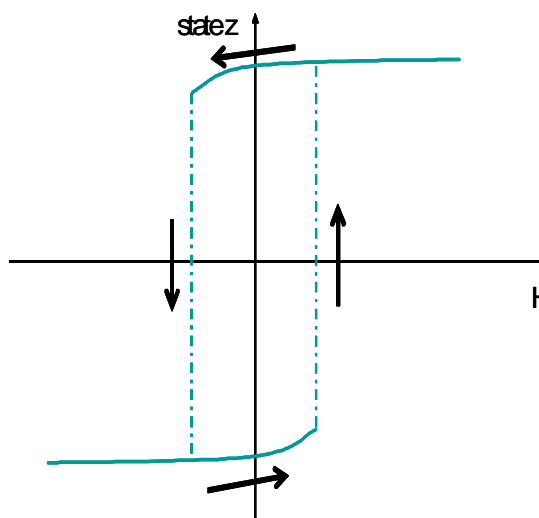


FIGURE 3

The hysteresis effect plays a central role in the learning dynamics based on the minimization of the time loop error, see Chapter 4.

3. Systems with delay

We have considered so far the case that the response for the sensors occurs immediately after the motor command is sent to the robot. In practice there is always a finite delay time which is about four steps in the case of the Khepera robot. So we have to consider time-delay systems which will be seen to display some surprising effects. In the dynamical systems approach to robotics this has not found sufficient attention so far so that we will consider our very simple system in some detail.

3.1. Delay with one step. Let us consider a system with one step delay, which is described by the following equation

$$x_{t+1} = \psi(x_{t-1}) + \xi_t$$

where we use as before

$$\psi(x) = a \tanh(cx)$$

Let us define S which is called the full orbit as

$$S = \{x_t | t = 0, 1, 2, \dots\}$$

In the system without delay, i.e. if $x_{t+1} = \psi(x_t)$, then $x_t = \psi^t(x_0)$, where $\psi^t(x) = \psi(\psi^{t-1}(x))$ and so on (we drop the noise for the moment), so the solution will be completely defined by the initial value x_0 . However in the system with one step delay x_0 is mapped as

$$x_t = \psi^n(x_0)$$

where $t = 2n$, i.e. x_0 defines the values x_2, x_4, \dots which is the suborbit

$$S_0 = \{x_t | t = 0, 2, 4, \dots\}$$

The initial value x_1 is mapped correspondingly as

$$x_t = \psi^n(x_1)$$

where $t = 2n + 1$, i.e. x_1 defines the values x_1, x_3, \dots which is the suborbit

$$S_1 = \{x_t | t = 1, 3, 5, \dots\}$$

The complete orbit S is composed of two independent suborbits and in practical applications one often encounters this splitting into suborbits.

3.1.1. *Fixed points of the suborbits.* Let us consider of a positive feed back loop, i.e. $R > 1$. The fixed point equations are $x_{t+1} = \psi(x_{t-1})$, so if $x_0 > 0$ ($x_0 < 0$), then S_0 converges to the positive (negative) fixed point. Now one can consider following different cases:

- 1.: x_0 and x_1 have the same sign, then both of S_0 and S_1 converge to the same fixed point.
- 2.: x_0 is positive and x_1 negative, then S_0 converges to the positive fixed point and S_1 converges to the negative one.

What was discussed about the fixed points and their stability without delay carries over to the system with delay since the fixed point equations stay the same and also the linearized equations which are the basis of the stability analysis. In the simple model case where $z = Rg(z)$ the stability of the fixed points depends again on the value of $L = Rg'(z)$ taken at the fixed point. In particular, in the case that $R > 1$ if one of the initial values is positive and the other one is negative, then the positive one converges to the positive fixed point $z^* = q$, but the negative one converges to the negative fixed point $z^* = -q$, i.e. in the case of initial values with opposite sign the trajectory S jumps between the two fixed points $z^* = \pm q$. This behavior is also conserved in the case of strong noise which actually might wash out this property, see Fig. 4

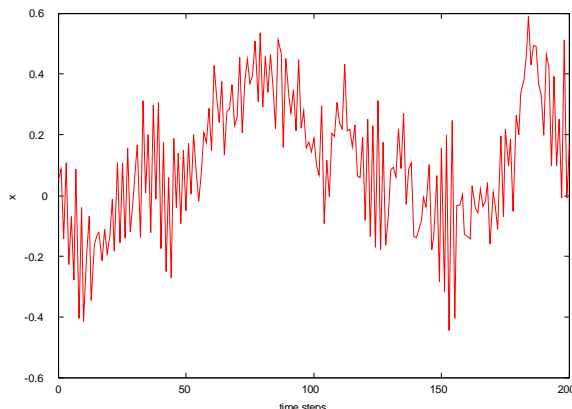


FIGURE 4. The jumping of the state between the positive fixed point and the negative one in the case of strong noise.

3.2. Delay of more than one step. Let us consider the case of a delay of T time steps (where $T = 4$ is realistic in the case of our Khepera robot), i.e. consider the system

$$x_t = \psi(x_{t-T}) + \xi_t \quad (3.1)$$

Let us call

$$S = \{x_t | t = 0, 1, \dots\}$$

the (full) orbit and

$$S_m = \{x_t | t = m, T + m, 2T + m, \dots\} \quad (3.2)$$

a suborbit of the system 3.1, where $m = 0, 1, \dots, T-1$. According to eq. 3.1 each suborbit corresponds to a full orbit without delay. Without the noise each suborbit is written explicitly as

$$x_t = \psi^n(x_m) \quad (3.3)$$

where $t = Tn + m$ and ψ^n stands for the n fold iteration of the function ψ .

The fixed point equation for a fixed point in orbit S_m reads

$$x^{(m)} = \psi(x^{(m)}) \quad (3.4)$$

according to eq. 3.1. Of course the fixed points and their property agree with the no delay case. As seen from eq. 3.3 the reached fixed point depends exclusively on the initial condition for the orbit considered.

In our example where

$$\psi(x) = a \tanh(cx)$$

we know that for $R > 1$ the sign of the FP in a suborbit is given by that of the initial value for the suborbit. With $R > 1$ in the deterministic case each of the orbits converges towards its fixed point the sign of which depends on the sign of the starting value alone. Let us call the sequence of the signs of the initial values x_0, \dots, x_{T-1} the signature of the initial conditions of the full orbit. Then, asymptotically the full orbit is a period- T dynamics with each state x_t , $t = 0, 1, 2, \dots$ at a fixed point the signs of the FPs being given by the signature of the initial conditions.

When including weak noise the state in each suborbit will fluctuate around its fixed point. However if the noise is strong enough, the state in any of the suborbits may occasionally switch sign and if the noise is not too strong will stay there for a longer time. Hence the full orbit is an irregular (modulated by fluctuations) period T behavior where however the signature of the period can switch occasionally. This is clearly seen by the computer simulations, see the Figures 5, ?? and Fig6. For the robot this means that it moves for some time into one direction (all signs positive or negative), then goes into a jumpy mode (different signs in one period), may switch between different jumping modes (signatures) and also may return to the steady mode and so on.

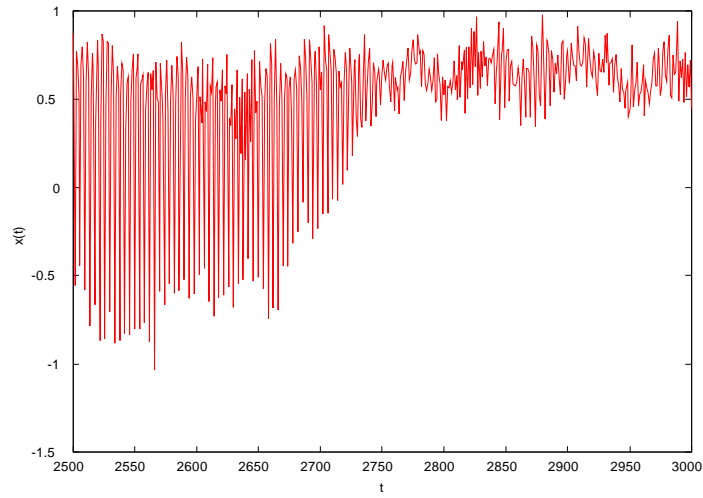
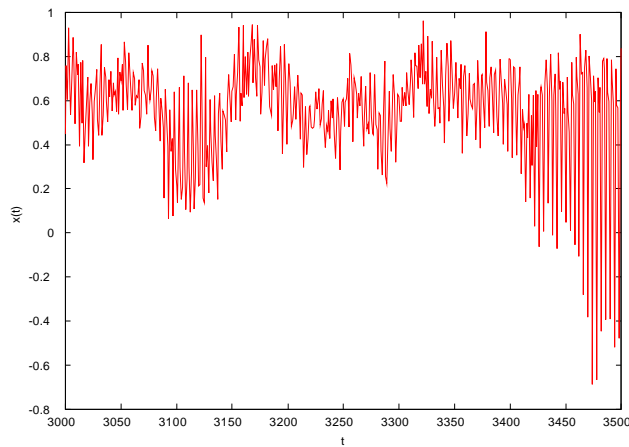


FIGURE 5. Full orbit of the system $x_{t+1} = a \tanh(cx_{t-3}) + \xi_t$ with $c = 1.2$, ξ is a Gaussian noise with strength 0.15. The system makes a transition from the jumpy to the steady mode.



Transition from the steady to the jumpy mode.

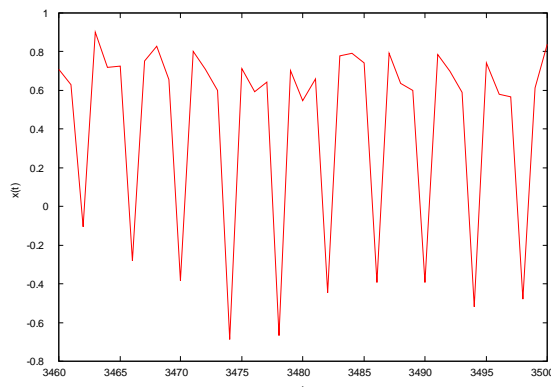


FIGURE 6. The full orbit in a small interval of time. We see explicitly here that three of the suborbits are fluctuating about the positive fixed point and the remaining suborbit fluctuates about the negative fixed point.

3.2.1. *How to overcome the jumpy mode.* When in the jumpy mode the velocity of the robot jumps between positive and negative values in one time step. Physically, this is a rather nasty behavior which after some time may destroy the robot. It is therefore of interest that this mode can be damped by a simple averaging procedure. Let us average the input into the controller over a time window of N steps (where $N \leq T$), i.e. the controller outputs are

$$y_t = K(\bar{x}_t)$$

where

$$\bar{x}_{t'} = \frac{1}{N} \sum_{i=0}^{N-1} x_{t'-i}$$

so that the dynamics now is

$$x_t = \psi(\bar{x}_{t-T}) + \xi_t \quad (3.5)$$

instead of equation 3.1. Our computer simulations show that the jumpy mode disappears and the state converges to the steady state, cf. Fig 7 which is without noise.

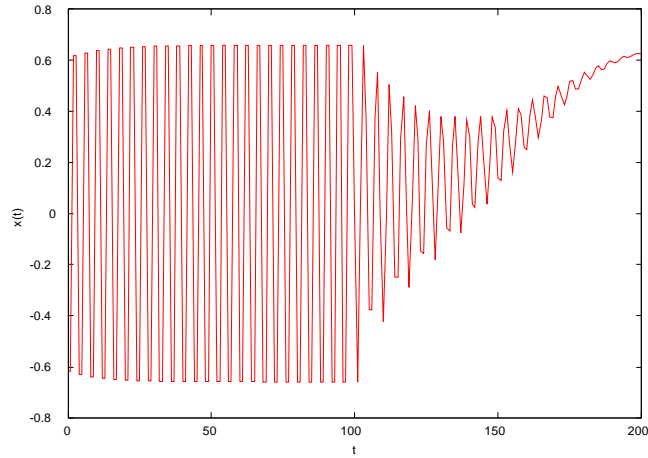


FIGURE 7. The transition of the state from the jumpy mode to the steady mode after using the time smoothing procedure over $N = 2$ steps. The process is without noise. Initial conditions have been $x_0 = x_1 = -x_2 = -x_3 = x^*$ where $x^* = 0.65$ is at the FP.

Using the time smoothing procedure with $N = 4$, we observe that the transition of the state from the jumpy mode to the steady mode occurs faster than in the case of $N = 2$. This can be seen by comparing Fig. 7 with Fig. 8

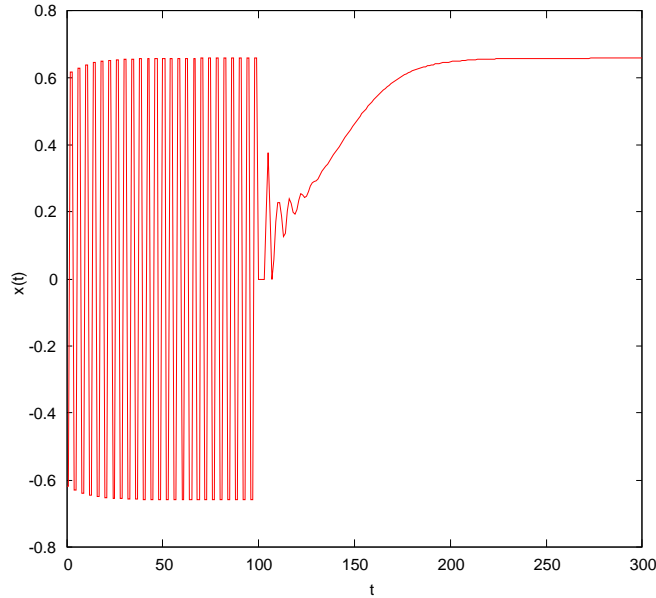


FIGURE 8. The transition of the state from the jumpy mode to the steady mode using time smoothing process with $N = 4$ occurs faster (in comparison with $N = 2$)

In the case of weak noise we observe again the transition of the state from jumpy mode to fluctuation around the steady state after using the time smoothing procedure over $N = 2$ steps, cf. Fig. 9.

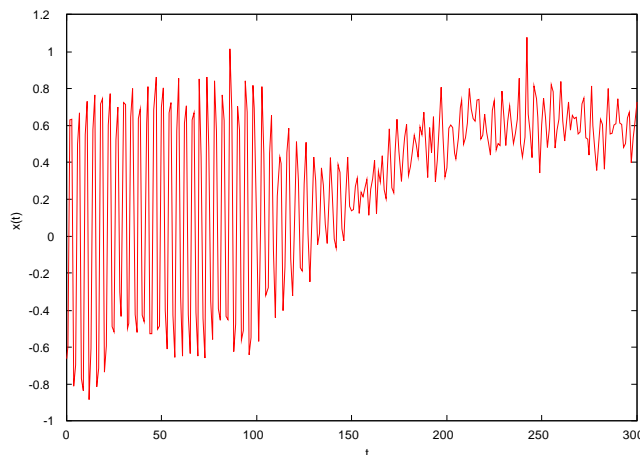


FIGURE 9. The transition of the state from the jumpy mode to the fluctuation around the steady state after using the time smoothing procedure over $N = 2$ steps and using the process with noise of strength $\xi^2 = 0.1$. The initial conditions have been $x_0 = x_1 = -x_2 = -x_3 = 0.65$, where $x^* = 0.65$ is the fixed point.

3.2.2. *Experiment with the real robot.* The use of the time smoothing procedure to overcome the jumpy mode is verified practically with a Khepera robot. We used moderate noise of strength 0.15 in order to get the jumpy mode first and then by using the time smoothing procedure we obtain the desired transition to the steady state fluctuation around the negative fixed point, see the Fig. 10. The simulations have shown that the time averaging over a short period of time reliably destroys the jumpy mode and makes the robot behavior more steady.

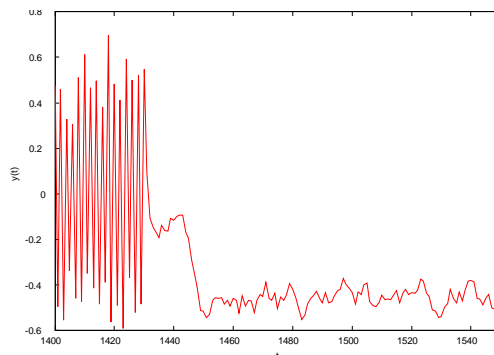


FIGURE 10. Stabilization of the behavior with a Khepera robot.

CHAPTER 3

The approach to self-referential systems

The present chapter will introduce the self-referential systems as a systematic approach to the self-organization of behavior for autonomous robots. In order to do so we at first formulate the dynamical system description in a more general context, where the robot is equipped with a simple self-model. Then we will define adaptive systems and built on this the concept of self-referential dynamical systems. This is followed by the discussion of some formal properties of these systems.

1. Dynamical system formulation

In order to give a concrete example we consider a robot in a given environment. We always deal with discrete times $t = 0, 1, 2, \dots$. At each instant of time we observe the sensor values $x_t \in R^n$. An example is the Khepera robot with $n = 10$:

$$x = (v_l, v_r, IR_1, \dots, IR_8)^T$$

where

- v_l and v_r are the wheel velocities of the left and right wheel, respectively, as measured by the wheel counters.
- IR_i is the value of the infrared sensor i , $0 \leq IR_i \leq 1$.

1.1. The controller. The robot has a controller with output $y \in R^m$ realized by a function $K : R^n \rightarrow R^m$

$$y = K(x) \tag{1.1}$$

y_1, y_2 being the target wheel velocities in the Khepera case ($m = 2$). At each instant of time t the robot measures the vector of its sensor values x_t . In the course of time we get a time series of sensor values. Our controller is to be adaptive, i.e. it depends on a set of parameters $C \in R^C$. In practice the controller is a neural network with the synaptic strenghts and the biases of the neurons as parameters. The behavior of the robot and hence the nature of the time series of sensor values depends on these parameters in an essential way.

1.2. The world model. Our robot is to have a certain ability for cognition. We postulate that cognition is to understand what happens. The prediction of the future sensor values requires some kind of understanding, i.e. we use a time series predictor as the cognitive instance in our approach.

1.2.1. *The forward model.* Prediction is realized by a forward model predicting the new sensor values x_{t+1} on the basis of the old sensor values x_t and motor values y_t . This is realized by a function $F : R^n \times R^m \rightarrow R^n$, i.e.

$$x_{t+1} = F(x_t, y_t) + \xi_t \quad (1.2)$$

where $\xi_t \in R^n$ is the model error. In practice we realize F by a neural network which can be trained by any algorithm of supervised learning using the measured values x_{t+1} as target outputs of the net. The most common tool is gradient descent based on the error

$$E = \|x_{t+1} - F(x_t, y_t)\|^2 \quad (1.3)$$

where $\|\dots\|$ is the Euclidean norm in general.

Assuming that the world model depends on a set of parameters A we may train the model by gradient descent as

$$\Delta A = -\varepsilon \frac{\partial E}{\partial A} \quad (1.4)$$

In an on-line scenario, the set of parameters A is updated in each time step, i.e. with each training instance given by a pair $(x_{t+1}; x_t, y_t)$ being available at time t .

1.2.2. *Modelling the dynamics of the time series.* Assuming F to be known we now may also formulate a model dynamics for the time series of the sensor values x_t . Using Eq. 1.1 in F we introduce the function $\psi : R^n \rightarrow R^n$ where

$$\psi(x) = F(x, K(x))$$

so that the dynamical law for the time series of the x_t may be modelled as

$$x_{t+1} = \psi(x_t) + \xi_t \quad (1.5)$$

where ξ_t is the model error again. ψ can be called a model of the time series the robot produces by its movements.

2. Adaptive systems

The dynamics depends on the parameters C of the controller K . Changing C changes the behavior of the model. A system is adaptive if there is an objective function measuring the distance from the current to a desired behavior. (This might be as abstract as increasing the survival properties of the system). One realization is parameter dynamics as gradient flow

$$\Delta C_t = -\varepsilon S(x_t, C_t)$$

where

$$S = \frac{\partial}{\partial C} E$$

so that we have the combined dynamics

$$\begin{aligned} x_{t+1} &= \psi(x_t, C_t) + \xi_t \\ C_{t+1} &= C_t - \varepsilon S(x_t, C_t) \end{aligned} \quad (2.1)$$

and we can also include the parameter dynamics of the world model according to eq. 1.4. This involves the problem that the learning of the controller and that of the world model influence each other. In particular the controller would have to adapt in such a way that the world model is provided

with good learning examples (pairs $x_{t+1}; x_t, y_t$) so that it learns as much as possible about the dynamics of the sensorimotor loop.

The problem however is with finding the objective function E . In supervised learning case we may assume that there is a teacher which in each instant of time provides the controller outputting $y_t = K(x_t)$ with a target output $y_t^{(teach)}$ so that we can use a gradient descent on the error

$$E^{(teach)} = \left\| y_t^{(teach)} - K(x_t) \right\|^2$$

which depends on the parameters C of the controller in an explicit way.

In general the teaching values are difficult to produce so that methods of unsupervised learning are of much value. We are going to give a pertinent approach in the following. The aim is at this stage of the development to find an approach which realizes something like the self-exploration of the behavior of the agent, in particular the self-exploration of the possibilities of its physical body. Our approach is via self-referential dynamical systems.

3. Self-referential systems

The aim of our work is to find objectives which are entirely inherent to the agent and not given from outside. Hence the system is to find its own reference to the world and this is why we call them self-referential systems. In more detail, we consider behavior as a dynamical reference to the world. Adaptation improves this reference to the world in the direction of a special performance (survive, e.g.). Self-referential systems on their hand are adaptive systems which define their behavior (dynamical reference to the world) by their dynamics alone. A special criterion is the dominance of the internal perspective. These systems have no program, no purpose and no goal. The aim is to create self-referential systems with emerging behaviors which are closely related to the world the agent is acting in.

At the formal level self-referentiality is established by the requirement that the objective function E is formulated entirely only in the sensor states and their dynamics modelled by ψ . In particular there should be no fixed goal states or any other domain specific information be involved in the formulation of E . Before presenting the approach due to Der et al. we consider at first some naive examples how such objectives could be obtained.

3.1. Smoothness of behavior. An example which is into our direction is given by the requirement that the robot is to move such that the sensor values stay as constant as possible. If x is the vector of sensor values the objective E is

$$E = \|x_{t+1} - x_t\|^2$$

The transition from sensor values x_t to x_{t+1} of course depends on the controller so E is a function of C . We could also use a larger time horizon over which we require the sensor values to be constant. However the above objective has the trivial "do nothing" behavior as a stable solution, since it keeps the sensor values constant, in a static world at least. Thus we have to endorse the robot with a basic activity, for instance the forward velocity is set constant and the controller has only to learn to output an appropriate turn velocity. It has been shown in [7] by way of examples with real robots

that the objective indeed generates smooth controlled behaviors of the robot in complex worlds. In particular stable wall following and object pushing behavior have been observed under different conditions. This is in agreement with the paradigm since under such a behavior the sensor values stay more or less constant. However, from the point of view of the self-referential system or the true self-organization of behavior, setting the forward velocity is a domain related prescription which we actually want to avoid.

3.2. Predictability. Another more general approach is formulated in control so that the consequences of the actions taken are well predictable. This requires the availability of a world model as proposed in eq. 1.2. This means that we consider the SM dynamics

$$x_{t+1} = \psi(x_t) + \xi_t \quad (3.1)$$

where $\psi(x) = F(x, K(x))$ so that $\hat{x}_{t+1} = \psi(x_t)$ is the estimate of the new sensor values given by the world model F . We use

$$E_{t+1} = \|x_{t+1} - \psi(x_t)\|^2 \quad (3.2)$$

as the error for the adaptation of the parameters. Using eq. 3.1 we find

$$E_{t+1} = \|\xi_t\|^2$$

so that there is no explicit dependence on the parameters of the controller. In order to get a direct learning signal we must extend the estimation of the new sensor values over more than one time step, i.e. consider

$$E_{t+1} = \|x_{t+1} - \psi(\psi(x_{t-1}))\|^2 \quad (3.3)$$

where $\hat{x}_{t+1} = \psi(\psi(x_{t-1}))$ is obtained from the iteration of eq. 3.1 with the noise suppressed.

Obviously in both cases E fulfills the requirement that it depends only on the sensor values itself together with the model of the state dynamics ψ . In order to estimate the consequences of such an approach we consider the elementary one-dimensional SM loop, $x_{t+1} = ay_t + \xi_{t+1}$ or

$$x_{t+1} = ag(Cx_t) + \xi_{t+1}$$

so that $F(x, y) = ay$,

$$\psi(x) = ag(Cx) \quad (3.4)$$

so that $\psi : R^1 \rightarrow R^1$ and a measures the strength of the response of the world to the actions of the controller. We put $a = 1$ in the following discussion without restriction of generality. Using eq. 3.3 the error E is

$$E_{t+1} = \|x_{t+1} - \psi(\psi(x_{t-1}))\|^2 = \|\psi(\psi(x_{t-1}) + \xi_t) + \xi_{t+1} - \psi(\psi(x_{t-1}))\|^2 \quad (3.5)$$

Using Taylor expansion with the assumption of weak noise we get

$$E_{t+1} = \|x_{t+1} - \psi(\psi(x_{t-1}))\|^2 = \|\xi_{t+1} - L(x_{t-1})\xi_t\|^2 \quad (3.6)$$

where

$$L(x) = \frac{\partial}{\partial x}\psi(x) \quad (3.7)$$

Assuming moreover that the noise is uncorrelated over the time steps and is of zero mean, i.e. $\overline{\xi_t \xi_{t+1}} = 0$ we find in the average over the noise

$$E_{t+1} = D_t L^2(x_{t-1}) + D_{t+1}$$

where

$$D_t = \overline{\xi_t^2}$$

Note that we encountered L already in Chapter 2.

In the specific case of eq. 3.4 we get

$$L(x) = Cg'(Cx)$$

so that the learning rule for the parameter C reads

$$\Delta C = -\varepsilon \frac{\partial E}{\partial C} = -\varepsilon (g'(z) + zg''(z)) \quad (3.8)$$

where $z = Cx$. In the generic case that $g(z) = \tanh(z)$ we have $g''(z) = -2g(z)g'(z)$ so that the learning rule simplifies to

$$\Delta C = -\varepsilon Dg'(z)(1 - 2zg(z)) \quad (3.9)$$

In order to discuss this we use the results of Chapter 2, i.e.

$$z = \pm \sqrt{3 \frac{C-1}{C}} \approx \pm \sqrt{3\delta} \quad (3.10)$$

where $C = 1 + \delta$ with $0 < \delta \ll 1$. Let us assume for a further discussion that ε is so small that the time for the change of C is much longer than for the dynamics of x . In that case one can assume that the state is always converged to one of the stable fixed points so that we can use in eq. 3.8 the value for z at the fixed point. Then we see that for $0 < C < 1$ we have

$$\Delta C = -\varepsilon D$$

meaning that C is driven towards $C = 0$. In order to discuss the situation with $C > 1$ we must at first find the zero of the right hand side of the learning rule, i.e.

$$1 = 2zg(z)$$

with numerical solution $z = \pm 0.7717$ and using this in $z = Cg(z)$ we find numerical $C_{crit} = 1.1911$. Using the approximation we find instead $z = \pm 0.79623$ and using this in eq. 3.10 we find $C_{crit} = 1.2113$. Hence the agreement is very good from the practical point of view.

Returning to eq. 3.9 we find that if $C < C_{crit}$ we have $\Delta C < 0$ so that C is driven towards $C = 0$ as before. If $C > C_{crit}$ we find on the other hand that C increases towards infinity where $z = \pm 1$. What we learn from this is that, depending on the initial conditions for C , we are driven towards either $z = 0$ or $z = \pm 1$ where the fixed points have maximum stability. This result is in agreement with our paradigm of maximum predictability of the results of the actions undertaken since the predictability is best if the systems stabilizes against the influence of unpredictable events like our noise ξ_t .

For the robot this would mean that it either stalls as a result of the learning procedure or moves with maximum velocity either forward or backward without being able to switch between these modes of behavior. In other word one may say that we have realized in this way an ultrastable system in the sense of the homeostatic principle of Ross Ashby.

3.3. The time loop error. However, an ultrastable system is not our aim. Instead we want a robot with self induced activities which are a sensitive response to the sensor values it produces by its activities. It has been shown by the work of R. Der and his coworkers that there is a simple way to get such a system. This consists in not using the prediction error as in eq. 3.2 but instead the so called reconstruction error, i.e. find the reconstructed value \hat{x}_t from the currently observed x_{t+1} . If ψ is invertible the reconstructed state \hat{x}_t may be defined as

$$\hat{x}_t = \psi^{-1}(x_{t+1}) \quad (3.11)$$

or

$$x_{t+1} = \psi(\hat{x}_t) \quad (3.12)$$

leading to the objective

$$E_t = \|x_t - \hat{x}_t\|^2 \quad (3.13)$$

With $\hat{x} = x + v$ we can also write

$$\psi(x) + \xi = \psi(x + v) \quad (3.14)$$

i.e. shift the input in order to correct the output of the model. If ψ is not invertible this inverse task has in general none or even more than one solution. In either case one tries to solve the task as good as possible, i.e. one tries to minimize the distance between x_{t+1} and $\psi(x + v)$ so that

$$v = \arg \min_u \|x_{t+1} - \psi(x + u)\|$$

In the ambiguous case (several solutions) one has to provide a heuristics on the choice of v , each choice producing a different kind of behavior of the agent. In the present work we always will work with the invertible case.

The reconstruction error is defined in terms of the shift vector v as

$$E = v^T v = Tr(vv^T) \quad (3.15)$$

where v is interpreted as a row matrix (vector) and Tr is the trace of a matrix. E is called also the time loop error because it is the error obtained from starting with state x_t , go one step into the future by the true dynamics and then go back by the model dynamics ψ , cf. eq. 3.11. Note that E is a function of the current values of both the state variable x and the parameters C of the controller and the parameters A of the world model, i.e. $E = E(x_t, C_t, A_t)$.

We now define the parameter dynamics by the requirement that the reconstruction error is minimized. Using gradient descent this leads to the combined dynamics in the space $R^n \times R^C$, i.e. we have to consider the dynamical system

$$\begin{aligned} x_{t+1} &= \psi(x_t, C_t) + \xi_t \\ C_{t+1} &= C_t - \varepsilon S(x_t, C_t) \end{aligned} \quad (3.16)$$

where $S = \partial E / \partial C_t$ and the second equation stands for the set of equations for the parameters C .

4. Explicit expressions

Consider the defining equation for the shift vector v

$$x_{t+1} = \psi(x) + \xi = \psi(x + v) \quad (4.1)$$

for the small noise case so that (Taylor expansion)

$$\psi(x + v) = \psi(x) + L(x)v \quad (4.2)$$

where

$$L_{ij}(x) = \frac{\partial}{\partial x_j} \psi_i(x)$$

is seen to be the Jacobian matrix of the state dynamics as known by the model ψ of the sensorimotor loop. Consequently

$$\xi = L(x)v$$

and obtaining v means now "only" to find the inverse of the matrix L . This might cause troubles if the matrix is singular.

In the regular case (which we will assume throughout) the time loop error is

$$E = v^T v = \xi_t^T L^{-1T} L^{-1} \xi_t = \xi_t^T (LL^T)^{-1} \xi_t$$

Introducing the correlation matrix of the noise $D = \xi \xi^T$ (suppressing the index t) we can also write E as a trace

$$E = Tr \left((LL^T)^{-1} D \right) \quad (4.3)$$

which is sometimes beneficial for formal considerations (exploiting the cyclic invariance of the trace). Introducing the weighted matrix norm

$$Tr (SS^T D) = \|S\|_D^2$$

we write

$$E = \|L^{-1}\|_D^2 \quad (4.4)$$

More explicit expressions will be given below.

5. Formal considerations

The specific form Eq. 4.4 of our objective function defines a number of properties at the formal level.

5.1. Some explicit results. Consider the time loop error of eq. 4.4. We use

$$L^{-1} = \frac{1}{\det L} \tilde{L} \quad (5.1)$$

where the matrix element \tilde{L}_{ij} is given by $(-1)^{i+j}$ times the cofactor Λ_{ij} which is given by the determinant obtained from eliminating both the row i and the column j of the matrix L^T . Then

$$E = \frac{1}{\det^2 L} Tr \left(\tilde{L}^T \tilde{L} D \right) \quad (5.2)$$

With uniform orthogonal noise we have the special case that D is proportional to the unit matrix. Using $D = \sigma^2 \mathbf{1}$ (where σ^2 measures the noise strength) we obtain

$$E = \sigma^2 \frac{\|\tilde{L}\|^2}{\det^2 L} \quad (5.3)$$

Note L is the Jacobian matrix of the state dynamics as it is known in terms of the model ψ and that $\det L < 1$ means contracting, $\det L > 1$ expanding dynamics. Minimizing E means increasing $\det L$ while minimizing $\text{Tr}(\tilde{L}^T \tilde{L} D)$.

5.2. Persistence of initial values. An interesting aspect is also seen if we use the singular value decomposition of the Jacobian matrix, i.e. write

$$L = SPU \quad (5.4)$$

where S and U are orthogonal matrices and V is diagonal. We then immediately find that

$$E = \text{Tr}\left((SP^2S^T)^{-1} D\right) = \text{Tr}\left(P^2 \tilde{D}\right) \quad (5.5)$$

where $\tilde{D} = S^T D S$. Obviously the matrix U is not adapted by gradient descending E . This is independent on the properties of the noise.

Assuming now that the noise is orthogonal and of equal strength in all channels we see that E boils down to

$$E = \sigma^2 \sum_{i=1}^n p_i^{-2} \quad (5.6)$$

where p_i are the diagonal elements of P . Gradient descent will increase these unrestricted. Adding a damping term to the gradient rule we get

$$\Delta p_i = \varepsilon \sigma^2 p_i^{-3} - \lambda p_i$$

which is converged if

$$p_i^4 = \frac{\varepsilon \sigma^2}{\lambda}$$

Assuming further that the dynamics is linear, i.e.

$$x_{t+1} = SPUx_t + \xi_t$$

we easily see that the case that $p_i < 0$ leads to dynamics alternating in sign in the corresponding dimension. This is to be considered as non-physical since the robot can not follow such rapidly changing commands. We therefore have to initialize all $p_i > 0$ which also means that after convergence

$$p_i = p = \sqrt[4]{\frac{\varepsilon \sigma^2}{\lambda}} \quad (5.7)$$

or $P = p\mathbf{1}$ and the Jacobian matrix after convergence is $L = pSU$ so that the dynamics is

$$x_{t+1} = pRx_t + \xi_t \quad (5.8)$$

where $R = SU$ is also an orthogonal matrix. Hence the state dynamics consists of a rotation of the vector x_t and a rescaling by a factor p . Note that the rotation is persisting from the initial conditions of the matrix C .

5.3. The exploration scenario. Interpretation: The parameter dynamics tends to stretch the state vector into all dimensions by the same factor p . This could be interpreted as the tendency of looking into all directions of the space with the same attention or as the uniform exploration of the space by the state vector.

If the noise is not uniform we may rewrite Eq. 5.5 as

$$E = \sum_{i=1}^n \frac{d_{ii}}{p_i^2} \quad (5.9)$$

where d_{ii} are the diagonal elements of the transformed noise matrix $S^T D S$. With the damping term added the gradient descent

$$\Delta p_i = \varepsilon d_{ii} p_i^{-3} - \lambda p_i$$

yields

$$p_i^4 = \frac{\varepsilon d_{ii}}{\lambda}$$

Here we see that the exploration rate is larger in the directions of strong noise. Noting that the noise actually is the model error we can say that the parameter dynamics is such that the dimensions of the state space where the model is bad (large ξ) are more extensively explored than those where the modeling is certain. This is an interesting consequence of the present approach.

6. Fixed point flow from self-referential dynamics

Up to now only linear systems were considered. The full flavor of the self-referentiality is seen only if the systems are nonlinear. We will in the thesis describe a number of interesting effects and see that the behavior modes the system are emerging from a subtle interplay between state and parameter dynamics which depends very much on the specific realization of the controller and the world model. However in the case that the time scales for the state and the parameter dynamics are well separated, we can draw some general conclusions on the behavior of the system.

Consider the case that the learning rate for the parameter dynamics is small and that the system given by ψ has stable fixed points (or other attractors). Consider the case of small learning rates ε in the combined dynamics

$$\begin{aligned} x_{t+1} &= \psi(x_t, C_t) + \xi_t \\ C_{t+1} &= C_t - \varepsilon S(x_t, C_t) \end{aligned} \quad (6.1)$$

where

$$S(x, C) = \frac{\partial}{\partial C} E(x, C)$$

The dynamics 6.1 in general has two different time scales. If we choose ε small enough the parameter dynamics can be made very slow as compared to the time scale of the state dynamics. As is well known from synergetics what happens is that the slow variables tend to slave the fast ones. In particular if the state dynamics with fixed parameters has a fixed point x^* as solution of

$$x^* = \psi(x^*, C_t)$$

we have

$$x^* = \chi(C_t) \tag{6.2}$$

and

$$Q(C) = S(\chi(C), C)$$

so that

$$C_{t+1} = C_t - \varepsilon Q(C_t) \tag{6.3}$$

which is a closed dynamics for C . In this way we have a separation of state and parameter dynamics, the parameters playing the role of the master variables which slave the state dynamics.

As a consequence, the dynamics of C generates a fixed point flow. The state x_t of the system is

$$x_t = \chi(C_t) \tag{6.4}$$

However the parameter dynamics generally tends to run into the direction of low stability (see Chapter 4), eventually fixed points can disappear, so that the slaving principle is not any longer valid and state dynamics jumps to another fixed point where the scenario restarts.

CHAPTER 4

One-dimensional system with learning

We studied before in Chapter 2 the dynamics of the system with fixed parameters (c and H). In the following sections we will study the dynamic of the the same system but with the learning of its parameters.

1. Parameter dynamics for the model system

Let us consider again the case that our robotic system is modelled by the dynamical system

$$x_{t+1} = ag(cx_t + H) + \xi_{t+1} \quad (1.1)$$

so that $\psi(x) = ag(cx + H)$. The approximate solution of the equation

$$\psi(x) + \xi = \psi(x + v)$$

yields

$$v = \frac{\xi}{L(x)} \quad (1.2)$$

where $z = cx + H$,

$$L(x) = Rg'(z) \quad (1.3)$$

and the time loop error is

$$E = v^2 = \frac{\xi^2}{L^2}$$

the parameter dynamics for the controller is by gradient descent given as (absorbing the factor of 2 into ε)

$$\begin{aligned} \varepsilon^{-1}\Delta c &= \frac{\xi^2}{L^3} \frac{\partial}{\partial c} L = \frac{E}{L} \frac{\partial}{\partial c} L \\ \varepsilon^{-1}\Delta H &= \frac{E}{L} \frac{\partial}{\partial H} L \end{aligned}$$

In the case of $g(z) = \tanh z$ we have $g''(z) = -2g'(z)g(z)$ so that we may also write

$$\begin{aligned} \varepsilon^{-1}\Delta c &= \frac{E}{c} (1 - 2zg(z)) \\ \varepsilon^{-1}\Delta H &= -2\frac{E}{c}g(z) \end{aligned}$$

or

$$\begin{aligned} \Delta c &= \varepsilon \frac{E}{c} (1 - 2zy) \\ \Delta H &= -2\varepsilon \frac{E}{c} y \end{aligned} \quad (1.4)$$

both z and y being taken at time t , i.e. $y = g(cx_t + H)$.

What is immediately seen is that the update has a singularity at $c = 0$ so that this value can not be crossed by the parameter dynamics. The case $c < 0$ however is specified by the fact that there we have a jumping dynamics because of $x_{t+1} = g(cx_t)$ and $g(z) = -g(-z)$ so that positive x are converted into negative ones and vice versa. We therefore consider the case $c < 0$ as non-physical regime which is to be avoided by using in the following $c > 0$ as the initial condition for the parameter dynamics.

1.1. No bias. In order to better understand the system we study at first the case that the bias $H = 0$. We will see that we obtain the convergence to a fixed point slightly above the bifurcation regime. These considerations have been already published elsewhere but are important for understanding the higher dimensional systems so that we will give the results here in some detail. We essentially follow the presentation in [8].

When comparing eq. 1.4 with the corresponding expression for the parameter dynamics obtained from the prediction error, see eq. 3.9 of Chapter 3 we see that the essential difference is in the sign of the update. We have again to differ between the cases $R < 1$ and $R > 1$. Starting with $0 < R < 1$ we have the fixed point $z = 0$ and hence

$$\Delta c = \varepsilon \frac{E}{c}$$

which means that c is increasing and approaching the bifurcation point. Once there the new fixed points $z = \pm q$ are emerging. With c still close to 1 the term $1 - 2zy$ is still > 0 so that c is increasing further and reaches a stationary value if $1 - 2zy = 0$ which as discussed earlier is the case at $c = 1.1911$ where $z = \pm q$ with $q = 0.7717$. Using the approximation $q = \sqrt{3}(c - 1)$, which is valid in leading order of $\delta = c - 1$, we find

$$c = \frac{3}{2} - \frac{1}{6}\sqrt{3} = 1.2113$$

and $z = \frac{1}{2}\sqrt{6 - 2\sqrt{3}} = 0.79623$ as noted before.

Hence the system adapts itself to a regime slightly above the bifurcation point. This means that the system takes decisions (it goes to one of its fixed points) but on the other hand is still open to the environment since it can be switched by the noise (environmental influences) to the other alternative. The latter point would be not the case if the system is further away from the bifurcation point.

2. Including the bias dynamics

Without bias the system is seen to converge to a simple fixed point dynamics since the parameter for c converges. This is different if we include the bias dynamics as well. The essential point is seen from remembering the hysteresis properties of the loop dynamics described in Chapter 2. The dynamics for the bias H is

$$\Delta H = -\varepsilon \frac{E}{c} g(z)$$

Assuming $z > 0$ the value of H is seen to be driven to negative values. Because of the hysteresis effect the value of z will stay on the positive side

until the critical value $H = -H_c$ is reached. Then the state z will switch to negative values which leads to an increase of the bias until the value $H = H_c$ is reached and the state again changes sign. As a consequence we obtain an oscillatory system characterized by a limit cycle in the combined space $\{z, c, H\}$. The value of c is also slightly oscillating but these oscillations are not essential for the phenomenon since the latter are also obtained if we keep $R > 1$ fixed. Atypical run of a numerical simulation is given in the Figs.1 and 2.

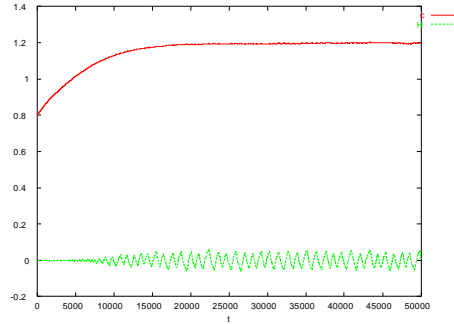


FIGURE 1. From this graph we observe c converges after increasing for some time to the value ≈ 1.2 . Also in this graph we observe the oscillations of H .

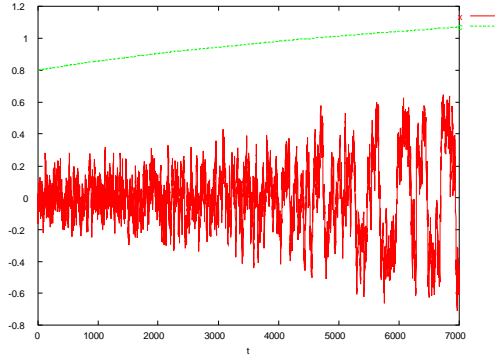


FIGURE 2. From this graph we observe that the state variable x start to oscillate only after c becomes greater than 1. At this point the system has two fixed points and the learning of h will drive the system to switch between its two fixed points.

From these figures one observe that c increases and as long as its value still less 1, the state variable x stays at fixed point of the system which is equal to 0 and only when c becomes greater than 1, the state variable start to oscillate. The reason for this oscillations is that if $c > 1$ and $H > 0$, then the state variable converges after some time to its positive fixed point (which

is stable), but decreasing H (which is supported by its learning rule if the state is positive) till $H = -H_c$ as discussed in Sec.2 and Sec.2.2 will result in the switching of the system to its negative fixed point and the opposite will occur if $H < 0$ which is known as hysteresis effect.

2.1. Analysis of the oscillating system. The above findings have been reported already in papers by Der et al. In the present thesis we include a dynamical systems analysis. In order to keep this as simple as possible we keep the value of $c > 1$ fixed (put $a = 1$ so that $R = c$ here) so that we get the combined dynamics of state and parameters as

$$\begin{aligned} z_{t+1} &= cg(z_t) + H_t \\ H_{t+1} &= H_t - \mu g(z_t) \end{aligned}$$

where $\mu = \varepsilon E/c$. Close to the bifurcation point z is not large so that we may linearize $g(z) \rightarrow z$. We obtain the linear system

$$\begin{aligned} z_{t+1} &= cz_t + H_t \\ H_{t+1} &= H_t - \mu z_t \end{aligned}$$

with Jacobian matrix

$$J = \begin{pmatrix} c & 1 \\ -\mu & 1 \end{pmatrix}$$

with eigenvalues

$$\lambda_{1/2} = \frac{1}{2}c + \frac{1}{2} \pm \frac{1}{2} \sqrt{(c-1)^2 - 4\mu}$$

Oscillations set in if the eigenvalues become complex which is the case if

$$(c-1)^2 < 4\varepsilon \frac{E}{c}$$

Independently on the value of $c > 1$ and the value of E (which is determined by the noise = modeling error ξ) this can always be achieved if ε is chosen large enough. Moreover what the result also discloses is the fact that the frequency of the oscillations depends largely on the strength of the noise. With $\mu \gg c-1$ we find that the frequency (given by the imaginary part of the eigenvalue) is proportional to $\sqrt{\mu}$.

2.2. Learning the world model . The dependence of the frequency of the oscillations on the strength of the modeling error will now be shown to influence positively the learning of the world model. In practical applications the modelling error can change over orders of magnitude depending of the situation the robot is in. If this is one which he has not yet "understood" (large modeling error) it will increase the frequency of its oscillations so that it explores the situation more actively. In the present work we do not consider learning of the world model $F(x, y)$ which in our simple case means $F(x, y) = ay$ where the constant a is to be learned. a reflects the strength of the sensoric response to the action y of the controller. One can imagine that a higher frequency of the robots forward and backward motion will give more information on this constant. In more detail, let us assume that the true dynamics is given by $x_{t+1} = \alpha y_t + \xi_t^{(0)}$ whereas we the system uses the constant a so that

$$\xi = (\alpha - a)y + \xi^{(0)}$$

where $\xi^{(0)}$ is the remaining modeling errors due to true noise, e.g. Hence the error is immediately seen to be proportional to the difference between true and the model value of the response strength.

The parameter a can be learned by gradient descending ξ^2 , i.e.

$$\Delta a = \kappa \xi y$$

and this can be learned with a constant value of y so that there is no advantage from the error proportional frequency of the robot's motion. However the situations changes immediately if the model is a little less trivial for instance if the sensor value has a bias, too. Assume the true dynamics is given by

$$x_{t+1} = \alpha y_t + \beta + \xi_t^{(0)}$$

so that our model is to be used as $F(x, y) = ay + b$ so that $\xi = (\alpha - a) + \beta - b + \xi^{(0)}$ so that the

$$\Delta a = \kappa \xi y$$

$$\Delta b = \kappa \xi$$

Using $\overline{\xi^{(0)}} = 0$ we obtain, in the average over the noise, the convergence condition for the learning dynamics as $\xi = 0$, i.e. (assuming $y \neq 0$)

$$(\alpha - a) y + \beta - b = 0$$

which has no unique solution. The situation improves however, if y changes, convergence towards the correct values being the faster the higher the frequency of y . Hence the self-induced oscillations of the learning system proves immediately valuable for the learning of the world model. This interaction between model and controller learning is even more intensive for higher dimensional systems with more complicated world models.

Two-dimensional systems

In this chapter we will study in some detail the case of 2-dimensional systems. The practical application is for instance with a two-wheel robot like the Khepera. We will see that the parameter dynamics, introduced by our general principle, produces a number of interesting phenomena which are of immediate importance for developmental and behavioral robotics. In order to prepare these considerations we start with discussing the dynamics of 2-dimensional systems with fixed parameters and introduced the learning dynamics afterwards.

1. Dynamics with fixed parameters

We consider by way of example a robot with two wheels, equipped with a controller $K : R^2 \rightarrow R^2$ so that

$$y_t = K(x_t) \quad (1.1)$$

defines the target velocities of the wheels at time t . The vector $x_t \in R^2$ contains the measured velocities as obtained by the wheel counters. The controller is to be realized by a simple one-layer neural network, i.e.

$$K(x) = g(Cx + H) \quad (1.2)$$

where g is a vector function defined by its components as $g_i(z) = g(z_i)$. We assume that the wheels essentially execute the target velocities prescribed by the controller so that we write

$$x_{t+1} = Ay_t + \xi_t \quad (1.3)$$

where A is essentially the unit matrix (apart from some hardware constants) in the case considered. The model error ξ contains all effects due to friction, slip, and any noise due to discretization, measurement errors and the like.

1.1. The dynamical system. Using Eq. 1.1 in 1.3 we obtain the stochastic, time-discrete dynamical system

$$x_{t+1} = Ag(Cx_t + H) + \xi_t \quad (1.4)$$

so that the function $\psi : R^2 \rightarrow R^2$ is given by

$$\psi(x) = AK(Cx + H) \quad (1.5)$$

The Jacobian matrix is

$$L_{ij} = g'(z_i) C_{ij}$$

or

$$L = G'(z)C = \begin{pmatrix} C_{11}g'_1 & C_{12}g'_1 \\ C_{21}g'_2 & C_{22}g'_2 \end{pmatrix} \quad (1.6)$$

with the diagonal matrix $G'(z) = \text{diag}[g'_1, g'_2]$ where $g'_i = g'(z_i)$.

In many cases it is convenient to formulate the dynamics directly in terms of the activations, i.e. consider

$$z_{t+1} = H + CAg(z_t) + \eta_t \quad (1.7)$$

where

$$z = Cx + H$$

is the vector of activations (post synaptic potential) of the neurons and $\eta_t = C\xi_t$. In the two dimensional system we have (rename CA as C)

$$\begin{aligned} z_1^{t+1} &= H_1 + C_{11}g(z_1^t) + C_{12}g(z_2^t) + \eta_1^t \\ z_2^{t+1} &= H_2 + C_{21}g(z_1^t) + C_{22}g(z_2^t) + \eta_2^t \end{aligned} \quad (1.8)$$

(we write the time as an upper index if we consider the components of the vector). The Jacobian matrix in this case is

$$L_{ij} = C_{ij}g'(z_j)$$

or

$$L = CG'(z) = \begin{pmatrix} C_{11}g'_1 & C_{12}g'_2 \\ C_{21}g'_1 & C_{22}g'_2 \end{pmatrix} \quad (1.9)$$

The difference between the two Jacobian matrices is that in Eq. 1.6 the rows and in Eq. 1.9 the column of the matrix C are multiplied by the g' factors. This however does neither change the trace, the determinant nor the eigenvalues so that we can use either form in the further analysis.

1.2. The bifurcation scenario. The dynamics of the system Eq. 1.8 has been studied intensively by several authors since it may be interpreted as the dynamics of a recurrent neural network with two neurons, cf. [44]. As is well known the eigenvalues of a two-dimensional matrix can be written in terms of the trace T and the determinant D of the matrix as

$$\lambda_{1,2} = \frac{1}{2}(T \pm \sqrt{(T^2 - 4D)}) \quad (1.10)$$

The eigenvalues λ are functions of the network parameters. The dynamics is asymptotically stable if the two eigenvalues lie inside the unit circle in the complex plane, however it bifurcates if an eigenvalue λ – during parameter variation – leaves the unit circle. Depending on the exact location where $\lambda(T, D)$ crosses the unit circle, the following three bifurcation types can be distinguished:

- Saddle node (fold) bifurcation $\lambda = 1$: It creates or destroys a pair of fixed points.
- Period doubling (flip) bifurcation $\lambda = -1$: It creates or destroys a two-cycle from a fixed point.
- Neimark-Sacker bifurcation $\lambda = \exp(\pm i\omega)$ where $\exp(\pm i\omega) \neq 1$ for both signs: It creates an asymptotically stable invariant curve with periodic or quasi-periodic dynamics on it, depending on whether the rotation number is rational or irrational, see [32] and the discussion below.

According to the above definitions, the boundary of the stability of the fixed point of the system 1.6 in the (T, D) plane is given by a triangle Fig.1 bounded by the three straight lines $T - D = 1$, $T + D = -1$, $D = 1$. For instance, along the line $T + D = -1$ (on which the eigenvalues are real

and one of which is -1) there will be a period-doubling bifurcation. Along the line $D = 1$, $|T| < 1$ (on which the two eigenvalues are complex and of modulus 1), there will be Neimark-Sacker bifurcations from a fixed point attractor to a periodic or quasi-periodic attractor, see [55].

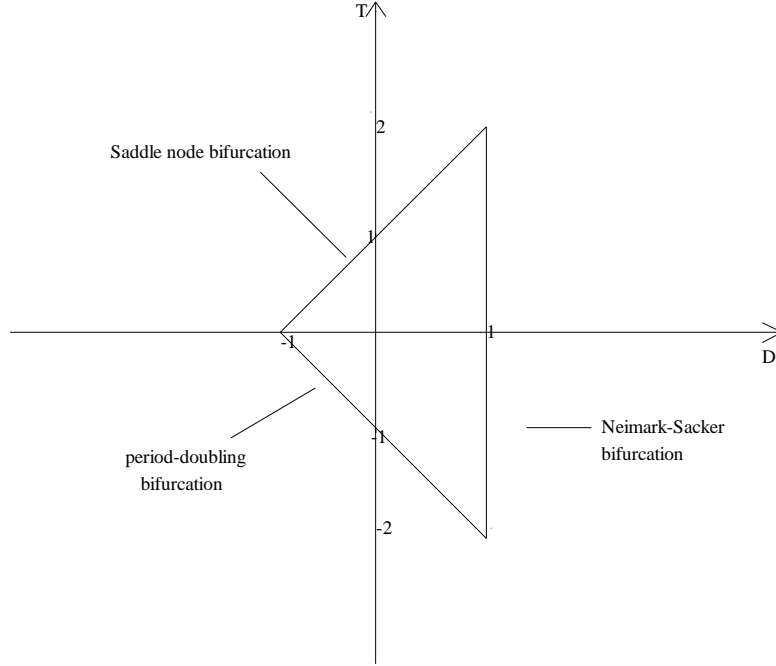


FIGURE 1. The domain of the stability of the fixed point x^* is given by the triangle bounded by the three lines $T - D = 1$, $T + D = -1$, and $D = 1$.

1.3. Properties of $SO(2)$ networks with fixed parameters. A specific form of the above two-neuron dynamics is obtained if C is essentially a member of the special orthogonal group, i.e.

$$C = uQ(\phi) \quad (1.11)$$

where

$$Q(\phi) = \begin{pmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{pmatrix}$$

is the rotation matrix. This is of interest since then the network becomes a nonlinear oscillator if $u > 1$, cf. [45]. Moreover we will find below that under certain conditions our learning dynamics drives the parameters to just this structure. Let us consider at first the case of small values of z where the system Eq. 1.8 can be linearized and one gets (dropping the noise for the moment)

$$z_{t+1} = uQ(\phi) z_t \quad (1.12)$$

This means that the state vector is rotated in each step with the angle ϕ and multiplied by the factor u . The solution of Eq. 1.12 is

$$z_t = u^t Q(t\phi) z_0 \quad (1.13)$$

If the starting vector is

$$z_0 = \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix}$$

then the state vector after t iterations will be

$$z_t = u^t \begin{pmatrix} \cos(\theta + t\phi) \\ \sin(\theta + t\phi) \end{pmatrix} \quad (1.14)$$

meaning that the vector is rotated by the angle $t\phi$.

If $2\pi = k\phi$, then $\cos(k\phi + \theta) = \cos(\theta)$ and $\sin(k\phi + \theta) = \sin(\theta)$ so that the state vector returns to its initial direction after k steps, i.e. with respect to the direction of the state vector we have a periodic behavior with frequency $\omega = 1/k$ or

$$\omega = \frac{\phi}{2\pi} \quad (1.15)$$

If there is no k such that $2\pi = k\phi$ we have a quasi-periodic behavior since the angle of the state vector never returns exactly to its initial direction.

Eq. 1.14 also directly shows that the length of the state vector is decreasing (damped oscillation) if $u < 1$ and increasing if $u > 1$. The case of $u = 1$ is of special interest since the amplitude is conserved under the dynamics, i.e. the vector rotates with constant length and we have a periodic or quasi-periodic oscillation with constant amplitude.

The present considerations are valid for the linearized dynamics. This is correct if $\|z_0\| \ll 1$ and $u \leq 1$. However with $u > 1$ the length of z increases and after some time the linearization is not longer valid. However the expansion of z is confined by the nonlinearity introduced by the function $g(z)$ in eq. 1.4 so that the frequency is still roughly given by eq. 1.15. More details can be found by numerical simulations.

1.3.1. *The effect of the bias on the behavior of the system.* As we have seen above, when using a C matrix with the orthogonal matrix C of Eq. 1.11 if $u > 1$ we get an oscillatory behavior with frequency $\omega \approx \frac{\phi}{2\pi}$. This is true as long as the bias H has the value 0. However we will demonstrate now that the value of H has significant influence on the behavior. For instance when using $C_{11} = C_{22} = 1.2$ and $C_{12} = -C_{21} = -0.1$ corresponding to $\phi = 0.083$, $\omega \approx 0.01321$, and $\alpha = 1.2$ we observe in simulations that already when using H as small as $H_1 = H_2 = 0.02$ the oscillations are destroyed. Instead we have a convergence of both x_1 and x_2 to a fixed point as shown from Figure 2. This most sensitive dependence of the oscillations on the value of the H vector was already observed in the thesis by R. Haschke [18], see also [19]. Interestingly, they established that the oscillatory region as found from the theory is much larger than that obtained by computer simulations.

Our numerical studies corroborate these findings. We will see further below that, including the learning dynamics of H into the combined system of state and parameter dynamics, has a tremendous effect onto the the combined dynamics and introduces completely new phenomena into the state dynamics (frequency wandering, see below). The present findings path the way for these more complicated effects.

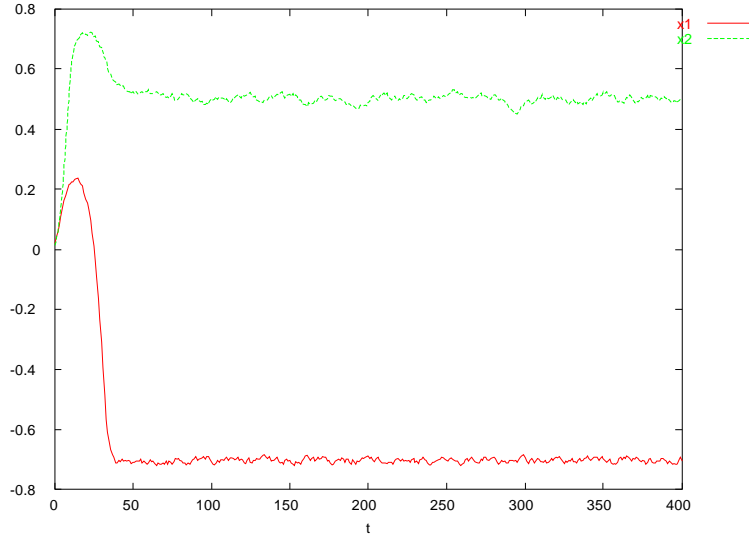


FIGURE 2. The influence of the bias h on the stability of a limit cycle in an $SO(2)$ network. The matrix C is of orthogonal structure with supercritical prefactor. Nevertheless the expected oscillations of x_1 and x_2 are suppressed due to the presence of the bias H which is as small as $h_1 = h_2 = 0.02$.

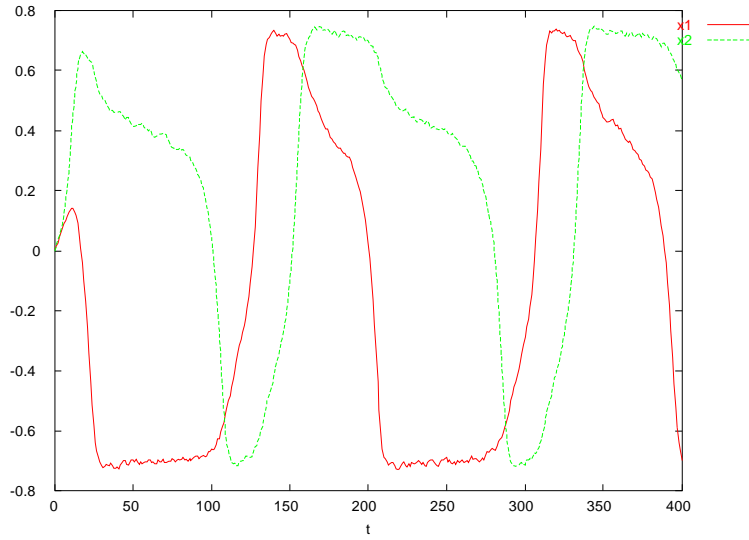


FIGURE 3. The influence of the bias H on the stability of a limit cycle in an $SO(2)$ network: With the bias H sufficiently small, $H_1 = H_2 = 0.01$, the state vector is now oscillating nonlinearly as expected from the structure of the C matrix.

2. Including learning I: The linear case

We start with the case of linear $K(x)$, i.e.

$$K(x) = Cx + H$$

and put $A = \mathbf{1}$, i.e. $\psi(x) = Cx$ so that the Jacobian matrix $L = C$ and our fundamental equation for the time loop (reconstruction) error reads

$$\xi = Cv$$

and we remember that

$$E = v^T v = Tr(vv^T)$$

2.1. Parameter dynamics. If the matrix C^{-1} exists we have the full reconstruction error as

$$E = Tr\left((CC^T)^{-1} D\right)$$

where $D = \xi\xi^T$ is the correlation matrix of the noise given by its matrix elements

$$D_{ij} = \xi_i \xi_j$$

In most cases we work with the averaged over the noise matrix. If the noise is uncorrelated over the channels and is of zero mean this means that $D = \text{diag}[\overline{\xi_1^2}, \overline{\xi_2^2}]$ or

$$D_{ij} = \delta_{ij} \overline{\xi_i^2}$$

which holds also for higher dimensions.

In the present two dimensional case the inversion of C can be done explicitly

$$C^{-1} = \frac{1}{\chi} \begin{pmatrix} C_{22} & -C_{12} \\ -C_{21} & C_{11} \end{pmatrix}$$

where the determinant is

$$\chi = C_{11}C_{22} - C_{12}C_{21}$$

so that

$$E = \chi^{-2} Tr \left(\begin{pmatrix} C_{22}^2 + C_{21}^2 & -C_{22}C_{12} - C_{21}C_{11} \\ -C_{22}C_{12} - C_{21}C_{11} & C_{12}^2 + C_{11}^2 \end{pmatrix} D \right)$$

If D is diagonal, then

$$E = \chi^{-2} \left((C_{22}^2 + C_{21}^2) D_{11} + (C_{12}^2 + C_{11}^2) D_{22} \right)$$

The corresponding parameter dynamics however is already rather complicated so that we consider special cases. However quite generally we may say that the sign of the determinant can not be changed by the parameter dynamics due to the singularity of E at $\chi = 0$. In robotic applications it is appropriate to always work with $\det \chi > 0$ since otherwise the controller produces actions with alternating sign in the time step. We will therefore in most cases also assume that $\det \chi > 0$ as a starting condition for the matrix C .

2.2. Uniform noise – Emergence of an $SO(2)$ dynamics. Assume uniform and orthogonal noise and $\overline{\xi_1^2} = \overline{\xi_2^2} = 1$ (by rescaling) so that $D = \mathbf{I}$, and using the gradient descent rule on E , the update rules of C simplify to (we add again a small damping term)

$$\Delta C = -\varepsilon \frac{\partial}{\partial C} \frac{\mu^2}{\chi^2} - \gamma C$$

where the derivative is a 2×2 matrix

$$\left(\frac{\partial}{\partial C} E \right)_{ij} = \frac{\partial}{\partial C_{ij}} E$$

and

$$\mu^2 = \text{Tr}(C^T C) = \sum C_{ij}^2$$

where μ is the Frobenius norm of the matrix C . More explicitly

$$\begin{aligned} \varepsilon^{-1} \Delta C_{11} &= \frac{\mu^2}{\chi^3} C_{22} - \frac{1}{\chi^2} C_{11} - \gamma C_{11} \\ \varepsilon^{-1} \Delta C_{22} &= \frac{\mu^2}{\chi^3} C_{11} - \frac{1}{\chi^2} C_{22} - \gamma C_{22} \end{aligned} \quad (2.1)$$

We get a similar scenario for the non-diagonal elements for which we have

$$\begin{aligned} \varepsilon^{-1} \Delta C_{12} &= -\frac{\mu^2}{\chi^3} C_{21} - \frac{1}{\chi^2} C_{12} - \gamma C_{12} \\ \varepsilon^{-1} \Delta C_{21} &= -\frac{\mu^2}{\chi^3} C_{12} - \frac{1}{\chi^2} C_{21} - \gamma C_{21} \end{aligned} \quad (2.2)$$

The argument is based on two quantities K , k defined as

$$K = \frac{1}{\chi^2} \left(\frac{\mu^2}{\chi} - 1 \right) \quad (2.3)$$

$$k = \frac{1}{\chi^2} \left(\frac{\mu^2}{\chi} + 1 \right) \quad (2.4)$$

where $k > 0$ and $K = 0$ if $\mu^2 = \chi$, i.e. the squared norm is equal to the determinant. Note that $\mu^2 = 2\chi$ if the matrix is of the orthogonal structure, cf. eq. 1.11.

Introducing $r = C_{11} + C_{22}$ and $s = C_{12} - C_{21}$ we find from the update rules that

$$\varepsilon^{-1} \Delta r = (K - \gamma) r \quad (2.5)$$

$$\varepsilon^{-1} \Delta s = (K - \gamma) s \quad (2.6)$$

Obviously both r and s increase until the condition of stationarity

$$\frac{\mu^2}{\chi} = \gamma \chi^2 + 1$$

is reached. In the orthogonal case this means convergence is reached if $u^4 = 1/\gamma$ due to eq. 1.11.

On the other hand, introducing $p = C_{11} - C_{22}$ and $q = C_{12} + C_{21}$ we have

$$\varepsilon^{-1}\Delta p = -(k + \gamma)p \quad (2.7)$$

$$\varepsilon^{-1}\Delta q = -(k + \gamma)q \quad (2.8)$$

which means unconditional decay until $p = q = 0$ is reached, i.e. until $C_{11} = C_{22}$ and $C_{12} = -C_{21}$ is reached. This proves the convergence to the orthogonal structure

$$C = uQ(\phi) \quad (2.9)$$

where

$$u = \|C\|$$

is increasing by the learning dynamics until $u = \sqrt[4]{1/\gamma}$ is reached and

$$\phi = \arctan \frac{C_{12}}{C_{11}}$$

converges to a fixed value depending on the initial conditions for the C matrix.

The explicit proof of the convergence of the controller matrix to the $SO(2)$ structure is a first result of our analysis of the parameter dynamics emerging by gradient descending the time loop error in a two-dimensional system. This can also be observed in practice and means that the robot starts to get into some kind of periodic motion. Depending on the frequency, this corresponds to regular motion patterns of the robot in space. Again this is only valid for the linearized dynamics. We will see below how the confinement effect introduced by the nonlinearity will bring also the explosion of u to stop after some time and produces much more complex behaviors than that of pure oscillations.

3. Including learning II: The nonlinear case

In the linear approximation, it was proved that, under certain conditions for the noise, the C matrix converges to an orthogonal matrix multiplied by a factor, see Eq. 1.11. This means for the state dynamics that the state vector will spiral outward once u exceeds the critical value 1. In the nonlinear system there are much more different dynamical regimes which can be realized from the combined dynamics, i.e. state + parameter dynamics. We will show in the following that there are different kinds of limit cycles, corresponding to both regular and irregular oscillations, and moreover that there are also metastable limit cycles. As in the one-dimensional case, the inclusion of additional parameters, the bias, deeply changes the nature of the combined dynamics and introduces completely new phenomena based on hysteresis effects. We will reestablish this situation in the realm of limit cycles below.

For the sake of simplicity, we use in the following the case of uniform, orthogonal noise so that the matrix D is used as $D = \mathbf{1}$. We start our considerations with the case that the bias H of the neurons is not subject to the parameter dynamics so that $H = 0$ throughout.

3.1. Combined dynamics without bias. Let us consider now the full dynamics given by Eq.1.4 where we use $g(z) = \tanh(z)$. The Jacobian matrix is given by Eq. 1.6, i.e.

$$L_{ij} = g'_i(z_i)C_{ij}$$

or

$$L = G'(z)C$$

in matrix notation. As already indicated, we do not use a bias (threshold) in the neuron function in the present section so that

$$z = Cx$$

The parameter dynamics for the matrix element of C is as usual

$$\Delta C_{ip} = -\epsilon \frac{\partial E}{\partial C_{ip}} \quad (3.1)$$

where ϵ is the learning rate. Despite of these simplifications, the compound system, consisting of the parameter and system dynamics, displays already rather complex behavior as is seen from computer simulations. The derivatives in eq. 3.1 can still be carried out explicitly so that there is no problem for the computer simulations. We will in the following consider some specific cases which are typical for the system.

One point of interest is that without bias the emerging dynamics depends largely on the initial conditions for the matrix C . In practical applications one therefore should include a small damping term in the update of C . Two different behaviors are observed depending on the initial C . Before studying special cases, we will give some general considerations on the nonlinear bias-free case.

3.1.1. General properties. One point of interest is that without bias the emerging dynamics depends largely on the initial conditions for the matrix C . This can be seen from the following considerations. We have

$$E = Tr \left((LL^T)^{-1} D \right)$$

and $E = Tr \left((LL^T)^{-1} \right)$ in the $D = \mathbf{1}$ case. Similar as in the linear case considered in Chapter 3 we use the singular value decomposition of L . In the nonlinear case the singular value decomposition (SVD) can be made more explicit in the case of our specific $L = G'C$ where G' is diagonal. We introduce the SVD of C as $C = SPU$ so that

$$L = G'SPU$$

With $D = \mathbf{1}$ this means that

$$\begin{aligned} Tr \left((LL^T)^{-1} \right) &= Tr \left((G')^{-2} (CC^T)^{-1} \right) = Tr \left((G')^{-2} SP^{-2}S^T \right) \\ &= \sum_i (P_{ii} \tilde{g}'_i(z))^{-2} \end{aligned} \quad (3.3)$$

where the $\tilde{g}'_i(z)$ are the diagonal elements of the matrix $SG'S^T$

$$\tilde{g}'_i(z) = (SG'(z)S^T)_{ii}$$

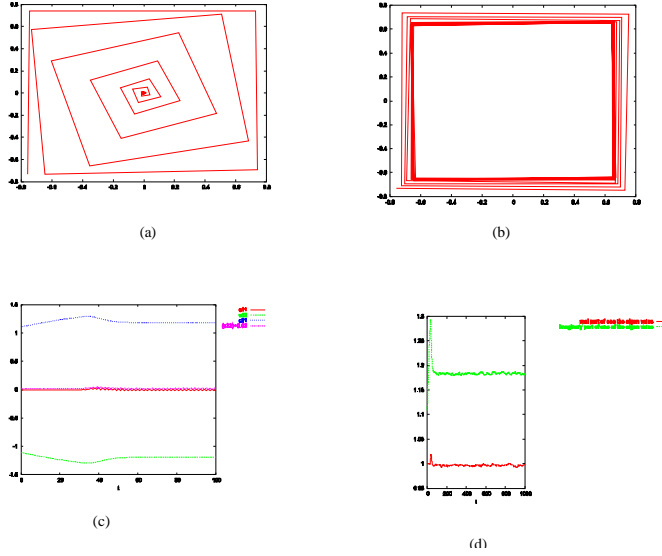


FIGURE 4. In (a) we see the trajectory of the state vector when starting with $x_1 = x_2 = 0$, learning rate = 0.005 , $c_{11} = 0$, $c_{12} = -1.1$, $c_{21} = 1.1$ and $c_{22} = 0$. It expands and reaches after some time a stable period-4 cycle. In (b) we observe that the trajectory of the state vector converges towards a stable limit cycle. (c) shows the convergence of the C matrix towards an orthogonal structure with appropriate prefactor. The convergence towards the stable limit cycle is also reflected by the behavior of the eigenvalues of the matrix C . In (d) we observe that after small number of steps, the imaginary part of the eigenvalue converges to 1.2 and the real value is almost 0.

Note that $\tilde{g}'_i(z)$ strongly depend on the parameters of S but, more importantly, they also depend on all matrix elements of C via the dependence of G' on $z = Cx$. Hence, the parameter dynamics now also has access to the parameters of U and will change the latter in most cases whereas U was conserved in the linear case, see Sec. 5.

As we will see, this effect is so strong that it destabilizes limit cycles based on an orthogonal matrix C . However there is one exception, namely that the values of g'_i are the same at all instants of time. This is the case if there is a period 4 oscillation, see Sec. 3.1.2. In most other cases we obtain irregular oscillations, see Sec. 3.1.3.

3.1.2. Regular oscillations. In order to get the regular oscillations we start with a matrix C close to the case of a period 4 oscillation. In particular, we use the initial values $x_1 = x_2 = 0$, $C_{11} = C_{22} = 0$, $C_{12} = -C_{21} = 1.1$, together with $\epsilon = 0.005$, and $D = \mathbf{1}$. In Fig.4, we observe that C preserves its orthogonal structure with the prefactor u converging after a few steps. Also x_1 and x_2 oscillate regularly (periodic behavior).

Using eq. 3.2, the nature of this specific limit cycle can be elucidated by noting that the orthogonal structure 1.11 with $\phi = \pi/2$ reads

$$C = u \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \quad (3.4)$$

Consider the vector $s = (1, 1)^T$ and

$$Qs = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, Q^2s = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, Q^3s = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, Q^4s = s$$

Assume that at some time t_0 we have $z_t = \alpha s$. Introduce for $t > t_0$

$$s_{t+1} = Qs_t$$

where $s_{t_0} = s$ and put tentatively $z_t = \alpha_t s_t$. Using $g(z) = \tanh z$ we have

$$g(\alpha s) = \begin{pmatrix} g(\alpha s_1) \\ g(\alpha s_2) \end{pmatrix} = g(\alpha) \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

for any vector s with $s_i = \pm 1$. For all $t > t_0$ we have using $z_{t+1} = \alpha_{t+1} s_{t+1}$ (according to our presupposition)

$$\begin{aligned} \alpha_{t+1} s_{t+1} &= z_{t+1} = Cg(z_t) = uQg(\alpha_t s_t) = ug(a_t)Qs_t \\ &= ug(a_t) s_{t+1} \end{aligned}$$

so that quite generally we obtain

$$\alpha_{t+1} = ug(a_t) \quad (3.5)$$

With $u > 1$ this converges towards a fixed point obtained by the solution of $\alpha = u \tanh(\alpha)$ where with u close to one the approximation $\alpha = \sqrt{3(u-1)}$ is valid, see Sec. 2. Obviously for any coupling strength $u > 1$ there is an α so that the dynamics of z boils down to

$$z_{t+1} = Qz_t \quad (3.6)$$

with

$$z_{t_0} = \alpha s$$

This result has an important consequence for the parameter dynamics of C . In fact, due to eq. 3.6 the components z_i of the state vector z are changing only their sign under the update dynamics so that $g'(z_t) = \text{const}$ and the components $g'_i = 1 - g_i^2$ of the vector function $g'(z)$ are of the same value, $g'_i = 1 - \tanh^2(\alpha_t)$. The Jacobian matrix for

$$x_{t+1} = g(Cx_t)$$

is

$$L = G'(z_t)C = u(1 - \tanh^2(\alpha_t))Q$$

and with orthogonal and uniform noise ($D = \overline{\xi^2}$) we obtain the time loop error as

$$E = \frac{\overline{\xi^2}}{u^2(1 - \tanh^2(\alpha_t))^2} = \frac{\overline{\xi^2}}{\Lambda^2}$$

which is independent on the matrix elements of Q . The dependence on the parameter u of C is given via eq. 3.5 so that using eq. 3.5

$$\begin{aligned}\Delta u &= \varepsilon \frac{E}{\Lambda} \frac{\partial u g'(\alpha_t)}{\partial u} = \varepsilon \frac{E}{\Lambda} \left(g' + u g''(\alpha_t) \frac{\partial \alpha_t}{\partial u} \right) \\ &= \varepsilon \frac{E g'}{\Lambda} (1 - 2u g(\alpha_t) g(\alpha_{t-1}))\end{aligned}$$

With α taken at the respective fixed point (slow learning rate $0 < \varepsilon \ll 1$) we find using the fixed point relation $\alpha = u g(\alpha)$ that u is stationary where $1 = 2\alpha g(\alpha)$ from which u is obtained as $u = a/g(\alpha)$. This repeats more or less the arguments of Chapter 4, from where we know that $u = 1.1911$.

In concluding one may say that, once the matrix C is of the form eq. 3.4, the learning leaves the matrix Q invariant (at least is stationary with respect to Q) and the feed-back strength factor u converges towards a value slightly above the bifurcation point where the Sacker-Neimark bifurcation takes place. The fact the specific form of Q is indeed a stable fixed point of the learning dynamics is seen from the computer simulations where the four cycle is found to be an attractor with seemingly a large basin of attraction.

In general, we may say that, without bias, the classical Sacker-Neimark bifurcation is emerging in the combined dynamics in the case of period 4 oscillations, i.e. if the initial conditions of C are appropriate. In most other cases we get more or less irregular oscillations, see the following.

3.1.3. Irregular oscillations. We have discussed that the combined dynamics of state and controller parameters will end up in a stable 4-cycle when starting close to the special orthogonal structure 3.4. If one is sufficiently far away from this one gets the phenomenon of irregular oscillations which are of different nature in that they are not supported by the special form of the C matrix but emerge from a pure interplay of the state and the parameter dynamics in a similar way to the effects observed in the one-dimensional system with bias dynamics. Using for instance the values $x_1 = x_2 = 0$, $\epsilon = 0.005$, $D = \mathbf{1}$ and

$$C = \begin{pmatrix} 1 & 0.2 \\ 0.05 & 0.7 \end{pmatrix}$$

as initial condition, one observes from Fig.5 that the matrix C does in fact not converge to any orthogonal structure. Instead the non-diagonal matrix elements are seen to oscillate coherently. The system dynamics is seen to exhibit irregular oscillations which are very sensitive to the noise. It is to be noted that the oscillations result from the combination of both the state and the parameter dynamics. As a consequence the frequency of the oscillations depends on the value of ε and the strength of the noise.

The oscillations can be understood by the results of the one-dimensional systems with bias, where we have seen, that due to the nonlinearity, the bias is driven such that the nonlinearity is decreasing. As a consequence, the bias is always moving such that the fixed point is destabilized more and more until the state jumps to the other fixed point of the bistable system. Together with the hysteresis property this leads to an oscillation of the state dynamics. We observe here a similar scenario, where each neuron uses the output of its neighbor as a secondary input which acts as a kind of bias.

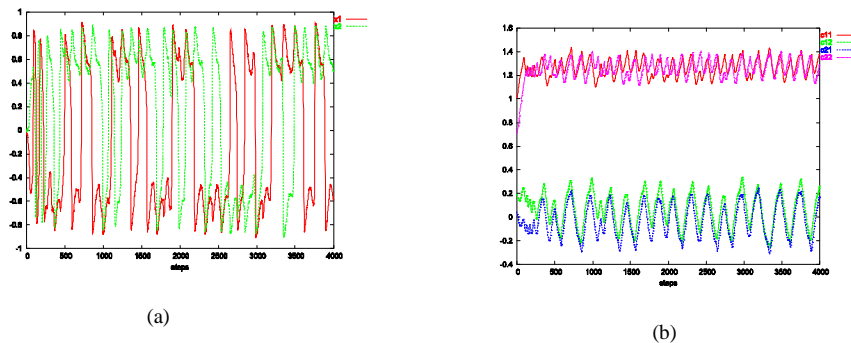


FIGURE 5. From (a), we observe that x_1 and x_2 exhibit irregular oscillations. The initial conditions which are used $x_1 = x_2 = 0$, learning rate = 0.005 and $c_{11} = 1$, $c_{12} = 0.2$, $c_{21} = 0.05$ and $c_{22} = 0.7$. In (b), we observe that the c matrix does not converge to the orthogonal structure, instead the non diagonal elements of c oscillate with about the same value.

Note that we may consider the parameter H in the one-dimensional case also as the synaptic weight of a secondary input of strength 1 (called a tonic input in neuroscience) into the neuron.

This combined dynamics phenomenon is of some interest for the practical applications since it makes the dynamics particularly sensitive to the effect of the noise. Please remember that noise means in our case the modeling error so that the system may be said to be particularly sensitive to the relation of the behavior to the environment which is measured by the modeling error.

3.1.4. *Switching oscillators.* Besides the irregular oscillatory behavior we still observe another dynamical regime which we call the switching oscillatory regime. This one is obtained for example when starting from an orthogonal C with $\phi \neq \frac{\pi}{2}$, in the simulations we used

$$C = \begin{pmatrix} 1.1 & -0.1 \\ 0.1 & 1.1 \end{pmatrix}$$

The regime is qualified by the fact that one neuron is oscillating, while the other one is fluctuating around a fixed point. Then in irregular time intervals the two neurons change their role, see Fig. 6.

As seen from the figures, the phenomenon is based on the fact that the diagonal element (which is regulating the feed-back strength in the loop) of the fixed state neuron is a little higher, which is seemingly sufficient in order to keep the neuron at the fixed point. When the oscillator is switched, the role of the diagonal elements are switching as well so that the other neuron is kept fluctuating around the fixed point.

The oscillations are again of a similar nature as that in the one-dimensional case. In the present situation the oscillating neuron uses the more or less constant output of the other neuron as a kind of tonic input, i.e. the oscillations of the non-diagonal element, coupling the oscillating to the fixed state

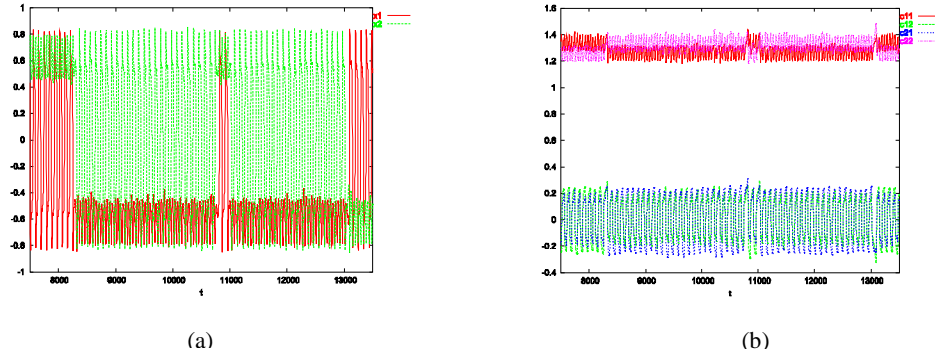


FIGURE 6. In (a), we observe that at different time steps each of x_1 and x_2 makes a transition between two different regims namely oscillation about one of the fixed points and the other one is the oscillation between the fixed points. From (b), we observe that the diagonal elements allmost have the same value and the nondiagonal elments oscillate about 0.

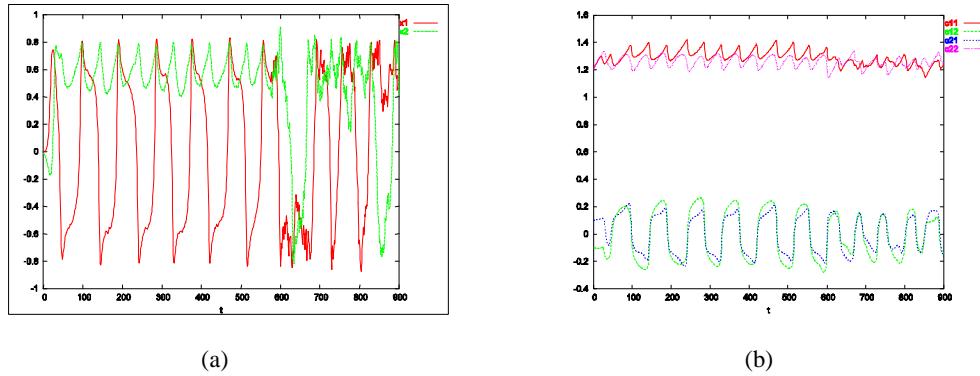


FIGURE 7. As we observe from (a), in the beginning x_1 oscillates and x_2 stays at the positive fixed point. However by increasing the noise strenght at step 590 , x_2 leaves the fixed point and start to oscillate. From (b), we observe that the diagonal elementes of C oscillate about 1.2 , while the nondiagonal ones oscillate about 0 and have the same sign.

neuron, correspond to the oscillations of the bias in the one-dimensional case.

The effect of noise on the switching oscillators: The attractor of the system of switching oscillators has a small basin of attraction because in the simulation we observe that a little increase of the noise strength makes the system leave this attractor. A typical run is depicted in Figure 7.

4. Including the bias dynamics

In the one-dimensional system we have seen, that the inclusion of the bias dynamics introduces hysteresis effects which change the fixed point system into a limit cycle. Similar effects are to be expected in the two-dimensional case. Let us consider the state z as

$$z_i = \sum_{s=1}^2 C_{is} x_s + H_i \quad (4.1)$$

the bias H_i being updated now also by gradient descending the error E , i.e.

$$\Delta H_i = -\varepsilon \frac{\partial E}{\partial H_i}$$

The introduction of the bias enriches the combined dynamics (system plus parameter dynamics) tremendously. In order to get some systematics into the wealth of phenomena we at first keep C fixed and study the effect of updating H only and then we will study the behavior of the system by updating both C and H

4.1. Bias dynamics in an $SO(2)$ network. We have seen in Sec.1.3.1 that, depending on the value of H , the system either exhibits nonlinear oscillations of fixed frequency or it converges to a fixed point. Including the dynamics of H results in an interesting behavior, namely that there coexist two limit cycle attractors with different frequencies, with basins of attraction defined in terms of the initial values of H and x . In the simulation, we use an orthogonal structure

$$C = u \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix} \quad (4.2)$$

We choose ϕ so that the linear system would correspond to a frequency of ≈ 0.01 Hz. The nonlinearity is seen to produce two different behaviors of the system depending on the initial conditions on H (small values) or x . In the first we get a regular oscillation with a frequency ≈ 0.02 Hz, i.e. much higher than that dictated by the matrix C (≈ 0.01), which would be realized with $H = 0$ fixed. This one is realized however in the second case.

The different nature of these two scenarios is understood by looking at the phase shift between the state variables. In the high frequency case the phase shift is in accordance with that given by C as seen from Figure 8 (a), whereas in the second case the phase shift is the opposite one as seen from (b) of the same Figure. This demonstrates the relevance of the phase shift for the frequency of the oscillatory behavior. The phase shift in an essential way influences the interplay between the state and the H dynamics, therefore it is responsible for the different frequencies of the limit cycle attractors.

The decisive influence of the phase shift on the behavior of the system will be seen to play an essential role also in the more complicated cases considered below. Thus this seems to be an essential ingredient of the present approach. In particular, one may use the phase shift in order to switch between these two limit cycle attractors. In other words this opens the possibility of the robot to react in a definite way to environmental influences. In fact, if the system is in one of its attractors, a simple shift in phase between

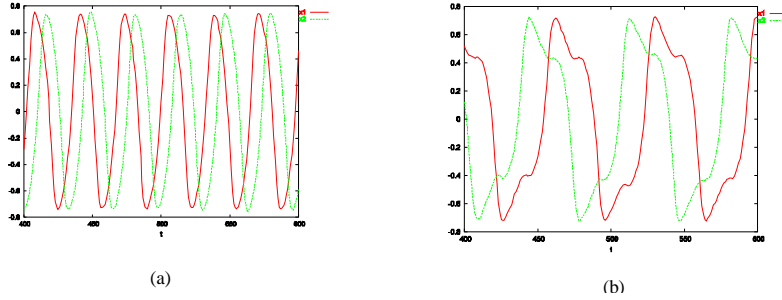


FIGURE 8. (a) shows that x_1 and x_2 oscillate with higher frequency than that calculated from the used matrix c . The phase shift between x_1 and x_2 is in accordance with that dictated by the used c . The graph is obtained under initial condition $x_1 = x_2 = 0$, $h_1 = h_2 = 0$, learning rate = 0.005, regularization factor = 0.1, $c_{11} = 1.2$, $c_{12} = -0.1$, $c_{21} = 0.1$ and $c_{22} = 1.2$. However (b) shows x_1 and x_2 oscillate with frequency ≈ 0.01 that calculated from the used matrix c . The phase shift between x_1 and x_2 is the opposite of that dictated by the used c . The used initial values are $x_1 = x_2 = 0$, $h_1 = 0.01$, $h_2 = 0.0$, learning rate = 0.005, regularization factor = 0.1, $c_{11} = 1.2$, $c_{12} = -0.1$, $c_{21} = 0.1$ and $c_{22} = 1.2$.

the sensor values x_1 and x_2 may lead to a switching of the frequencies making the robot to try a different dynamical regime.

4.2. Full parameter dynamics. Frequency wandering. We are now going to study the combined dynamics of state and parameter variables, i.e. that of x , C , and H which takes place in a 8-dimensional space. The most interesting effect when including the bias together with the C dynamics was discovered from computer simulations. At the phenomenological level the effect can be considered as if the system wanders through its different frequencies and we call it therefore the frequency wandering effect.

In order to understand how this effect is produced we consider a typical run of a numerical simulation as given in the Fig.9. The first observation is that, most over the time, the matrix C essentially realizes an orthogonal structure like eq.4.2. However the non-diagonal elements of C and hence the angle ϕ are not constant but display also a periodic behavior on a much larger time scale. Formally, this is the reason for the frequency wandering since the angle ϕ of the orthogonal matrix is connected with the frequency. However, we also see that there is no unique relation between ϕ and the frequency of the state oscillations. Instead there are again two possible frequencies related with each angle ϕ .

However we have already seen in Sec. 4.1, where the C matrix was fixed and only the H dynamics included, that in the nonlinear system this relation is valid only if the phase relation between the states is in agreement with the one given by the C matrix, see Sec. 4.1. In the present case of the full dynamics in the x, C, H space the interplay between the variables is of still more complicated nature.

Let us start the discussion with Fig. 9 (a) where we see in the time 1 : 1700 one half-period of the C parameter cycle. In this region the non-diagonal elements at first increase from zero to some maximum close to 1 (corresponding to an increase of the angle ϕ) and then decrease again to zero. As discussed above, the frequency of the state dynamics should roughly be given by $\omega = \phi/2\pi$. This relation between ϕ and the frequency of the state oscillations are indeed observed in the time interval where the angle ϕ decreases, i.e. time steps 940:1700.

However this is not the case during the first half of this time region (1 : 940) where the angle ϕ increases but the frequency decreases, see (c) Fig.9. This effect is understood if we look at the phase relations between the state variables x_1 and x_2 . In the region 1 : 1700 the angle ϕ is always negative so that the phase relation between x_1 and x_2 is such that x_1 precedes x_2 . However this is only the case in the second part of the interval considered, i.e. 900 : 1700 where one could say that the dynamics is as it would be without the additional H dynamics, see (a) and (d) of Fig.9.

However in the first interval the phase relation is inverted what can only be understood by stipulating an interplay between the state and the bias dynamics. Apparently the H dynamics counteracts the state dynamics generated by the $SO(2)$ like structure of the controller matrix C . The interplay is such that the frequency decreases although the angle of rotation ϕ steadily increases. A further interplay between state and parameter dynamics is also obtained in the interval 900 : 1700 seen by the steady decrease of the frequency during the time evolution of the system. This effect could not be obtained from the H free dynamics studied in Sec. 3.1 since there is no such systematic change of the frequency once the orthogonal structure is reached.

4.2.1. Stability of the frequency wandering effect. We have found the above behavior for the special initial condition for C , cf. Eq. 4.2. In order to discuss the stability of the phenomenon we used different matrices C as initial conditions. In all cases we observed the same phenomenon so that we may say that it is largely independent on the choice of the initial C .

However the number of steps after which the system exhibits this behavior depends on the initial C . In Fig10 two different initial C are used in order to demonstrate this.

4.2.2. Discussion. The inclusion of the bias dynamics obviously has dramatic effects on the combined dynamics and thus on the behavior of the system. There is an interesting parallel to the one-dimensional case. In the latter we have seen, that without H dynamics the system converges towards a fixed point. The role of the H dynamics was seen to produce a smooth switching between the fixed points so that altogether a periodic motion resulted.

In the two-dimensional case considered here without bias dynamics the system in the majority of cases converges towards a stable limit cycle behavior, see Sec. 3.1 the frequency of which essentially depends on the initial conditions. The role of the bias dynamics was seen to change in a periodic way the frequency of the limit cycle so that we can say that the system self-organizes a sweep through its limit cycles and in this way explores its dynamic possibilities.

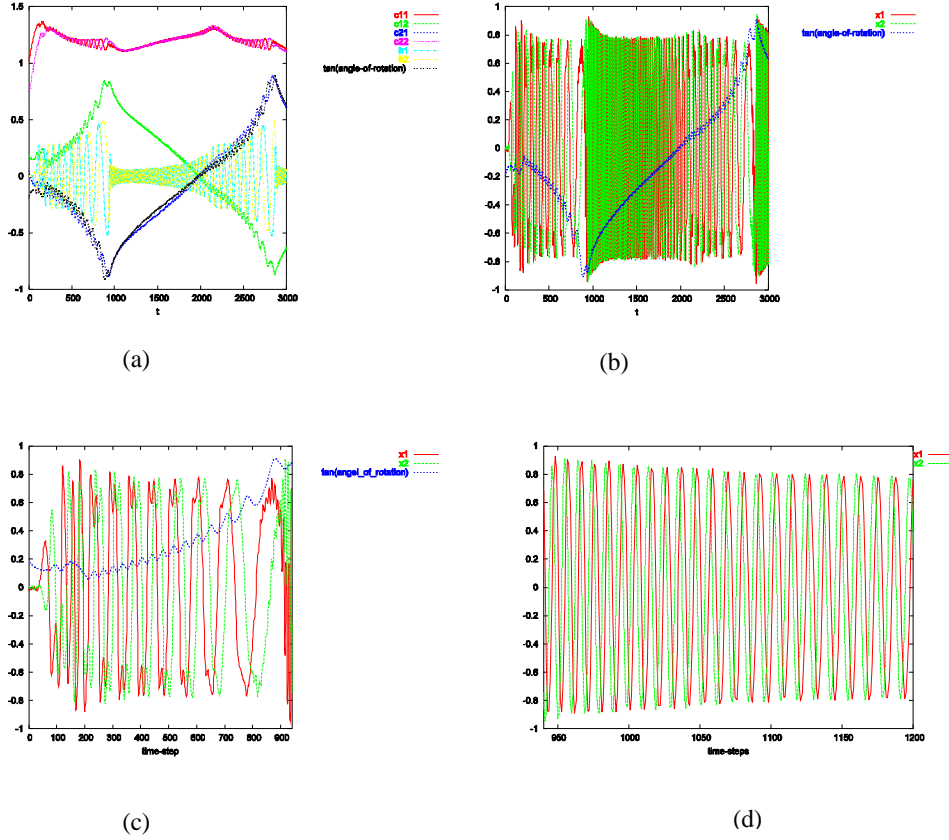


FIGURE 9. In (a), we observe that the matrix c converges to an orthogonal structure in which its nondiagonal elements change periodically and h exhibits oscillations with periodically changing amplitude. The amplitude of h is large in the region of the "wrong" phase relation between x_1 and x_2 . After this the matrix c generates high frequency oscillations so that the h dynamics can not follow. The graph is obtained under initial condition $x_1 = x_2 = h_1 = h_2 = 0$, learning rate = 0.005, regularization factor = 0.1, $c_{11} = 1$, $c_{12} = 0.2$, $c_{21} = 0.05$ and $c_{22} = 0.7$. In (b), we observe that x_1 , x_2 exhibit oscillations with a non constant frequency. The changing of this frequency is influenced by h . In (c), we observe that x_1 precedes x_2 in the period 0:900- which corresponds to a anticlockwise rotation (positive angle increase). However this is the opposite of what is expected from the sign of angle of rotation (blue dotted line) which is negative. This shows that the h dynamics must have an essential influence on the rotation. As a result we observe a decreasing frequency in the period 0:900. In (d) x_2 precedes x_1 in this period as expected from the sign of the angle of rotation. Hence the dynamics is governed essentially by the matrix c so that the h dynamics has no essential effect.

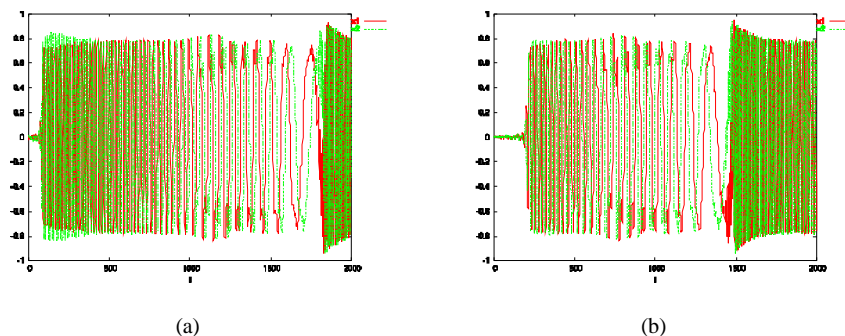


FIGURE 10. In (a) we observe that x_1, x_2 start to oscillate after 100 steps (the initial condition was $x_1 = x_2 = h_1 = h_2 = 0$, learning rate = 0.005, regularization factor = 0.1, $c_{11} = 0.5$, $c_{12} = -0.1$, $c_{21} = 0.7$ and $c_{22} = 1$). However (b) shows that x_1, x_2 start to oscillate after 200 steps (the initial condition was, $x_1 = x_2 = h_1 = h_2 = 0$, learning rate = 0.005, regularization factor = 0.1, $c_{11} = 0.1$, $c_{12} = 0$, $c_{21} = 0$ and $c_{22} = 0.2$).

5. Correlated noise

The influence of the noise (modeling error) on the dynamics is of much importance for the practical applications. For a discussion we consider in the following the role of correlations of the noise in the different channels. We use for this purpose again uniform noise, i.e. ξ_i , $i = 1, 2$ is randomly chosen from the interval $[-0.1, 0.1]$. We consider at first the case of no bias so that the dynamics is entirely determined by the matrix C .

5.1. Parameter dynamics without bias.

5.1.1. *Correlated noise.* As the first case we consider correlated noise obtained by choosing the ξ_i randomly, the correlations being introduced by inverting the sign of one of the two noise components so that the signs are the same. Studying the above system using correlated noise shows that both of x_1 and x_2 will also be correlated as shown from Figure 11 (a).

Although the noise is rather weak, these correlations are quite strong so that the states are synchronized, as is seen from Fig. 11. The state dynamics is again coupled to the parameter dynamics by the combined dynamics effect.

5.1.2. *Anticorrelated noise.* If we choose again the noise randomly but anticorrelated the signs of the noise variables, we find the behavior depicted in Fig.?? which clearly demonstrates that x_1 and x_2 will be anticorrelated as well. Comparing (a) with (b) in that Figure we recognize the combined dynamics effect again. Moreover by comparing Fig.??(b) and 11(b), we see that the correlations in the noise also strongly influence the interplay between the parameter and the system dynamics. While in the latter case (anticorrelated noise) the diagonal elements are correlated with the diagonal ones they appear anticorrelated in the former case.

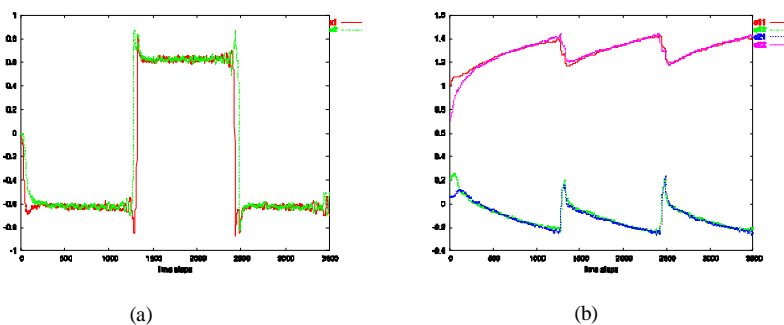


FIGURE 11. (a) shows the correlation of x_1, x_2 under correlated noise, (b) the corresponding C matrix. We observe from (b) that the nondiagonal elements of C will be anticorrelated with the diagonal ones.

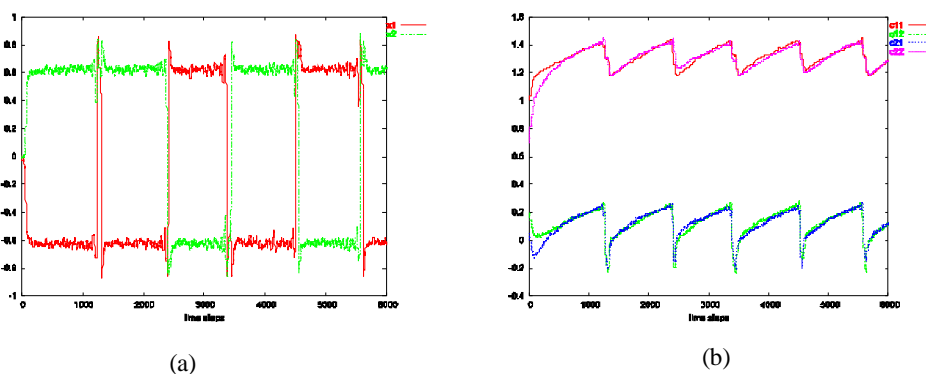


FIGURE 12. (a) shows anticorrelation of x_1, x_2 under anti-correlated noise. (b) shows the corresponding c matrix. We observe from (b) that the non diagonal elements of c will be correlated with the diagonal ones.

Discussion: As already mentioned the dynamics is without bias. However as seen in the Figures there are oscillations of the state variables x_1 and x_2 which are different from the regular oscillations obtained with an orthogonal structure. Instead, similarly as in Sec. 3.1.3, the oscillations of the state result from the oscillations of the non-diagonal elements of C . However the correlations in the noise make the non-diagonal elements nearly identical. The difference between the correlated and the anticorrelated case is in the phase relation between the diagonal and the non-diagonal elements of C .

In any case we see that the correlations in the noise have decisive influence on the dynamics of the combined dynamical system. This is a desirable effect from the point of view of robotics since it shows that the system deploys a rich dynamics which however is still related to the environment, in particular via the modeling error of the world model, so that we may say

that the system actively explores itself and its environment in order to improve the world model, see the related discussion with the frequency effect in Chapter 4 on one-dimensional systems.

In the following section the effect of the noise on the behavior of the system in which both the C and H parameters are learned. First we study the behavior of the system using anticorrelated noise and second the correlated one.

5.2. Parameter dynamics with bias. Including the bias dynamics was shown to enrich much the sensitivity of the system because in the simulation different behaviors are observed depending essentially on the actual values of the state, the parameters of the system and the noise. In order to introduce the phenomena we consider a simulation run as depicted in Fig. 13. One of the behaviors that the system can exhibit that is observed in the period [60:38000]. In this period the C matrix has real eigenvalues - as indicated from Fig. 13(a)-but x_1 and x_2 oscillate regularly, therefore this oscillations can only be a result of the interplay between the state and the H dynamics. In this period we observe also that x_1 and x_2 are anticorrelated, see Fig13(b). This anticorrelation between x_1 and x_2 can be understood as an effect of the noise only, because in 3.1 there was a phase shift between x_1 and x_2 .

Another behavior that the system can exhibit that is observed in the period [38000:40000]. In this period we observe that the C matrix has another structure which has complex eigenvalues. Also x_1 and x_2 oscillate with continuously decreasing frequency and phase shift is not as dictated by the actual C matrix.

The last behavior is that observed in the period [40000:40500]. In this period we observe that the matrix C has almost the same value as in the previous period, however x_1 and x_2 oscillate with high frequency and phase shift as dictated by the current C matrix.

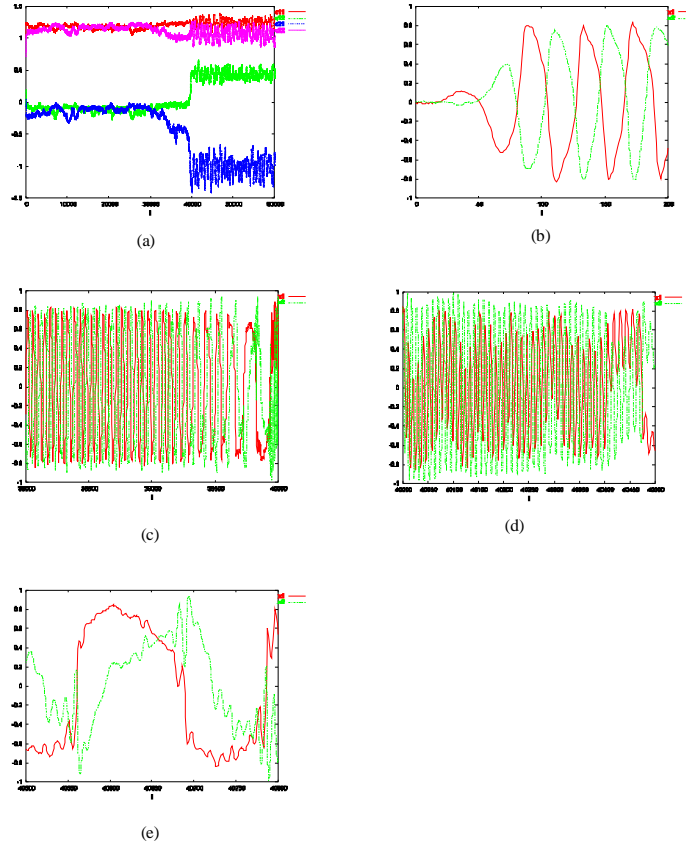


FIGURE 13. (a) shows that depending on the actual state of the system, the c matrix may make a transition from a structure that does not support the oscillation of the state variables to that one support the oscillations of the state variables. From (b) we observe that after number of steps- depending on the actual condition- x_1 and x_2 oscillate regularly whereas the c matrix in this period has real eigenvalues. Also we observe in this graph that x_1 and x_2 are anticorrelated. (c) shows that in the period 38000:39800 we observe that x_1 and x_2 oscillate with continuously decreasing frequency and the phase shift is the opposite that dictated by the actual c matrix. (d) shows the period 40000:40470, in this period we observe that x_1 and x_2 oscillate with high frequency and the phase shift is that dictated by the actual c matrix. The initial condition was $x_1 = x_2 = 0$, $h_1 = h_2 = 0.0$, learning rate = 0.005, regularization factor = 0.1, $c_{11} = 1$, $c_{12} = 0.2$, $c_{21} = 0.05$ and $c_{22} = 0.7$.

Discussion:

We see that in Sec. 4.2 that there is a systematic and periodicity change of the C , H and of the frequency of oscillations of the state variables, however including the anticorrelated or correlated noise influences this systematics essentially. In fact, as seen above the actual state of the system determines

whether it goes into the anticorrelation or correlation regime or into high frequency regime with correct (with respect to C) phase shift or into the low frequency regime with wrong phase shift.

6. Application - The barrel robot

The frequency wandering effect can also nicely be seen in applications. By way of example we consider a simplified version of a spherical robot, the rolling barrel as constructed by Georg Martius in ODE. Inside the barrel (cylinder) we have two orthogonal axis with a mass on each driven by a linear motor. The two controller outputs set the position of the masses and the sensor values are the angles of the axis with the z -axis of the space coordinate system (inclination angles). The world model has to predict the new sensor values given the controls y and we use a linear model as before, i.e.

$$x_{t+1} = Ay_t + \xi_{t+1}$$

the matrix A being trained on-line by gradient depending the error function

$$U = \|x_{t+1} - Ay_t\|^2 \tag{6.1}$$

The system is started in a *tabula rasa* condition for both the world model and the controller, meaning that both of them are initialized as the unit matrix with small random non-diagonal matrix elements. Learning runs parallel for both model and controller as described above. The effect of the combined state and parameter dynamics is seen in the video <http://robot.informatik.uni-leipzig.de/Videos/Barrel/2006/barrel.avi>.

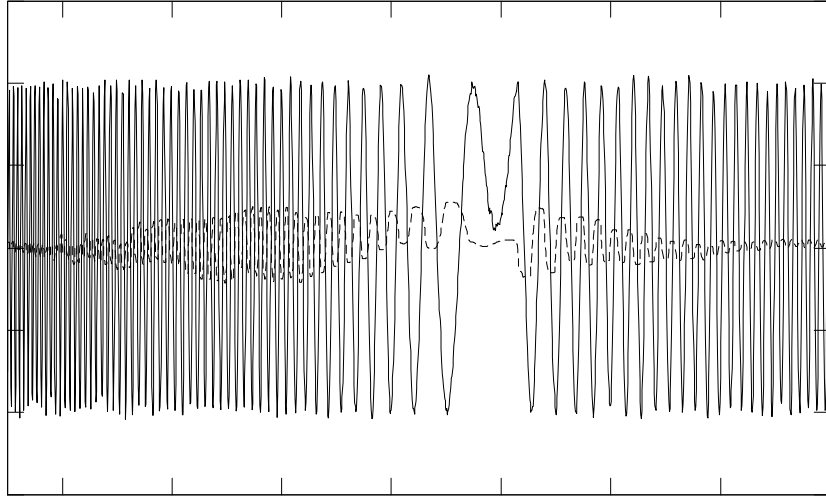


FIGURE 14. The sensor value $x_1(t)$ in the time interval 2500 until 4000 together with the values of H_1 (dashed line). The region covers the event where the robot from a very high speed mode slows down actively and then inverts its velocity and rolls backward with increasing velocity. The sensor measures the angle of one of the internal axis with the space coordinate system so that the frequency of the sensor values directly measures the rolling velocity of the barrel.

The barrel is seen after an initial period of some indecision to start a rolling motion with constantly increasing velocity¹. Then after some time it starts decreasing the velocity and eventually stops and rolls backwards in the same scenario. This repeats more or less periodically. On the phenomenological level one might say that the robot is able to both accelerate and brake or that it actively investigates its space of velocities.

¹I thank Georg Martius very much for the computer simulations.

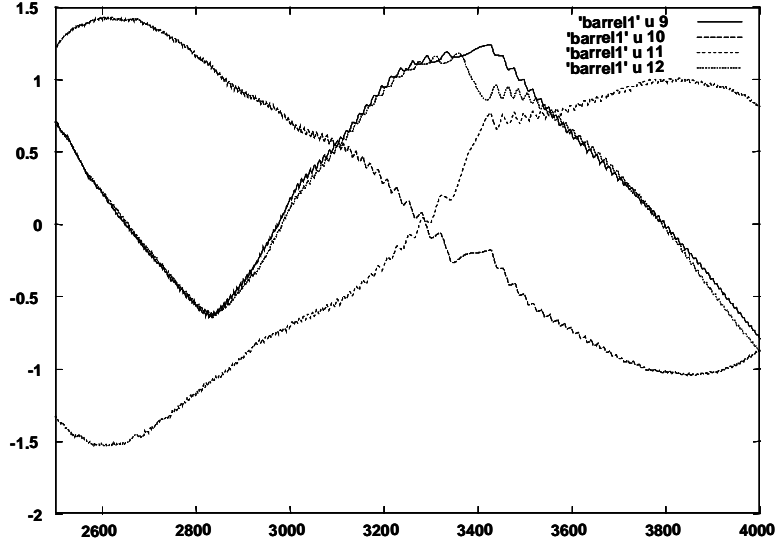


FIGURE 15. The time course of the controller matrix C in the time interval of slowing down, stopping and accelerating again with inverted velocity. The diagonal matrix elements are $C_{11} \approx C_{22}$ whereas $C_{12} \approx -C_{21}$ so that we have essentially an orthogonal matrix with a rotation angle continuously changing over time over the full range $-\pi < \phi < \pi$, negative values of ϕ corresponding to the deceleration phase. This is the frequency wandering effect with an embodied agent.

When looking at the time evolution of the C values, one recovers the effect observed in Sec. 4.2 with the hypothetical system, i.e. a system without physical embodiment, whereas the barrel is a massive object with quite a complicated physics due to the shifting masses driving it. Although qualitatively similar to the agent without embodiment, there are also clear differences, in particular in the change of the C matrix, which, with the specific matrix structure realized in the simulations, may be qualified by the angle

$$\phi = \arctan \frac{C_{12}}{C_{11}}$$

in a good approximation. In the non embodied case the angle changed between $-\pi/2 < \phi < \pi/2$ whereas now it is seen to vary through the entire range of ϕ values, i.e. using that the C matrix is given by $\phi \bmod 2\pi$ we have ϕ values as $-\pi < \phi < \pi$.

In particular because of its large mass there are strong inertia effects, which have to be dealt with by a controller that wants to roll the barrel with differing velocities in a closed loop control paradigm based on the sensor values which are the angles of the internal axes. The emerging strategy of the controller obviously is to rotate the angle ϕ in order to accelerate and decelerate.

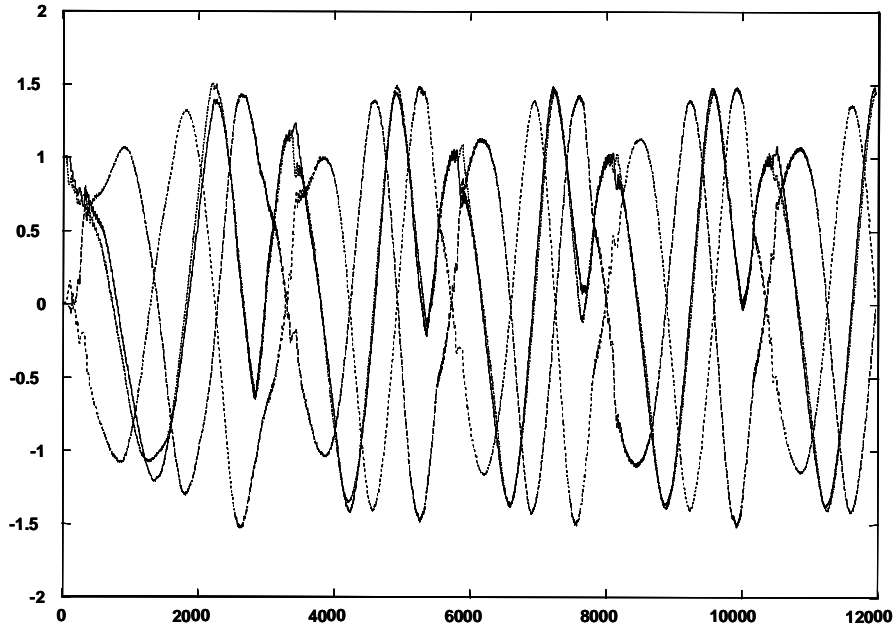


FIGURE 16. The behavior of the C matrix over a very long interval of time. Most of the time we have the orthogonal structure, where C is essentially a rotation matrix, $C_{11} \approx C_{22}$ whereas $C_{12} \approx -C_{21}$ so that rotation angle ϕ periodically wanders through the full interval $-\pi < \phi < \pi$, negative values of ϕ corresponding to the deceleration phases.

CHAPTER 6

Three-dimensional systems

In Chapter 5 we investigated the two-dimensional system and we showed that the system exhibits interesting properties. In particular, the system was shown to actively investigate its space of possibilities. For instance, it is able under certain conditions to sweep through its range of accessible frequencies. In the application with the barrel bot this lead to the emergence of a controlled acceleration and deceleration of the rolling mode. In other applications demonstrated by the videos on <http://robot.informatik.uni-leipzig.de/research/videos/> the robots are seen to try different modes of motions to explore itself or to feel its body.

The investigations in Chapter 5 have largely been centered on the peculiarities of the two-dimensional system, in particular the structure of the orthogonal matrix. In the present chapter we try to investigate the three-dimensional system in order to see which of those properties survives in the new context. We use the same paradigm for the gradient flow of the parameters of the controller, i.e. we study a system with dynamics

$$x_{t+1} = Ay_t + \xi_{t+1}$$

where the noise ξ (model error) has mean zero but may be of any kind. The 3×3 matrix A measures the response of the sensors to the actions $y \in R^3$ of the controller where

$$y_t = K(x_t) = g(Cx_t + H)$$

with C a 3×3 matrix and g a vector function, defined by $g_i = g(z_i)$ with $z = Cx + H$. We write $AK(x) = \psi(x)$ as before, find the shift vector v by the solution of

$$\xi = L(x)v$$

where

$$L_{ij} = \sum_{k=1}^3 A_{ik} g'(z_k) C_{kj}$$

which in matrix notation reads $L = AG'(z)C$ with the diagonal matrix $G'(z) = \text{diag}[z_1, z_2, z_3]$. The time loop error reads as before

$$E = v^T v = \xi^T (LL^T)^{-1} \xi = \text{Tr} \left((LL^T)^{-1} D \right)$$

The inversion of the matrix LL^T can still be done explicitly so that one gets an explicit expression for E in terms of the parameters of the matrix C and H . The updates

$$\Delta C = -\varepsilon \frac{\partial}{\partial C} E, \quad \Delta H = -\varepsilon \frac{\partial}{\partial H} E$$

have been obtained in the computer program by numerical differentiation.

1. General observations

Let us consider for the moment the case of uniform and orthogonal noise so that D is proportional to the unit matrix. As explained in Chapter 5 if we have an orthogonal structure for C , i.e.

$$C = u Q(\alpha, \beta, \gamma)$$

where $Q^T = Q^{-1}$ is an $O(3)$ matrix depending on three parameters, the time loop error is in the linear case ($g(z) = z$)

$$E = \frac{1}{u^2} \text{Tr}(D)$$

so that E does not depend on the parameters of Q . As we have seen in 2-d, see Chapter 5, the nonlinearities destroy this independence so that the parameters change under the gradient dynamics.

Many of the phenomena present already in two dimensions survive also in the three-dimensional case, but we can give here only a narrow outlook on these phenomena. We concentrate on the case of the regular oscillations and the frequency wandering effect.

2. Regular oscillations

Suppressing the bias learning, we observed in Chapter 5, that under certain condition, the system converge to a stable cycle of period 4. The reason was that when applying a matrix C which is essentially a rotation about an angle $\phi = \pm\pi/2$ on a vector with components of equal absolute value, the effect of C is just a sign flip of certain components of the vector. This is of course also possible in the d -dimensional cases. In fact this means that one has to find an orthogonal transformation Q so that $Q^n = \mathbf{I}$ and

$$s^T Q s = d$$

where $|s_i| = 1 \forall i = 1, \dots, d$. Obviously the number of possibilities and the value of n increases with increasing dimension so that with increasing dimension, there are more and more possibilities for our system to support an n -cycle which is stable under the combined dynamics. In the simulation we observe that in the 3-dimensional system, the learning rules drive the parameters to regions of different cycles of the system, indeed. The cycle attracted to depends on the initial conditions of the parameters and the state and also on the noise. Moreover we observe, that under stronger noise the system may also switch between different cycles. In Figure 1, we see two different cycles exhibited by the system in two different periods of time, (c) displaying the C value during these two period.

A preliminary result of our simulation is that seemingly cycles of lower periodicity are more stable than the higher ones.

3. The frequency wandering effect

As in the two-dimensional case the stable limit cycle behavior is obtained if the dynamics of the bias is switched off. When including the latter we get again the frequency wandering effect, however in a modified form. A typical run of a numerical simulation is given in Figure 1(a). In this graph, four distinct regions can be distinguished. In the first region 1500:1800,

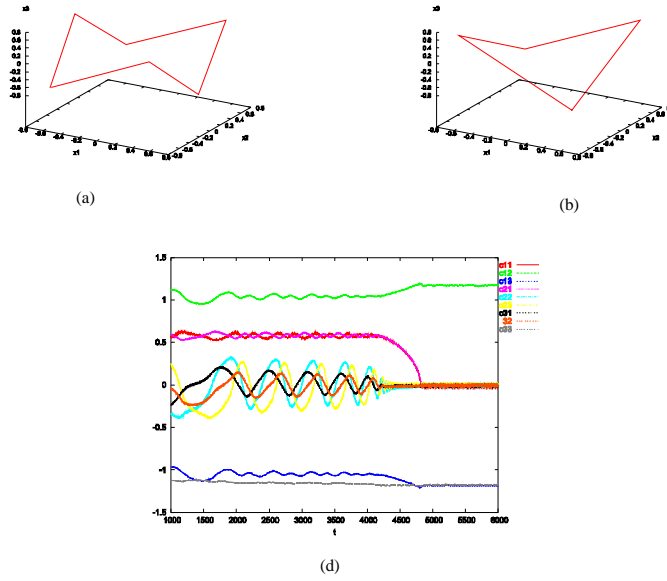


FIGURE 1. In (a) we see the period-6 cycle exhibited by the system in the interval of time before 4000 . In (b) we see the period-4 cycle exhibited by the system in the interval of time after 4000. (d) shows the values of C in the period during which the system makes the transition between its two cycles.

we observe that x_2 , x_3 oscillate with very low frequency while x_1 oscillates with higher frequency. Also we observe that the phase shift between x_2 , x_3 is such that x_3 precedes x_2 , see (d) in the that figure. In the second region 1800:2200, x_2 , x_3 oscillate with very high frequency at the beginning followed with continuously decreasing one and the phase shift is opposite to that of the previous period, but x_1 still oscillates with the same frequency as in the in the first regions. The third region 2600:3000 is characterized by the oscillation of x_1 and x_3 with very low frequency and the phase shift between them is such that x_1 precedes x_3 , while x_2 oscillates with higher frequency. In the fourth region, x_1 , x_3 oscillate with very high frequency at the beginning. Then the frequency is decreasing and the phase shift is opposite to that of the third region, see (b). However, x_2 still oscillates with the same frequency as in the in the third region.

Figure 2 demonstrates that the frequency wandering effect is also present in the three-dimensional system, however it does not involve all three neurons but instead realizes in subsystems of two neurons, with a switching between the subsystems after some time. As in the 2-d system, the effect is found to be rather stable, i.e. it emerges under a broad range of initial conditions provided the noise must not be too strong.

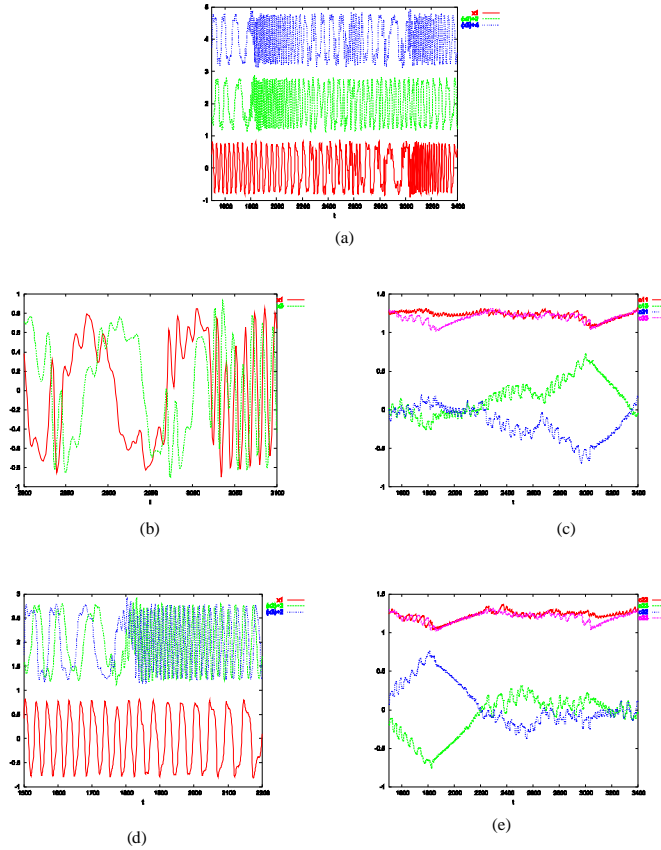


FIGURE 2. The frequency wandering effect in three-dimensional systems, (a) showing the state variables in the time interval 1500 through 3200, (c) and (e) the matrix elements of C for the 1-3 and the 2-3 submatrices, respectively. From (e) we see that in the time interval 1500 through 2200 we have the frequency wandering effect between neurons 1 and 3 whereas the effect switches afterwards to the subsystems of the neurons 2 and 3, see (c). Figures (b) and (d) display the state variables in the time intervals 2800 up to 3100 and 1500 to 2200 which are the pertinent regions for the frequency wandering phenomenon. These figures also make the relation between the phase shifts clear in the sense explained in the two-dimensional case.

Natural gradient parameter dynamics

So far we have used the ordinary gradient descent of the error function in the learning. As it is well known, the gradient of a function gives the steepest direction of this function, but this is helpful only if the parameter space of this function is the Euclidean space with orthonormal coordinate system. In the other cases when the parameter space is not Euclidean and has the Riemannian metric structure, the optimal direction for descending the function is given by another gradient namely the natural gradient as introduced by Amari into the learning theory, see [1]. The natural gradient method will be seen to help us in restructuring the general approach such that the exploration properties of the robot can be modified into desired directions by defining an appropriate metrics in the parameter space.

1. The natural gradient method

Let $S = \{w \in R^n\}$ be the parameter space on which a function $E(w)$ is defined and S is a nonlinear manifold, then the length of a vector w in parameter space is defined as

$$\|w\|_\gamma = \sqrt{\sum_{i,j} \gamma_{ij} w_i w_j}$$

where $\Gamma = (\gamma_{ij})$ is called the Riemannian metric tensor which depends in general on w and the space is called Riemannian space. In the following we will prove that the steepest ascent direction of $E(w)$ in a Riemannian space is given by the natural gradient $\tilde{\nabla}E(w) = \Gamma^{-1}\nabla E(w)$ where Γ^{-1} is the inverse of the metrics Γ and

$$\nabla E = \frac{\partial E}{\partial w} = \left(\frac{\partial E(w)}{\partial w_1}, \dots, \frac{\partial E(w)}{\partial w_n} \right)^T$$

is the ordinary gradient as used before.

The optimal direction for descending $E(w)$ at w is defined by the vector dw that maximize $E(w + dw) - E(w)$ under the constraint $\|dw\|_\gamma^2 = \varepsilon^2$ with ε a small constant. Using the Lagrangian method we want to maximize

$$\{E(w + dw) - E(w)\} - \lambda\{\|dw\|^2 - \varepsilon^2\}$$

which by means of

$$E(w + dw) - E(w) = (\nabla E(w))^T dw$$

leads to maximizing

$$(\nabla E(w))^T dw - \lambda dw^T \Gamma dw$$

An extremum with respect to the small vector dw is found if

$$\frac{\partial}{\partial dw} \{(\nabla E(w))^T dw - \lambda dw^T \Gamma dw\} = 0$$

This implies

$$\nabla E(w) - 2\lambda \Gamma dw = 0$$

hence

$$dw = \frac{1}{2\lambda} \Gamma^{-1} \nabla E(w)$$

and our gradient **descent** rule using the natural gradient consequently is

$$\Delta w = -\varepsilon \tilde{\nabla} E(w)$$

where

$$\begin{aligned} \tilde{\nabla} E(w) &= \Gamma^{-1} \nabla E(w) \\ dw &\propto \Gamma^{-1} \nabla E(w) \end{aligned}$$

and the natural gradient is taken as

$$dw = \Gamma^{-1} \nabla E(w)$$

In the following we will consider a very simple example in order to show how this method can be applied in order to make the robot include the sensor values according to their reliability.

2. Application: One neuron several channels

Let us consider the case of one controller neuron with n sensoric channels.

2.1. The sensorimotor loop. The sensorimotor loop can be written as

$$x_{i,t+1} = a_i y_t + \xi_{j,t} \quad (2.1)$$

where $x \in R^n$, $y \in R^1$ so that our usual sensorimotor dynamics $x_{t+1} = \psi(x_t) + \xi_t$ specifies to

$$\psi_i(x) = a_i y = a_i g(z) = a_i g\left(\sum_{p=1}^n c_p x_p + H\right)$$

The Jacobian L in this case is

$$L_{ij} = \frac{\partial \psi_i}{\partial x_j} = a_i c_j g'(z)$$

or

$$L = R g'(z)$$

where¹

$$R = a c^T$$

From our general principle we have

$$\xi = Lv \quad (2.2)$$

The Jacobian matrix L is not invertible, so the shift v is not uniquely identified. We may remove the ambiguity by making an assumption on the

¹In the present Section, $c \in R^1$ so that we use again a small letter.

direction of v . Because of eq. 2.1 it is appropriate to put v proportional to a so that we put

$$v = u\hat{a}$$

where $\hat{a} = a/\|a\|$ (meaning that \hat{a} is a unit vector in the direction of a), so that

$$Lv = ug'ac^T\hat{a} = ug'\|a\|\hat{a}c^T\hat{a} = ug'R\hat{a} \quad (2.3)$$

where now

$$R = \sum_{p=1}^n a_p c_p = a^T c$$

Using this customized v in our error formula we get

$$E = v^T v = u^2$$

In order to find u we multiply the 2.2 by \hat{a} so that

$$\xi \cdot \hat{a} = Rg'(z)u$$

hence

$$E = u^2 = \frac{(\xi \cdot \hat{a})^2}{(Rg'(z))^2} = \alpha\Lambda^{-2}$$

the gradient is obtained as

$$\begin{aligned} -\frac{\partial E}{\partial c_p} &= -2u \frac{\partial u}{\partial c_p} = 2\alpha\Lambda^{-3} \frac{\partial \Lambda}{\partial c_p} \\ &= 2u^2 \Lambda^{-1} \frac{\partial \Lambda}{\partial c_p} \end{aligned}$$

or

$$-\frac{\partial E}{\partial c_p} = 2u^2 \Lambda^{-1} (a_p g' + Rg'' x_p) \quad (2.4)$$

$$= \mu a_p (1 - 2Ry^2) \quad (2.5)$$

where $g'' = -2gg'$, $x_p = a_p y$ and $\mu = \frac{2u^2}{R}$.

The parameter dynamics in the normal gradient descent scenario reads

$$\Delta c_i = \mu a_p (1 - 2Ry^2)$$

which is stationary at $1 = 2Rg(z)$ and the fixed point relation is as before $z = Rg(z)$, see Chapter 4. It is appropriate to add a small damping term in order to damp away remnants of the initial conditions which are not changed by the parameter dynamics due to the ambiguity of the definition of v . Using a penalty term as $\|c\|^2$ which is aimed at suppressing large values of c , the time loop error now is

$$E = u^2 + \lambda \sum c_i^2 \quad (2.6)$$

We find

$$\Delta c_i = \mu a_i (1 - 2Ry^2) - \lambda c_i \quad (2.7)$$

Considering this at the stationary state where $\Delta c = 0$ we get our first result

$$c_i = k a_i \quad (2.8)$$

where

$$k = \frac{\mu}{\lambda} (1 - 2Ry^2) \quad (2.9)$$

Hence the contribution of the sensor i to the sensorimotor dynamics is proportional to the response strength a_i of the sensor to the actions y , see also [8].

We conclude this paragraph with a discussion of the value of k . In the present many channel case, the fixed point equation

$$z = Rg(z) \quad (2.10)$$

is as in Chapter 4, with $R = ka^2$ where k is considered unknown so far. Using eqs. 2.8,2.10 in eq. 2.7, we have to consider eq. 2.10 together with

$$1 - 2zg(z) = \frac{\lambda}{\mu}k = \frac{\lambda}{\mu a^2}R \quad (2.11)$$

With $\lambda = 0$ we find the solution given already in Chapter 4. Let us call that one z, k (where $R = ka^2 = 1.1911$, $z = 0.7717$) and the one for finite λ as $z + \delta z$ and $k + \delta k$. From eq. 2.8 we find in lowest order

$$\delta z = -\frac{\lambda}{2\mu(g(z) + zg'(z))}k = \frac{\lambda}{2\mu a^2 g(z)(1 + Rg'(z))}R$$

Using this in $z + \delta z = a^2(k + \delta k)g(z + \delta z)$ we find

$$\begin{aligned} \delta k &= \frac{1}{a^2 g(z)}(1 - Rg'(z))\delta z = \frac{\lambda k}{2\mu g^2(z)a^2(1 + Rg'(z))} \\ &= 0.183 \frac{\lambda k}{2\mu x^2} \end{aligned}$$

so that

$$\delta k = \alpha \frac{\lambda}{\mu}, \quad \delta z = \beta \frac{\lambda}{\mu}$$

where α, β are of the order of 1 so that the dependence of the fixed point on the value of λ/μ is smooth.

2.2. Natural gradient parameter dynamics. In order to use the natural gradient we must specify the metrics of the Riemannian space first. For the present purpose we use the metrics via the noise matrix $D = \overline{\xi\xi^T}$ as

$$\gamma_{ij} = D_{ij} = \overline{\xi_i \xi_j} \quad (2.12)$$

In the following we will use the natural gradient descent to derive the learning rules of the parameters c for the case of one neuron and many channels. Using the metrics given by eq. 2.12 we at first have to change the penalty term in the time loop error which was defined as the length of the vector c , cf. eq. 2.6. According to our new metrics in parameter space we put

$$E = u^2 + c^T D c$$

In general, this matrix will depend on the parameters itself since the latter control the behavior and the model error ξ of the world model of course will depend largely on the quality of the behavior. In the present approach this is not the case, we are just going to sketch the consequences of using the natural gradient descent.

The natural gradient in the present case reads

$$\tilde{\nabla} E = D^{-1} \nabla E \quad (2.13)$$

which produces the parameter dynamics

$$\Delta c = \varepsilon\mu(1 - 2Ry^2)D^{-1}a - \lambda c \quad (2.14)$$

With two channels this reads using the explicit form of D^{-1}

$$\begin{pmatrix} \Delta c_1 \\ \Delta c_2 \end{pmatrix} = \frac{\mu\varepsilon}{\det(D)}(1 - 2Ry^2) \begin{pmatrix} \overline{\xi_2^2} & -\overline{\xi_1\xi_2} \\ -\overline{\xi_1\xi_2} & \overline{\xi_1^2} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} - \lambda \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \quad (2.15)$$

The dynamics of eq. 2.14 is stationary if

$$c = kD^{-1}a$$

where k was given above. this is our second result meaning that the c is not proportional to a itself as in eq. 2.8 but instead to the vector $D^{-1}a$. The consequences of this new relation is seen most easily in the case of uncorrelated noise

$$D_{ij} = \delta_{ij}\overline{\xi_i^2}$$

where we get immediately

$$c_i = \frac{k}{\overline{\xi_i^2}}a_i$$

This is the main result showing that the strength of the coupling c_i of channel i is proportional to the response strength of the sensor i divided by the strength of the noise in that channel. The quotient

$$\frac{a_i}{\overline{\xi_i^2}}$$

may be considered as a measure of the feasibility of the sensor so that the control exerted by the neuron in channel i is proportional to this feasibility, more noise channels being less contributing to the sensorimotor dynamics. The theoretical results are well corroborated by the computer simulations, see the Figs. 1 and 2.

3. An algorithm for updating the inverse noise matrix.

With respect to practical applications, the matrix D can be obtained on-line by a moving average procedure according to

$$D_{t+1} = \eta\xi\xi^T + (1 - \eta)D_t$$

where D_t is the average noise matrix in time step t and η^{-1} defines the width of the time window for the averaging. Moreover, it would of course be helpful to have also an algorithm for the update of the inverse matrix D^{-1} . Otherwise the matrix D_t would have to be inverted in each time step which is not feasible since the number of channels may be very high. In practical applications, the sensor values for instance might be the pixels of a camera. For this purpose we may use the Sherman-Morrison formula for inverse matrix updates. The latter states that if we have an $n \times n$ matrix A which is to be changed as

$$A \rightarrow A + wv^T$$

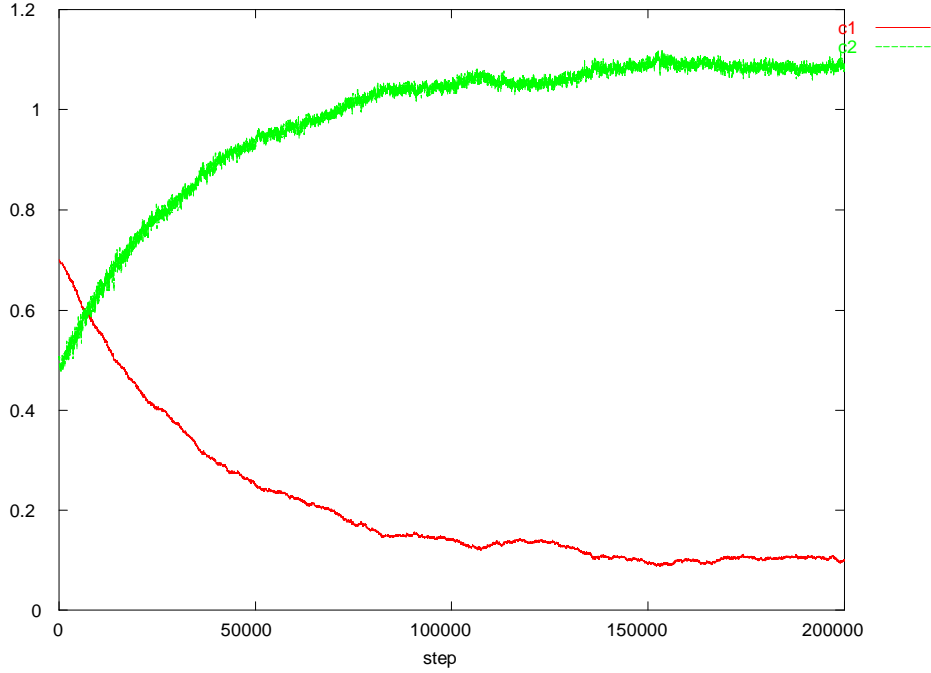


FIGURE 1. Natural gradient dynamics for the case of one neuron and two channels. The parameters are $a_1 = a_2$ and $(\overline{\xi_1^2})^2 = 10(\overline{\xi_2^2})^2$. This graph shows that starting from $c_1 = 0.7$, $c_2 = 0.5$ the c values converge towards the expected relation, i.e. the c of the channel with larger noise strength is about 10 times smaller than that of smaller noise strength.

where u and v are $n \times 1$ column vectors then the inverse of the new matrix is given as

$$(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^T A^{-1}}{1 + v^T A^{-1}u}$$

Let us put $A = (1 - \eta)D_t$ and $uv^T = \eta\xi\xi^T$. So, if D^{-1} is known initially, the update of D_t^{-1} reads as

$$D_{t+1}^{-1} = \frac{1}{(1 - \eta)} \left(D_t^{-1} + \frac{\eta}{(1 - \eta)} \frac{D_t^{-1}\xi\xi^T D_t^{-1}}{(1 + (\frac{\eta}{1 - \eta})\xi^T D_t^{-1}\xi)} \right)$$

(the denominator is a scalar). In this way we can get the metrics for our natural gradient descent in an on-line learning process. This has been used in the computer simulations and has proven a very reliable algorithm.

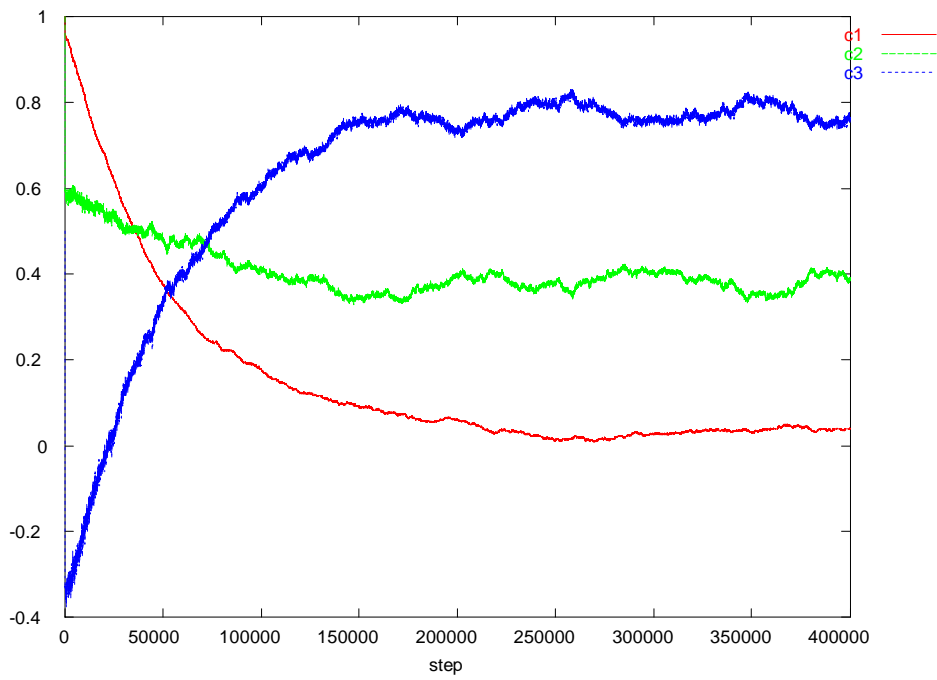


FIGURE 2. This graph shows that the channel with largest noise strength(c_1) has the smallest value. The used learning rate is 0.0002, damping term 0.1 and $\gamma = 0.001$. The initial condition is $x_1 = x_2 = x_3 = 0$, $c_1 = c_2 = 1$ and $c_3 = 0.5$

CHAPTER 8

Summary and outlook

The present thesis is a theoretical contribution to a systematic approach to the self-organization of behavior in autonomous robots. It is deeply rooted in the dynamical system approach to robotics, cognitive science, and artificial intelligence and investigates in some detail the properties of such systems based on a general paradigm. The paradigm tries to give the robot a maximum sensitivity to its sensor values together with a maximum predictability of future sensor values resulting from the actions taken by the robot. Formally, this leads to a parameter dynamics for the neural network controlling the robot which is based on the gradient descent on the so called time loop error, see Sec. 3.3 of Chapter 3. The latter is formulated entirely in terms of the dynamics of the dynamical system describing the robot in its environment (state dynamics). Hence the parameters for the state dynamics are driven by the state dynamics itself and this is why we call them self-referential dynamical systems.

The thesis studies such systems in some detail. The results obtained may be summarized in the following way. A first result is obtained in a very simple sensorimotor loop with delays. In such systems, under the closed loop control paradigm used in the present thesis, one often encounters rapid oscillations of the robot which may be strong enough to destroy the robot. These oscillations have been shown to arise from the time delays and it was demonstrated that by a simple smoothing procedure the problem can be solved. The further chapters of the thesis are devoted to the study of the self-referential systems. On a quite general basis, see Chapter 3, one may argue that the self-referential system tries to maximize its exploration rate in the dimensions of the state space where the model error is large. In this way the robot is enabled to gather more information in the dimensions which are not well covered by the world model, see Sec. 5.3 of Chapter 3. With a good separation of the time scales for the parameter dynamics (learning) and the state dynamics it was shown that the self-referential system generates a fixed point flow of the state variables towards the bifurcation points with occasional jumps to other fixed points, see Sec. 6 of Chapter 3. This is the generalization of the well known situation in the one-dimensional case (with learning of both the bias and the feed-back strength parameters) which was demonstrated in earlier papers to lead to a limit cycle behavior. The thesis adds to these well known results an example for the learning of the world model as driven by the active exploration behavior of the robot in the sense mentioned above, see Sec. 2.2 of Chapter 4.

The main body of results is obtained for the case of a two-dimensional system, see Chapter 5. In the linear case it could be shown that the controller matrix C converges for a large set of initial conditions to an $SO(2)$ structure

leading to a periodic or quasiperiodic state dynamics with frequency largely depending on the initial condition of the C matrix. Nonlinearities are shown to largely modify this picture. In order to get a systematics at first the case of learning only the controller matrix C is considered. Several different phenomena could be identified which are realized alternatively depending on the initial conditions and the reaction of the environment. Noteworthy are (i) the case of regular oscillations where a period-4 cycle was shown to be particularly stable; (ii) the case of irregular oscillations with the non-diagonal elements playing the role of a bias dynamics; and (iii) the switching oscillator scenario where one neuron is oscillating and the other one keeps itself at the fixed point, the role of the neurons interchanging after some time. In the latter cases the frequency is seen to be regulated by both the learning rate and the reactions of the environment (noise) so that the behavior of the robot is strongly coupled to the behavior of the world.

The most interesting effects were found for the case that each neuron also has a bias driven as well by the gradient dynamics on the time loop error. The emerging bias dynamics can be considered as an additional internal state dynamics for the controller which increases the complexity of possible behaviors of the robot far beyond a simple reactive behavior. The most prominent effect has been found in form of the so called frequency wandering, which makes the robot to continuously sweep through its accessible periodic behaviors. This is a clear example of its exploration capabilities.

In a practical application, the so called barrel robot, the frequency wandering effect is also realized however in a different way dictated by the embodiment of the agent which is seen to react on the parameter dynamics in a definite way. The results show how the parameter dynamics driving the internal state of the controller sensitively adapts to the specific properties of the body it is controlling. On the phenomenological level one might say that the robot is able to both accelerate and decelerate or that it actively investigates its space of velocities. The sensitive dependence of the combined state-parameter dynamics has also been made explicit by considering the case of noise over the channels, see Sec. 5 of Chapter 5.

In the three dimensional case several of the effects found in two dimensions could be shown to survive in a modified form. In particular, in the case of no bias a theoretical and computer simulation study was carried out showing the persistence of the regular oscillations with higher order periodicity. When the bias dynamics is included, the frequency wandering effect is also observed, however it does not involve all three neurons but instead realizes in subsystems of two neurons, with a switching between the subsystems after some time. The effect is found to be rather stable, i.e. it emerges under a broad range of initial conditions provided the noise is not too strong.

A modification of the general approach has been presented in Chapter 7 by using the so called natural gradient for generating the parameter dynamics from the time loop error. The natural gradient as introduced by Amari into learning theory gives the optimal direction for descending a function if the metric of the space is Riemannian. By using the noise matrix D as the metric tensor we could show that, under the closed loop control paradigm, control is realized on the basis of sensor values weighted according to their

feasibility. In the most simple case of uncorrelated noise, this measure is given by the response strength of the sensor divided by the strength of the noise. In more general cases the controls are given by the vector of response strengths a multiplied by the matrix D^{-1} . Moreover an algorithm was proposed and tested in simulations for the on-line learning of this inverse noise matrix.

The present thesis has revealed that the time loop error is a constructive method for the self-organization of a robot behavior which is both explorative and sensitive to the environment. Most of the investigations, however, have been carried through under the assumption that the world (robot + environment) is more or less trivial, i.e. we have proprioceptive sensors only which report on the result of the actions plus some noise. This is reflected in the linear expression $x_{t+1} = Ay_t + \xi_t$ of the sensorimotor dynamics with ξ being a pure (and mostly small) random number. An exception has been the barrel robot which is a physical object with large inertia effects and a very complicated relation between motor and sensor values. Nevertheless a strong parallel to the results with the idealized world could be established in the presence of the frequency wandering effect. It would be very interesting to repeat these investigations with different kinds of embodied agents.

Bibliography

- [1] S. Amari. Natural gradients work efficiently in learning. *Neural Computation*, 10, 1998.
- [2] R. D. Beer. A dynamical systems perspective on agent-environment interaction. *Artif. Intell.*, 72(1-2):173–215, 1995.
- [3] R. D. Beer. Dynamical approaches to cognitive science. *Trends in Cognitive Sciences*, 4(3):91–99., 2000.
- [4] A. Bredenfeld, H. Jaeger, and T. Christaller. Mobile robots with dual dynamics. *ERCIM News*, 42, 2001.
- [5] J. Buchli, L. Righetti, and A. Ijspeert. A dynamical systems approach to learning: a frequency-adaptive hopper robot. In *Proceedings of the VIIIth European Conference on Artificial Life ECAL 2005*, Lecture Notes in Artificial Intelligence, pages 210–220. Springer Verlag, 2005.
- [6] M. Cohen. The construction of arbitrary stable dynamics in nonlinear neural networks. *Neural Networks*, 5:83–103, 1992.
- [7] R. Der. Self-organized acquisition of situated behavior. *Theory Biosci.*, 120:179–187, 2001.
- [8] R. Der, F. Hesse, and R. Liebscher. Contingent robot behavior generated by self-referential dynamical systems. *Autonomous robots*, 2005. submitted.
- [9] R. Der, F. Hesse, and G. Martius. Learning to feel the physics of a body. In *Proceedings CIMCA'2005*, 2005.
- [10] R. Der, F. Hesse, and G. Martius. Rocking stumper and jumping snake from a dynamical system approach to artificial life. *J. Adaptive Behavior*, 14:105 – 116, 2005.
- [11] R. Der, F. Hesse, and G. Martius. From motor babbling to purposive actions: Emerging self-exploration in a dynamical systems approach to early robot development. In *Proc. Simulation of Adaptive Behavior - SAB'06*, volume 14, 2006. to appear.
- [12] R. Der, G. Martius, and F. Hesse. Let it roll – emerging sensorimotor coordination in a spherical robot. In L. M. Rocha, editor, *Artificial Life X*, pages 192 – 198. MIT Press, 2006.
- [13] W. Ebeling and R. Feistel. *Evolution of Complex Systems*. Deutscher Verlag der Wissenschaften, Berlin, 1989.
- [14] A. Feldman. Superposition of motor programs, I. Rhythmic forearm movements in man. *Neuroscience*, 5:81–90, 1980.
- [15] M. D. G. Schöner and C. Engels. Dynamics of behavior: Theory and applications for autonomous robot architectures. *Robotics and Autonomous Systems*, 16:213–245., 1995.
- [16] H. Haken. *Advanced Synergetics*. Springer, Berlin, 1987.
- [17] H. Haken. *Synergetics of Cognition*. Springer, Berlin, 1990.
- [18] R. Haschke. *Bifurcations in Discrete-Time Neural Networks – Controlling Complex Network Behaviour with Inputs*. PhD thesis, Bielefeld University, Sep 2003.
- [19] R. Haschke and J. J. Steil. Input space bifurcation manifolds of recurrent neural networks. *Neurocomputing*, 64C:25–38, 2005.
- [20] M. Hulse, S. Wischmann, and P. F. Structure and function of evolved neuro-controllers for autonomous robots. *Connection Science*, 16(4):249–266, 2004.
- [21] M. Hülse and F. Pasemann. *Dynamical Neural Schmitt Trigger for Robot Control*, volume 2415 of *Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, New York, 2002.

- [22] F. Iida, G. J. Gomez, and R. Pfeifer. Exploiting body dynamics for controlling a running quadruped robot. In *International Conference of Advanced Robotics (ICAR 05)*, 2005.
- [23] F. Iida and R. Pfeifer. Cheap rapid locomotion of a quadruped robot: Self-stabilization of bounding gait. *Intelligent Autonomous Systems*, 8, 2004.
- [24] F. Iida, R. Pfeifer, L. Steels, and Y. Kuniyoshi, editors. *Embodied Artificial Intelligence, International Seminar, Dagstuhl Castle, Germany, July 7-11, 2003, Revised Papers*, volume 3139 of *Lecture Notes in Computer Science*. Springer, 2004.
- [25] A. J. Ijspeert, J. Nakanishi, and S. Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *ICRA*, pages 1398–1403, 2002.
- [26] F. Kaplan and P.-Y. Oudeyer. Maximizing learning progress: An internal reward system for development. In *Embodied Artificial Intelligence*, pages 259–270, 2003.
- [27] D. Kim. Self-organization for multi-agent groups. *International Journal of Control, Automation, and Systems*, 2:333–342, 2004.
- [28] D.-H. Kim and J.-H. Kim. A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer. *Robot. Auton. Syst.*, 42(1):17–30, 2003.
- [29] A. S. Klyubin, D. Polani, and C. L. Nehaniv. Organization of the information flow in the perception-action loop of evolved agents. In *Evolvable Hardware*, pages 177–, 2004.
- [30] S. Kotosaka and S. Schaal. Synchronized robot drumming by neural oscillators. In *Proceedings of the International Symposium on Adaptive Motion of Animals and Machines*, 2000.
- [31] Y. Kuniyoshi, Y. Yorozu, Y. Ohmura, K. Terada, T. Otani, A. Nagakubo, and T. Yamamoto. From humanoid embodiment to theory of mind. In *Embodied Artificial Intelligence*, pages 202–218, Berlin, Heidelberg, New York, 2003. Springer.
- [32] Y. A. Kuznetsov. *Elements of Applied Bifurcation Theory*. Springer, Berlin, Heidelberg, 2004.
- [33] K. Lerman. A model of adaptation in collaborative multi-agent systems. *Adaptive Behavior*, 12:187–198,, 2004.
- [34] K. Lerman and A. Galstyan. Automatically modeling group behavior of simple agents. Agent Modeling Workshop, AAMAS-04, New York, NY., 2004.
- [35] K. Lerman, A. Martinoli, and A. Galstyan. A review of probabilistic macroscopic models for swarm robotic systems. Self-organization of Adaptive Behavior04, Santa Monica, CA, 2004.
- [36] M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini. Developmental robotics: a survey. *Connect. Sci.*, 15(4):151–190, 2003.
- [37] M. Lungarella and O. Sporns. Information self-structuring: Key principle for learning and development. In *Proceedings 2005 IEEE Intern. Conf. Development and Learning*, pages 25–30, 2005.
- [38] T. McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 9, No., 2,:62–82, 1990.
- [39] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato. Learning from demonstration and adaptation of biped locomotion. *Robotics and Autonomous Systems*, 47:79–91, 2004.
- [40] S. Nolfi and D. Floreano. *Evolutionary Robotics*. MIT Press, Cambridge, MA, 2000.
- [41] P.-Y. Oudeyer. The self-organization of speech sounds. *Journal of Theoretical Biology*, 233(3):435–449, 2005.
- [42] P.-Y. Oudeyer. *Self-Organization in the Evolution of Speech*. Studies in the Evolution of Language. Oxford University Press, 2006.
- [43] P.-Y. Oudeyer, F. Kaplan, V. V. Hafner, and A. Whyte. The playground experiment: Task-independent development of a curious robot. In D. Bank and L. Meeden, editors, *Proceedings of the AAAI Spring Symposium on Developmental Robotics, 2005, Pages 42-47, Stanford, California, 2005.*, 2005.
- [44] F. Pasemann. Complex dynamics and the structure of small neural networks network. *Computation in Neural Systems*, 13:195–216, 2002.

- [45] F. Pasemann, M. Hild, and K. Zahedi. SO(2)-networks as neural oscillators. In J. Mira and J. Alvarez, editors, *Computational Methods in Neural Modeling*, pages 144–151, Berlin, Heidelberg, New York, 2003. Springer.
- [46] R. Pfeifer and J. Bongard. *How the Body Shapes the Way We Think. A New View of Intelligence*. MIT Press, 2006.
- [47] R. Pfeifer and F. Iida. Embodied artificial intelligence: Trends and challenges. In *Embodied Artificial Intelligence*, pages 1–26, Cambridge, MA, 2003. Bradford Books, MIT Press.
- [48] R. Pfeifer and C. Scheier. *Understanding Intelligence*. Bradford Books, 1999.
- [49] R. F. Port and T. V. Gelder, editors. *Mind as Motion*. The MIT Press, Cambridge, MA, 1995.
- [50] L. Righetti and I. A.J. Programmable central pattern generators: an application to biped locomotion control. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, 2006. In press.
- [51] L. Righetti, J. Buchli, and A. Ijspeert. Dynamic hebbian learning in adaptive frequency oscillators. *Physica D*. In Press.
- [52] J. Schmidhuber. Completely self-referential optimal reinforcement learners. In *ICANN (2)*, pages 223–233, 2005.
- [53] S. Singh, A. G. Barto, and N. Chentanez. Intrinsically motivated reinforcement learning. In *Proceedings of Advances in Neural Information Processing Systems (NIPS) 17*, volume 17, 2005.
- [54] A. Steinhage. *Dynamical Systems for the Generation of Navigation Behavior*. PhD thesis, Ruhr-Universit at Bochum, Germany, 1997.
- [55] B. H. Stewart and J. M. Thompson. *Nonlinear Dynamics and Chaos*. John Wiley and Sons, 2002.
- [56] J. Tani. Learning to generate articulated behavior through the bottom-up and the top-down interaction processes. *Neural Networks. The Official Journal of the International Neural Network Society, European Neural Network Society, Japanese Neural Network Society.*, 16(1):11–23, 2001.
- [57] J. Tani and M. Ito. Self-organization of behavioral primitives as multiple attractor dynamics: A robot experiment. *IEEE Transactions of on Systems, Man, and Cybernetics Part A: Systems and Humans*, 33(4):481–488, 2003.
- [58] W. Tschacher and J. Dauwalder. *The Dynamical Systems Approach to Cognition: Concepts and Empirical Paradigms Based on Self-Organization, Embodiment, and Coordination Dynamics*. World Scientific Publishing Company, Singapore, 2003.
- [59] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen. Autonomous mental development by robots and animals. *Science*, 291:599 – 600, 2001.
- [60] S. Wischmann and F. Pasemann. From passive to active dynamic 3d bipedal walking - an evolutionary approach -. In M. Armada and P. Gonzalez de Santos, editors, *Proc. of the 7th Int. Conference on Climbing and Walking Robots (CLAWAR 2004)*, pages 737–744. Springer Verlag, 2004.