

# **Perspectives of the High Dimensional Dynamics of Neural Microcircuits from the Point of View of Low Dimensional Readouts**

Stefan Häusler<sup>\*+</sup>, Henry Markram<sup>+</sup>, and Wolfgang Maass<sup>\*</sup>

<sup>\*</sup>Institute for Theoretical Computer Science

Technische Universität Graz

Inffeldgasse 16b, A-8010 Graz, Austria

Tel.: +43 316 873-5811

Fax: +43 316 873-5805

E-mail: [haeusler@igi.tu-graz.ac.at](mailto:haeusler@igi.tu-graz.ac.at), [maass@igi.tu-graz.ac.at](mailto:maass@igi.tu-graz.ac.at)

<sup>+</sup>Brain Mind Institute

Ecole Polytechnique Federale de Lausanne

CH-1015 Lausanne, Switzerland

E-mail: [henry.markram@epfl.ch](mailto:henry.markram@epfl.ch)

April 24, 2003

## **Abstract**

We investigate generic models for cortical microcircuits, i.e. recurrent circuits of integrate-and fire neurons with dynamic synapses. These complex dynamic systems subserve the amazing information processing capabilities of the cortex, but are at the present time very little understood. We analyze the transient dynamics of models for neural microcircuits from the point of view of one or two readout neurons that collapse

the high dimensional transient dynamics of a neural circuit into a 1- or 2--dimensional output stream. This stream may for example represent the information that is projected from such circuit to some particular other brain area or actuators. It is shown that simple local learning rules enable a readout neuron to extract from the high dimensional transient dynamics of a recurrent neural circuit quite different low-dimensional projections, that even may contain "virtual attractors" which are not apparent in the high dimensional dynamics of the circuit itself. Furthermore it is demonstrated that the information extraction capabilities of linear readout neurons are boosted by the computational operations of a sufficiently large preceding neural microcircuit. Hence a generic neural microcircuit may play a similar role for information processing as a kernel for support vector machines in machine learning. We demonstrate that the projection of time-varying inputs into a large recurrent neural circuit enables a linear readout neuron to classify the time-varying circuit inputs with the same power as a complex nonlinear classifiers, such as for example a pool of perceptrons trained by the p-delta-rule, or a feedforward sigmoidal neural net trained by backprop, provided that the size of the recurrent circuit is sufficiently large. At the same time such readout neuron can exploit the stability and speed of learning rules for linear classifiers, thereby overcoming the problems caused by local minima in the error function of nonlinear classifiers. In addition it is demonstrated that pairs of readout neurons can transform the complex trajectory of transient states of a large neural circuit into a simple and clearly structured 2-dimensional trajectory. This 2-dimensional projection of the high-dimensional trajectory can even exhibit convergence to virtual attractors which are not apparent in the high dimensional trajectory.

## **1.Introduction**

Computation in biological neural circuits is often modeled by attractor neural networks with low dimensional internal state spaces, and analyzed from the point of view of a human observer with focus on easily discernible features such as convergence to an

attractor. However, the dynamics of real neural microcircuits, consisting of a few thousand neurons, represents a trajectory in a very high dimensional dynamical system. Due to its high dimensionality, new phenomena emerge which cannot be observed in the commonly studied 2- or 3-dimensional dynamical systems. Functionally most important are features of the dynamics of neural circuits that can be extracted by readout neurons, i.e. by neurons that receive inputs from hundreds or thousands of neurons in this circuit and transmit low dimensional projections of their transient dynamics to other brain areas, or to actuators. This article explores the possible relationship between the high dimensional dynamics of neural circuits and their neural readouts through computer simulations of generic cortical microcircuits.

Recently Maass, Markram & Natschlaeger proposed a general theoretical model, called *liquid state machine* [Maass et al., 2002], which represents a convenient framework for neural computations in real time for rapidly time varying continuous input functions. It does not require convergence to stable internal states or attractors, since information about past inputs is captured in the perturbations of a high dimensional dynamical system, i.e. in the continuous trajectory of transient internal states. First the input stream is projected into a sufficiently large neural circuit. In general different input streams will cause different trajectories of internal states of the system, i.e., the input streams are separated by the circuit. Secondly a memory-less readout learns to extract salient information from the high dimensional transient states of the circuit. In particular each readout can learn to define its own classes of equivalence of dynamical states within the neural microcircuit, and can then perform its task on novel inputs. Due to this principle of “readout assigned equivalent states of a dynamical system” an invariant readout can be possible despite the fact that the neural microcircuit may never re-visit the same state. Furthermore multiple readouts can be trained to perform different tasks on the same state trajectories of a recurrent neural circuit, thereby enabling parallel real-time computing. Good separation capability of the high dimensional dynamical system for different preceding inputs, in combination with an adequate readout, allows essentially any real-time computation on continuous and bounded time-varying inputs with fading memory to an arbitrary degree of precision. It is shown in [Maass et al., 2002] that a generic neural

microcircuit model tends to have fairly good separation property, due to the biologically realistic diversity of its components and its sparse but recurrent connectivity (“loops within loops”). Adaptivity within the microcircuit itself is not necessary in this context, although it may facilitate the task of the readout for a family of related tasks. This situation is analogous to that of choosing kernels for support vector machines, where there exist general purpose kernels that provide good performance for a large variety of tasks.

Whereas in [Maass et al., 2002] the potential readout capabilities of *pools* of neurons were explored, we investigate in this article the readout capability of *single* integrate-and-fire (I&F) neurons and of pairs of such neurons. It is shown that for a sufficiently large recurrent neural circuit a single neuron as readout achieves the same classification power for a binary classification task (as specified in section 3.2) as sophisticated multi-unit classifiers, such as pools of perceptrons with the p-delta-rule, see [Auer et al., 2002], voted perceptrons, see [Warmuth et al., 2002; Freund et al., 1999], feedforward sigmoidal neural nets trained by backprop. Hence one may argue that a generic neural microcircuit plays a similar role for neural computing as a high dimensional kernel for support vector machines in machine learning. In addition it is demonstrated that pairs of readout neurons can transform the complex trajectory of transient states of a large neural circuit into a simple and clearly structured 2-dimensional trajectory. This 2-dimensional projection of the high-dimensional trajectory can even exhibit convergence to virtual attractors which are not apparent in the high dimensional trajectory.

## **2.Methods**

We carried out computer simulations with a generic recurrent network of I&F neurons as described in [Maass et al., 2002]. The input to the network consisted of spike trains, which diverged to inject current into 30 % randomly chosen excitatory neurons. The amplitudes of the input synapses were chosen from a Gaussian distribution, so that each neuron in the recurrent microcircuit received a slightly different input. The input spike

trains were generated from randomly generated Poisson spike templates with a frequency of 20 Hz, where each spike in the template was moved by a Gaussian distribution with mean 0 and an SD of 4 ms.

We used randomly connected circuits consisting of I&F neurons, 20 % of which were randomly chosen to be inhibitory (see [Tsodyks et al., 2000]). Unless stated otherwise in the figure legend the circuit size was chosen to be 135 neurons. Parameters of neurons and synapses were chosen in accordance with biological data: membrane time constant 30ms, absolute refractory period 3ms (excitatory neurons), 2ms (inhibitory neurons), threshold 15mV (for a resting membrane potential assumed to be 0), reset voltage 13.5 mV, constant background current at 13.5 nA, input resistance  $1M\Omega$ .

Connectivity structure: The distribution of connection lengths was chosen to be biologically realistic, with primary local connections and a few longer connections. More precisely the probability of a synaptic connection from neuron  $a$  to neuron  $b$  (as well as that of a synaptic connection from neuron  $b$  to neuron  $a$ ) was defined as  $C \cdot e^{-D(a,b)/\lambda^2}$ , where  $\lambda$  is a parameter which controls both the average number of connections and the average distance between neurons that are synaptically connected (its value was fixed at 1.5 for all simulations reported in this article, independent of the size of the network). For the circuits consisting of 135 neurons we assumed that the neurons were located on the integer points of a  $15 \times 3 \times 3$  column in space, where  $D(a,b)$  is the Euclidean distance between neurons  $a$  and  $b$ . The neurons of the circuits used for the simulations for figure 2 and 3 were arranged in  $2 \times 2 \times 3$ ,  $3 \times 3 \times 6$ ,  $5 \times 5 \times 4$ ,  $5 \times 5 \times 8$ ,  $7 \times 7 \times 8$ ,  $7 \times 7 \times 12$  and  $7 \times 7 \times 16$  columns, whereas the columns of the networks for figure 6 had the size  $3 \times 3 \times 9$ ,  $3 \times 3 \times 11$ ,  $3 \times 3 \times 13$ ,  $3 \times 3 \times 15$ ,  $3 \times 3 \times 18$  and  $3 \times 3 \times 21$ . Depending on whether  $a$  and  $b$  were excitatory ( $E$ ) or inhibitory ( $I$ ), the value of C was 0.3 (EE), 0.2 (EI), 0.4 (IE), 0.1 (II).

In the case of a synaptic connection from  $a$  to  $b$  the synaptic dynamics were modeled according to the model proposed in [Markram et al., 1998], with the synaptic parameters  $U$  (use),  $D$  (time constant for depression),  $F$  (time constant for facilitation) randomly

chosen from Gaussian distributions that were based on biological data reported in [Gupta et al., 2000] and [Markram et al., 1998]. Depending on whether  $a, b$  were excitatory ( $E$ ) or inhibitory ( $I$ ), the mean values of these three parameters (with  $D, F$  expressed in second, s) were chosen to be .5, 1.1, .05 (EE), .05, .125, 1.2 (EI), .25, .7, .02 (IE), .32, .144, .06 (II). The scaling parameter A (in nA) was chosen to be 75 (EE), 150 (EI), -47 (IE), -47 (II). In the case of input synapses, the parameter A had a value of 18 nA. The SD of each parameter was chosen to be 50 % of its mean (with negative values replaced by values chosen from an appropriate uniform distribution). The time course of postsynaptic currents was modeled by an exponential decay  $\exp(-t/\tau_s)$  with  $\tau_s=3$  ms ( $\tau_s=6$  ms) for excitatory (inhibitory) synapses. The transmission delays between liquid neurons were chosen uniformly to be 1.5 ms (EE), and 0.8 for the other connections. For each trial the initial conditions of the circuit were randomly chosen (for each neuron in the circuit the membrane voltage was set at a value drawn from the uniform distribution over the interval [13.5 mV, 15 mV]).

We assumed that each readout neuron receives synaptic input from all neurons in the recurrent circuit. We defined the current *liquid state* of the circuit (using the terminology [Maass et al., 2002]) as the  $n$ -dimensional vector of contributions of the  $n$  neurons in the circuit to the membrane potential of a generic readout neuron at time  $t$  (assuming unit size weights and static synapses for this generic readout neuron). Technically these individual contributions to the membrane potential of a generic readout neuron are the outputs of a low pass filter with a kernel that decays exponentially with a time constant of 30 ms (reflecting the assumed 30 ms membrane time constant of the readout neuron), applied to the spike trains of the  $n$  neurons in the recurrent circuit. After training, the weights of a readout neuron have no longer uniform size, and hence each readout neuron defines a different projection of the high dimensional liquid states into one dimension. Therefore strictly speaking a readout neuron does not have full access to the real intrinsic state space of the recurrent network, which consists of the membrane potential of each I&F neuron and the fraction of available synaptic efficacy “R” and the running value of the utilization of synaptic efficacy “u” (for terminology see [Markram et al., 1998] ) of

each dynamical synapse. The trajectory of the recurrent neural circuit was modeled as a sequence of consecutive liquid states sampled every 20 ms. Each readout neuron defines in general a different projection of this trajectory of liquid states into a 1-dimensional trajectory, and correspondingly each pair of readout neurons defines a different projection of the high dimensional dynamics into 2 dimensions.

### **3.Results**

#### **3.1.Projecting the Input into a Larger Neural Circuit Increases the Classification Power of a Readout Neuron**

When a time-varying input, such as for example a Poisson spike train, is injected as input into a large recurrent circuit of  $n$  I&F neurons, it becomes difficult for a human observer to extract information about this input from the resulting dynamics of the circuit. Our computer simulations show that in contrast to that, the dynamics of neural circuits becomes easier to classify for a readout neuron when  $n$  is large. This effect is less surprising if one notes that the decision surface which is relevant for the decision whether an I&F neuron fires at some specific time  $t$ , can be approximated by a hyperplane in the state space of the dynamical system that models the recurrent circuit of I&F neurons. The number of degrees of freedom of this hyperplane grows with  $n$ . Furthermore an empirically well-supported result from statistical learning theory (see for example [Vapnik, 1998]) implies that the discrimination power of a perceptron (or hyperplane) increases when the inputs that need to be classified are first projected nonlinearly into a sufficiently high-dimensional space. This effect has been demonstrated by [Jaeger, 2001] in the context of artificial neural networks. We would like to argue that it may also play an important role in biological neural computation, and may contribute to the large and seemingly universal computational power of recurrent neural microcircuits. It may have received little attention so far in the analysis of neural computation, because it is not observable in small circuits. It also can usually not be observed in larger models for neural circuits if their architecture has been engineered by the modeler for a particular

task, since that often entails that the dynamics of that circuit is restricted to a lower dimensional subspace of its state space. For this reason we focus on the dynamics of generic models for recurrent neural microcircuits that reflect biological data in their connection statistics, and which have not been engineered for a particular purpose.

As a benchmark test for memory retrieval and pattern classification by neural circuits we considered the task illustrated in Figure 1. Four Poisson spike trains were randomly generated and fixed as spike pattern templates. More precisely, two such patterns, templates 1 and 2, were fixed for the time interval from 0 to 250 ms, and two other ones, templates 3 and 4, for the second interval from 250 to 500 ms. Input spike trains over 500 ms were randomly composed of noise variations (Gaussian distribution with mean 0,  $SD$  4 ms) of one of the templates 1, 2 in their first half followed by noisy variations of one of the templates 3, 4 in their second half. We defined that an input spike train belongs to class 1(2) if its first half was generated from the template spike train 1 (2), no matter whether its second half had been generated from. A readout neuron was required to carry out a classification at time  $t = 500$  ms, after a noisy variation of one of the 2 spike pattern templates 3 or 4 had been sent into the circuit and had "overwritten" the transient dynamics caused by the first pattern templates. One can therefore view the earlier pattern as one that sets the "context" for the second one, and it may be important from the functional point of view to recover this "context" at a later point in time. This classification task is relatively difficult, since it requires the integration of information over a time interval (and from a temporal distance) of 250 ms, which is fairly large compared to the membrane time constant of a single neuron (30 ms in our simulations). For this discrimination task only the weights of the synapses of the readout neuron were adapted, thus leaving the dynamics of the recurrent neural circuits unspecialized, potentially providing unbiased input to myriads of other readout neurons that are specialized to extract other information about the input to the circuit (see Fig. 8 in [Maass et al., 2002]). Several standard algorithms for single-unit and multi-unit neural classifiers were applied: pools of perceptrons with the p-delta-rule, see [Auer et al., 2002], voted perceptrons, see [Warmuth et al., 2002; Freund et al., 1999], and backprop for feedforward sigmoidal neural nets.

Panels a, b, and c of Fig. 2 show that each type of readout achieves a better performance when spike trains generated from input distributions as described above are injected into a larger recurrent circuit. Furthermore it can be seen that even for single neuron readouts (panels a, b) the classification error approaches 0 when the size of the recurrent circuit grows<sup>1</sup>. This effect is reminiscent of a frequently exploited effect in machine learning (more precisely in support vector machines and other kernel based methods). There one projects the given data first nonlinearly into a very high dimensional space. Within this high dimensional space the projections of the original data from different classes usually become linearly (or nearly linearly) separable, see [Vapnik, 1998]. But an essential difference to kernel-based methods in machine learning is that there the projection into a high dimensional space is not carried out explicitly, whereas in our neural model the nonlinear projection of the input stream into the high dimensional state set of the circuit may be viewed as the essential computational operation of the generic neural microcircuit model.

For a human observer the liquid states at time  $t = 500$  ms that result from input spike trains from the classes 1 and 2 look indistinguishable, like two sets of state vectors that are drawn from the uniform distribution over the state set. However readout neurons can be trained to recognize their inherent structural similarities, and are therefore able to classify also novel examples drawn from these classes. In order to demonstrate that their performance, which improves for larger sizes of the recurrent circuit, is due to this hidden structural similarity, and not to other scaling effects, the experiment were repeated with the same number of states drawn from a uniform distribution over the state set of the corresponding recurrent circuits (with a random assignment of class labels). Panel d of Fig. 2 shows that the readout can still be trained to classify states in the training set with an error of less than 50% (but much larger than for the previously considered classes). Furthermore this error on the training set decreases with the circuit size. But in the case

---

<sup>1</sup> Further work is needed to explore when exactly this occurs. It appears to depend both on values of parameters of the circuit (e.g.  $\lambda$ ) and on the type of inputs and the number of training examples.

of such randomly labeled state sets there is no generalization possible, hence the performance on test data from the same distribution yields an error of 50%.

The test that was applied here provides a generally applicable method for quantifying the characteristic inherent similarity of states within each of two classes  $A$ ,  $B$  of liquid states, even in cases where this inherent similarity of states can not be detected by a human observer. This method proposes to compare the classification performance of readouts that were trained to classify states from these two classes with that of the same type of readouts (using the same training algorithm) trained to classify states from two classes  $C$ ,  $D$  of the same size, whose elements were drawn from the uniform distribution over the state space (with randomly assigned class labels). The classification performance on these two other classes will in general also be better than random guessing, since the readout can store information about these particular two sets  $C$  and  $D$  in its weights. Hence its classification performance will improve with the number of weights in the readout, and hence with the size of the circuit. However if there is some structural similarity among states within one of the two classes  $A$ ,  $B$ , a trained readout neuron achieves a much higher performance for classification of states from these two classes. The difference in error rates achieved for the classification of states from the two original classes  $A$ ,  $B$  and the two classes  $C$ ,  $D$  quantifies how much common structure the readout can extract from each of the two original classes  $A$ ,  $B$  of liquid states.

In order to test the robustness of the classification capabilities of a readout neuron with regard to additional noise in the circuit we added Gaussian noise with mean 0 and standard deviation 0.5 nA and 1 nA respectively to the input currents of each neuron in the circuit at each simulation time step. This corresponds to Gaussian noise on the membrane potentials with a std of  $1/52$  ( $1/26$ ) of the voltage difference between the threshold and the membrane potential for a constant background current of 13.5 nA (reset voltage). As illustrated in figure 3 the classification error increased for the higher noise level, whereas the results for the lower noise level were comparable to those without this extra noise.

### 3.2. Finding Structure in Complex High Dimensional Trajectories of Circuit Dynamics

If one tracks the  $n$ -dimensional trajectory defined by the states of a recurrent circuit of  $n$  I&F neurons as a function of time, this trajectory is likely to resemble Brownian motion. However, from the point of view of readout neurons the same trajectory may have a simple and clear structure, and even converge to a "virtual attractor". The input to the recurrent circuit consists in the following always of 8 spike trains that are simultaneously injected into the circuit.

We focus on the information that *pairs* of readout neurons can extract from the high dimensional trajectory of liquid states of a recurrent neural circuit. The time course of the synaptic input to such readout neurons<sup>2</sup> is plotted as a curve in the plane, such as illustrated in Fig. 4 a. The approximate structure of this 2-dimensional curve is captured by the resulting spike trains of these 2 readout neurons (see Fig. 4 b, c), and can therefore be transmitted to other neural circuits.

In Figure 5 the input to two readout neurons is plotted in this way for three different time varying inputs to a recurrent circuit consisting of 135 I&F neurons. Depending on the choice of synaptic weights for these readout neurons, these 2-dimensional trajectories may look like Brownian motion (Fig. 5 a), trajectories that move fast into different attractor basins (Fig. 5 b), smooth trajectories with a characteristic dynamical structure (Fig. 5 c), or trajectories that are smooth and move into different attractor basins (Fig. 5 d). The first output (Fig. 5 a) is the typical result if the weights of the two readout neurons are randomly chosen. If one chooses the weights of these two readout neurons according to the Fisher discriminant [Duda et al., 2001], the resulting curves move for these three time-varying inputs to three different "virtual attractor basins". With a slight

---

<sup>2</sup> More precisely: the total contribution of the neurons in the recurrent circuit to the membrane potential of these two readout neurons, with an assumed membrane time constant of 30 ms.

variation of the Fisher discriminant (apply it to the sets of 2<sup>nd</sup> order derivatives with regard to time for points on these three trajectories, not to the sets of points on these trajectories) one gets the outputs of the two readout neurons that define the curves shown in Figure 5 c. These curves are quite smooth and exhibit a clear temporal evolution. By applying the Fisher discriminant to the union of points on the trajectories and their second derivatives with regard to time, one gets responses of the two readout neurons that combine the effects of Fig. 5 b and c: they move on smooth curves to different attractor basin (shown in Fig. 5 a). Although the Fisher discriminant is usually only viewed as a global optimization procedure, the resulting setting of the weights of the two readout neurons can also be approximated by an incremental learning algorithm: the MSE-algorithm, see section 5.8.2. in [Duda et al., 2001], which is local and unsupervised, and therefore not unrealistic from the biological point of view. For the simulations reported in this article we used the exact implementation of the Fisher discriminant. The results of Fig. 5 show that the low dimensional trajectory extracted by 2 readout neurons from a fairly large neural circuit may have little visible structural similarity with the high dimensional trajectory of transient states of that circuit, and may even move to "virtual attractors" that are not apparent from the high dimensional trajectory.

Analogously as with the classification task considered for Fig. 2, there exists an interesting scaling law, which prevents the observation of these effects in 2- or 3-dimensional dynamical systems, or in small models for neural circuits. The capability of pairs of readout neurons to extract smooth 2-dimensional trajectories from the very complex trajectory of firing activity in a recurrent circuit of I&F neurons increases with the number of neurons in this circuit (Fig. 6). The same input spike train was injected into recurrent neural circuits of varying size, and for each recurrent circuit a pair of readout neurons was optimized (as for Fig. 5 c) to generate a smooth 2-dimensional trajectory of their synaptic input. In order to show that the increased smoothness of the projections depends on the size of the network and is not due to a slow down of its dynamic the spike rasters with the responses of 10 randomly chosen neurons of each recurrent circuit to input as used for the simulations for figure 6 are illustrated in figure 7.

Figure 8 shows that the smooth large scale structure that pairs of trained readout neurons can extract from a complex dynamics of high dimensional circuits, may generalize to input where one of the eight simultaneously injected spike trains was replaced by a random Poisson spike train at each trial. This additional input may represent independent spatio-temporal information about the environment received from other areas in the neocortex. Two readout neurons were trained to respond with smooth trajectories of similar shape to the quite diverse high dimensional dynamics caused by this random spike train. In other words the readout was trained to assign equivalence classes in the internal state space, which contained all possible internal states at time point  $t$  that could result from different previously injected random spike trains. After training the readout neurons transformed the trajectories of liquid states that resulted from input composed of the same seven fixed spike trains but a previously not seen random spike train into closely related 2-dimensional projections (panels b-e). However their response is still highly selective and the trajectory of liquid states caused by entirely different input to the recurrent circuit (with the same firing rate) induces a completely different temporal evolution of membrane potentials in the readout (panel f).

### **3.3 Different Readouts may Create Diverse Virtual Attractor Landscapes**

We showed that different readout neurons can be trained to extract a diverse set of features from the same high dimensional neural activity of a sufficiently large recurrent circuit of I&F neurons. Hence for a specific information processing task it may not be necessary to manipulate this high dimensional trajectory itself. Instead, the number of degrees of freedom for readout neurons are chosen so large, that they can be trained to extract individualized smooth paths on virtual attractor landscapes from the same high dimensional circuit dynamics.

We call these attractors “virtual”, because they are not real attractors of the underlying dynamics but just look like attractors from the perspectives of certain low dimensional

projections. Furthermore these attractors are transient, i.e. they represent temporary attractors formed within the transient behaviour of the system. In other words: they are sets of states that attract certain trajectories during a certain time segment, but not permanently. Nevertheless, they may represent the result of a computation for a low dimensional readout. Hence the presence of virtual attractors makes it in principle possible to carry out particular computations needed by specific readouts without changing the dynamics of the recurrent circuit itself (thereby leaving it ready to serve as analog memory for other readouts with completely different tasks). The remarkable flexibility that remains when just the low-dimensional readouts are adapted for specific computational tasks is demonstrated in Figure 9 for 3 different pairs of readout neurons. For all three panels of Fig. 9 the inputs to the recurrent circuit (and the resulting circuit dynamics) are identical. However the temporal evolution of the readout responses has a different large-scale structure for each of the three pairs of readout neurons. Inputs to the recurrent circuit were three different spike trains A, B, C. The first pair of readout neurons was just trained to separate the trajectories resulting from these 3 inputs by smooth responses (panel a). The second pair was trained in addition to create a virtual common attractor for patterns A and B, but not for C (panel b). The third pair of readout neurons was trained to move the responses for patterns A and C to a common attractor, while keeping the trajectory for pattern B away from this attractor. Altogether Fig. 9 suggests an alternative to modeling neural dynamics by low-dimensional attractor neural networks: While the internal dynamics of a generic high-dimensional neural microcircuit may be extremely complex, different pools of readout neurons may use this high dimensional dynamics as a universal source of online information, and can be trained to extract low dimensional trajectories that move on virtual attractor landscapes. Since these virtual attractor landscapes may differ from readout to readout, one arrives in this way at a possible scheme for parallel real-time processing with the help of high dimensional dynamical systems.

#### **4.Summary**

Many tools and concepts that have been developed for the investigation of dynamical systems are very useful for analyzing low dimensional autonomous dynamical systems. However new effects have to be taken into account when one analyzes high dimensional dynamical systems such as those implemented by cortical microcircuits. Typically these systems are constantly bombarded with inputs from sensory neurons and other neural circuits, hence they are non-autonomous. Furthermore these systems have to compute in real-time, and therefore need to retrieve information for their computational tasks from trajectories of transient states of the circuit. Since the high dimensionality of the neural dynamics increases the capability of a readout neuron to select and represent specific components of the information, the high dimensional trajectory need not be engineered for a specific task. Rather, different readout neurons can extract completely different aspects for their specific task. In fact, from their point of view the high dimensional dynamics may even appear to move towards well-defined attractor basins, but this virtual attractor landscape may be a completely different one for each readout neuron.

Another beneficial aspect of the high dimensionality of the dynamics of neural microcircuits is the resulting boosting of the classification power of single readout neurons, which has been demonstrated in this article. This implies that very simple and robust learning algorithms, which cannot get stuck in local minima, can be used to train these readouts. The effects exhibited in this article may help to provide challenges and ideas for the development of a new theory of dynamical systems that is adequate for high dimensional non-autonomous systems with diverse components, and can therefore be used to analyze real-time computing in neural microcircuits.

**Acknowledgements:** We would like to thank Thomas Natschläger for valuable discussions, and Toni Schwaighofer for providing his code for voted perceptrons. Stefan Häusler has been partially supported by the Technische Universität Graz. Stefan Häusler and Wolfgang Maass have been partially supported by the Fonds zur Förderung der wissenschaftlichen Forschung in Austria (FWF), project P 15386.

## References

[Auer et al., 2002] Auer, P., Burgsteiner, H. and Maass, W. (2002). Reducing communication for distributed learning in neural networks. In José R. Dorronsoro, editor, Proc. of the International Conference on Artificial Neural Networks -- ICANN 2002, volume 2415 of Lecture Notes in Computer Science, pages 123-128. Springer, 2002.

[Duda et al., 2001] Duda, R. O., Hart, P. E., & Storck, D. G. (2001). *Pattern Classification*. New York: Wiley.

[Freund et al., 1999] Freund, Y., & Schapire, R. E. (1999). Large Margin Classification Using the Perceptron Algorithm. *Machine Learning* 37(3): 277-296

[Gupta et al., 2000] Gupta, A., Wang, Y., & Markram, H. (2000). Organizing principles for a diversity of GABAergic interneurons and synapses in the neocortex. *Science* 287, 2000, 273-278.

[Hertz et al., 1991] Hertz, J., Krogh, A., & Palmer, R. G. (1991). *Introduction to the Theory of Neural Computation*. (Addison-Wesley, Redwood City, Ca).

[Jaeger, 2001] Jaeger, H. (2001). The "echo state" approach to analyzing and training recurrent neural networks. Submitted for publication.

[Maass et al., 2002] Maass, W., Natschlaeger, T., & Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14(11):2531-2560, 2002.

[Markram et al., 1998] Markram, H., Wang, Y., & Tsodyks, M. (1998). Differential signaling via the same axon of neocortical pyramidal neurons. *Proc. Natl. Acad. Sci.*, 95, 5323-5328.

[Tsodyks et al., 2000] Tsodyks, M., Uziel, A., & Markram, H. (2000). Synchrony generation in recurrent networks with frequency-dependent synapses. *J. Neuroscience*, Vol. 20 RC50.

[Vapnik, 1998] Vapnik, V. N. (1998). *Statistical Learning Theory*. New York: Wiley.

[Warmuth et al., 2002] Warmuth, M. K., Rätsch, G., Mathieson, M., Liao, J., & Lemmen, C. (2002). Active learning in the drug discovery process. In Dietterich, T.G., Becker, S., & Ghahramani, Z., editors, *Advances in Neural information processing systems*, Vol 14. In press.

## Legends

**Figure 1:** Input distribution for the classification task. For each experiment four Poisson spike trains were randomly generated and fixed as spike pattern templates 1-4. Input spike trains over 500 ms were randomly composed of noise variations (Gaussian distribution with mean 0,  $SD$  4 ms) of one of the templates 1, 2 in their first half followed by noisy variations of one of the templates 3, 4 in their second half. An input spike train was defined to belong to class 1(2) if its first half was generated from the template spike train 1 (2), no matter whether its second half had been generated from. Three typical noisy variations of class 1 are shown in the lower part of the figure.

**Figure 2:** Demonstration that a larger recurrent neural circuit increases the classification power of a readout neuron. **a:** The readout was trained to carry out a classification of the previously injected input time series as described in Fig. 1 by means of the internal state of the recurrent network at time  $t = 500$  ms (after the complete spike train had been injected into recurrent circuit). The solid and dashed lines show for training and test inputs the error rates of a hyperplane (= perceptron) in the  $n$ -dimensional state space of the recurrent circuit, trained with the well-known delta learning rule (see [Hertz et al., 1991]), as a function of the size  $n$  of the recurrent circuit of I&F neurons. For each value of  $n$  100 randomly drawn recurrent circuits were generated. For each circuit a new input distribution was fixed and 800 input spike trains from this distribution were chosen for training, whereas 80 novel examples were used for testing. The error rates represent the averages over the performance of different circuits (error bars indicate  $SEM$ ). **b:** Corresponding results for the classification task applying the Fisher-discriminant algorithm (see [Duda et al., 2001]) instead of the delta learning rule to optimize the weights of a perceptron. The error rates were almost the same for this algorithm (whereas its computing time was much shorter). **c:** Corresponding results for the classification task applying multi-unit readouts instead of perceptrons: p-delta learning rule (using an array of 10 perceptrons, see [Auer et al., 2002]), backpropagation (applied to 10 feedforward sigmoidal neurons), and voted perceptrons (using an array of 10 perceptrons) implemented as outlined in [Warmuth et al., 2002]. The error rates were almost the same

as for perceptrons. **d**: Results of a control experiment. In order to show that a better readout performance for larger recurrent circuits can only be achieved if the complexity of the input stays constant (or grows at a lower rate than the circuit size), the same algorithms as in panel b were applied to compute an optimal separating hyperplane for randomly chosen internal states of the recurrent circuit. The training set consisted of 800 randomly drawn state vectors, drawn from the uniform distribution over all state vectors (with randomly assigned class labels) instead of 800 liquid states at time  $t = 500$  ms that resulted from injecting input spike trains from a fixed distribution into these circuits. The performance on these training sets decreased with the circuit size. But in the case of such randomly labeled state sets there was no generalization possible, hence the performance on test data from the same distribution (uniform distribution over all state vectors) yielded an error of 50%.

**Figure 3**: Results for the classification task of figure 2a with two levels of noise added to the input currents independently for all neurons in the circuit. For each simulation time step Gaussian noise with mean 0 and standard deviation 0.5 nA and 1 nA respectively was injected into the neurons. The classification error increased for the higher noise level, whereas the results showed little effect for the lower noise level .

**Figure 4: a**: A Typical 2-dimensional projection of the high dimensional trajectory of liquid states in a recurrent neural circuit, represented by the combined synaptic inputs from the neurons in this circuit to 2 readout neurons. The beginning of the trajectory is marked by a circle, with crosses on the curve at every 50 ms interval. **b, c**: Resulting spike trains of these 2 readout neurons.

**Figure 5**: Three trajectories of the synaptic input to 2 readout neurons (indicated by solid, dashed, and dotted lines) caused by three different input Poisson spike trains to the recurrent circuit during a time interval of 1 s. For randomly assigned synaptic weights these trajectories resemble Brownian motion as shown in panel **a**. For other values of the weights the same trajectories may appear well separated in space as illustrated in panel **b**, or smooth as displayed in panel **c**. For the calculations of the weights in both cases the

well-known Fisher discriminant algorithm was applied for panel b directly to the points of the trajectories and for panel c to their second time derivatives. A combination of both perspectives, enabling both classifications through clear spatial separation and smooth tracking of the trajectory (see panel d), can be achieved by applying the Fisher discriminant to the union of the two point sets used for panels b and c.

**Figure 6:** Dependence of the capability of 2 readout neurons to transform a trajectory of a recurrent circuit of  $n$  I&F neurons into a smooth low dimensional projection on the number  $n$  of neurons in the circuit. The input consisted of eight parallel injected random Poisson spike trains over a time interval of 2 s. In each case the same optimization method as for Fig. 5 c was applied to the weights of the 2 readout neurons.

**Figure 7:** Spike rasters with the responses of 10 randomly chosen neurons from the recurrent circuits of  $n$  I&F neurons, for a circuit input as used for the simulation for figure 6. The level of activity of these neurons does not diminish with circuit size. This shows that the increased smoothness of the low dimensional projections of the circuit trajectories shown in figure 6 is not the result of diminishing firing activity in the larger circuits.

**Figure 8: a)** The smooth large scale structure that pairs of trained readout neurons can extract from a complex dynamics of high dimensional circuits can generalize to novel input. One of eight fixed random input spike trains of length 0.5 s that were simultaneously injected into a recurrent circuit of 135 I&F neurons was replaced by a novel random Poisson spike train at each trial. The weights of the two readout neurons were chosen (similarly as in Fig. 5 c) in such a way that the trajectories have about the same shape from the point of view of these readout neurons. This generalized to previously not shown variations of the random input spike train (**b – e**). The trajectory caused by entirely different input to the recurrent circuit (with the same firing rate) induced a completely different temporal evolution of membrane potentials in the readout (**f**), what shows that the response of the two readout neurons was still highly selective.

**Figure 9:** Different pairs of readout neurons may create different virtual attractor landscapes. **a:** Two readout neurons were trained to respond to the 3 trajectories of liquid states caused by the injection of 3 different input spike trains during a time interval of 1 s to a recurrent circuit of 375 I&F neurons (in a 5x5x15 column) with smooth well-separated responses, like in Fig. 5c (beginnings of the 2 dimensional response-trajectories marked by circles). **b, c:** Responses of 2 other pairs of readout neurons to the same 3 trajectories of liquid states as in panel a. For panel b a pair of readout neurons were trained to move only for inputs *A* and *B* to a common attractor. For panel c another pair of readout neurons was trained to move only for inputs *A* and *C* a common attractor.

Figure 1

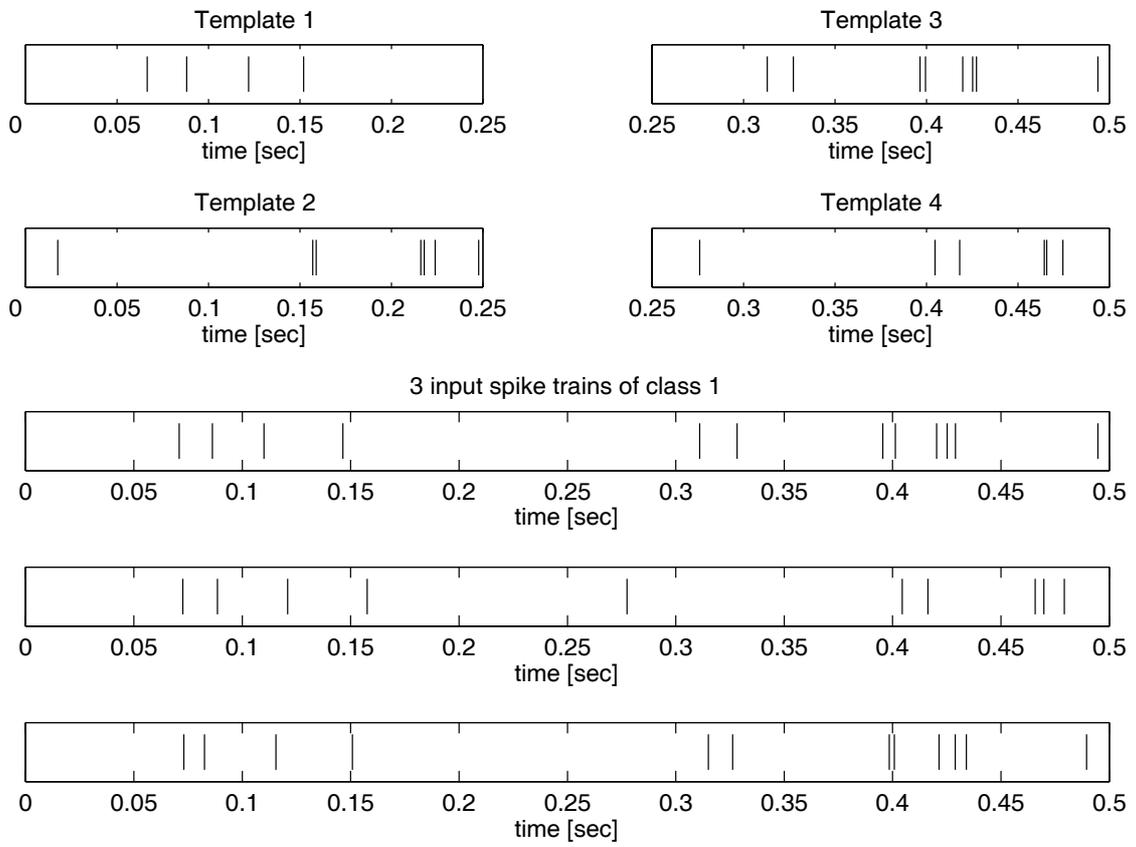
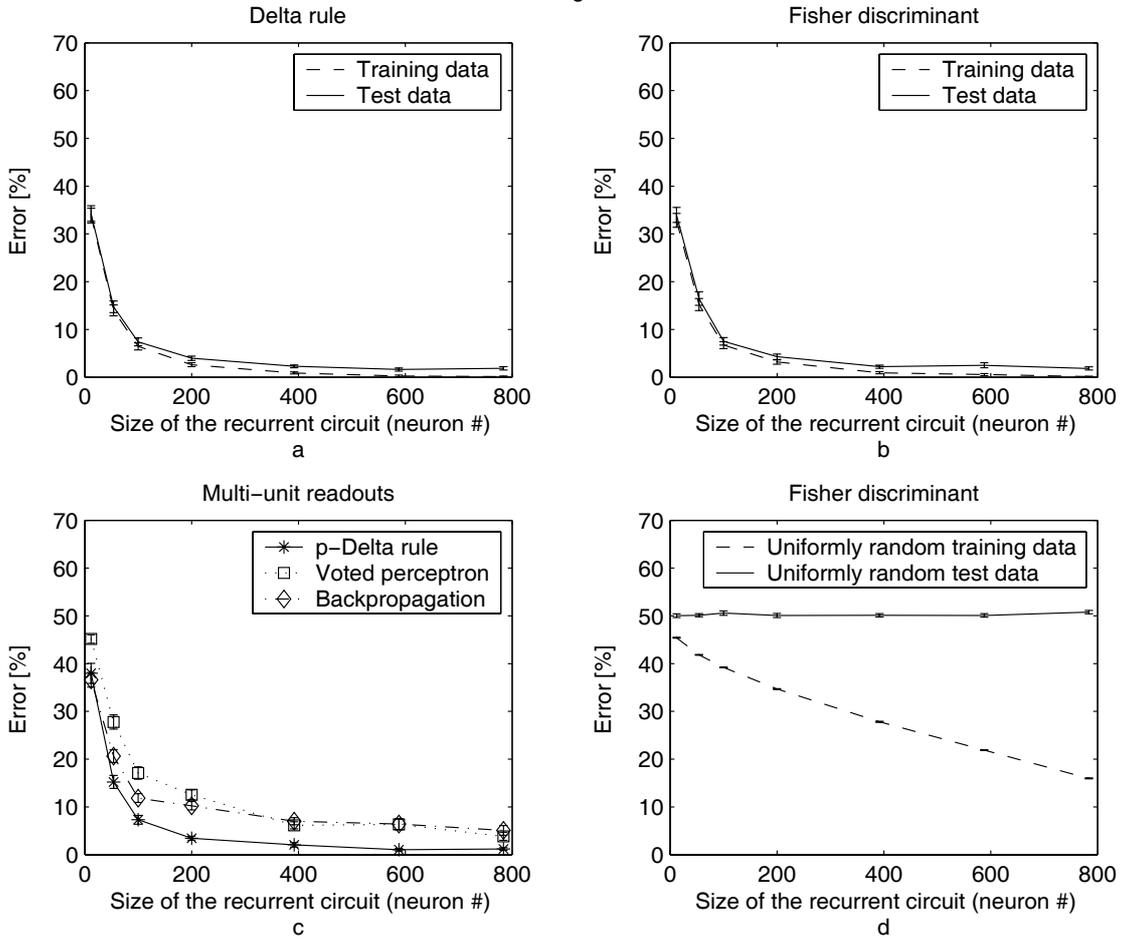


Figure 2



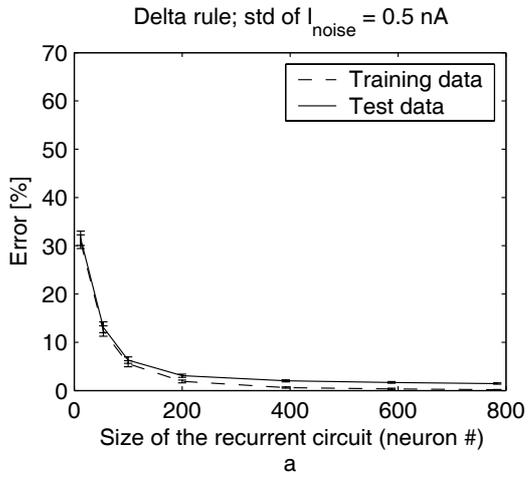


Figure 3

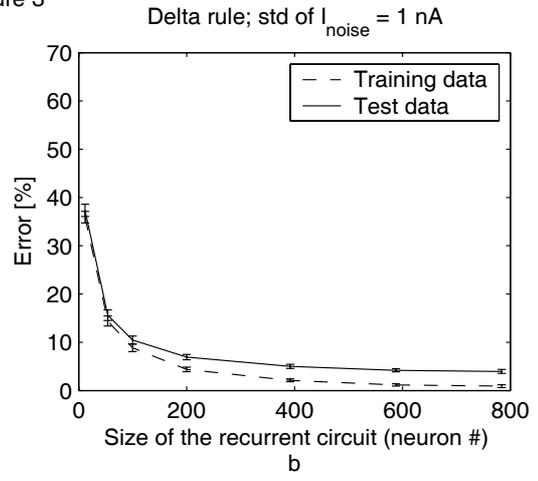


Figure 4

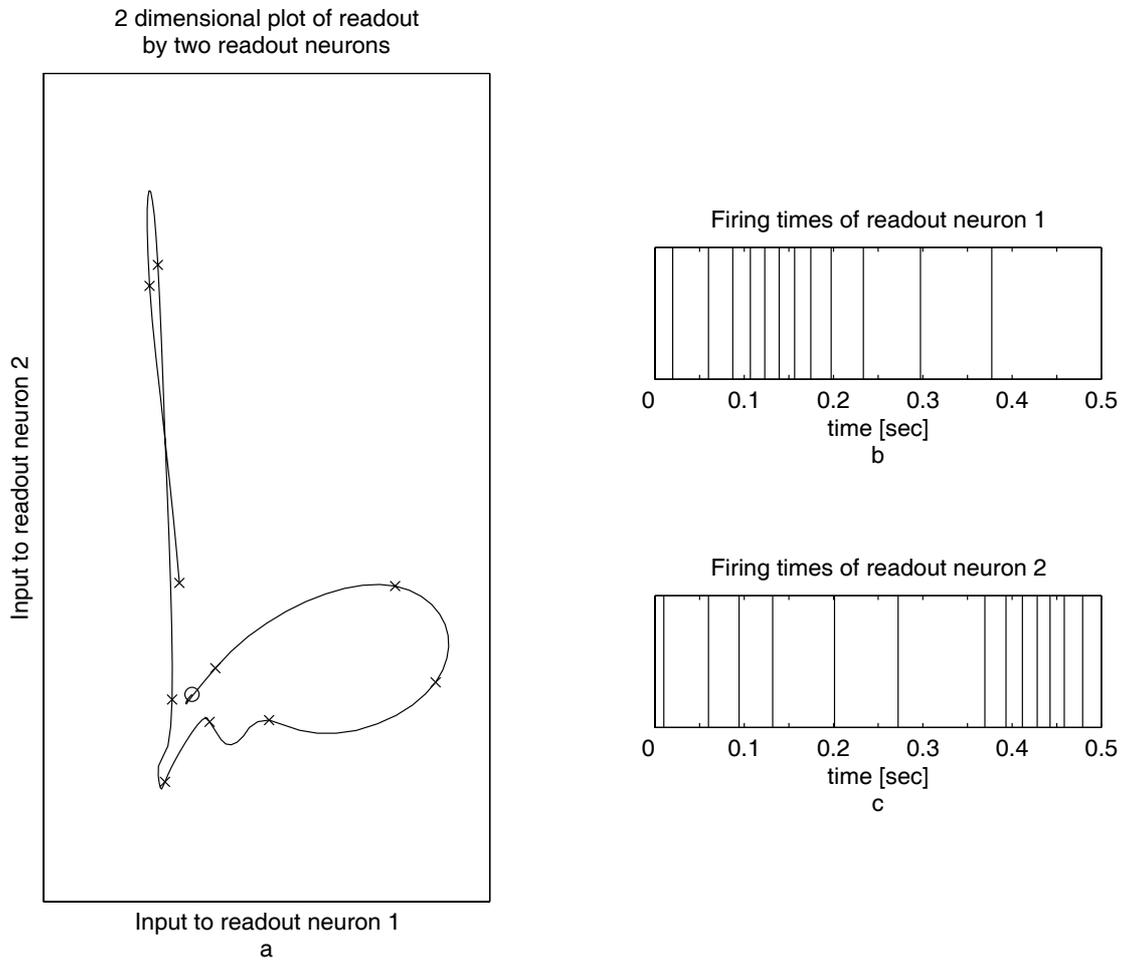
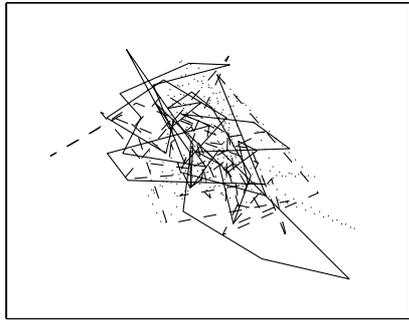
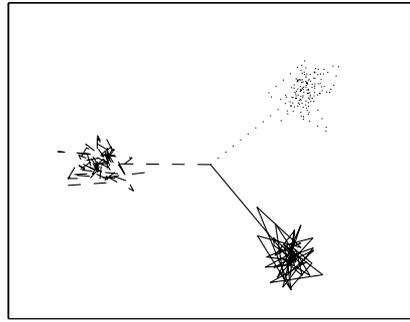


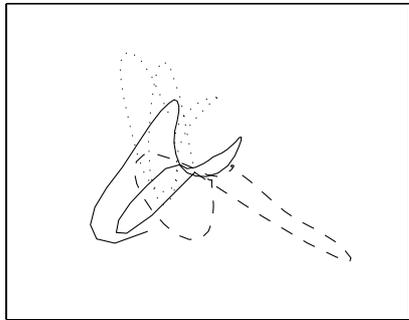
Figure 5



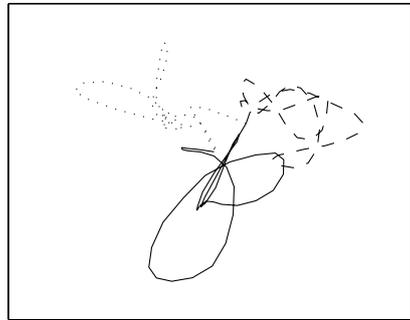
Untrained readout  
a



Readout trained for classification  
b

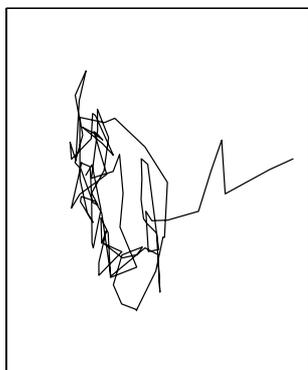


Readout trained to generate smooth trajectories  
c

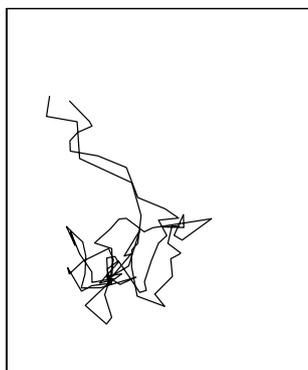


Readout trained for both  
d

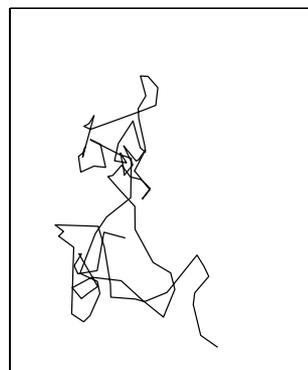
Figure 6



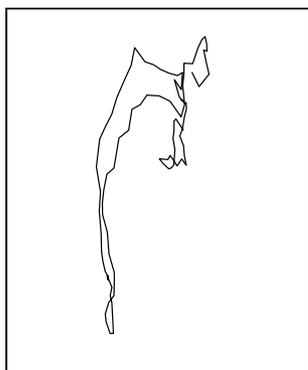
81 neurons  
a



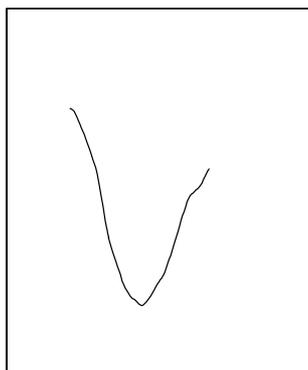
99 neurons  
b



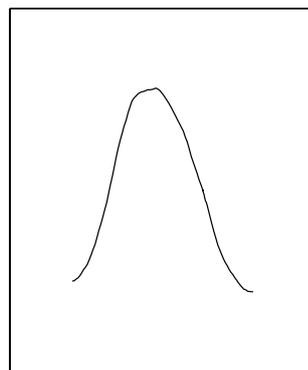
117 neurons  
c



135 neurons  
d



162 neurons  
e



189 neurons  
f

Figure 7

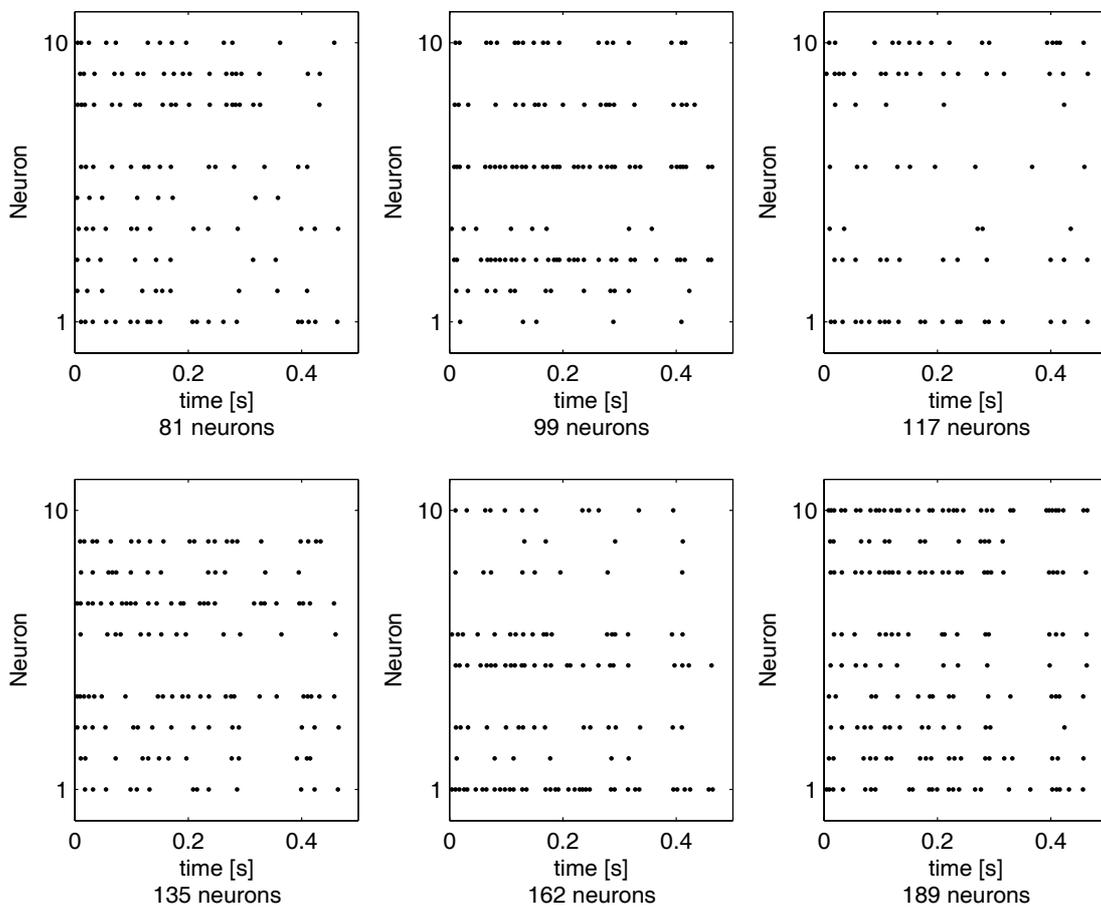
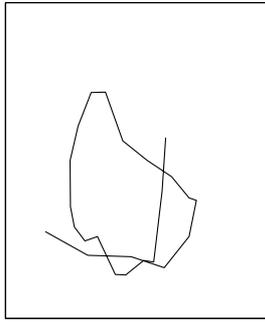


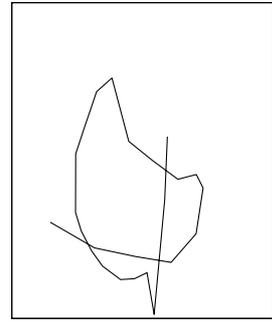
Figure 8



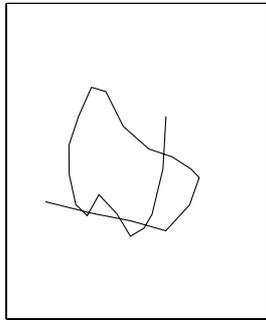
50 training spike train  
a



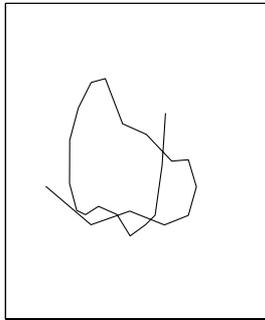
Test spike train 1  
b



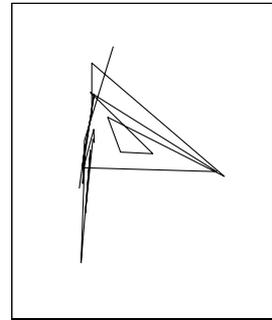
Test spike train 2  
c



Test spike train 3  
d



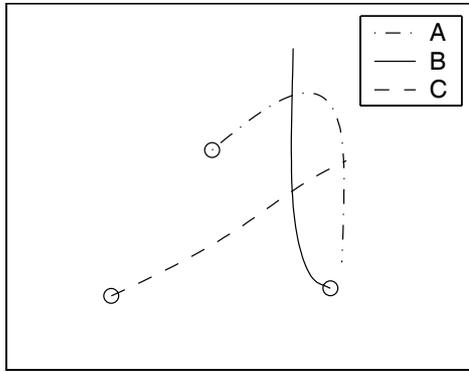
Test spike train 4  
e



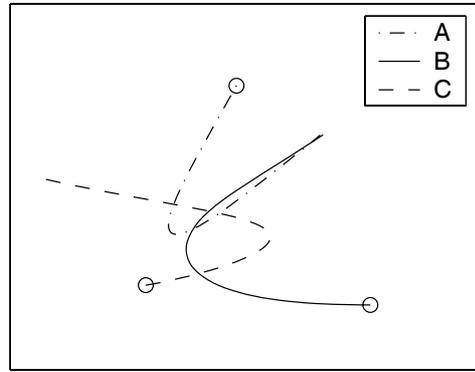
Random spike train  
f

Figure 9

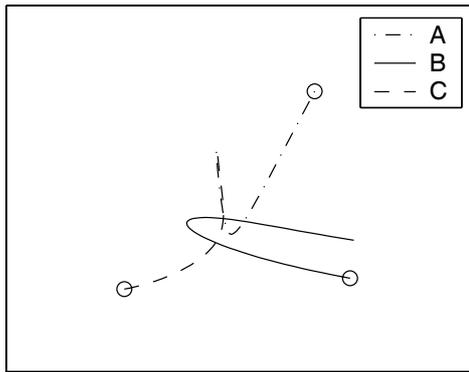
Virtual attractor landscapes



a



b



c