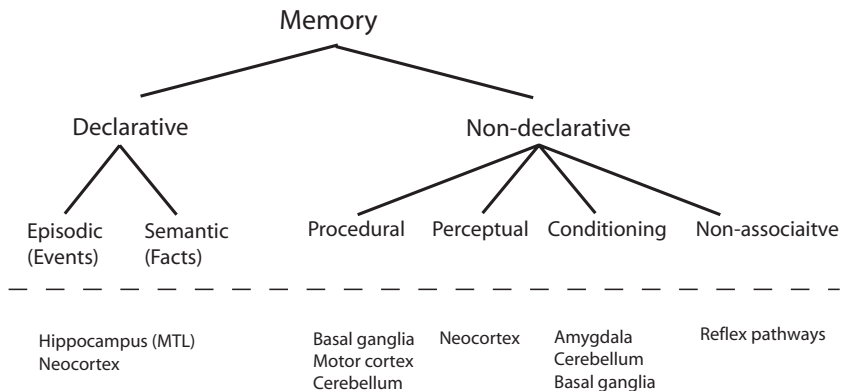


# Fundamentals of Computational Neuroscience 2e

December 31, 2009

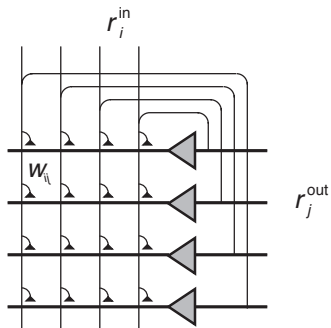
Chapter 8: Recurrent associative networks and episodic memory

# Memory classification scheme (Squire)

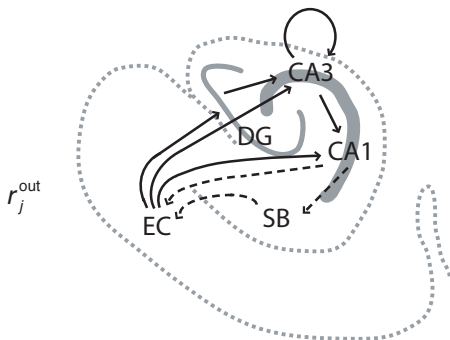


# Auto-associative network and hippocampus

A. Recurrent associator network



B. Schematic diagram of the Hippocampus



## Point attractor neural network (ANN)

**Update rule:**  $\tau \frac{du_i(t)}{dt} = -u_i(t) + \frac{1}{N} \sum_j w_{ij} r_j(t) + I_i^{\text{ext}}(t)$

**Activation function**  $r_i = g(u_i)$  (e.g. threshold functions)

**Learning rule**  $w_{ij} = \epsilon \sum_{\mu=1}^{N_p} (r_i^{\mu} - \langle r_i \rangle)(r_j^{\mu} - \langle r_j \rangle) - c_{ij}$

**Training patterns:** Random binary states with components

$$s_i^{\mu} \in \{-1, 1\}, r_i = \frac{1}{2}(s_i + 1)$$

**Update equations for fixed-point model** ( $du_i/dt = 0$ ):

$$s_i(t+1) = \text{sign} \left( \sum_j w_{ij} s_j(t) \right)$$

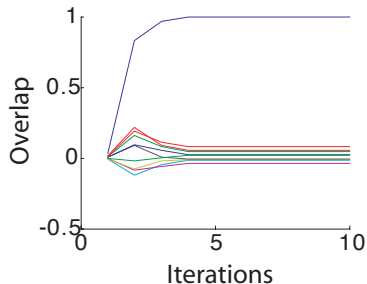
## ann\_cont.m

```
1  %% Continuous time ANN
2  clear; clf; hold on;
3  nn = 500; dx=1/nn; C=0;
4
5  %% Training weight matrix
6  pat=floor(2*rand(nn,10))-0.5;
7  w=pat*pat'; w=w/w(1,1); w=100*(w-C);
8  %% Update with localised input
9  tall = []; rall = [];
10 I_ext=pat(:,1)+0.5; I_ext(1:10)=1-I_ext(1:10);
11 [t,u]=ode45('rnn_ode_u',[0 10],zeros(1,nn),[],nn,dx,w,I_ext);
12 r=u>0.; tall=[tall;t]; rall=[rall;r];
13 %% Update without input
14 I_ext=zeros(nn,1);
15 [t,u]=ode45('rnn_ode_u',[10 20],u(size(u,1),:),[],nn,dx,w,I_ext);
16 r=u>0.; tall=[tall;t]; rall=[rall;r];
17 %% Plotting results
18 plot(tall,4*(rall-0.5)*pat/nn)
```

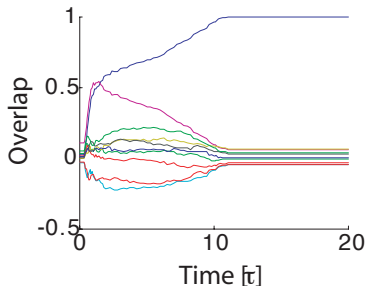
# ann\_fixpoint.m

```
1 pat=2*floor(2*rand(500,10))-1; % Random binary pattern
2 w=pat*pat'; % Hebbian learning
3 s=rand(500,1)-0.5; % Initialize network
4 for t=2:10; s(:,t)=sign(w*s(:,t-1)); end % Update network
5 plot(s'*pat/500)
```

A. Fixpoint ANN model

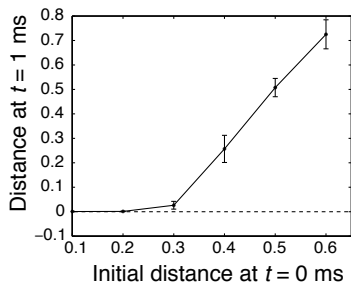


B. Continuous time ANN model

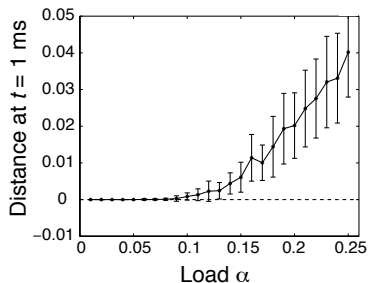


# Memory breakdown

A. Basin of attraction



B. Load capacity



# Probabilistic RNN

## Probabilistic update rule:

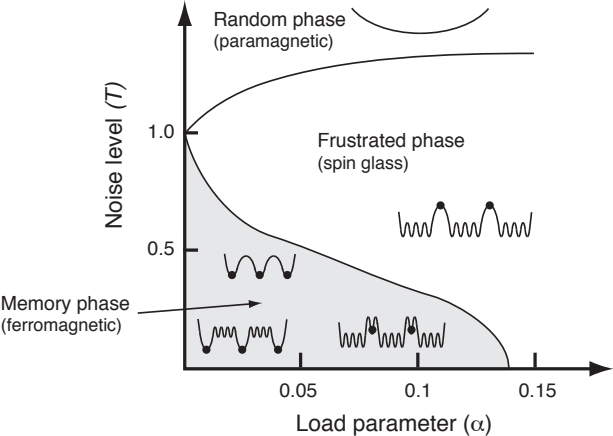
$$P(s_i(t) = +1) = \frac{1}{1 + \exp(-2 \sum_j w_{ij} s_j(t-1) / T)}$$

Recovers deterministic rule in  $\lim_{T \rightarrow 0}$

$$s_i(t) = \text{sign}\left(\sum_j w_{ij} s_j(t-1)\right)$$

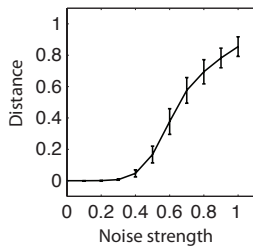


# Phase diagram

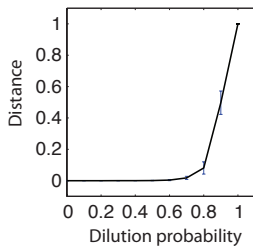


# Noisy weights and diluted attractor networks

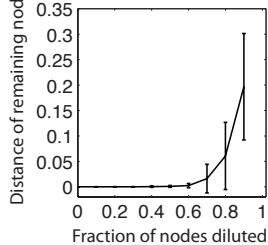
A. Noisy weights



B. Diluted weights



C. Diluted nodes



## How to minimize interference between patterns?

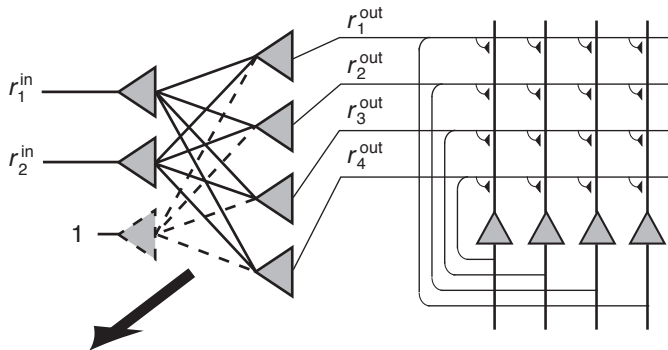
Associative memory in ANN is strongly influenced by interference between patters due to

- ▶ correlated patterns
- ▶ random overlap

Storage capacity can be much enhanced through decorrelating patterns. Simplest approach is generating sparse representations with expansion re-coding.

**Storage capacity:**  $\alpha_c \approx \frac{k}{a \ln(1/a)}$  (Rolls & Treves)

## Expansion re-coding (e.g. in dentate gyrus)

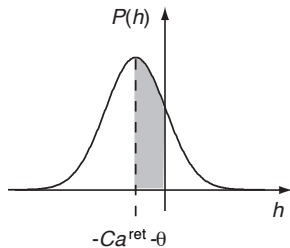


$$\begin{pmatrix} -1 & -1 & 0.5 \\ 1 & -1 & -0.5 \\ -1 & 1 & -0.5 \\ 1 & 1 & -1.5 \end{pmatrix}$$

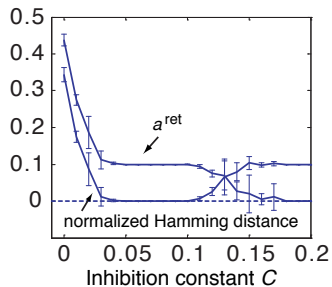
$r_1^{in}$	$r_2^{in}$	$r_1^{out}$	$r_2^{out}$	$r_3^{out}$	$r_4^{out}$
0	0	1	0	0	0
1	0	0	1	0	0
0	1	0	0	1	0
1	1	0	0	0	1

# Sparse pattern and inhibition

A. Probability density



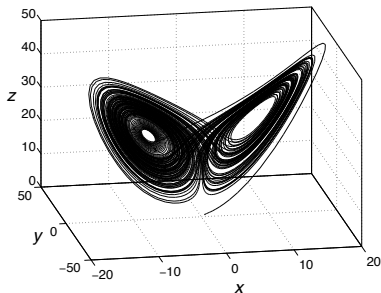
B. Sparse ANN simulation



## Three nodes with $\Sigma$ and $\Sigma - \Pi$ coupling (Lorenz attractor):

$$\frac{dx_i}{dt} = \sum_j w_{ij}^1 x_j + \sum_{jk} w_{ijk}^2 x_j x_k$$

$$\mathbf{w}^1 = \begin{pmatrix} -1 & a & 0 \\ b & -1 & 0 \\ 0 & 0 & -c \end{pmatrix} \quad \text{and} \quad \mathbf{w}^2 = \begin{cases} w_{213}^2 = -1 \\ w_{312}^2 = -1 \\ 0 \text{ otherwise} \end{cases}$$



## Cohen–Grossberg theorem

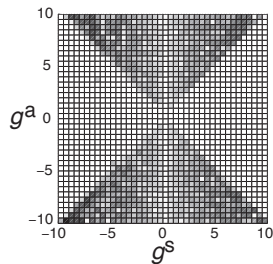
Dynamical system of the form  $\frac{dx_i}{dt} = -a_i(x_i) \left( b_i(x_i) - \sum_{j=1}^N (w_{ij}g_j(x_j)) \right)$

Has a Lyapunov (Energy) function, which guaranties point attractors, under the conditions that

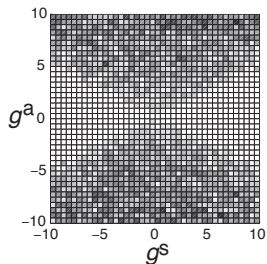
1. **Positivity**  $a_i \geq 0$ : The dynamics must be a leaky integrator rather than an amplifying integrator.
  2. **Symmetry**  $w_{ij} = w_{ji}$ : The influence of one node on another has to be the same as the reverse influence.
  3. **Monotonicity**  $\text{sign}(dg(x)/dx) = \text{const}$ : The activation function has to be a monotonic function.
- **more general dynamics possible with:**
- ▶ Non-symmetric weight matrix
  - ▶ Non-monotone activation functions (tuning curves)
  - ▶ Networks with hidden nodes

# Recurrent networks with non-symmetric weights

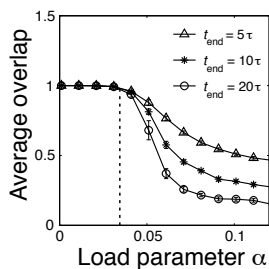
A. Unit components



B. Random components



C. Hebb–Dale network



→ **strong asymmetry is necessary to abolish point attractors**



## Further Readings

- Daniel J. Amit (1989), **Modelling brain function: the world of attractor neural networks**, Cambridge University Press.
- John Hertz, Anders Krogh, and Richard G. Palmer (1991), **Introduction to the theory of neural computation**, Addison-Wesley.
- Edmund T. Rolls and Alessandro Treves (1998), **Neural networks and brain function**, Oxford University Press
- Eduardo R. Caianello (1961), **Outline of a theory of thought-process and thinking machines**, in **Journal of Theoretical Biology** 2: 204–235.
- John J. Hopfield (1982), **Neural networks and physical systems with emergent collective computational abilities**, in **Proc. Nat. Acad. Sci., USA** 79: 2554–8.
- Michael A. Cohen and Steven Grossberg (1983), **Absolute stability of global pattern formation and parallel memory storage by competitive neural networks**, in **IEEE Trans. on Systems, Man and Cybernetics**, SMC-13: 815–26.
- MWenlian Lu and Tianping Chen, **New Conditions on Global Stability of Cohen-Grossberg Neural Networks**, in **Neural Computation**, 15: 1173–1189.
- Masahiko Morita (1993), **Associative memory with nonmonotone dynamics**, in **Neural Networks** 6: 115–26.
- Michael E. Hasselmo and Christiane Linster (1999), **Neuromodulation and memory function**, in **Beyond neurotransmission: neuromodulation and its importance for information processing**, Paul S. Katz (ed.), Oxford University Press.
- Pablo Alvarez and Larry R. Squire (1991), **Memory consolidation and the medial temporal lobe: a simple network model**, in **Proc Natl Acad Sci** 15: 7041-7045.