Connectionism, evolution, plasticity and the Brain

Prof.dr. Jaap Murre

University of Amsterdam

jaap@murre.com

http://neuromod.org

In simple animals, the ‘brain’ is determined completely by the DNA

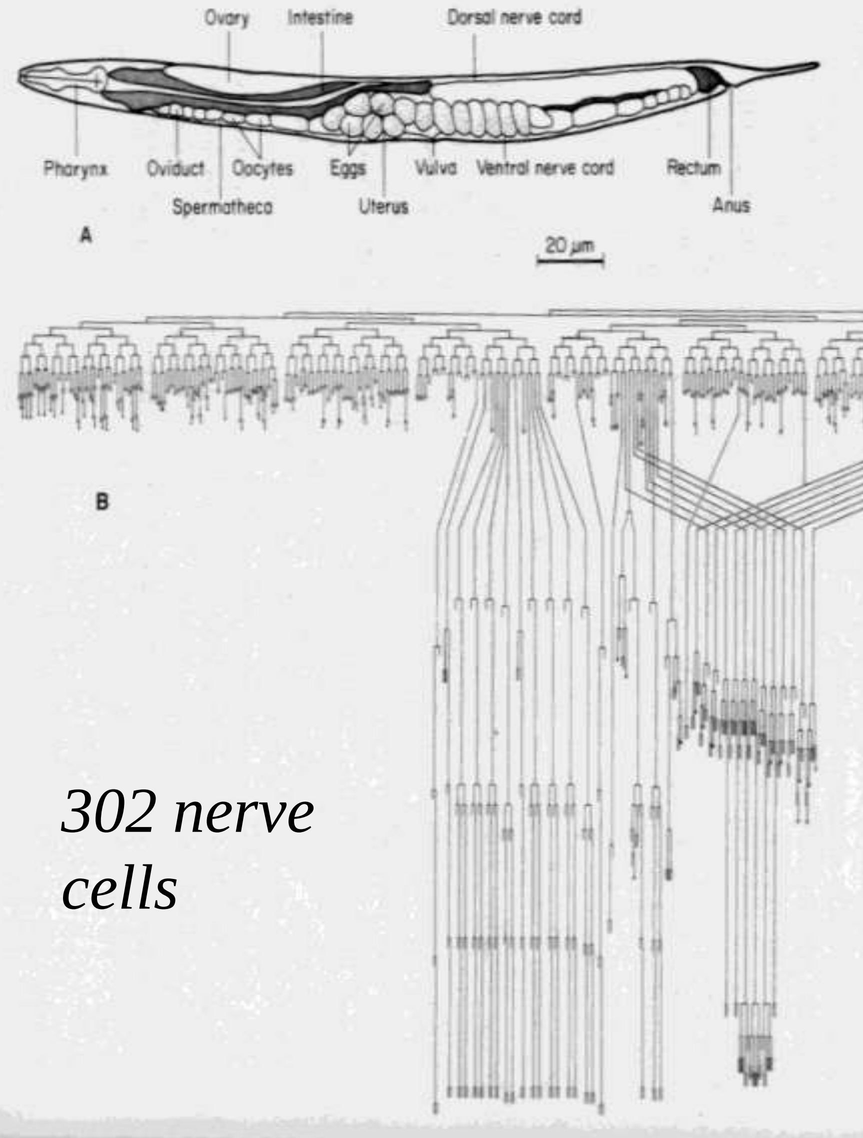
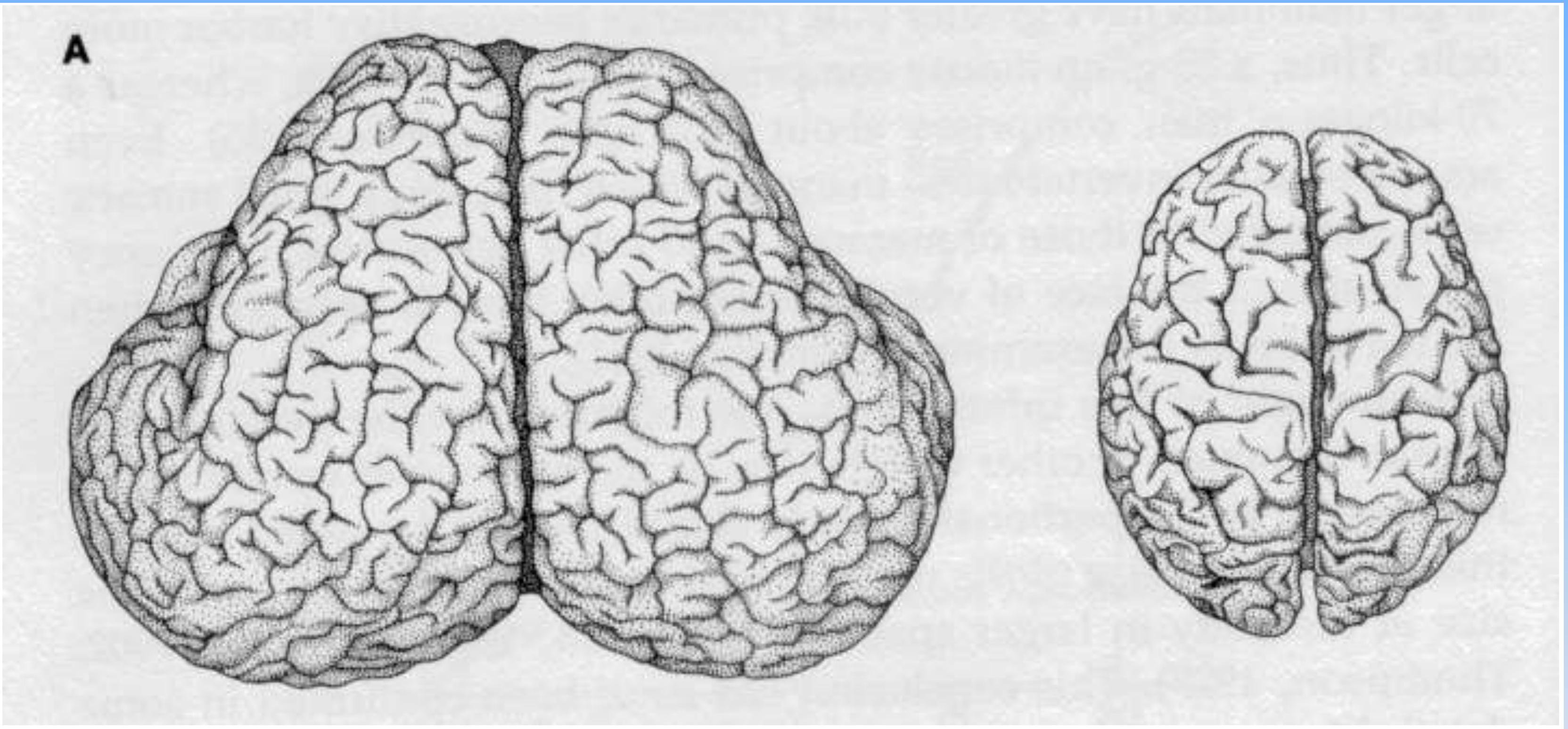


Figure 3.1. Determinate pattern of development in a small, simple animal, the nematode *C. elegans*. (*A*) The major features of the mature hermaphrodite. (*B*) The embryonic lineage, determined by microscopical observation of dividing cells in the embryo. The lineage reflects stereotyped sequences of cell division and differentiation, as well as the programmed death of particular cells. (*A* courtesy of R. Waterston; *B* from Sulston et al., 1983)

302 nerve cells

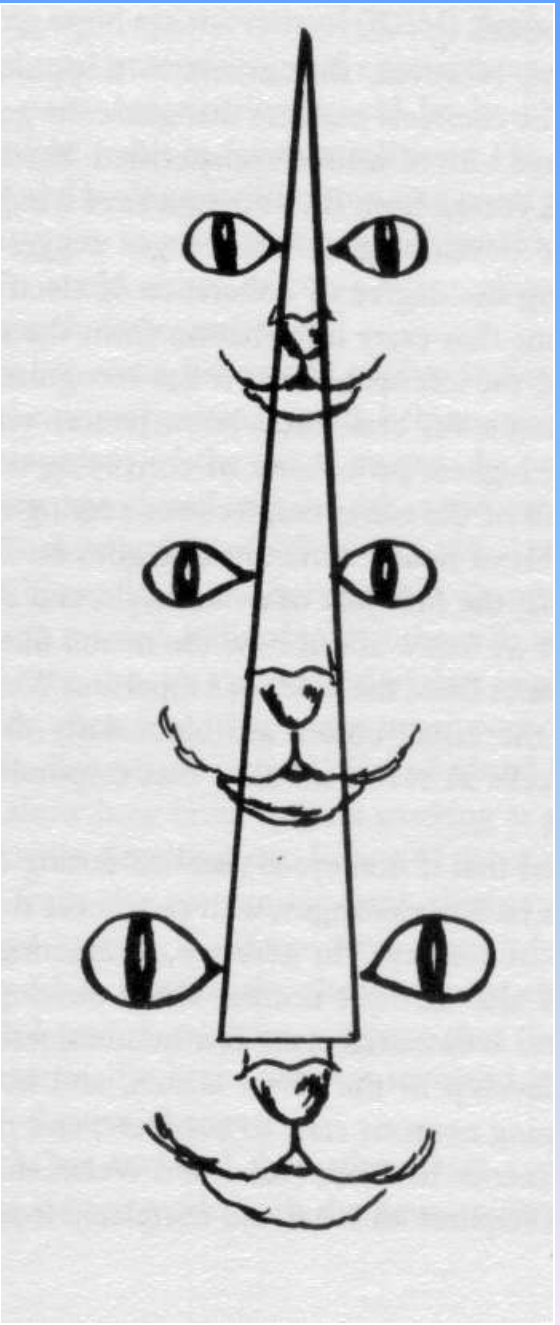
In larger animals other mechanisms of brain development must operate



Fin whale
(5 x human weight)

Human

The brain must remain plastic and adapt to the changing body during a lifetime



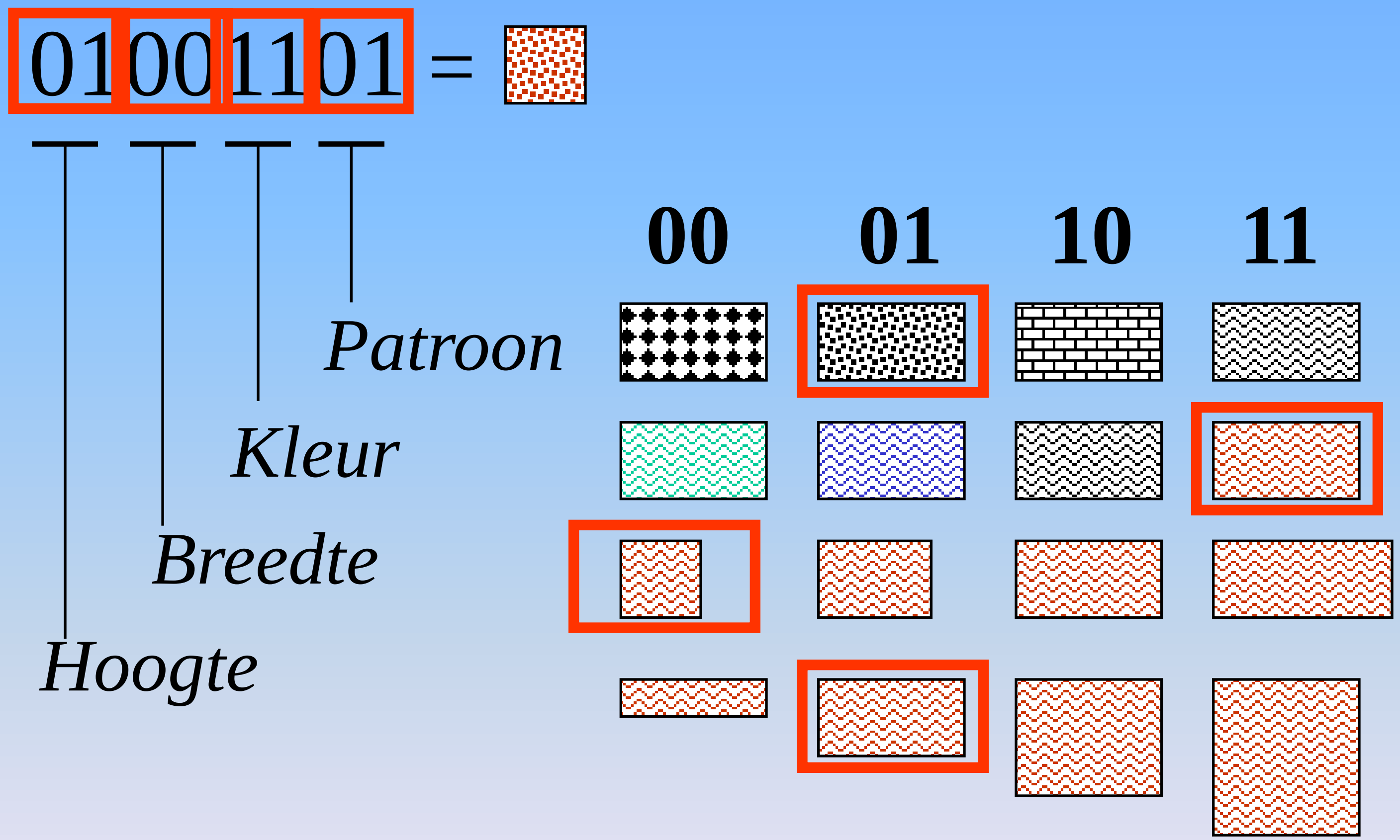
Plasticity and evolution

How the brain may come to encode important environmental constraints

Genetic algorithms: artificial evolution

- A solution is based on a binary string (cf. DNA)
  - For example: 010011011
- In nature a string is a coded building plan for an organism
- In (artificial) genetic algorithms anything at all can be encoded

Example of genetic coding



Populations of strings

- Genetic algorithms are based on populations of strings, for example:

01001101

00110101

01000101

00111010

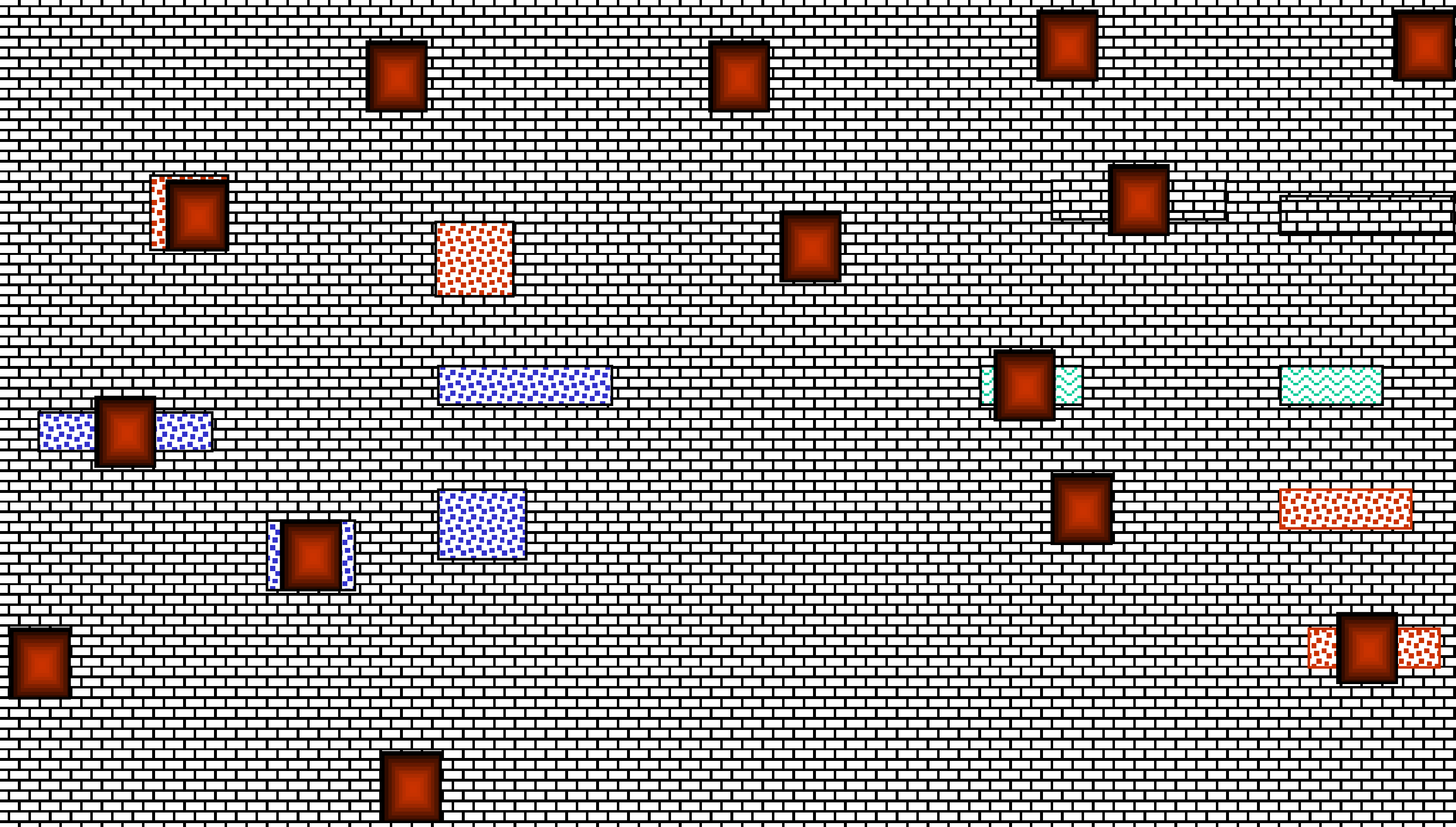
00010000

00101101

Fitness

- Every string has a certain fitness value
- For example: Fitness(010011011) = 0.67
- *Any* procedure may be used to arrive at a fitness value

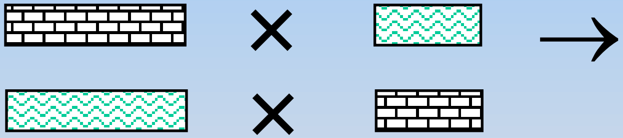
Example of a procedure to determine fitness: camouflage



Major genetic operators

- Reproduction and selection
- Cross-over
- Mutation
  - Mutation probability is usually very low

Cross-over

- **aaaaaaaa × bbbbbbbb →**
  aaaabbbb × bbbbaaaa
- **00111010 × 00010000 →**
  00110000 × 00011010
- 

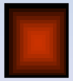
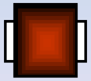
Sources of randomness in genetic algorithms

- Is *initial population* is generated randomly
- The *choice of operators* that will be applied is random
- Which *part of a string* will be selected for certain operators (e.g., mutation) is random

Genetic algorithms implement a *search process*

- The genetic search process works through selection of the fittest strings
- The process is driven by noise
- The process is structured through information exchange by the cross-over operator

Genetic building blocks

- Parts of a string that lead to good solutions are called schemata
  - In the example it was good to be short, so that **00***** is a schema
  - It is even better to also be not too high, so that 0000**** is is an even better schema
  - One could call this a ‘rock hiding schema’:  fits easily underneath , for example 

Color and pattern matter less for hiding under a ‘rock’

Short schemata are better than long ones and are called ‘building blocks’

- A short schema can better withstand cross-over and thus has a higher survival probability
  - For example, the schema **00**10 (short with stone pattern) is is very fit, but will easily be broken up
- Genetic algorithms are extremely efficient
  - in a population with 100 strings, about 1000.000 (n^3) schemata will be evaluated

The virtual creatures of Karl Sims

- Goal was to generate both the body and the brains of simple creatures
- Fitness was determined through specific tasks, such as swimming, walking or jumping
- The virtual creatures could not learn
- All aspects were artificial: their brain, their body, and their environment

Karl Sims' Virtual creatures

Large brains and evolution

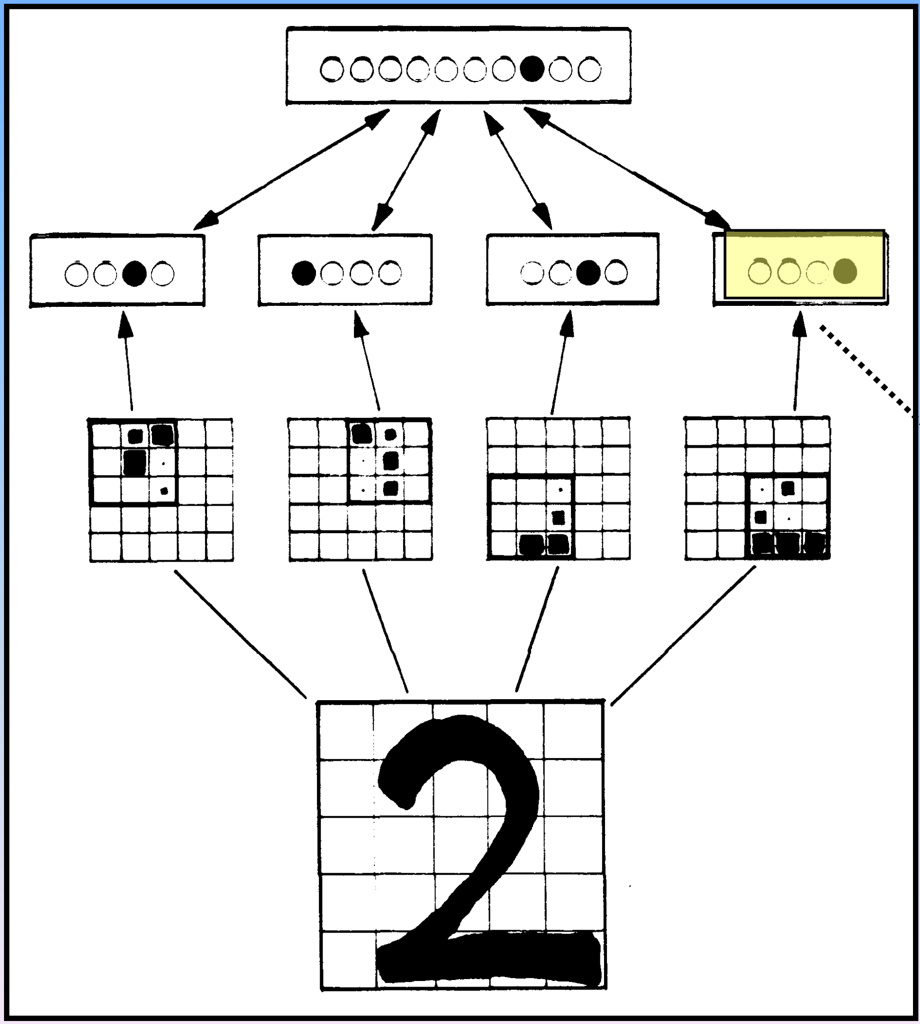
- Evolution determines the global or coarse structure of large brains
- Development carries out the DNA blue prints
- Even during development, the environment plays an important role
- Learning further imparts a fine structure on the coarse layout of the brain

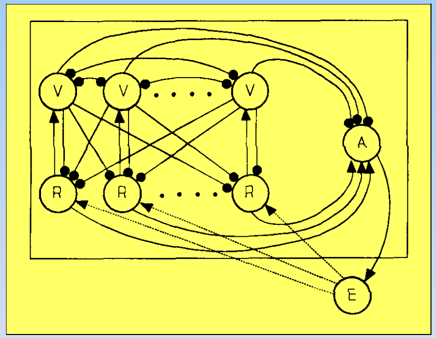
Example: The learning number recognizer by Happel and Murre

- Goal: Learning to recognize hand-written numbers:
  - Which architecture to use (with CALM modules)
  - Give examples so that the system can learn these (use a little bit of supervision)
  - Test the behavior with new, unseen examples. How well are they categorized?

What is a good architecture?

We achieved good recognition with a hand made architecture



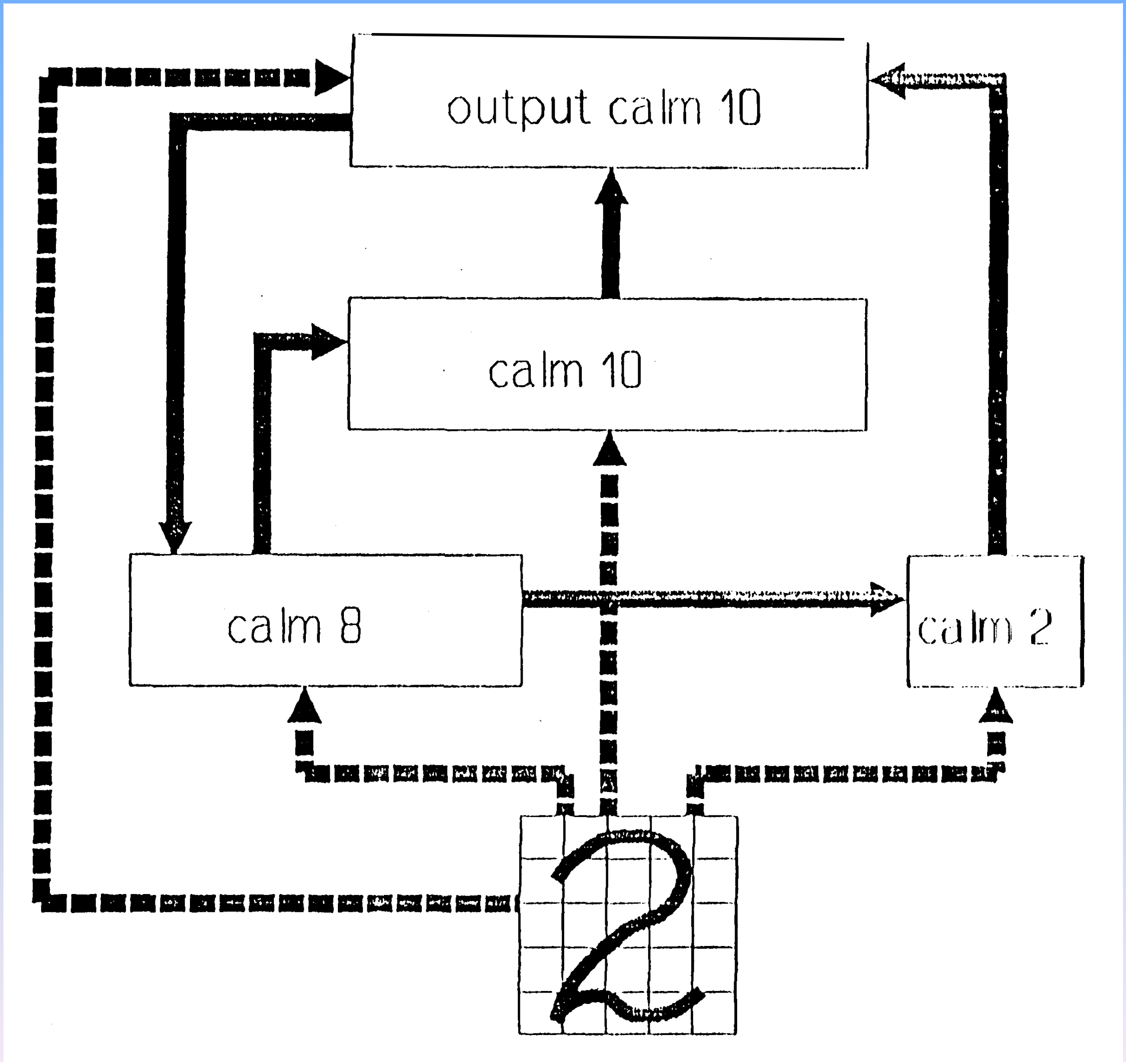


Architecture of a CALM module

Architecture found with a genetic algorithm

Much better architectures with a genetic algotihm were found

Many of these showed a broad similarity to the brain’s architecture



Selfrepairing neural networks

A framework for a theory of recovery from brain damage

Redundancy and repair

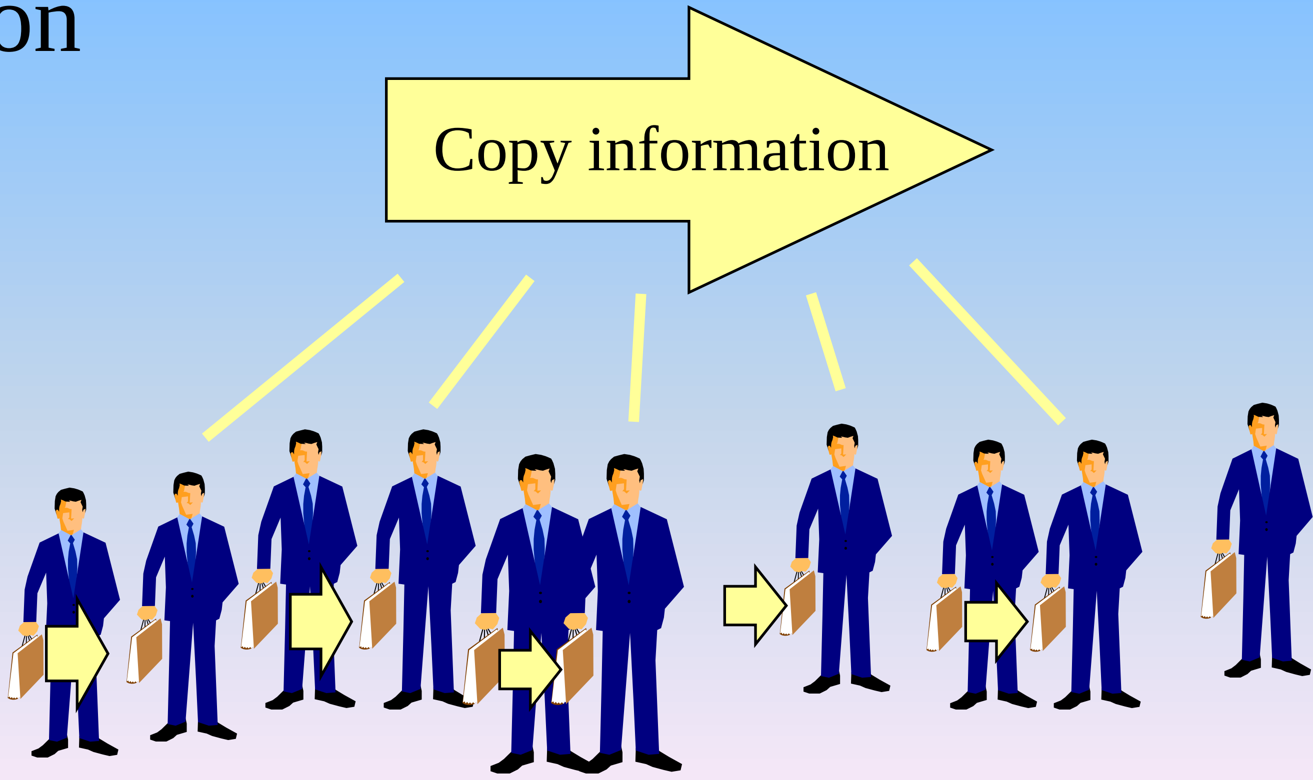
- *Redundancy* by itself does not guarantee survival
- Only a continuous repair strategy does
- Example: safeguarding a rare manuscript

Redundancy and repair example

- **Lesion**: Suppose there is a 50% loss rate

Redundancy and repair example

- **Repair**: At the end of each month new copies are made of surviving information



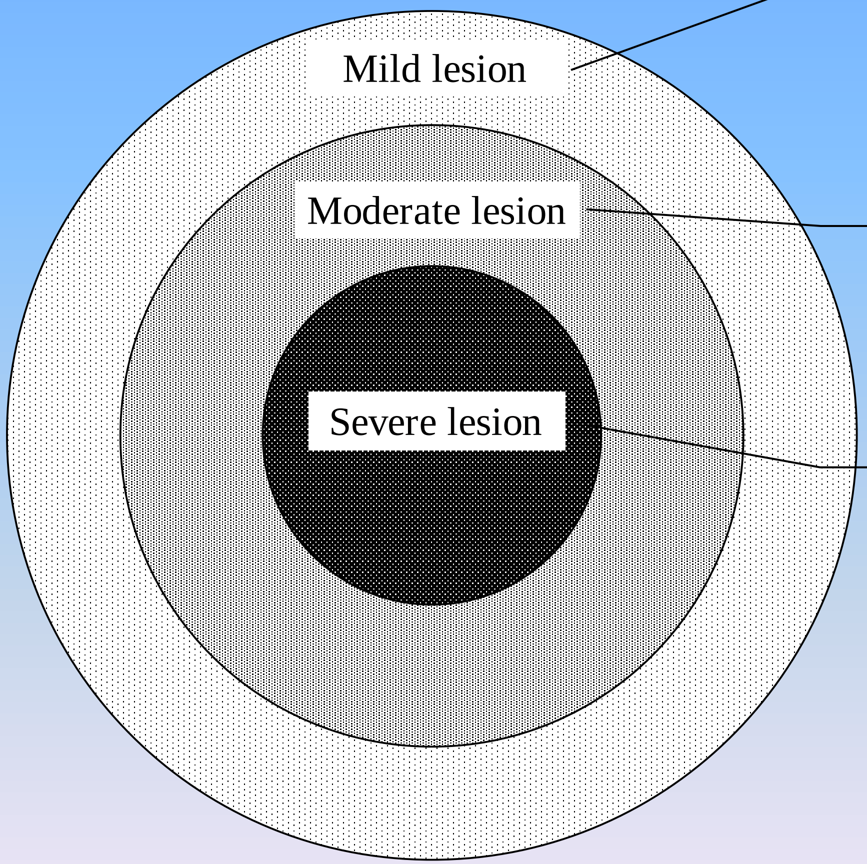
This process has a long life-time

- Monthly ‘lesion-repair’ continues for many months ...
- ... until all information is lost at the end of one unfortunate month
- Chances of this happening are very low
- The expected life-time of the manuscript in this example is *over 80 years*

Application

- Spontaneous recovery
- Guided recovery: rehabilitation from brain damage

Triage of recovery patterns

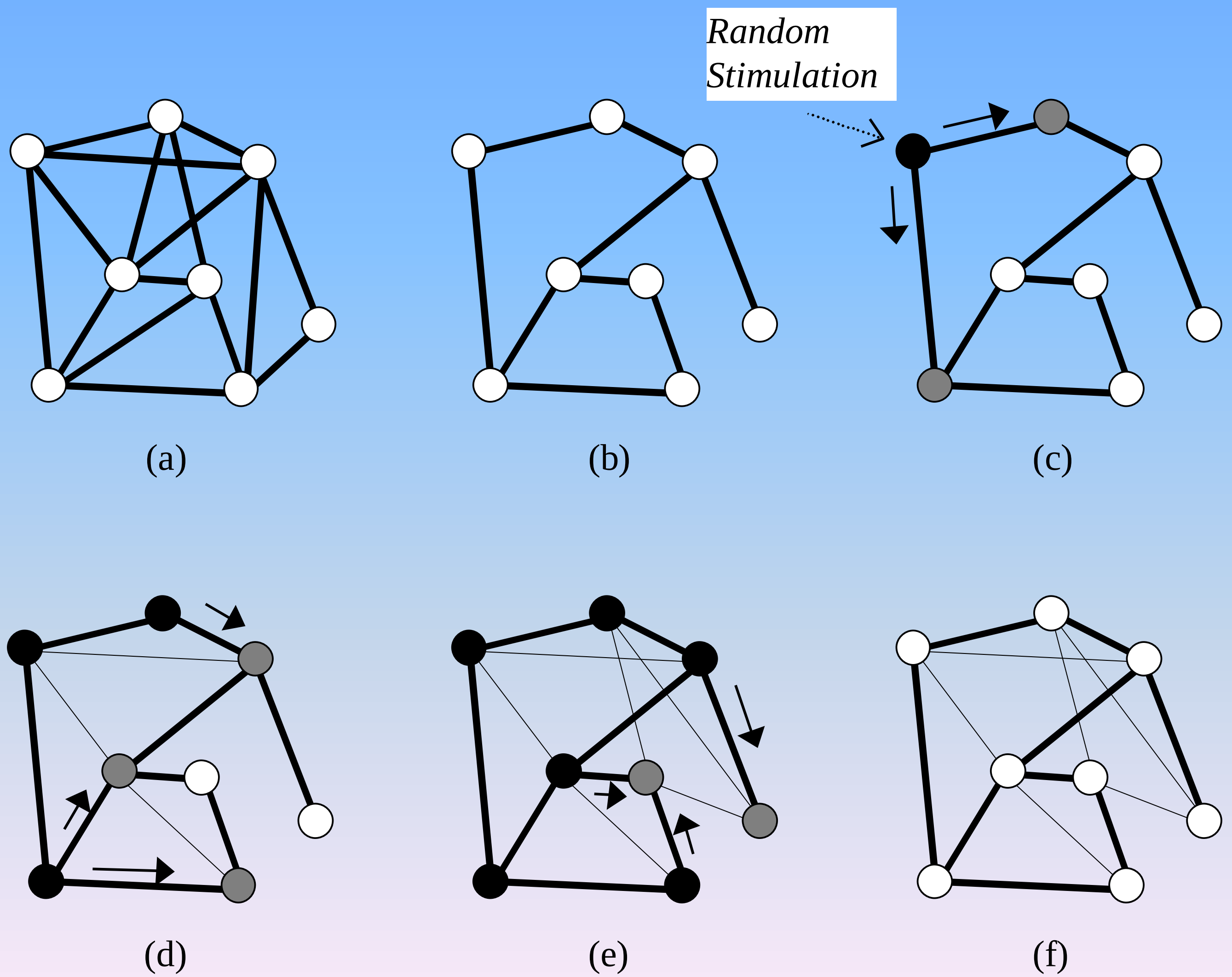


Spontaneous recovery through Hebbian learning

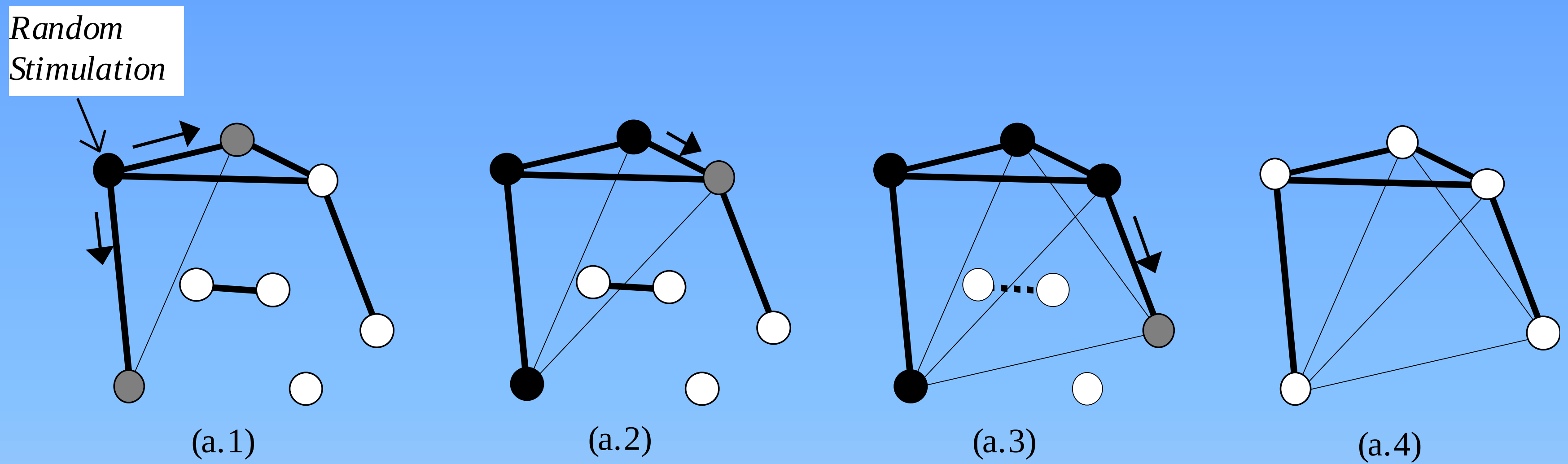
Guided recovery with patterned input

Compensation and functional reorganization

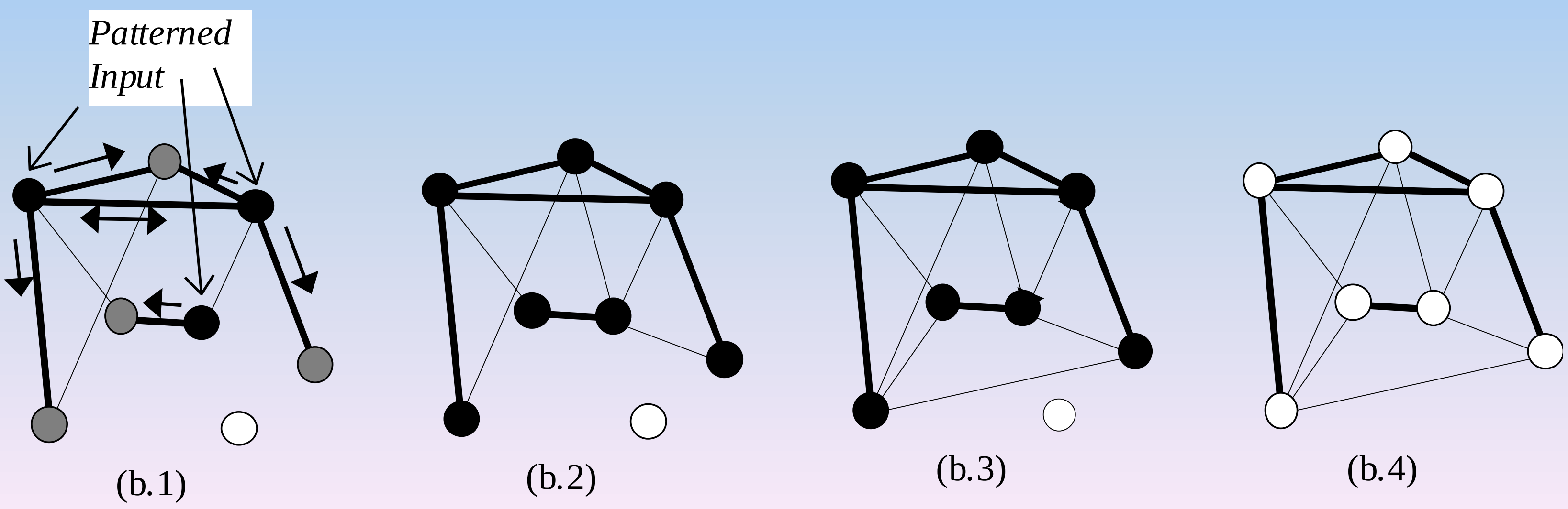
Rewiring and connectivity



a. Spontaneous recovery

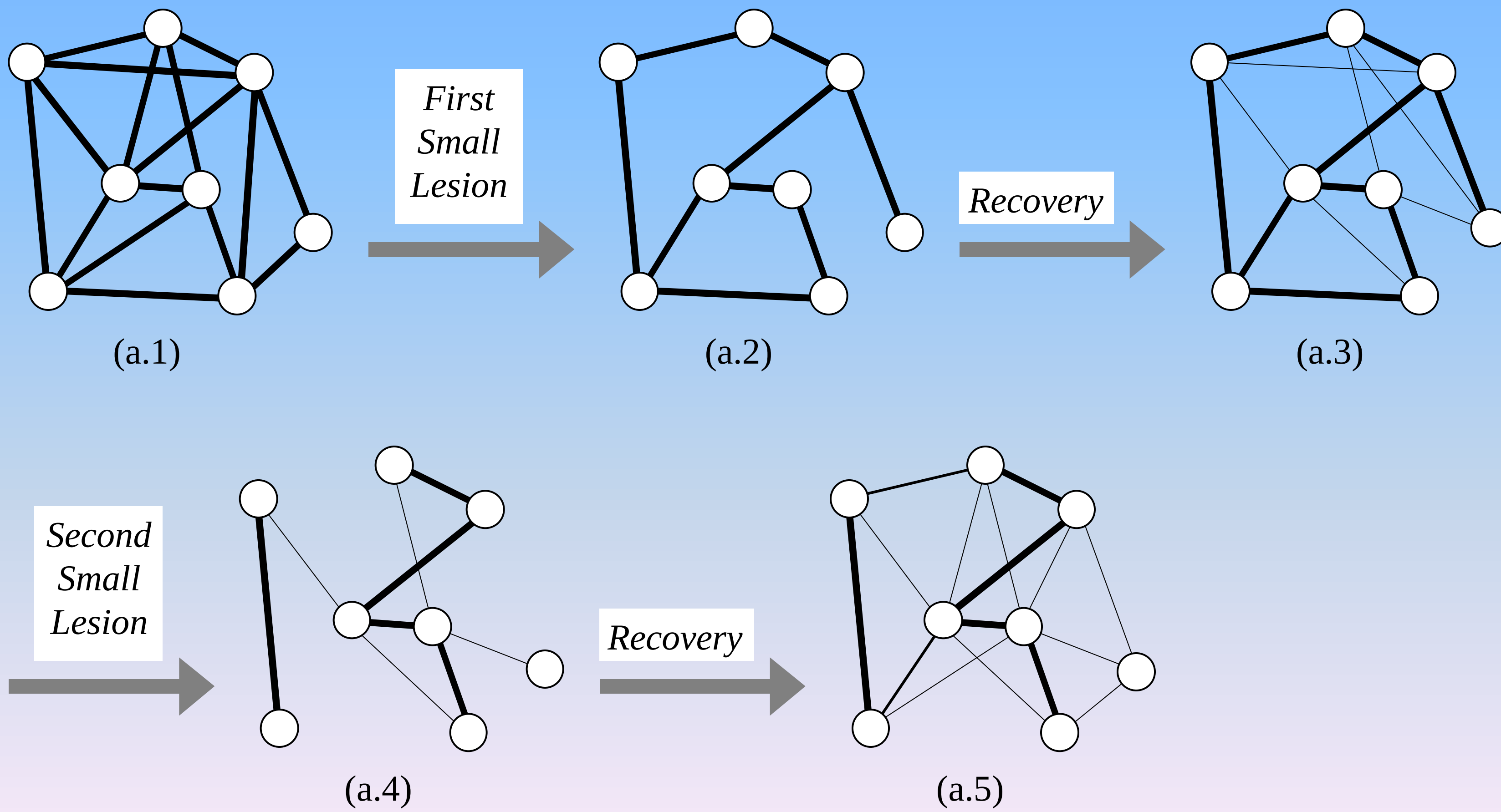


b. Recovery through patterned input



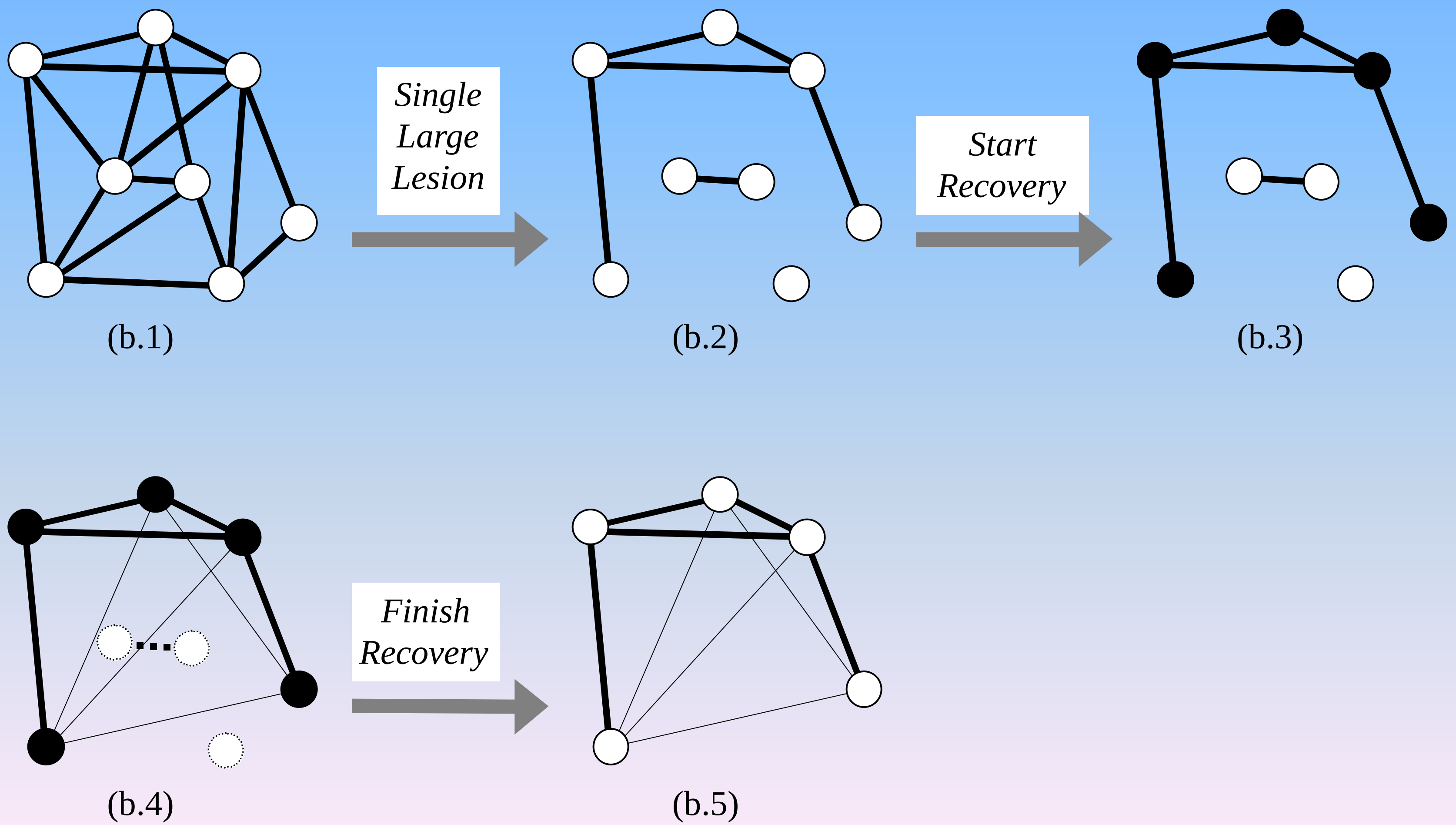
Serial-lesion effect

a. Two small lesions



Serial-lesion effect (continued)

b. A single large lesion



Self-repair in Hopfield networks

Successful repair reduces the perburburbation e_t with 50%

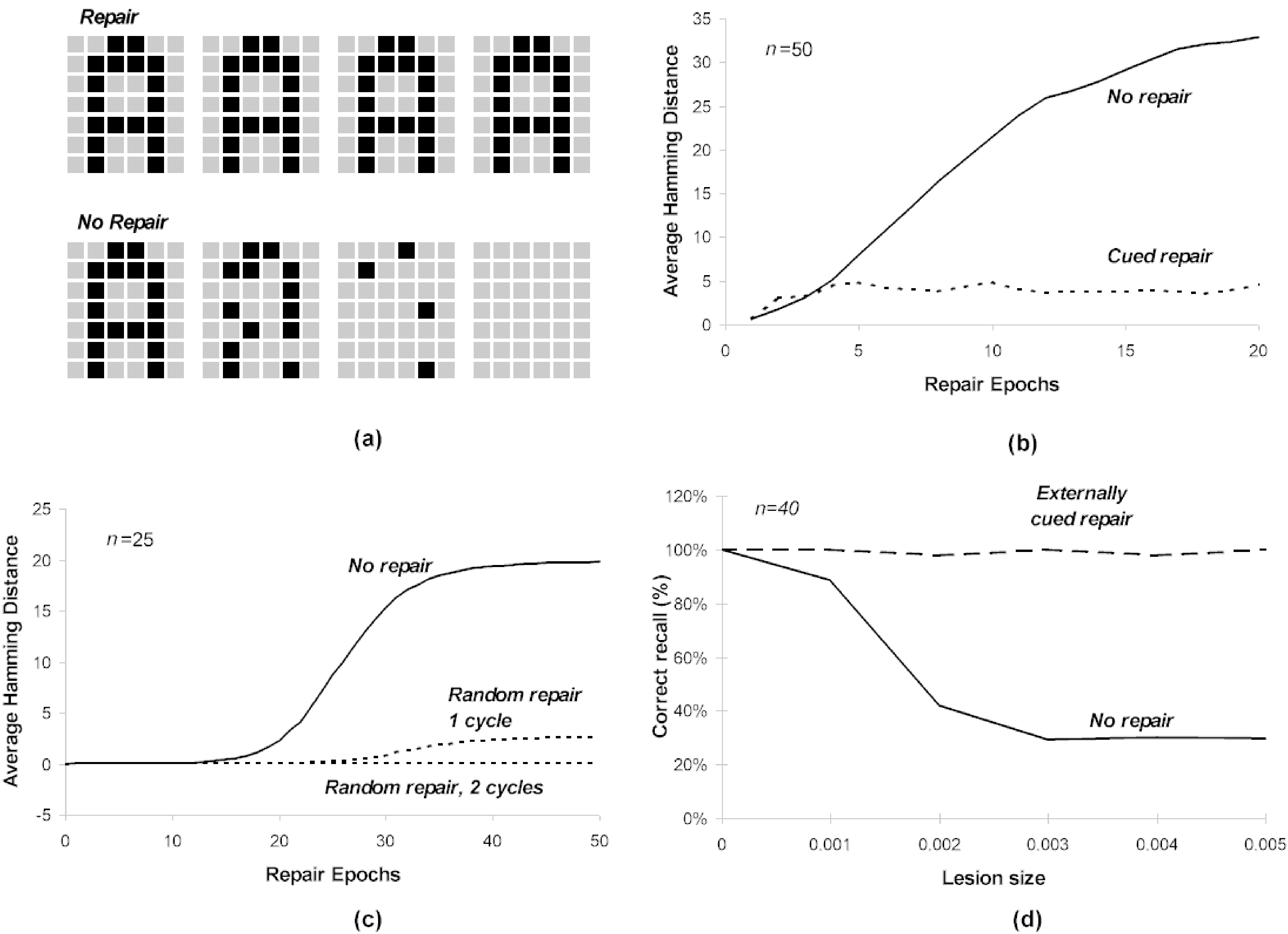
$$\tilde{T}_{ij}^{\tilde{S}} = (2\tilde{V}_i^{\tilde{s}} - 1)(2\tilde{V}_j^{\tilde{s}} - 1)$$

$$T_{ij}^{S}(t) + \tilde{T}_{ij}^{\tilde{S}}(t) + e_t\,.$$

$$\tilde{S} = S \qquad T_{ij}^{S}(t) = \tilde{T}_{ij}^{\tilde{S}}(t)$$

$$T_{ij}^{S}(t+1) = \{T_{ij}^{S}(t) + \tilde{T}_{ij}^{\tilde{S}}(t) + e_t\}/2 =$$

$$T_{ij}^{S}(t) + \tfrac{1}{2}e_t$$


Repair
No Repair
(a)
n=50
Average Hamming Distance
No repair
Cued repair
Repair Epochs
35
30
25
20
15
10
5
0
0
5
10
15
20
(b)
n=25
Average Hamming Distance
No repair
Random repair
1 cycle
Random repair, 2 cycles
Repair Epochs
25
20
15
10
5
0
-5
0
10
20
30
40
50
(c)
n=40
Externally
cued repair
No repair
Correct recall (%)
120%
100%
80%
60%
40%
20%
0%
Lesion size
0
0.001
0.002
0.003
0.004
0.005
(d)

Towards a scientifically based rehabilitation from brain damage

- How can *early* assessment be improved?
- How to improve triage classification?
- What is the role of arousal?
- Does learning consolidation play a role? Sleep?
- Can we identify (much) more precisely which brain areas are involved?