Marr-Albus Model of Cerebellum

Computational Models of Neural Systems

Lecture 2.2

David S. Touretzky

September, 2025

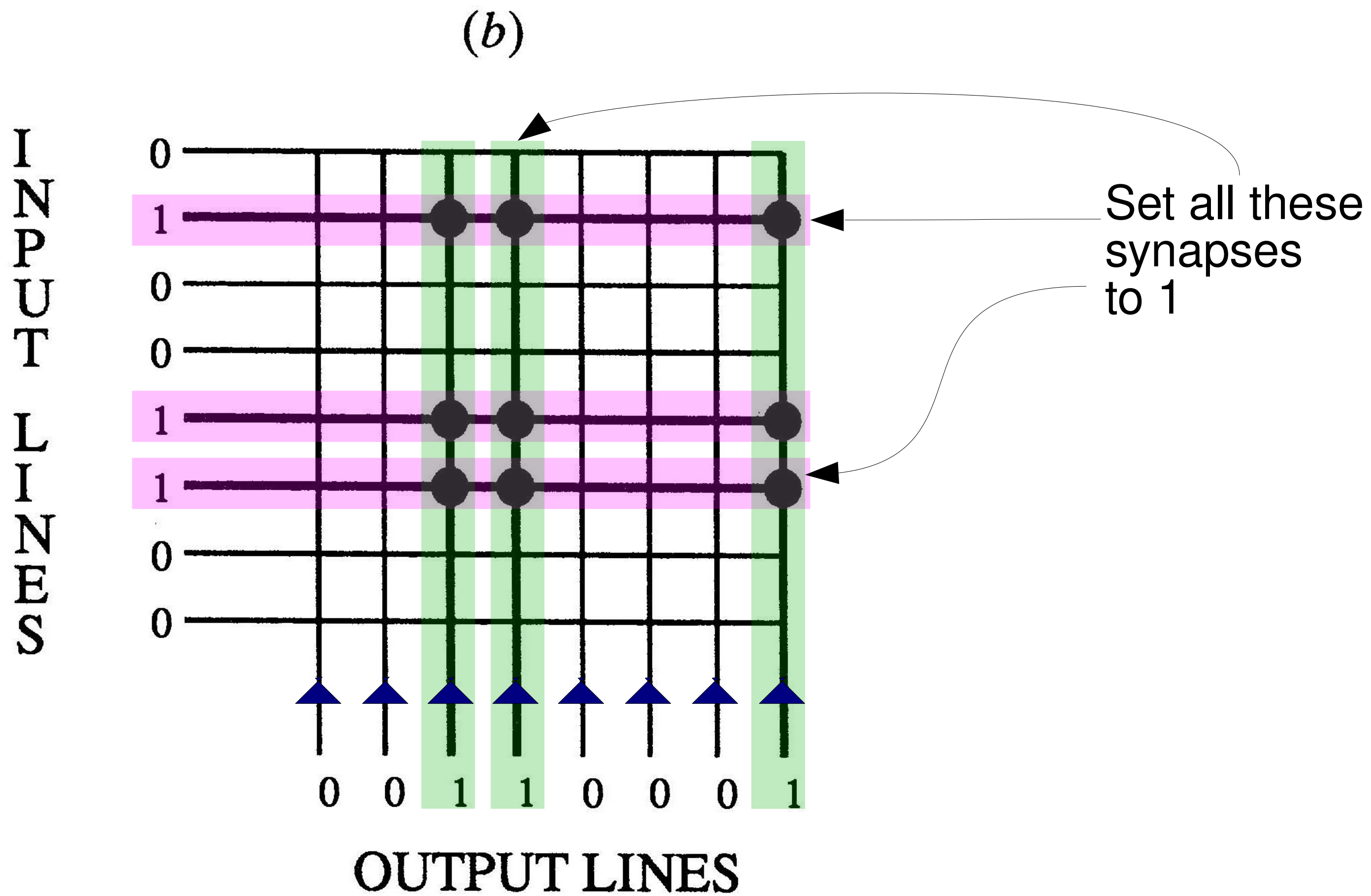
Marr's Theory

- Marr suggested that the cerebellum is an associative memory.
- Input: proprioceptive information (state of the body).
- Output: motor commands necessary to achieve the goal associated with that context.
- Learn from experience to map states into motor commands.
- Wants to avoid pattern overlap, to keep patterns distinct.

Albus' Theory

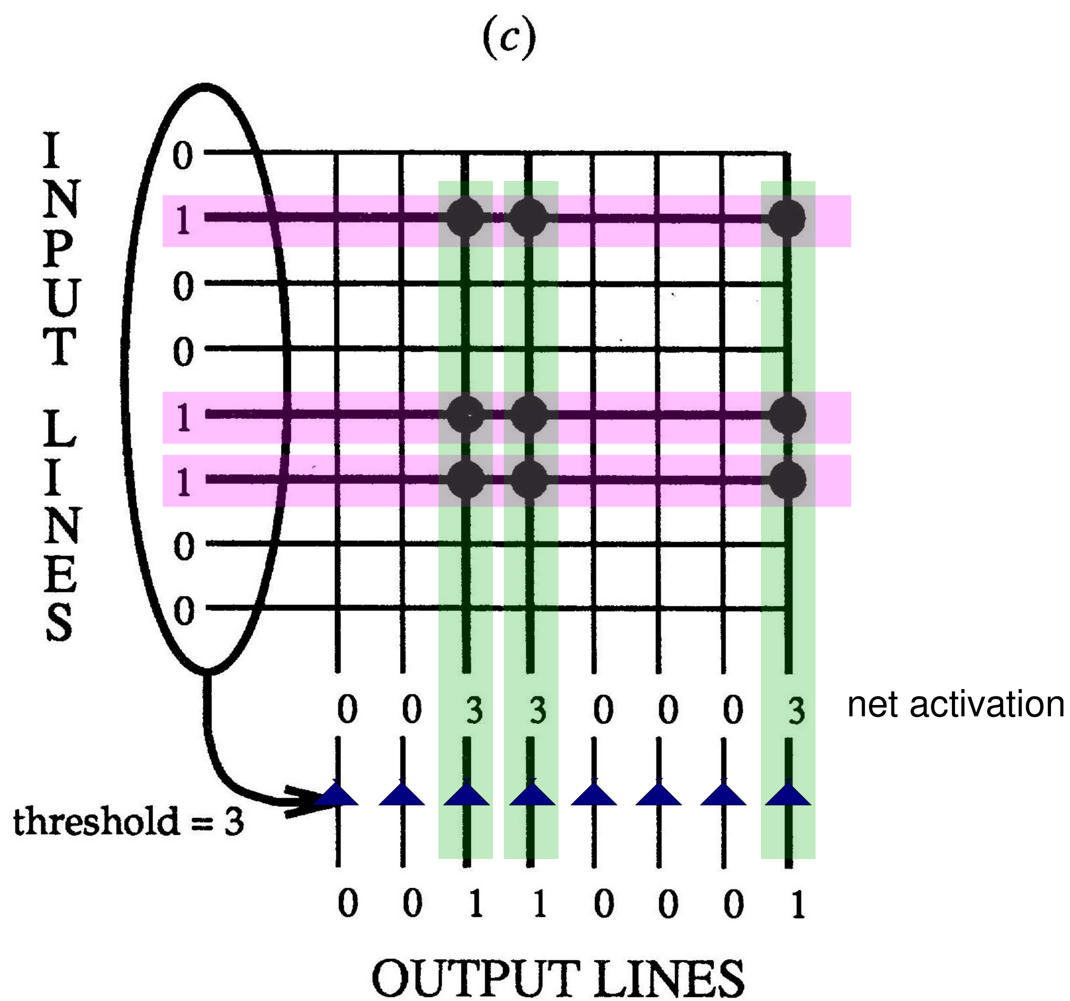
- Albus suggested that the cerebellum is a function approximator.
- Similar to an associative memory, but uses pattern overlap and interpolation to approximate nonlinear functions.
- Could explain how the cerebellum generalizes to novel input patterns that are similar to those for previously practiced motions.

Associative Memory: Store a Pattern

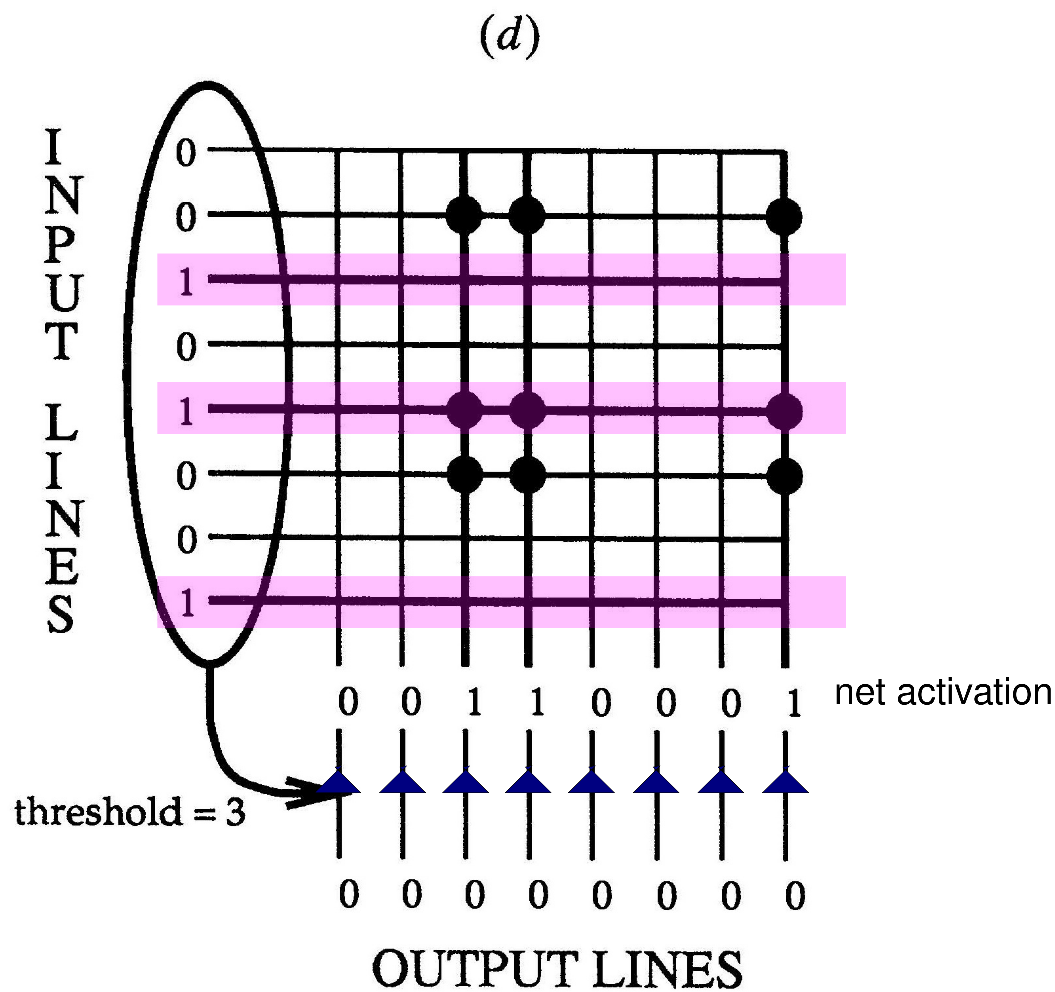


The input and output patterns don't have to be the same length, although in the above example they are.

Associative Memory: Retrieve the Pattern

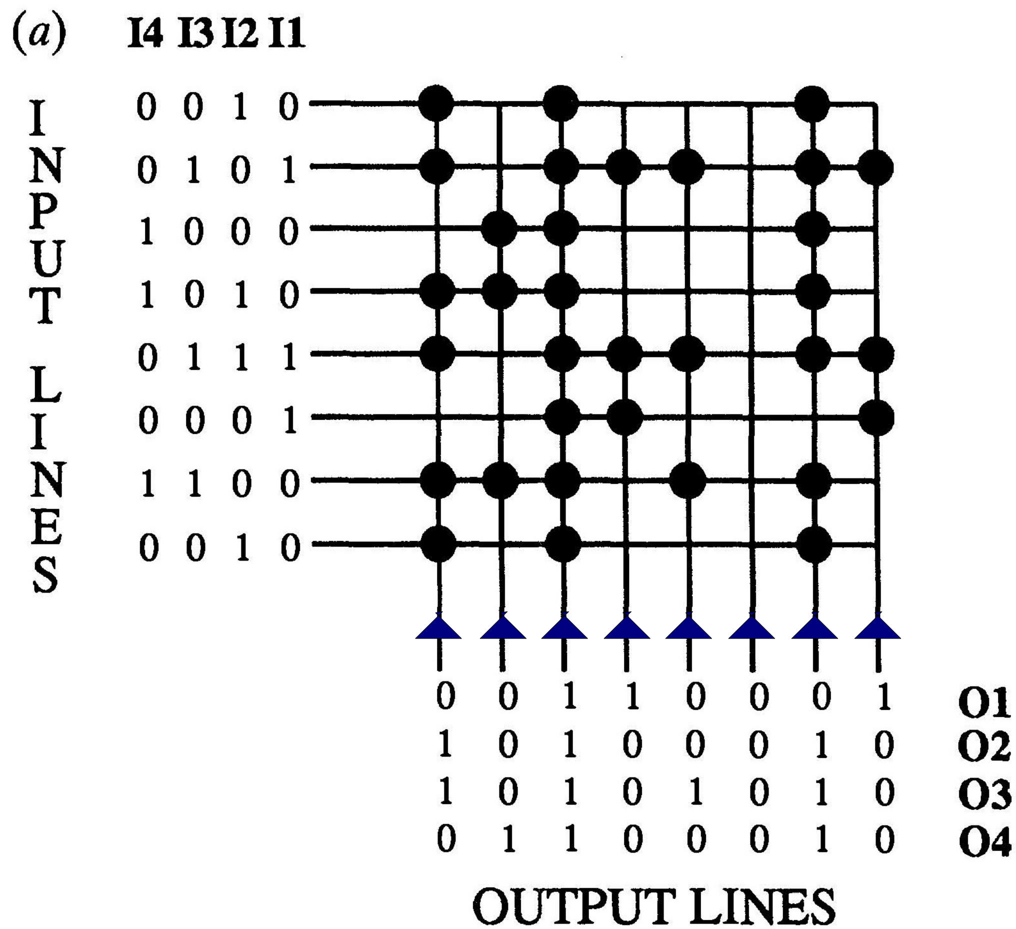


Associative Memory: Unfamiliar Pattern



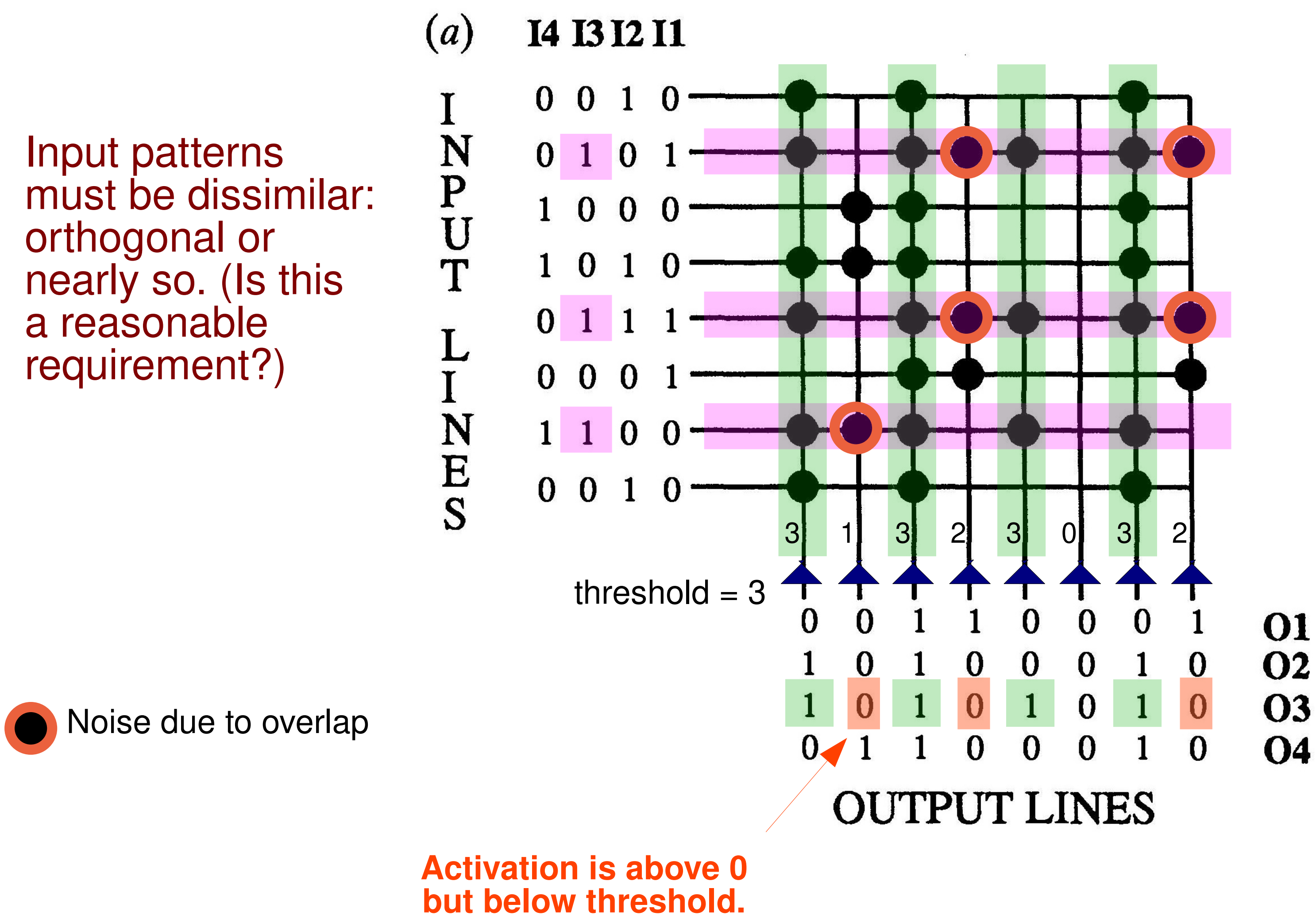
Storing Multiple Patterns

Input patterns must be dissimilar: orthogonal or nearly so. (Is this a reasonable requirement?)

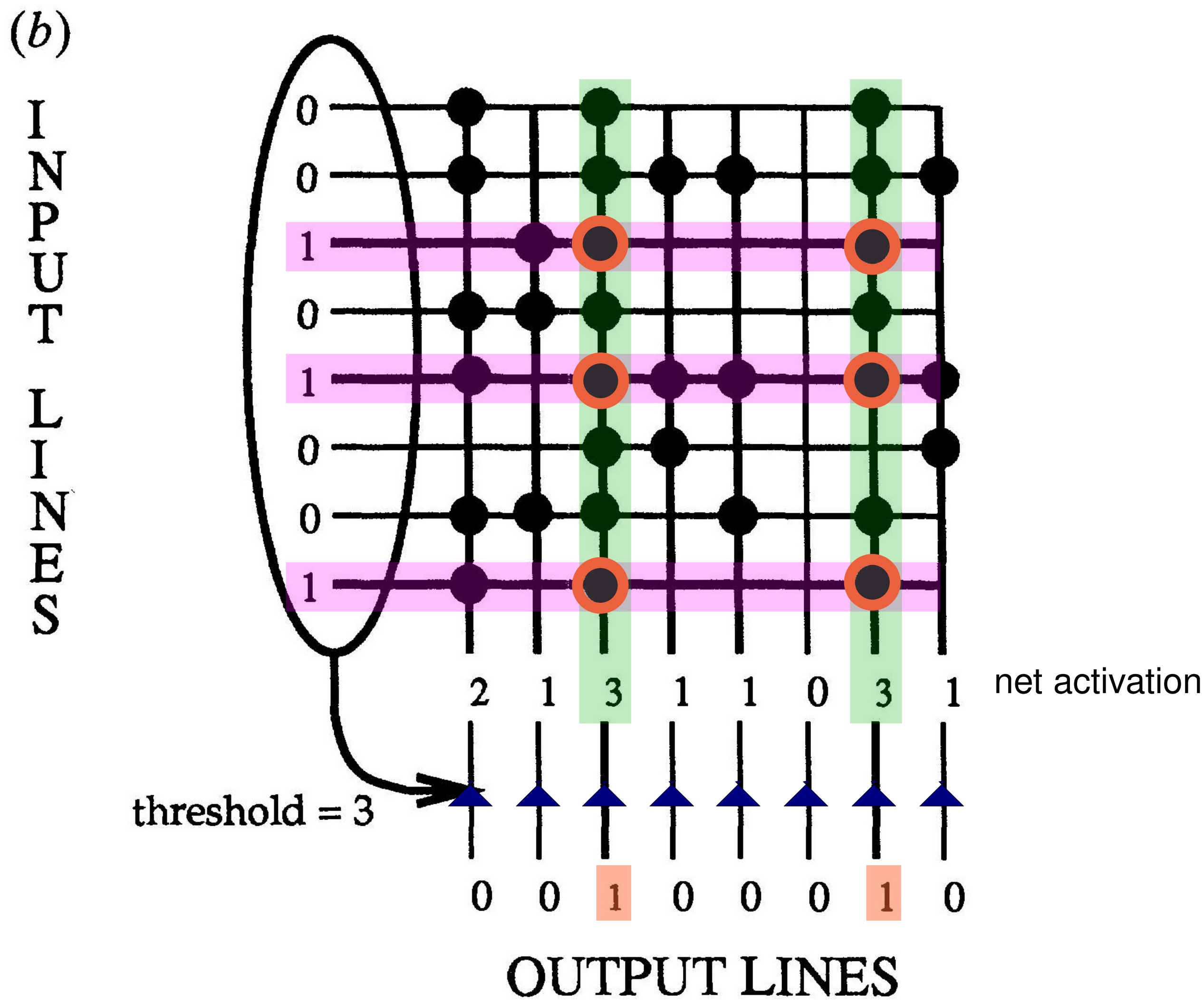


Storing Multiple Patterns

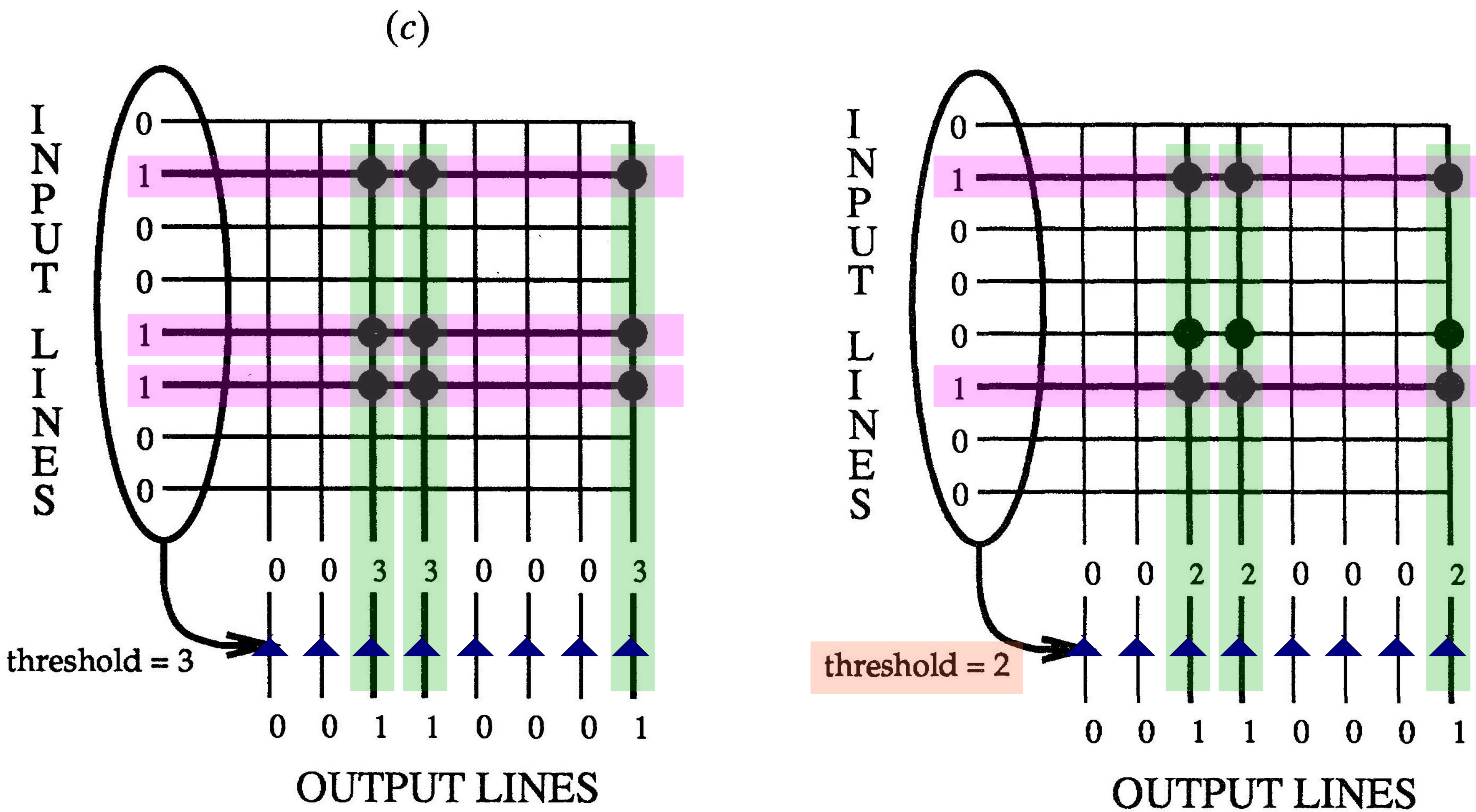
Input patterns must be dissimilar: orthogonal or nearly so. (Is this a reasonable requirement?)



False Positives Due to Memory Saturation

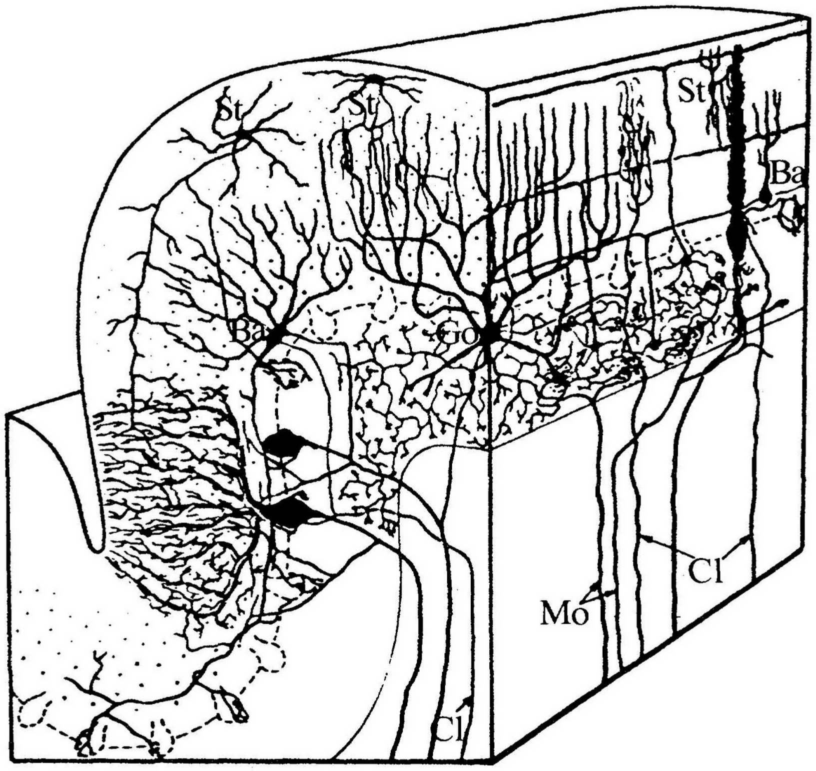


Responding To A Subset Pattern

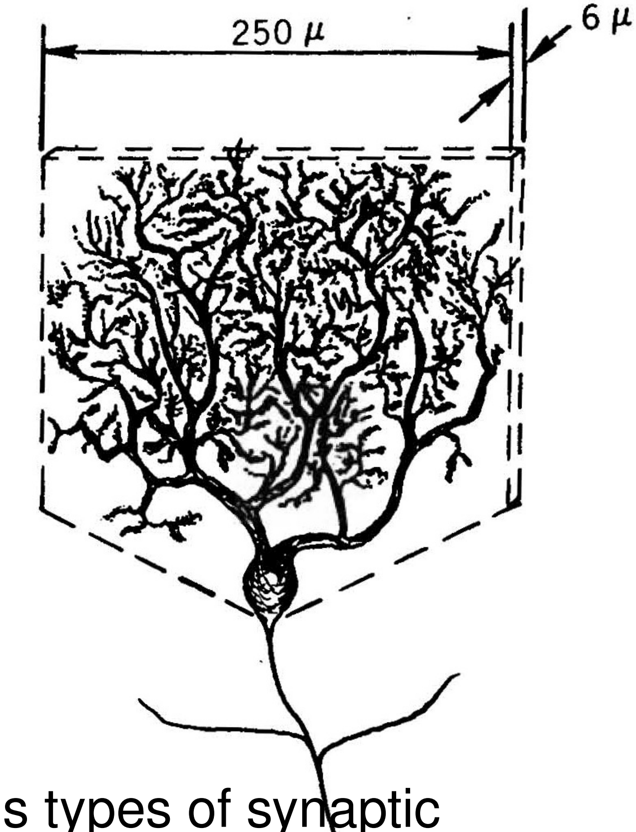


Training the Cerebellum

- Mossy fibers (input pattern)
  - Input from spinal cord, vestibular nuclei, and the pons.
  - Spinocerebellar tracts carry cutaneous and proprioceptive information.
  - Much more massive input comes from the cortex via the pontine nuclei (the pons) and then the middle cerebellar peduncle. More fibers in this peduncle than all other afferent/efferent fiber systems to cerebellum.
- Climbing fibers (teacher)
  - Originate in the inferior olivary nucleus.
  - The “training signal” for motor learning.
  - The UCS for classical conditioning.
- Neuromodulatory inputs from raphe nucleus, locus ceruleus, and hypothalamus.



Purkinje Cells



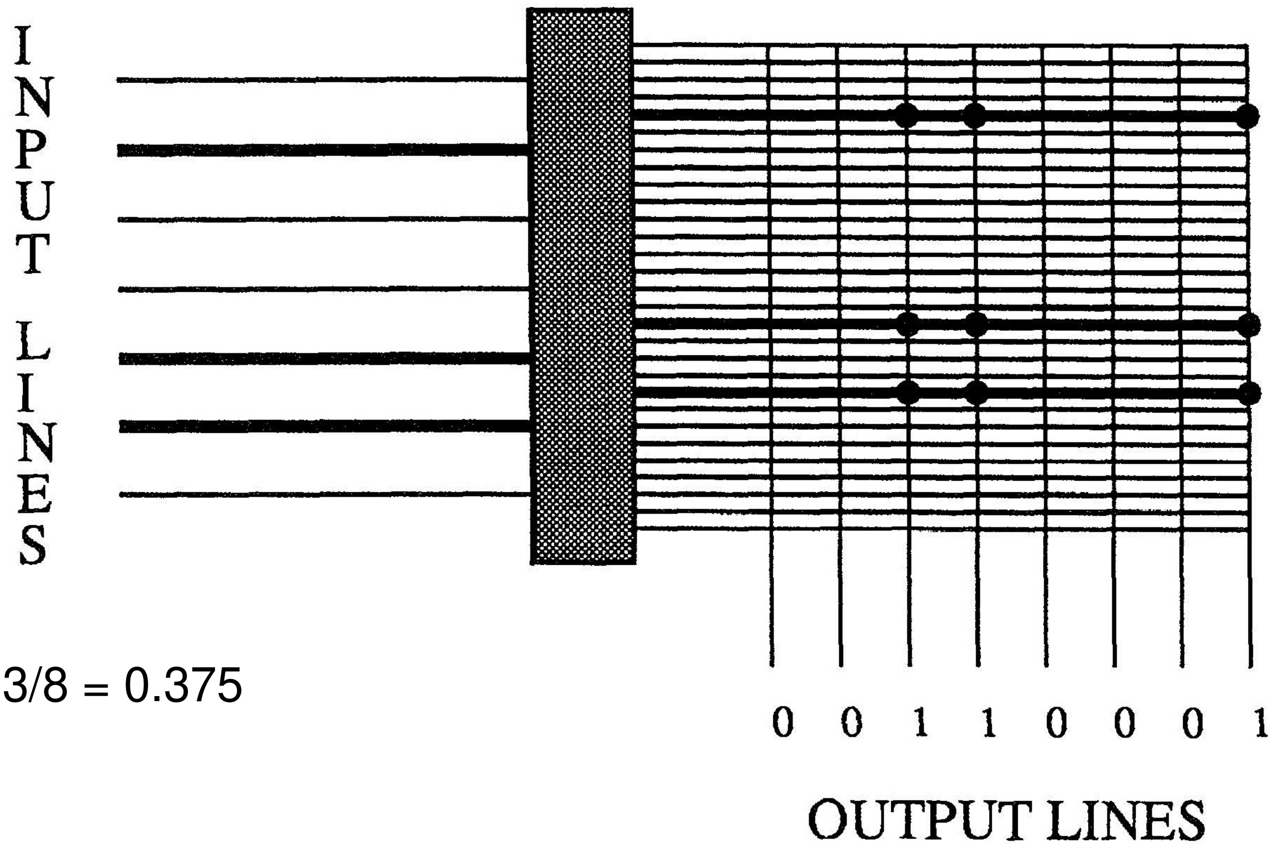
- The principal cells of the cerebellum.
- Largest dendritic trees in the brain: about 200,000 synapses.
- These synapses are where the associative weights are stored.
- Albus argues that basket and stellate cells should also have trainable synapses.
  - Beginning in the 1990s, evidence for various types of synaptic adaptation and long-term plasticity in stellate cell inputs and outputs has accumulated.
- Purkinje cells have recurrent collaterals that contact Golgi cell dendrites and other Purkinje cell dendrites and cell bodies.
- Purkinje cells make only inhibitory connections.

Input Processing

- If mossy fiber inputs made direct contact with Purkinje cells, the cerebellum would have a much lower memory capacity due to pattern interference.
- Also, for motor learning, subsets of an input pattern should not produce the same results as a supserset input. Subsets must be recoded so that they look less similar to the whole.
    - “cup in hand”, “hand near mouth”, “mouth open”
    - “cup in hand”, “mouth open” (don't rotate wrist!)
- Solution: introduce a layer of processing before the Purkinje cells to make the input patterns more sparse and less similar to each other (more orthogonal).
    - This is what granule cells are for!
- Similar to the role of the dentate gyrus in hippocampus.

Mossy Fiber to Parallel Fiber Transformation: "Conjunctive Coding"

- Same number of active lines, but a larger population of units, produces greater sparsity (smaller α) and less overlap between patterns.

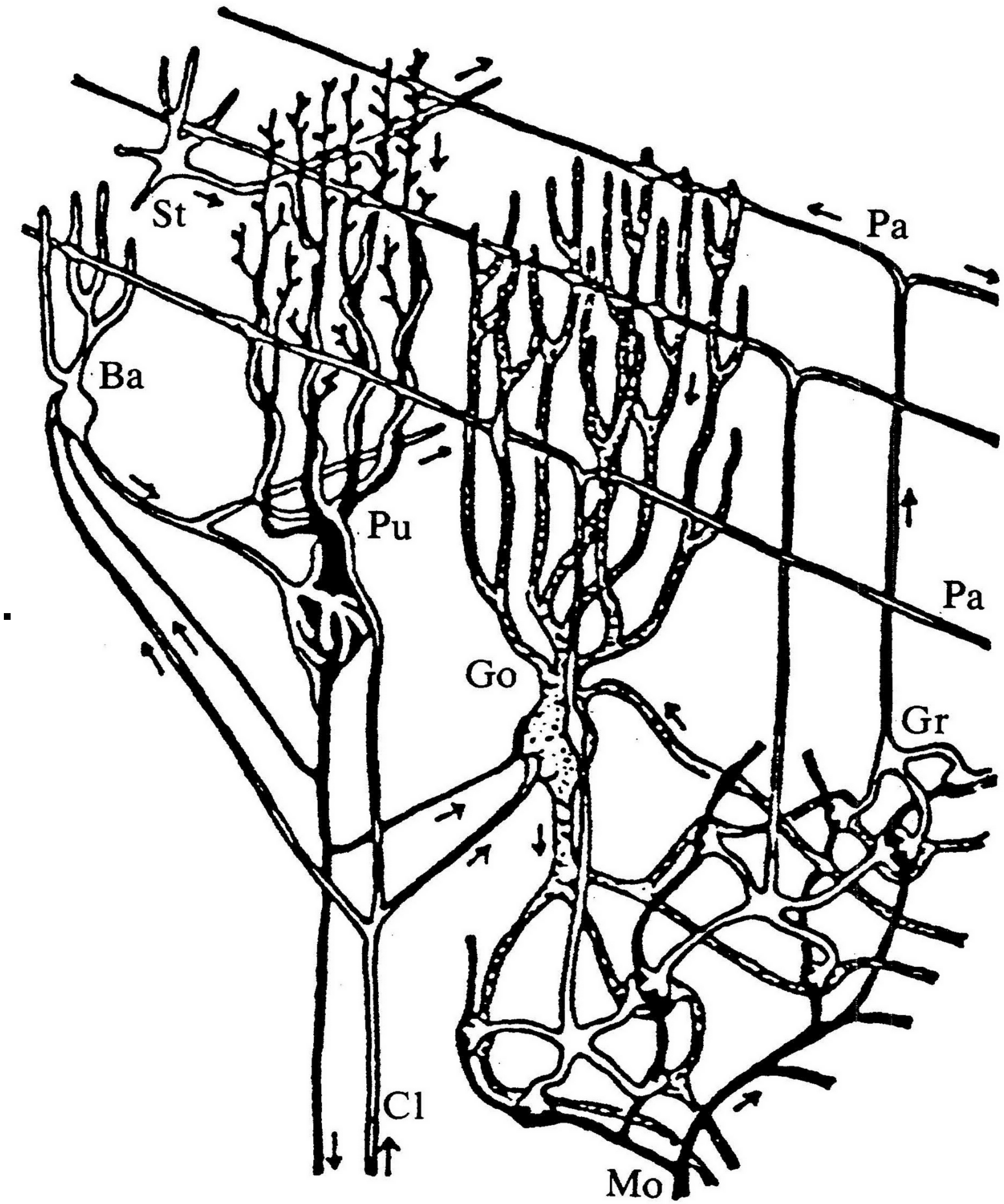


$\alpha_o = 3/29 = 0.103$

$\alpha_i = 3/8 = 0.375$

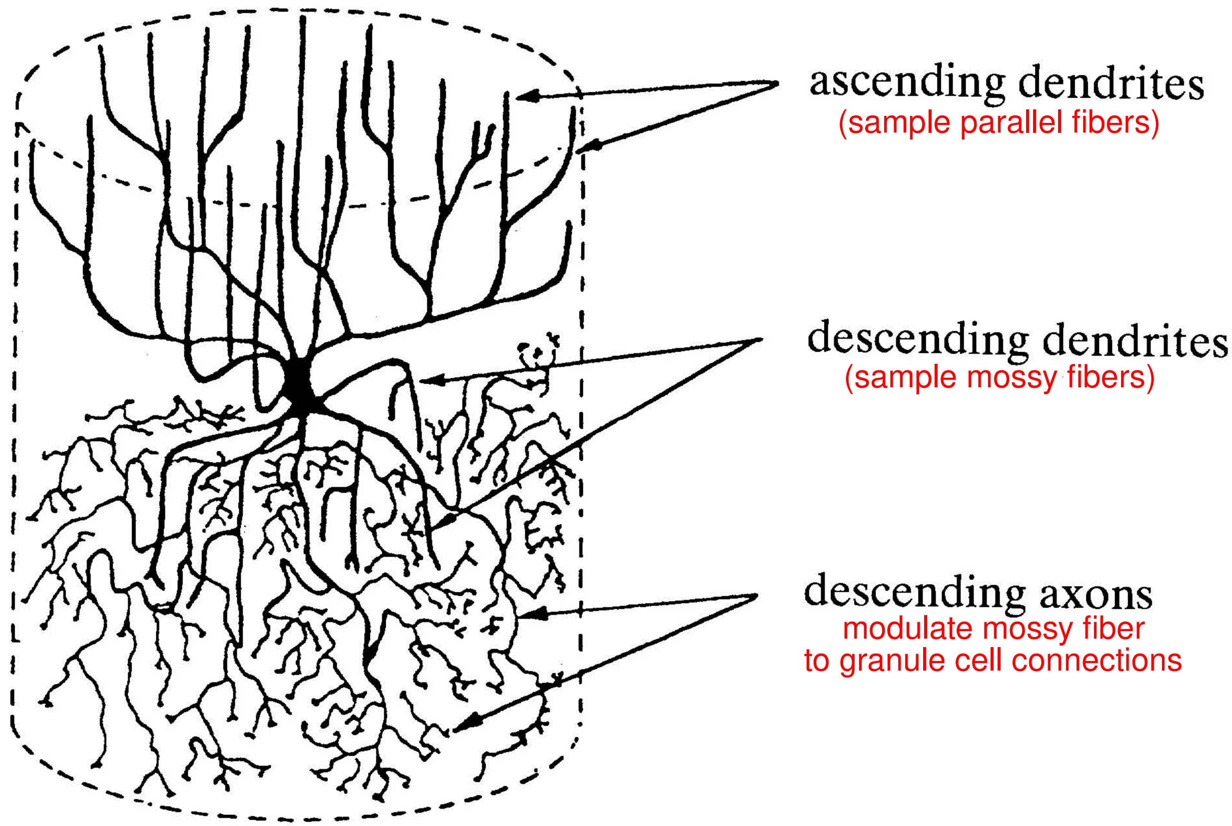
Recoding Via Granule Cells

- Mossy fibers synapse onto granule cells.
- Granule cell axons (called parallel fibers) provide input to Purkinje cells.
- Golgi cells are inhibitory interneurons that modulate the granule cell responses to produce 'better" activity patterns.

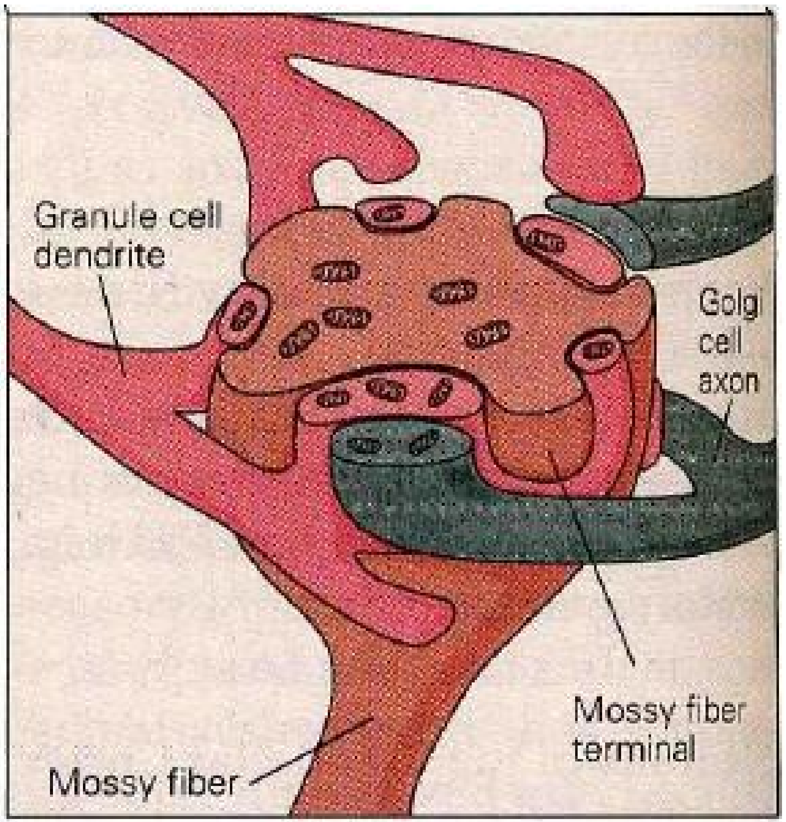


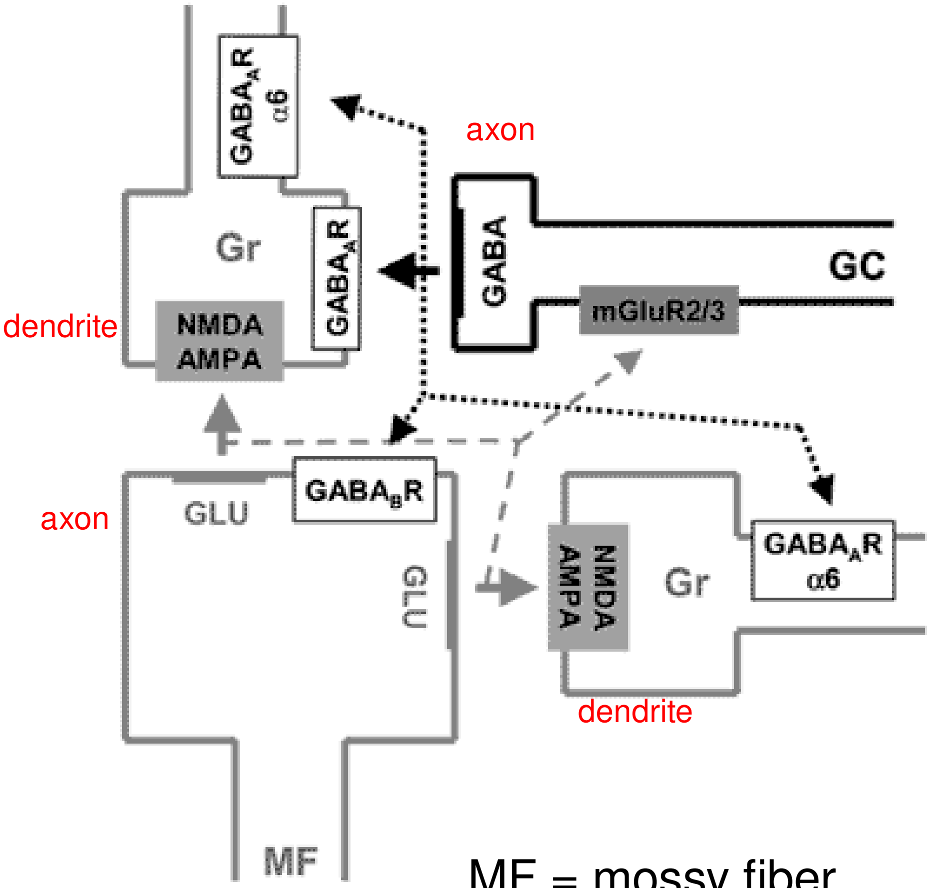
Golgi Cells

- Golgi cells monitor both the mossy fibers (granule cell inputs) and the parallel fibers (granule cell outputs).
- Mossy fiber input patterns with <u>widely varying</u> levels of activity result in granule cell patterns with roughly the <u>same</u> level of activity, thanks to the Golgi cells.



The Glomerulus

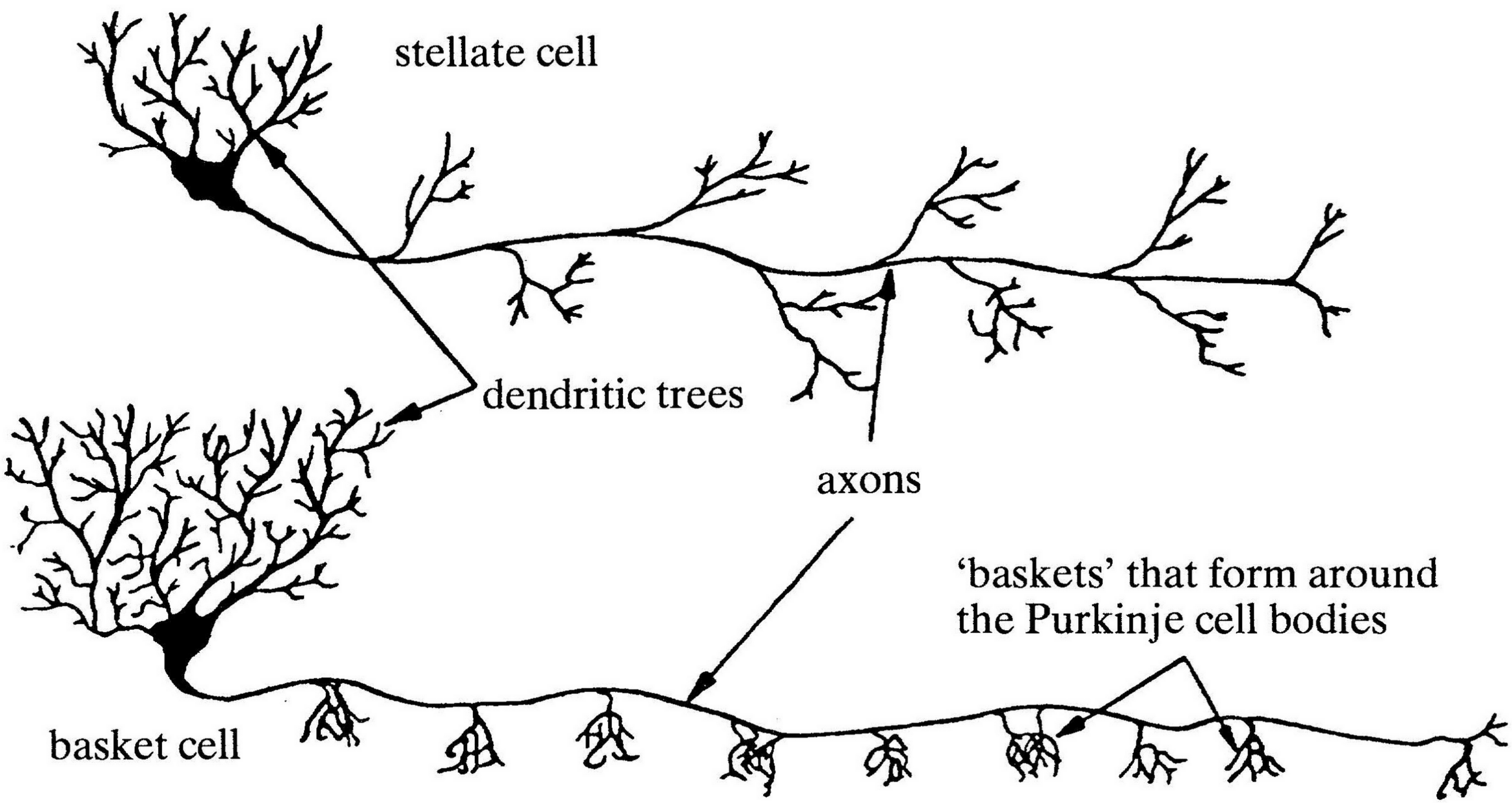




MF = mossy fiber
Gr = granule cell
GC = Golgi cell

Basket and Stellate Cells

- Inhibitory interneurons that supply short-range, within-beam inhibition (stellate) and long-range, across-beam inhibition (basket).



The Matrix Memory

- Weights: modifiable synapses from granule cell parallel fibers onto Purkinje cell dendrites.
- Thresholding: whether the Purkinje cell chooses to fire.
- Threshold setting: stellate and basket cells sample the input pattern on the parallel fibers and make inhibitory connections onto the Purkinje cells.
- Marr thought the synapses started out with zero or small weights that increased with learning (LTP).
- Albus' contribution: synapses should initially have high weights, not zero weights. Learning *reduces* the weight values (LTD).
- Since Purkinje cells are inhibitory, reducing their input means they will fire less, thereby dis-inhibiting their target cells.

Marr's Notation for Analyzing His Model

α_m is the fraction of active mossy fibers
α_g is the fraction of active granule cells (parallel fibers)
N_m, N_g are numbers of mossy fibers/granule cells

$N_m \alpha_m$ = expected # of active mossy fibers
$N_g \alpha_g$ = expected # of active granule cells

A fiber that is active with probability α transmits
$-\log_2 \alpha$ bits of information when it fires

$N_m \alpha_m \times -\log_2 \alpha_m$ = information content of a mossy fiber pattern
$N_g \alpha_g \times -\log_2 \alpha_g$ = information content of a granule cell pattern
(but assumes fibers are uncorrelated, which is untrue)

Marr's Constraints on Granule Cell Activity

1. Reduce saturation: tendency of the memory to fill up.

$$\alpha_g < \alpha_m$$

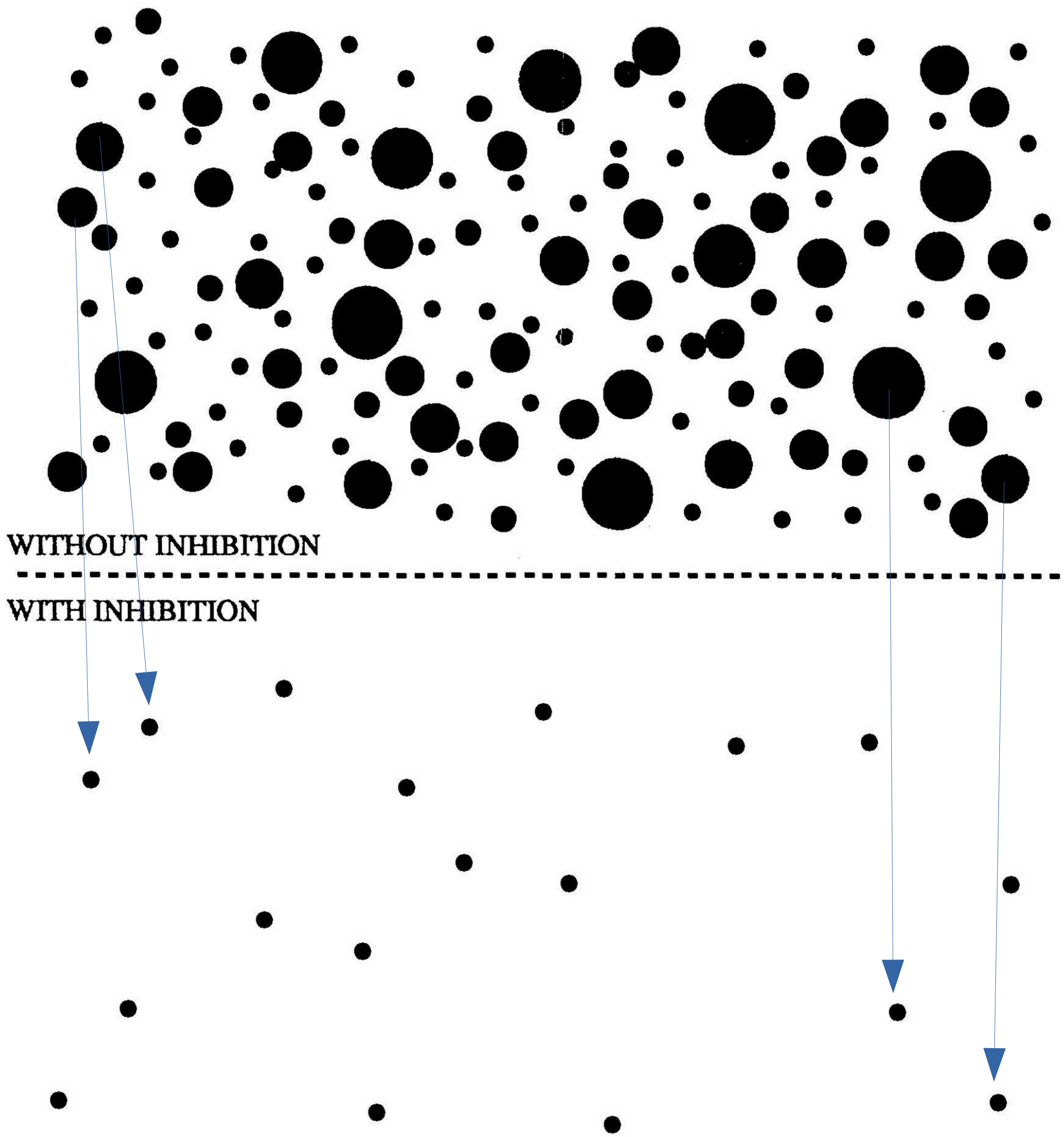
2. Preserve information. The number of bits transmitted should not be reduced by the granule cell encoding step.

$$-N_g \alpha_g (\log \alpha_g) \geq -N_m \alpha_m (\log \alpha_m)$$

$$-\alpha_g (\log \alpha_g) \geq -\frac{N_m}{N_g} \alpha_m (\log \alpha_m)$$

3. Pattern separation: overlap is an increasing function of α, so we again want $\alpha_g < \alpha_m$

Golgi Inhibition Selects Most Active Granule Cells



Summary of Cerebellar Circuitry

- Two input streams:
  - Mossy fibers synapse onto granule cells whose parallel fibers project to Purkinje cells
  - Climbing fibers synapse directly onto Purkinje cells
- Five cell types: (really 8 or more)
  1. Granule cells (input pre-processing)
  2. Golgi cells (regulate granule cell activity)
  3. Purkinje cells (the principal cells)
  4. Stellate cells
  5. Basket cells

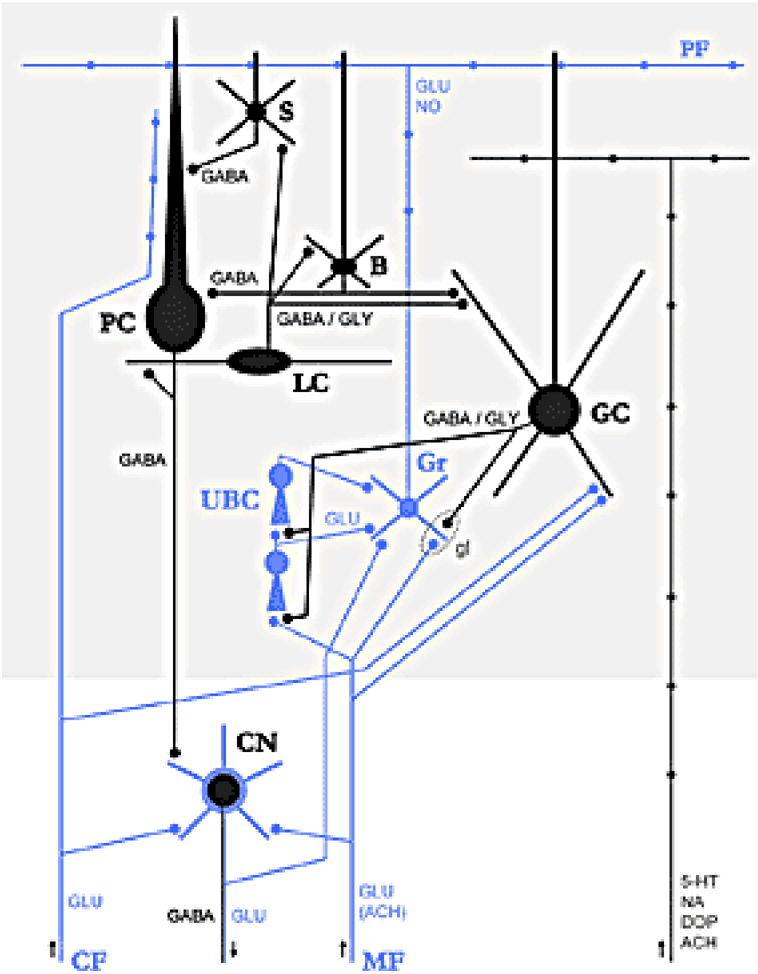
  Feed-forward inhibition of Purkinje cells
- One output path: Purkinje cells to deep cerebellar nuclei.
- But also recurrent collateral connections:

  Purkinje → Purkinje

  Purkinje → Golgi, basket, and stellate cells

New Cell Types Investigated Since Marr/Albus

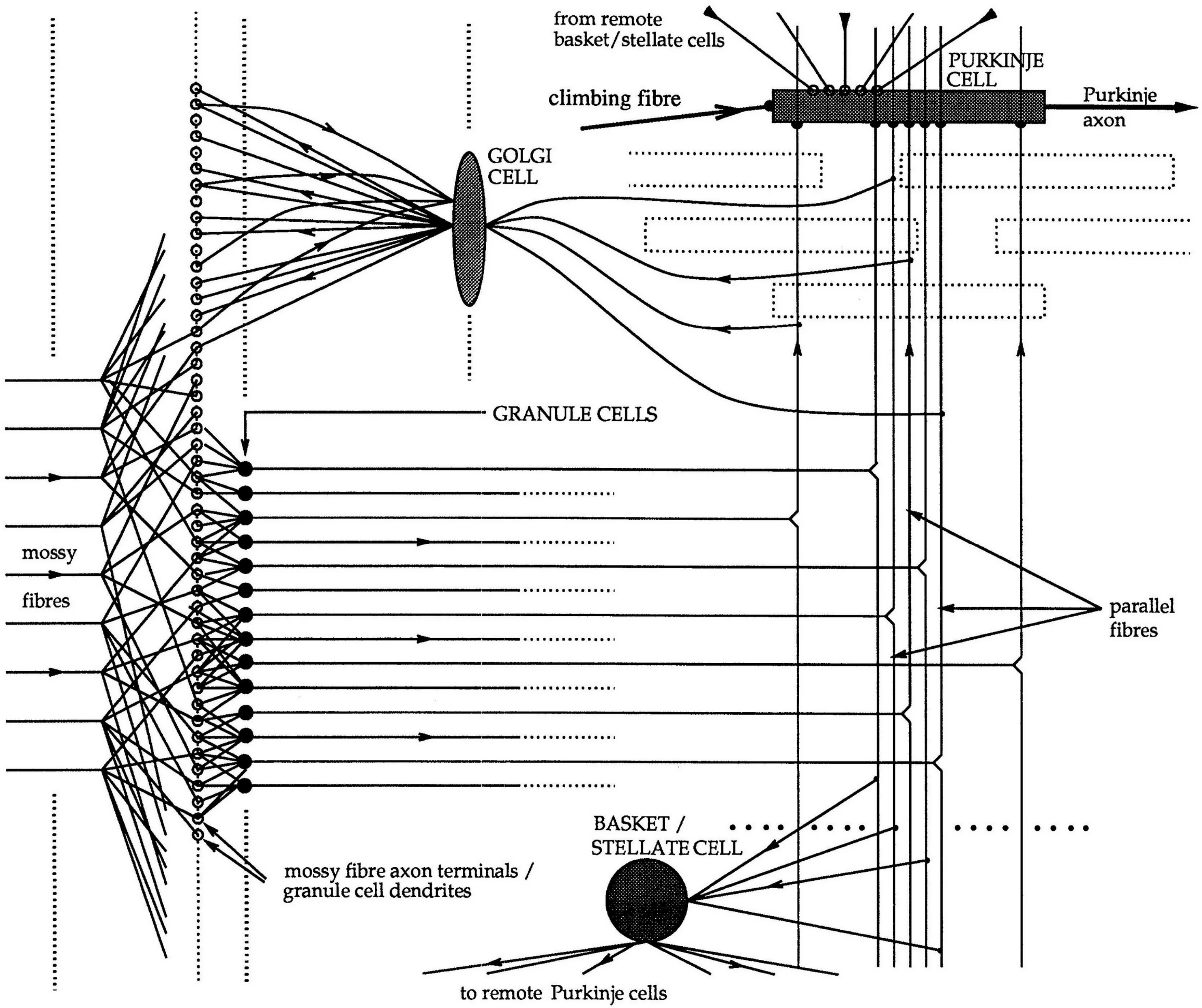
- Lugaro cells (LC): an inhibitory interneuron (GABA) that monitors the activity of Purkinje cells and targets Golgi, basket and stellate cells. May be involved in synchronizing Purkinje cell firing.
- Unipolar brush cells (UBC): excitatory interneurons. One mossy fiber input. Projects to granule cells and other UBCs. May serve to amplify mossy fiber effects on granule cells.
- Candelabrum cells: inhibitory interneuron that receives inputs from mossy fibers, granule cells, and Purkinje cells, and targets stellate and basket cells.



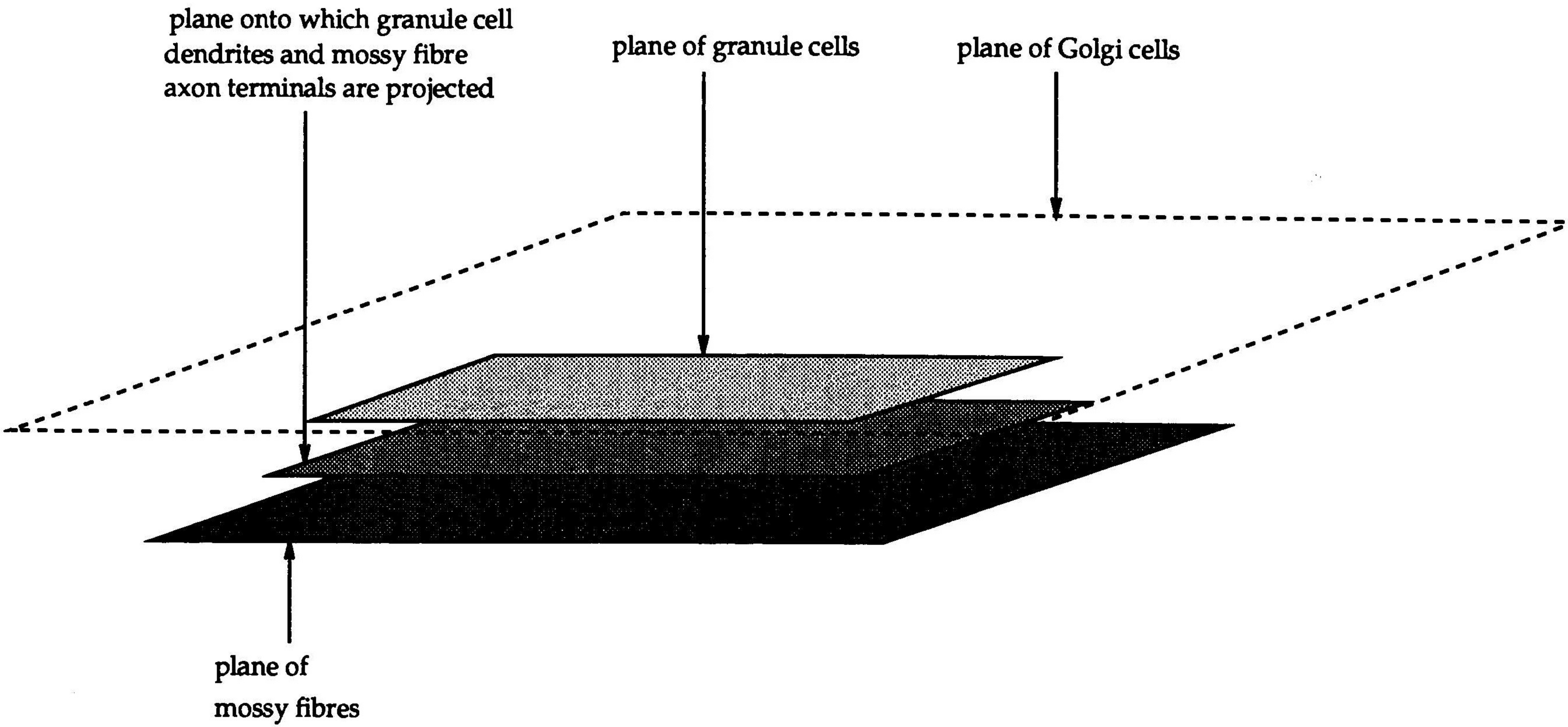
Tyrrell and Willshaw's Simulation (1992)

- C program running on a Sun-4 workstation (12 MIPS processor, 24 MB of memory)
- Tried for a high degree of anatomical realism.
- Took **50 hours** of cpu time to wire up the network!
  Then, 2 minutes to process each pattern.
- Simulation parameters:
  - 13,000 mossy fiber inputs, 200,000 parallel fibers
  - 100 Golgi cells regulating the parallel fiber system
  - binary weights on the parallel fiber synapses
  - 40 basket/stellate cells
  - 1 Purkinje cell, 1 climbing fiber for training

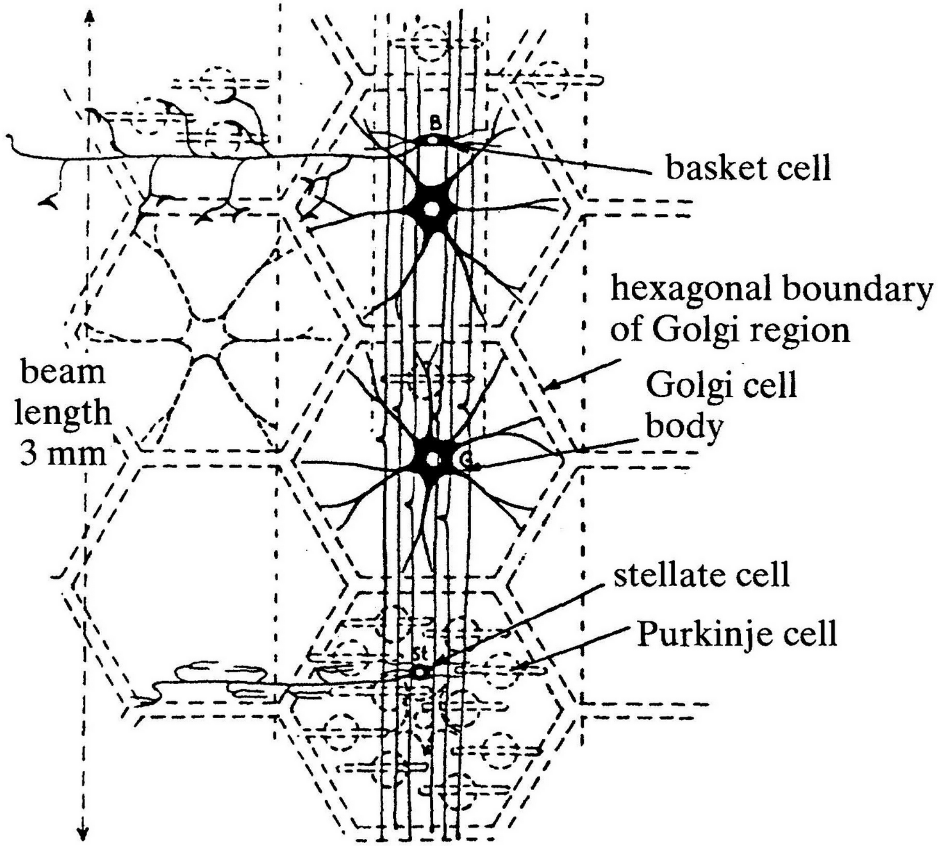
Tyrrell & Willshaw Architecture



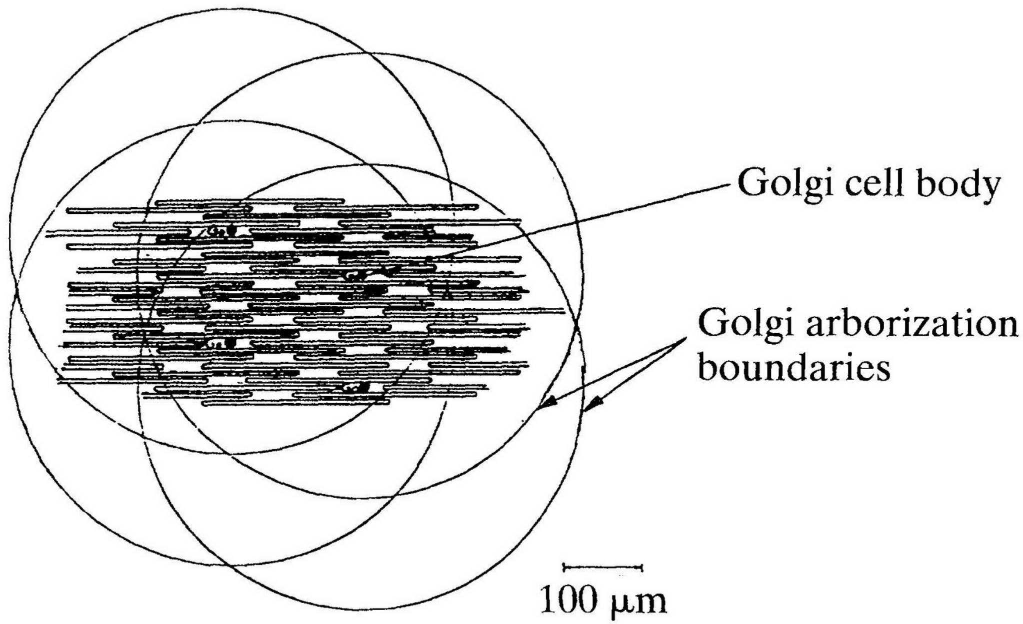
Geometrical Layout



Golgi Cell Arrangement

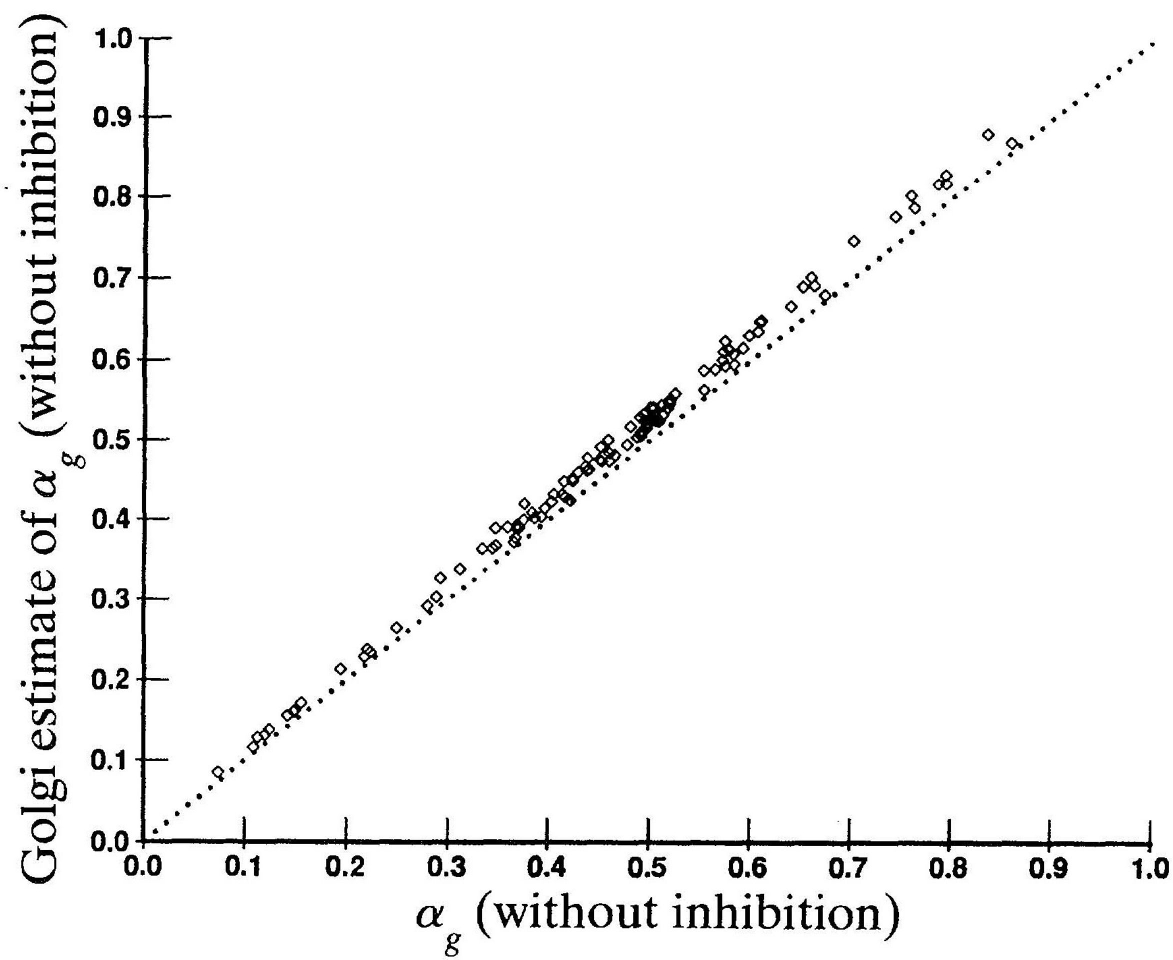


Marr



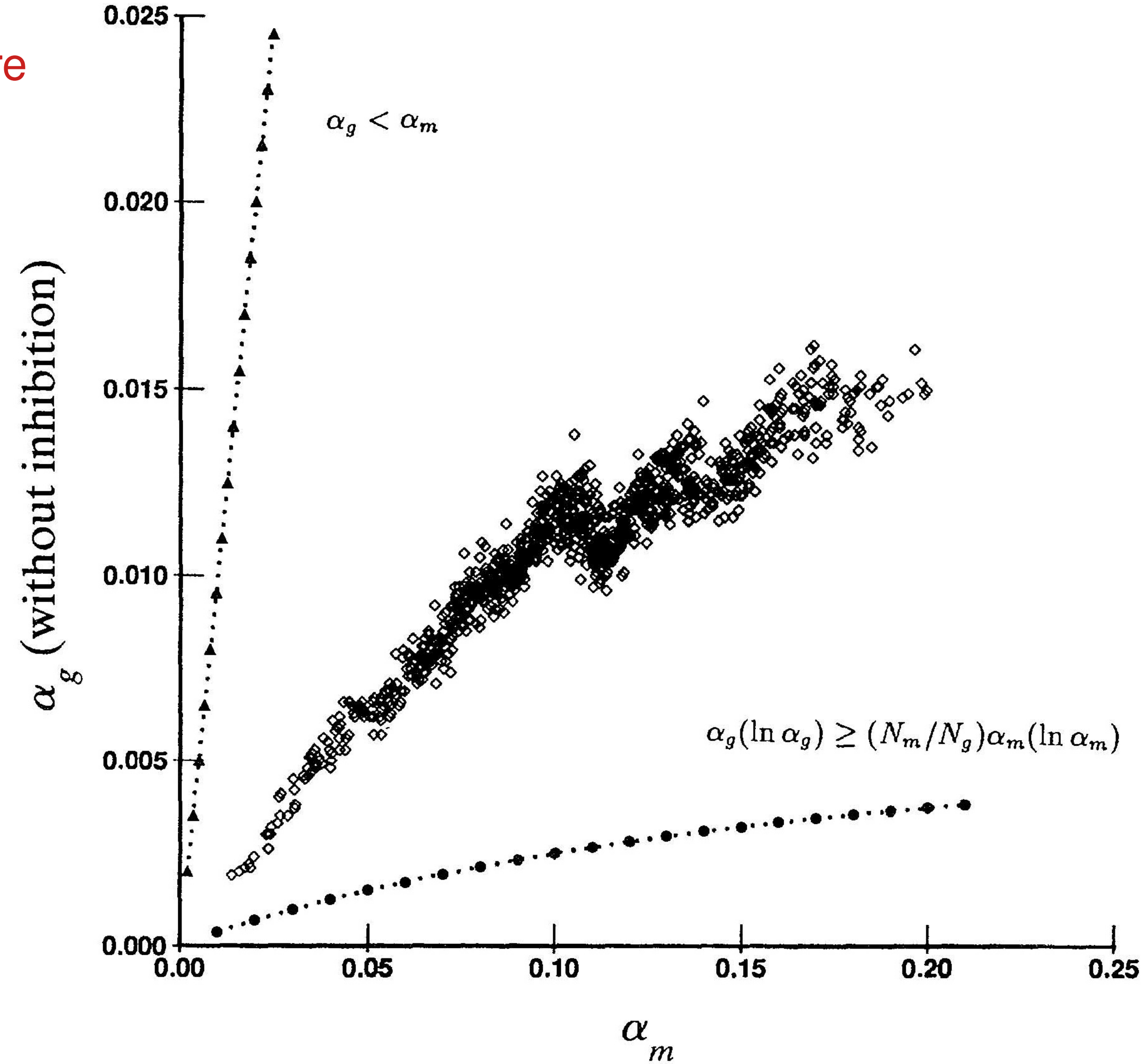
Albus

Golgi Cell Estimate of Granule Cell Activity

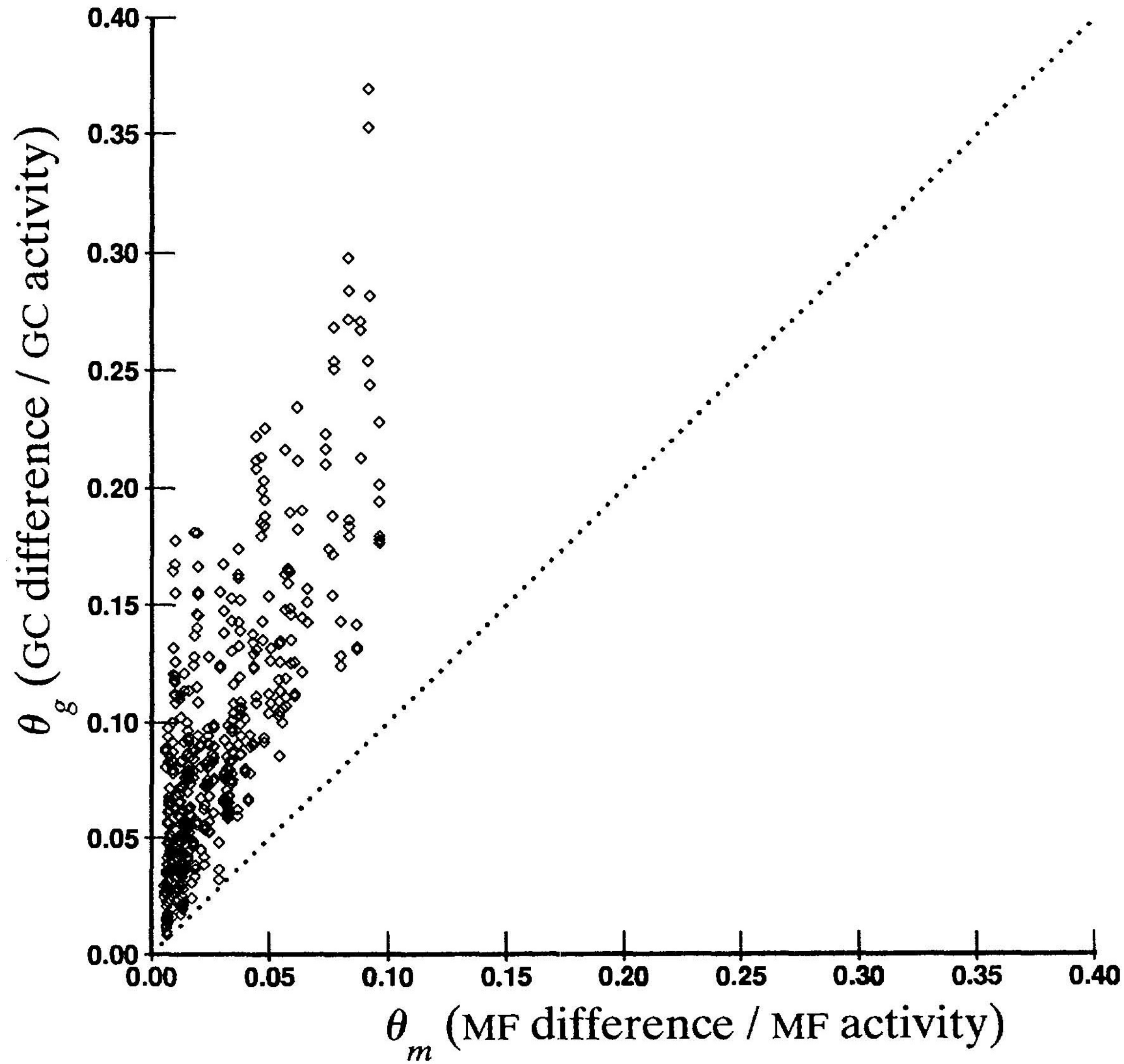


Golgi Cell Regulation of Granule Cell Activity

Note:
scales are
different!



Granule Cells Separate Patterns



Pattern Separation by Granule Cells

Let's look at how two patterns are transformed by the granule cells.

Mossy fibers: input pattern.
Parallel fibers: output pattern.

Mossy Fibers		Parallel Fibers
$\alpha_M = 3/6 = 0.5$		$\alpha_G = 4/10 = 0.4$
1 1 1 0 0 0	$\rightarrow$	**1 0 1 1 0 0 0 1 0 0**
0 1 1 0 0 1	$\rightarrow$	**0 0 1 0 1 1 0 0 0 1**
$\theta_M = 2 / 6 = 0.33$		$\theta_G = 6 / 8 = 0.75$

Patterns have become <u>more</u> <u>sparse</u>: $\alpha_G < \alpha_M$

Patterns have also become <u>more</u> <u>distinct</u>: $\theta_G > \theta_M$.

Tyrell & Willshaw's Conclusions

- Marr's theory can be made to work in simulation.
- Memory capacity: 60-70 patterns can be learned by a Purkinje cell with a 1% probability of a false positive response to a random input.
- Several parameters had to be guessed because the anatomical data were not yet available.
- A few of his assumptions were wrong, e.g., binary synapses.
- But the overall idea is probably right.
- The theory is also compatible with the cerebellum having a role in classical conditioning.

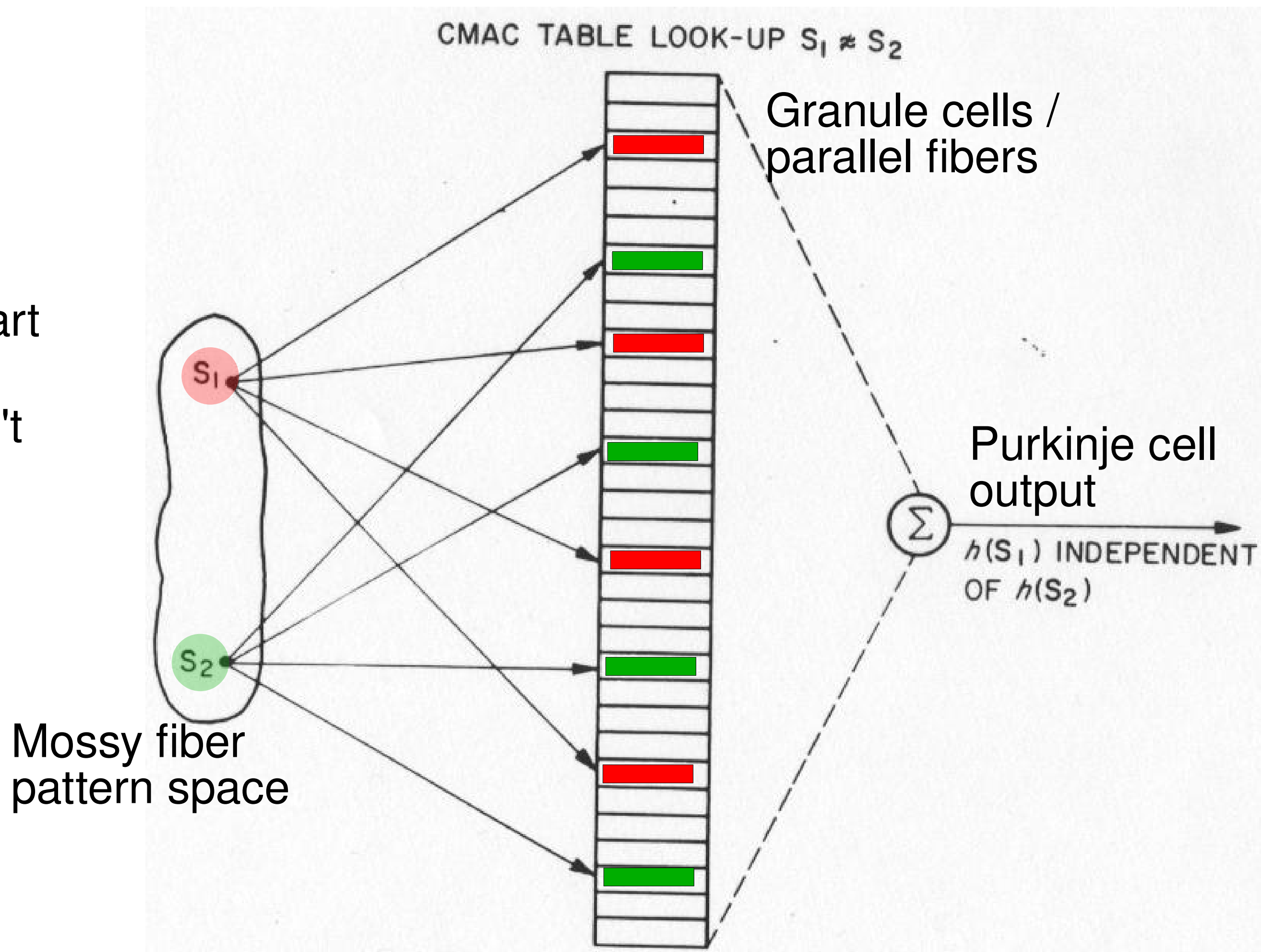
Marr's 3 System-Level Theories

- Cerebellum
  - Long-term memory but strictly “table lookup”.
  - Pattern completion from partial cues not desirable
- Hippocampus
  - Learning is only temporary (for about a day), not permanent.
  - Retrieval based on partial cues is important.
- Cortex
  - Extensive recoding of the input takes place: clustering by competitive learning.
  - Hippocampus used to train the cortex during sleep.

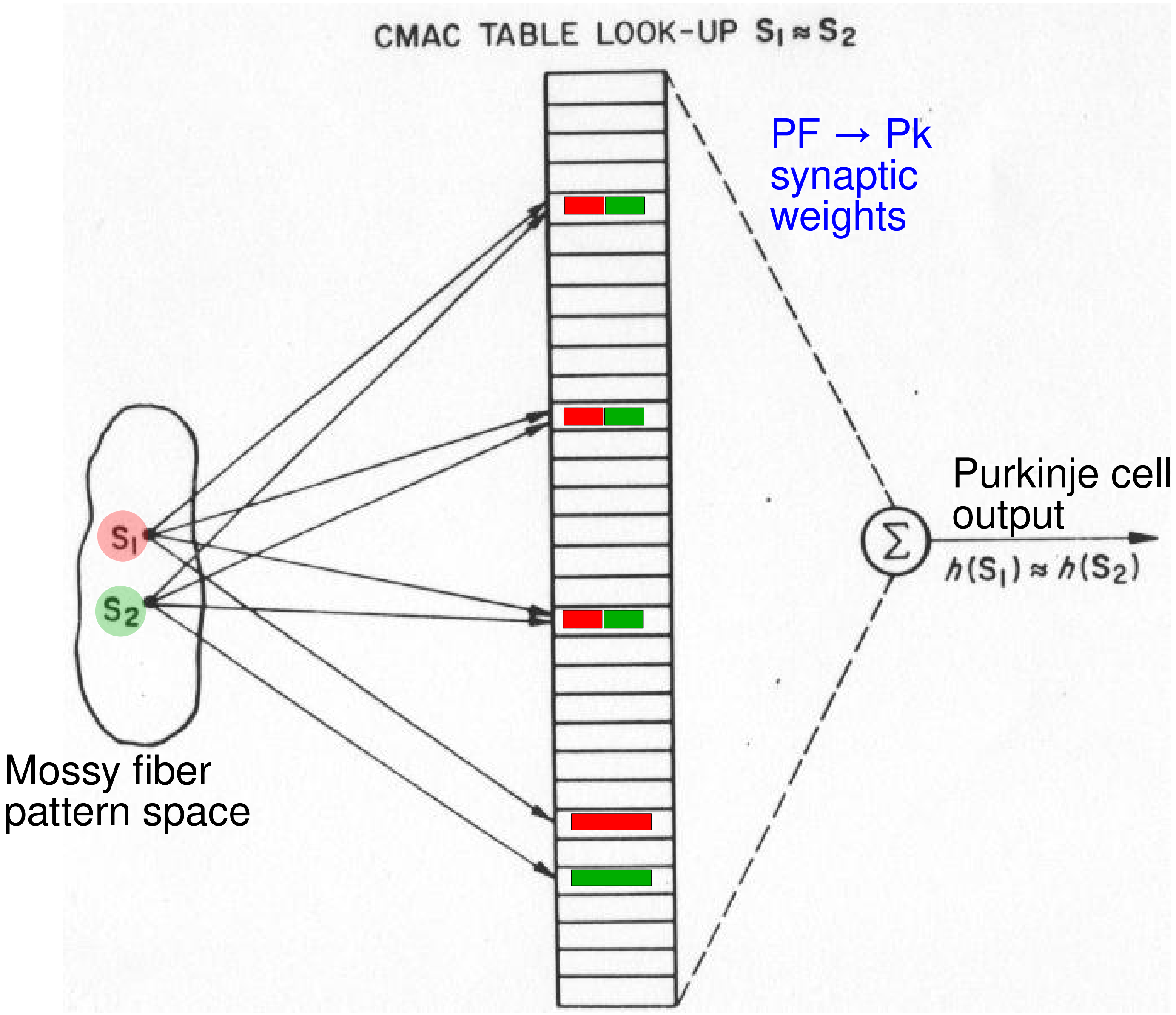
Albus' CMAC Model

- Cerebellar Model Arithmetic Computer, or
  Cerebellar Model Articulation Controller
- Function approximator using distributed version of table lookup. In machine learning this is called “kernel density estimation”.

S_1 and S_2 far apart in pattern space: table entries don't overlap.



Similar Patterns Share Representations



Learning a Sine Wave

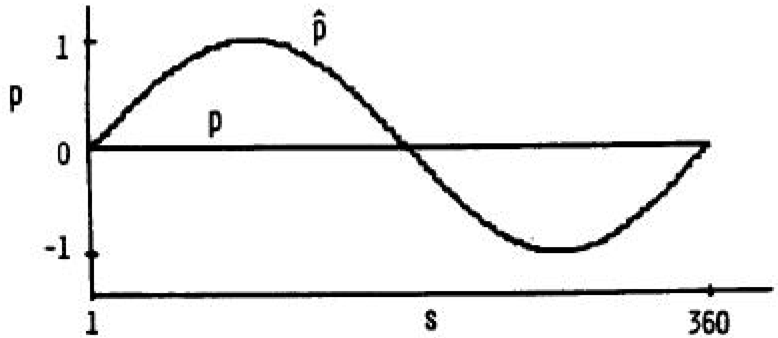


Fig. 1 ρ is the output from a one-input CMAC memory prior to any data being stored. $\hat{\rho}$ is the desired output. For this case the maximum error between ρ and $\hat{\rho}$ is 1.0 and the r.m.s. error is 0.707.

Fig. 3 After two data storage operations. Maximum error = 0.87 and r.m.s. error = 0.530.

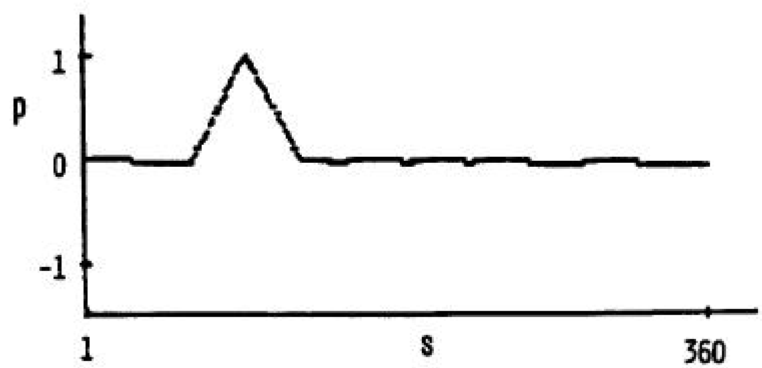


Fig. 2 The output of the CMAC memory after a single error correction data storage operation. ρ was set equal to 1.0 at s = 90. Maximum error is still 1.0 (at s = 270) and r.m.s error is now 0.625.

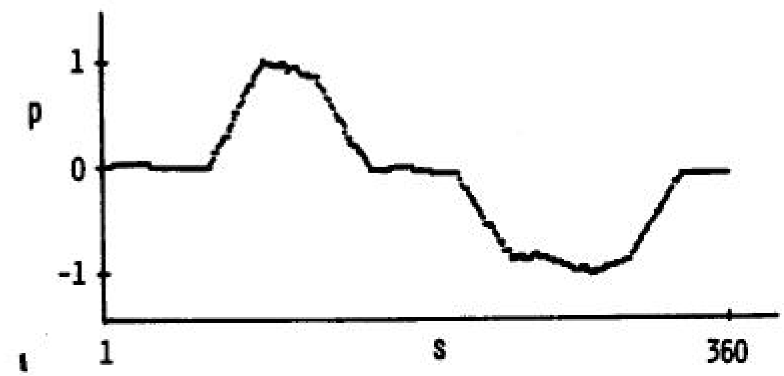


Fig. 4 After five data points are stored. Maximum error = 0.84 and r.m.s. error = 0.313.

Learning a Sine Wave

Fig. 5 After nine data points are stored. Maximum error = 0.33 and r.m.s. error = 0.081.

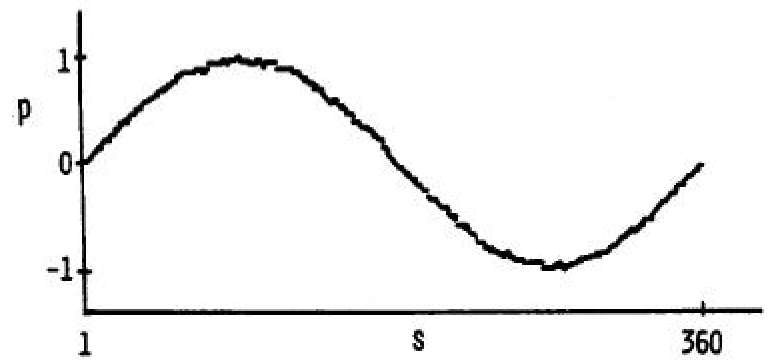


Fig. 6 After sixteen data points are stored. Maximum error = 0.09 and r.m.s. error = 0.033.

Learning 2D Data

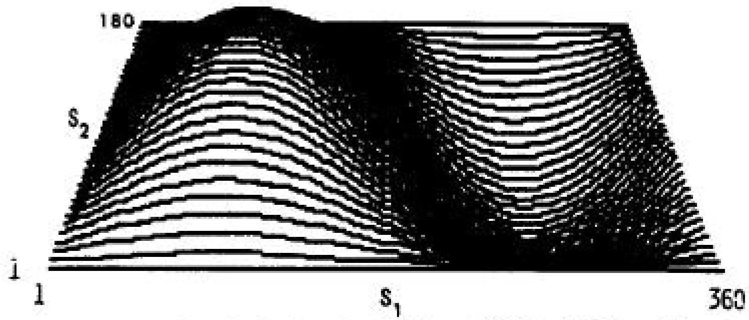


Fig. 7 A plot of a desired output $\hat{p}$ for a CMAC with two inputs.
$\hat{p} = \sin\left(\frac{2\pi s_1}{360}\right)\sin\left(\frac{2\pi s_2}{360}\right)$

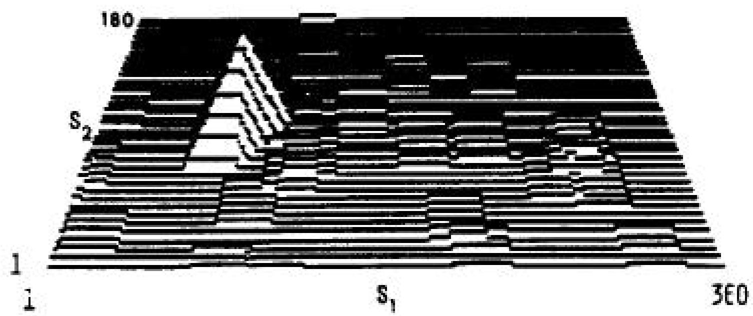


Fig. 8 The output of a two-input CMAC memory after a single error correction data storage operation. p was set equal to 1.0 at $s_1 = 90$, $s_2 = 90$.

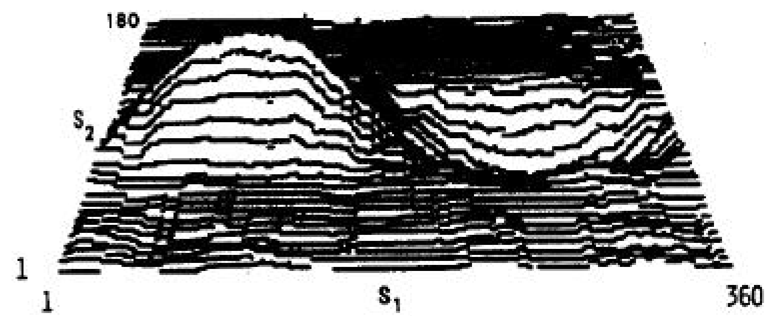
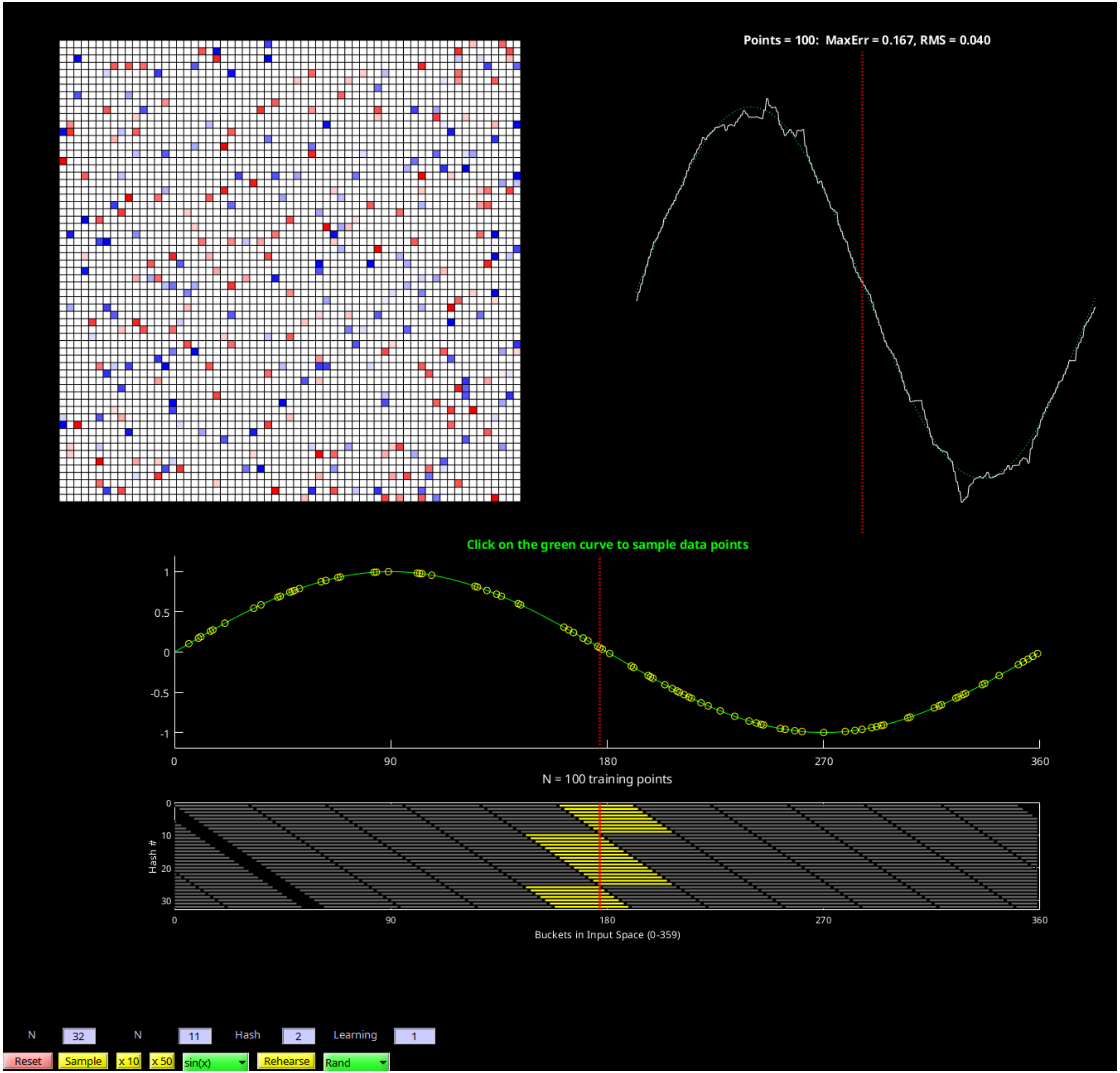
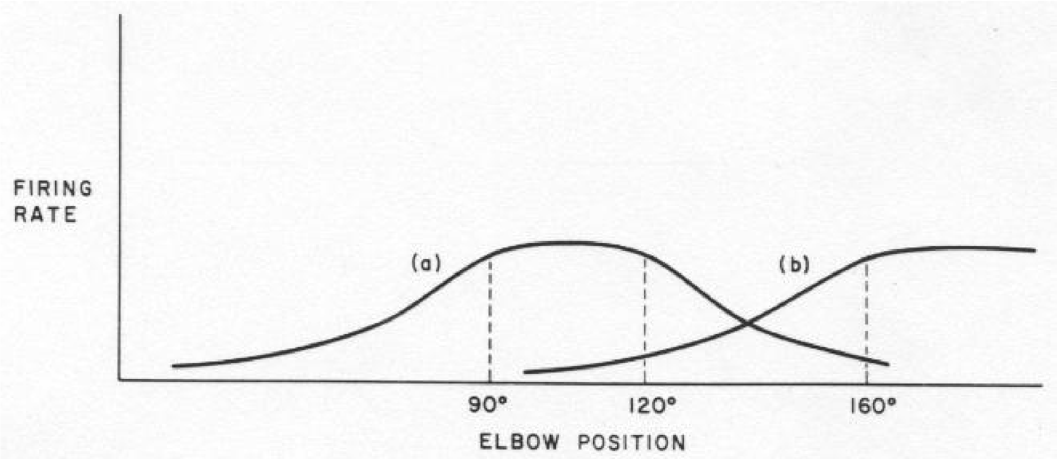
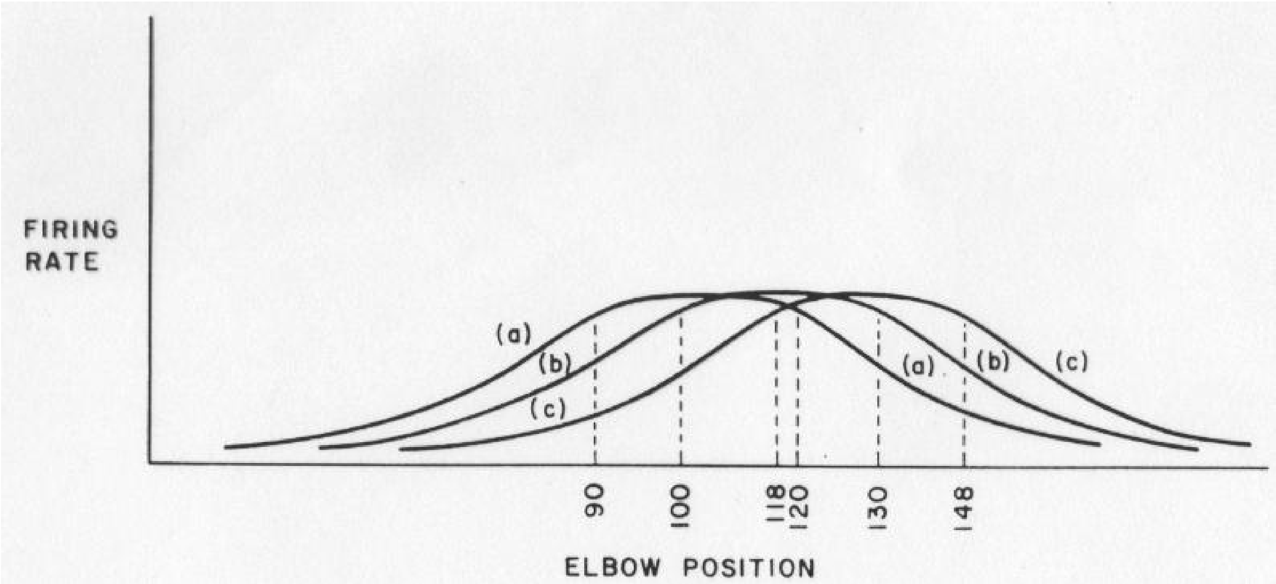


Fig. 9 The output of a two-input CMAC memory after sixteen data points were stored. A cross section of this figure in the $s_1 = 90$ plane is identical to Fig. 6.

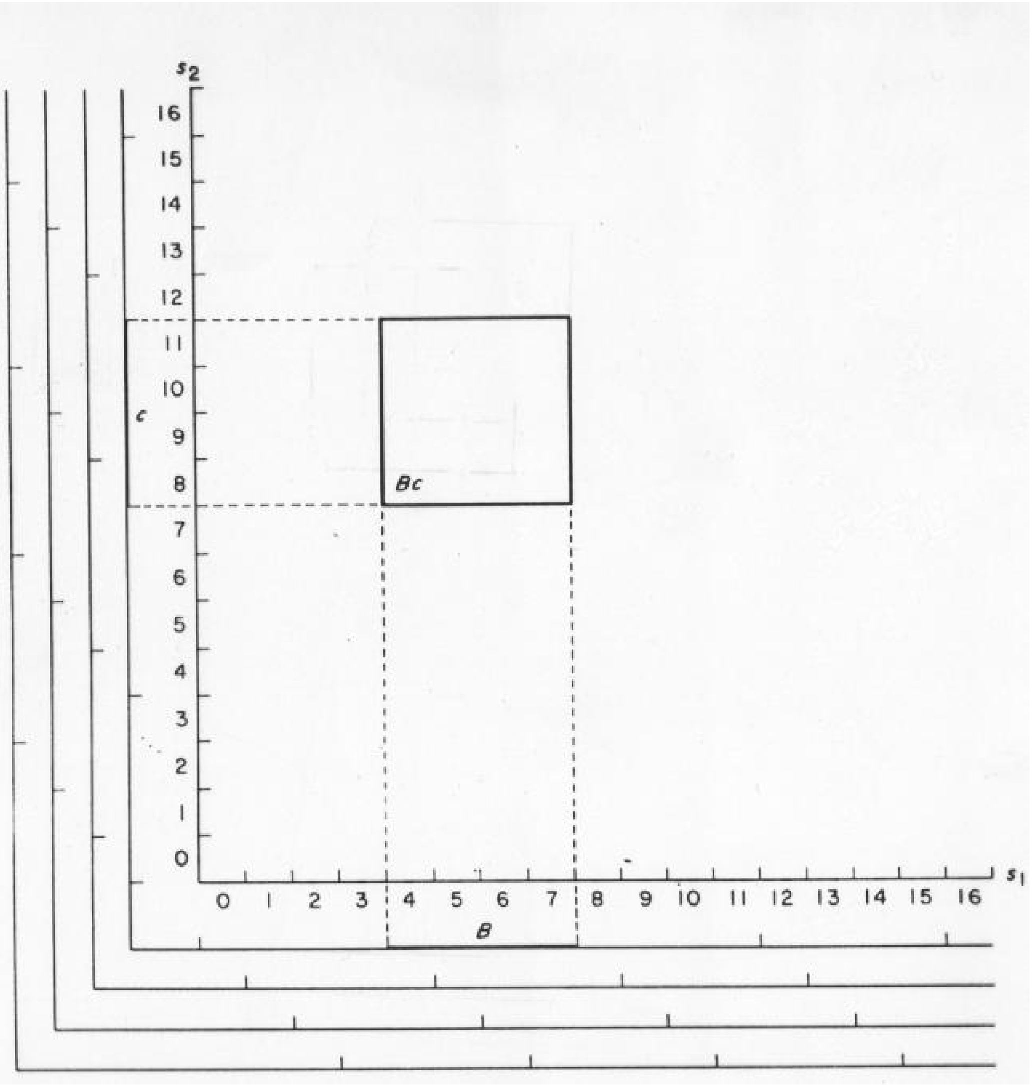

Points = 100: MaxErr = 0.167, RMS = 0.040
Click on the green curve to sample data points
1
0.5
0
-0.5
-1
0
90
180
270
360
N = 100 training points
Hash #
Buckets in Input Space (0-359)
N
32
N
11
Hash
2
Learning
1
Reset
Sample
x 10
x 50
sin(x)
Rehearse
Rand

Coarsely-Tuned Inputs Resemble Mossy Fibers

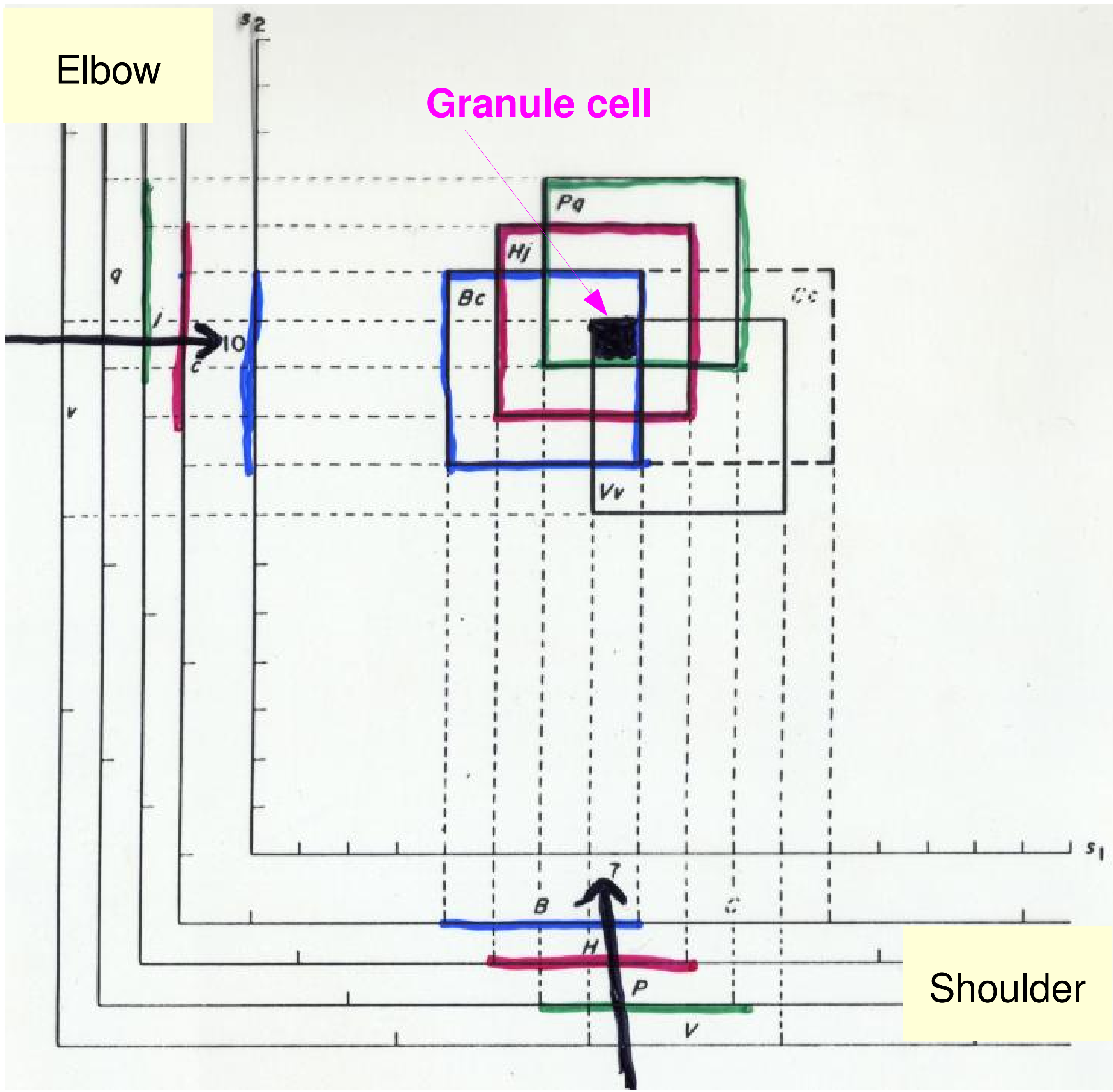




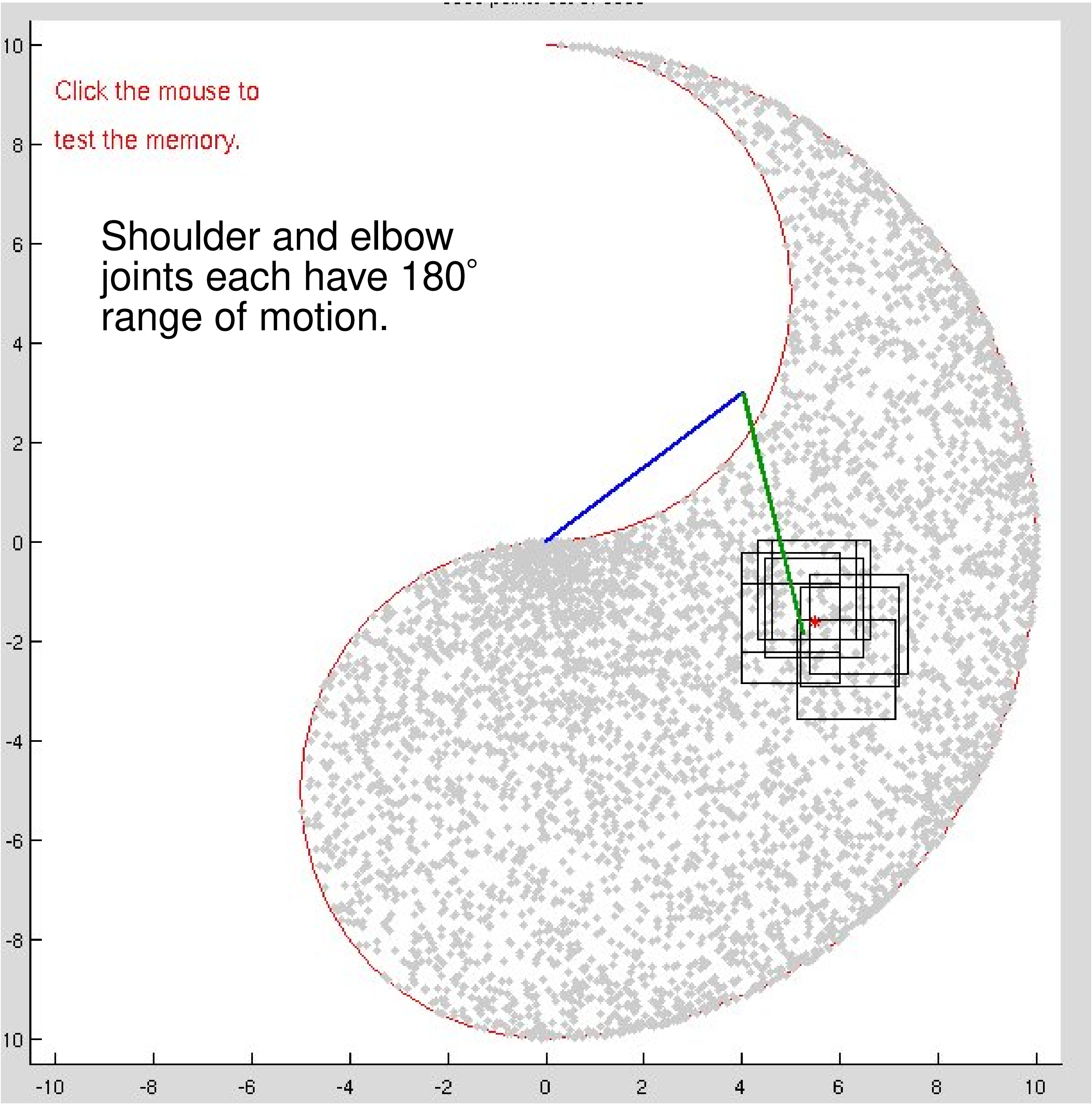
Coarse Tuning in 2D



Coarse Coding Using Overlapped Representations

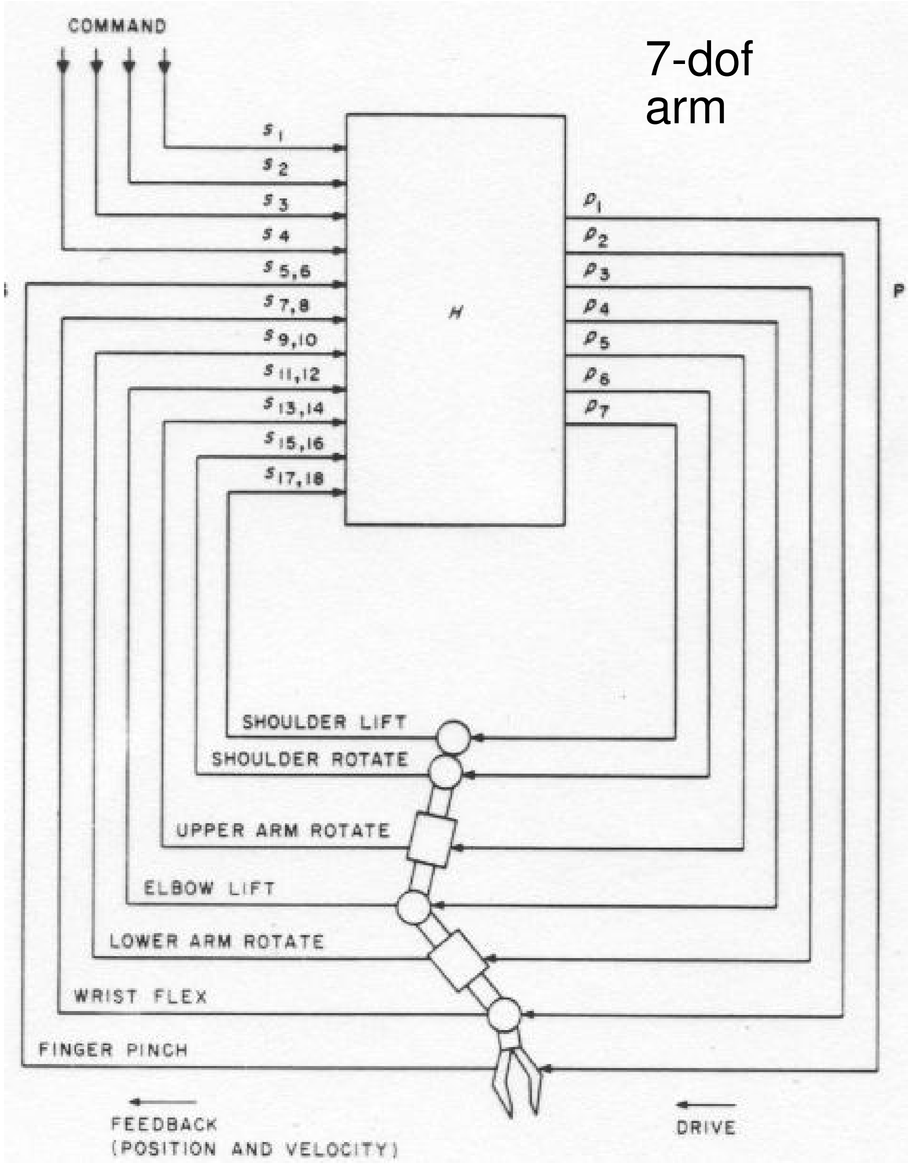


2D Robot Arm Kinematics



Higher Dimensional Spaces

Motor control is a high dimensional problem.



CMAC Learning Rule

1. Compare output value *p* with desired value *p**.
2. If they are within acceptable error threshold, do nothing.
3. Else add a small correction Δ to every weight that was summed to produce *p*:

g is a gain factor (learning rate) ≤ 1
A is the set of active weights

$$\Delta = g \cdot \frac{p^* - p}{|A|}$$

If $g=1$ we get one-shot learning.
Safer to use $g<1$ to ensure stability.

CMAC = LMS (Least Mean Square) Learning

- CMAC learning rule:

$$\Delta = g \cdot \frac{p^* - p}{|A|}$$

Implicit: rule only applies
to active units
(units in set A)

- LMS learning rule:

$$\Delta w_i = \eta \cdot (d - y) \cdot x_i$$

Explicit: learning
rate depends on
unit's activity level

- Same rule!

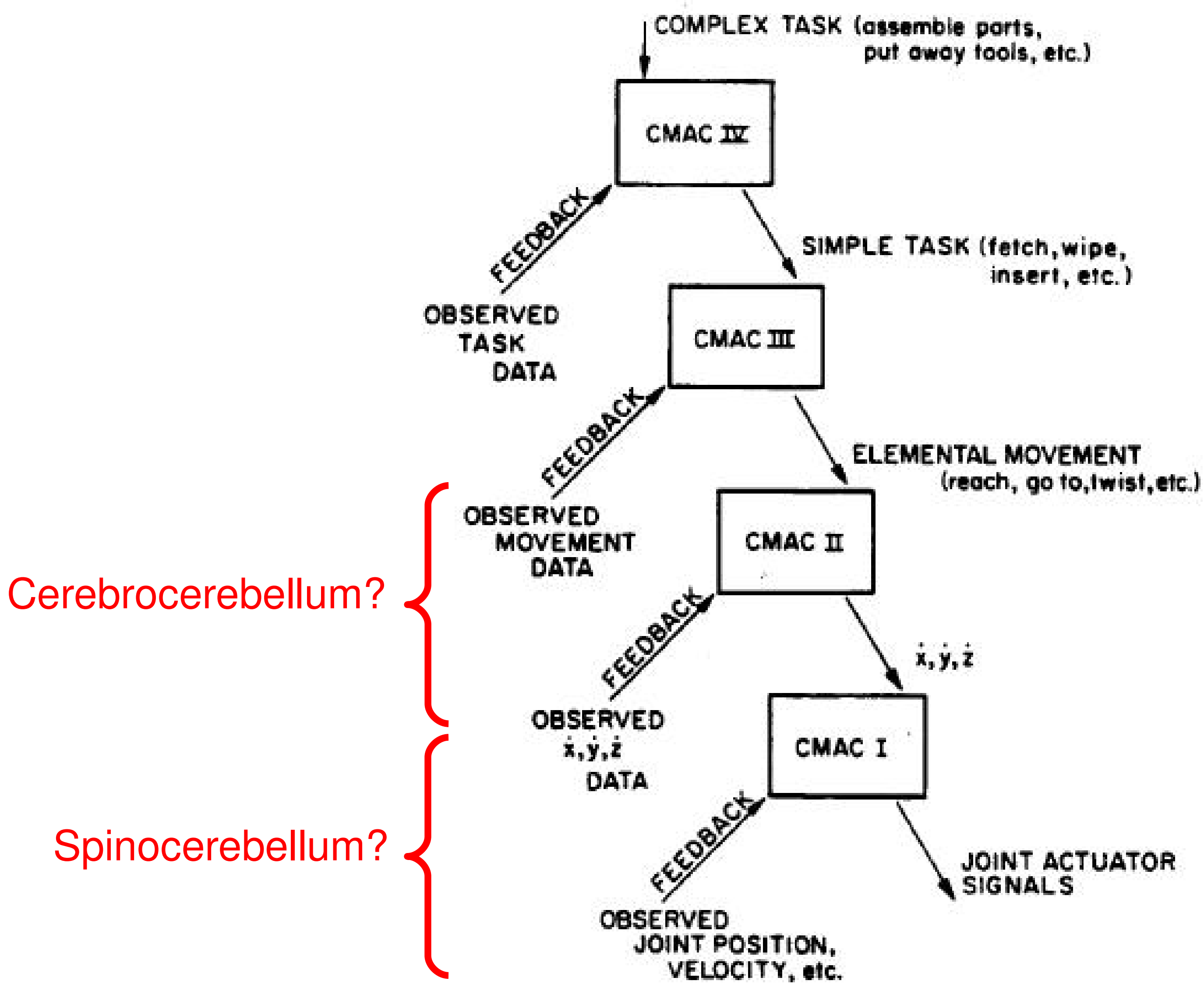
- LMS could be used to store linearly independent patterns in a matrix memory.

Albus: Why Should Purkinje Cells Use LTD?

1. Learning must be Hebbian, i.e., depend on Purkinje cell activity, not inactivity.
2. Climbing fiber = error signal.
   Climbing fiber fires → Purkinje cell should <u>not</u> fire.
3. Parallel fibers make excitatory connections onto Purkinje cells.

So: <u>reducing</u> the strength of the parallel fiber synapse when the climbing fiber fires will reduce the Purkinje cell's firing.

Application to Higher Order Control?



Compare Marr and Albus Models

Marr:

- Focus on single Purkinje cell recognizing N patterns
- Binary weights (incorrect)
- Binary output

Albus:

- Focus on PCs collectively approximating a function
- Continuous weights
- Continuous-valued output

Both use granule cells to recode input, decrease overlap.

Marr:

- Assumes learning by LTP

Albus:

- Requires learning by LTD

Both use static input and output patterns; no dynamics.

Newer Simulations using GPUs

- Mauk lab (2013): large scale simulation of cerebellum
  - 1024 mossy fibers; 1024 Golgi cells
  - 2^{20} (1,048,576) granule cells (vs. 50 billion in humans)
  - 32 Purkinje cells
  - 128 basket cells; 512 stellate cells
  - Simulated on an Nvidia GTX580 GPU
  - Eyeblink conditioning, pole balancing tasks

- Yamazaki & Igarashi (2013): real-time spiking simulation
  - 102,000 granule cells
  - 1024 Golgi cells, 16 Purkinje cells, 16 basket cells
  - Runs in real-time on Nvidia GeForce GTX580
  - Robot arm control application

Complications

- PF → Pk synapses show LTP as well as LTD
- Connectivity is more complex than these models provide for:
  - Pk cells project to other Pk cells
  - Deep cerebellar nuclei (DCN) cells project to Golgi cells
  - Deep cerebellar nuclei cells inhibit cells in the inferior olive
  - Inferior olive cells are electrotonically coupled
  - Unipolar brush cells excite granule cells
- Plasticity is not limited to PF → Pk synapses
  - Plasticity of connections onto interneurons
  - Plasticity within DCN
- DCN is complex
  - At least 6 cell types
  - Multiple neurotransmitters (glutamate, GABA, glycine)

Experimental Issues to Consider

Why do some papers report results that conflict with others?

- It's easier to record in slice than in intact animals.
  - But slices are missing some input pathways because those axons get severed.
  - Slice experiments require artificial stimuli; experiments done with intact animals can use natural stimuli.
- Recording in intact animals may require anesthesia.
  - Anesthesia alters the behavior of neurons.
- Although the cerebellum is common to vertebrates, there may be differences between species.