CN510: Principles and Methods of Cognitive and Neural Modeling

Adaptive Resonance Theory

Lecture 23

Instructor: Anatoli Gorchetchnikov <anatoli@bu.edu>
Teaching Fellow: Rob Law <nosimpler@gmail.com>

Adaptive Resonance Theory (ART)

Grossberg, S. (1975). A neural model of attention, reinforcement, and discrimination learning. *Studies of Mind and Brain, Chapter* 6.

Employs many of the elements that are presented in the next (1976) paper in a more distilled way as ART

Primary motivation for ART: a need to stabilize bottom-up incremental learning

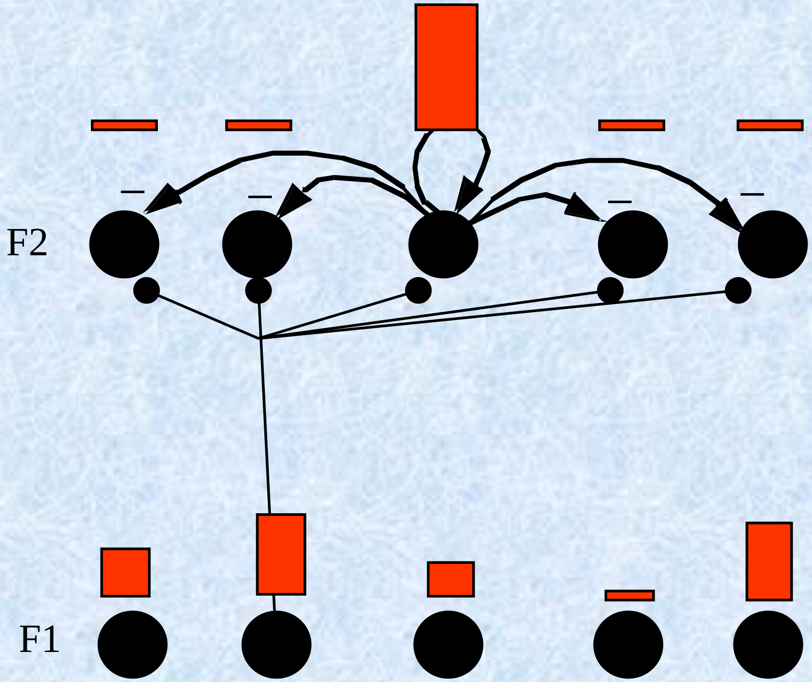
Recurrent Competitive Field

A set of neurons representing sensory inputs F1

A set of neurons representing input category F2

An instar (postsynaptically gated decay) learning between F1 and F2

$$\dot{w}_{ij} = \eta x_i y_j - \alpha y_j w_{ij}$$



Competition in F2

$$\frac{dy_j}{dt} = -Ay_j + \left(B - y_j\right)\left(f(y_j) + I_j\right) - \left(C + y_j\right)\sum_{k \neq j} f(y_k)$$

Why Categories?

Same response in F2 can represent many similar inputs in F1

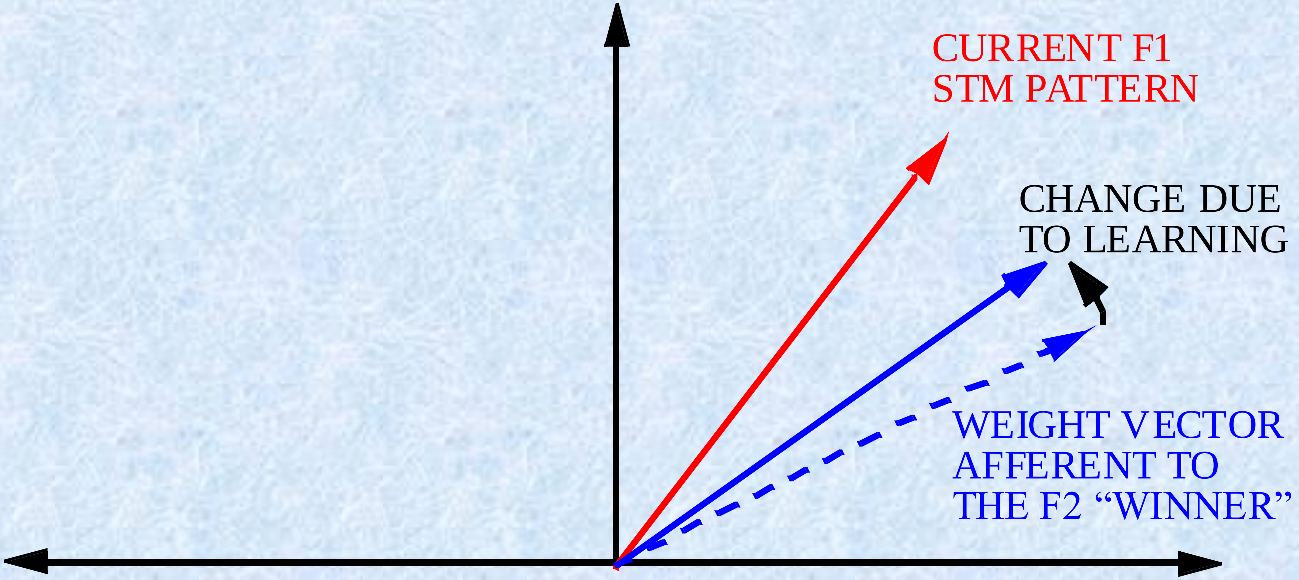
- Code compression and generalization
- Fast behavioral response: no need to create a new response for every new input

Categorical learning happens on various levels of cognition:

- Voice onset timing is interpreted in categorical fashion to distinguish voiced vs non-voiced consonants early in auditory perception
- Essentially any recognition task requires categorization

Issue with Stability in SOM

The afferent weight vector becomes more parallel to the current F1 STM pattern



Recall that we can set up a sequence of inputs so that the weight vector keeps rotating around and never stabilizes

Issues with Competitive Learning

Even with non-pathological data sequence it is possible that the weights will never stabilize, so the algorithm will never terminate

One approach is to reduce the learning rate, but then

- after some time we will not learn new patterns
- we will not be able to track changes through time

This leads to stability-plasticity dilemma

We want our system to be stable enough to remember previous patterns but plastic enough to learn new patterns

Suggested solution is an addition of a top-down adaptive filter

To Err Is Human

Lets suppose a pattern is presented to F1 and the wrong output is selected in F2

How can the system correct this error?

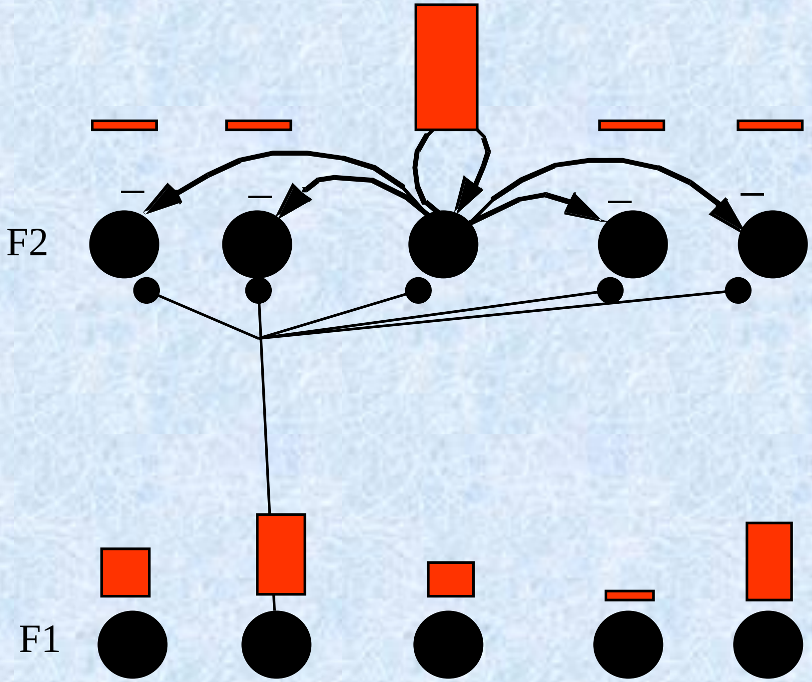
How can the system judge that the error has occurred?

During learning:

- statistics of the stimuli
- outcome feedback (utility)

After learning:

- consistency with prior learning



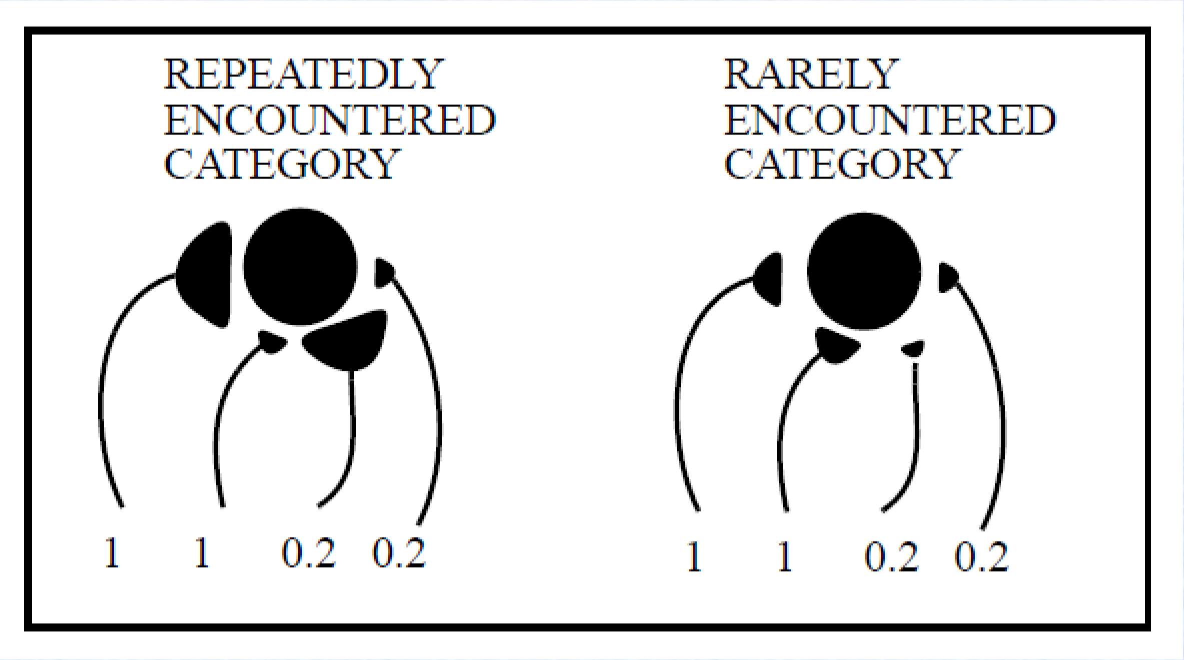
Why do we want to correct errors in the systems as fast as possible?

- Compressed code leads to a fast behavioral response, we do not want a wrong response
  - In fact, we would like the system to correct this error before producing a response (unlike supervised or reinforcement learning)
- Pairing of the input with a wrong category leads to recoding of the input, thus erasing the previously learned information

Why do we err?

Real world categories have

- Fuzzy boundaries
- Significant featural overlap
- Dense distributed coding
- Uneven frequency



Hit or Miss?

How can categorization error be detected and corrected without response feedback and even before any response happen?

The system already detects what is similar between current input and previous inputs that lead to the same response

The system also needs to monitor and detect what is different in the current input comparing to previous inputs that lead to the same response

In other words we need to monitor both the degree of match and the degree of mismatch

Suggested Solution

Add a mechanism that will check if the input pattern is good enough exemplar of the active category in F2

1. In addition to many-to-one learning of categories we add one-to-many (or to average of many) learning of expectation template
2. In addition to that we need a mechanism to compare the current input pattern with the expectation template

The latter mechanism cannot be at F2, we do not have featural representation of input there

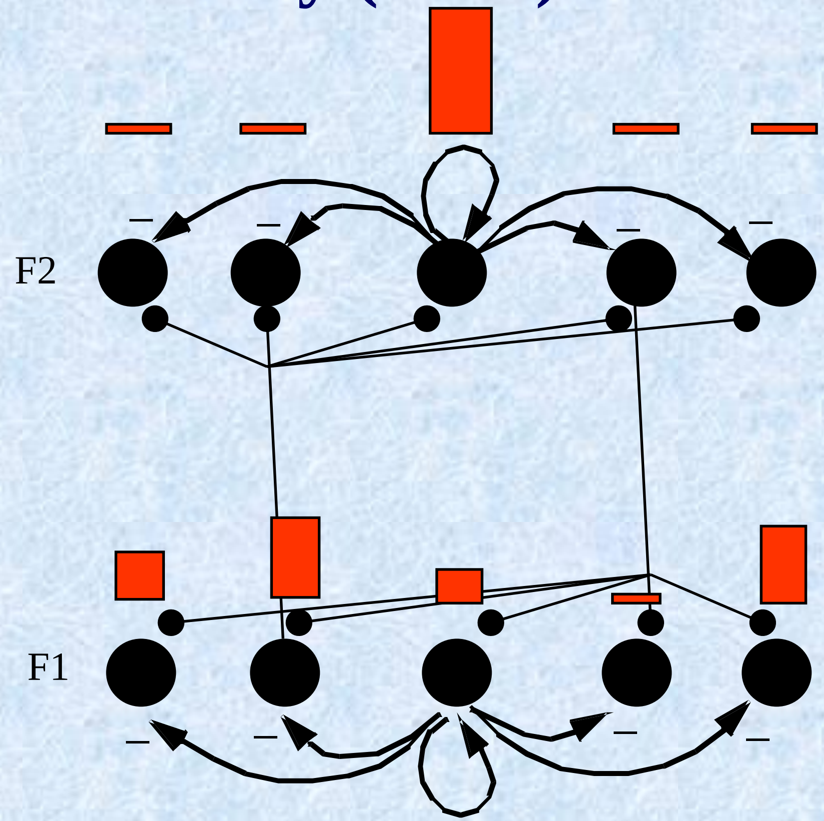
Can it be at F1?

Adaptive Resonance Theory (ART)

Now both F1 and F2 are RCFs, the network becomes almost symmetrical

But

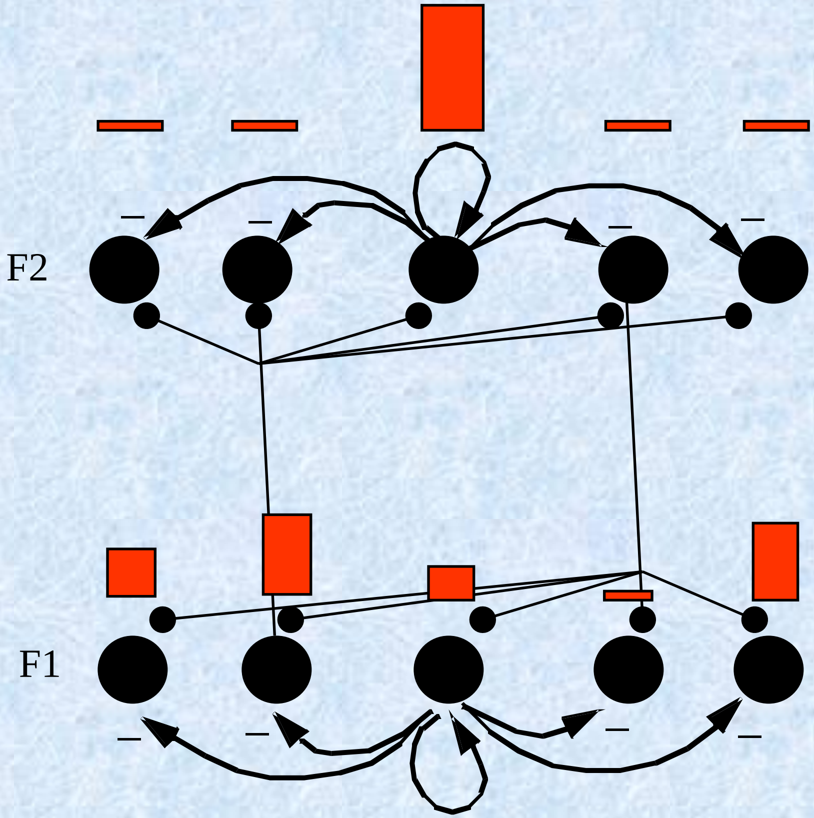
- F1 has linear signal function (preserves the pattern but normalizes it) while F2 has faster than linear signal function (WTA)
- Bottom up weights use instar, while top down weights use outstar, so both are gated by the F2 activity



$$\dot{w}_{ij} = \eta x_i y_j - \alpha y_j w_{ij}$$

$$\dot{w}_{ji} = \eta x_i y_j - \alpha y_j w_{ji}$$

- How feasible is this architecture?
- There is a lot of feedback in the brain
- Any sensory pathway through thalamus to cortex has a corresponding feedback from cortex to thalamus
- *Versace and Grossberg (2008)* developed SMART model that uses HH neurons, ART concepts, and modulatory reset



This paper got the “most cited” award in 2010

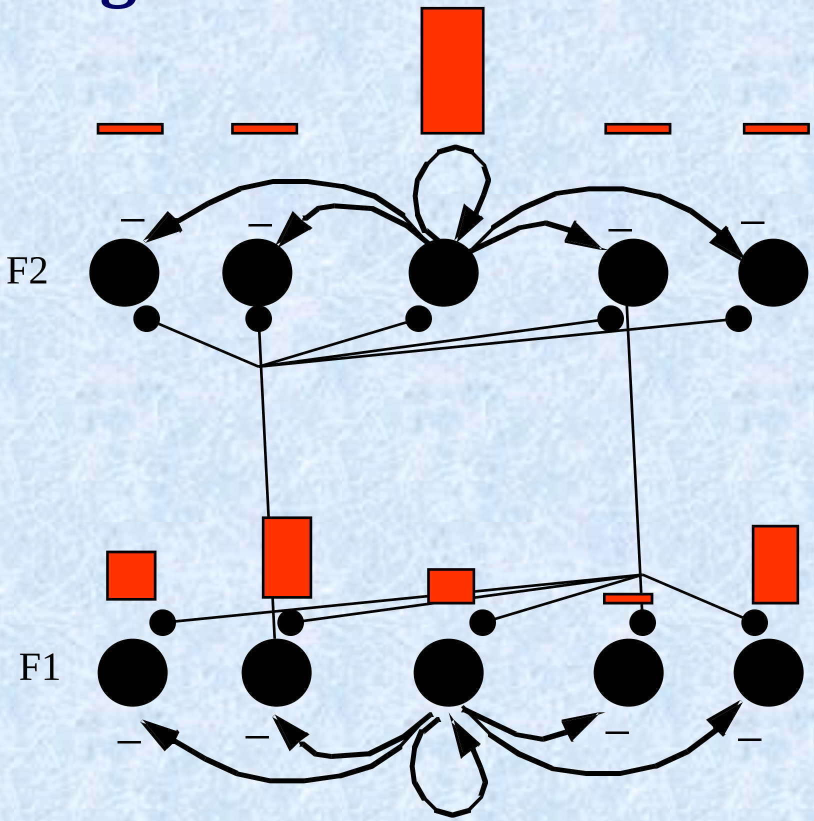
Learning

Two directions:

Feedforward (bottom-up) weights follow instar rule and learn the same way as in RCF

Feedback (top-down) weights follow outstar and are supposed to learn the same way as in outstar network

But: remember that outstar works well if and only if the activity in F1 is set from external source



$$\dot{w}_{ij} = \eta x_i y_j - \alpha y_j w_{ij}$$

$$\dot{w}_{ji} = \eta x_i y_j - \alpha y_j w_{ji}$$

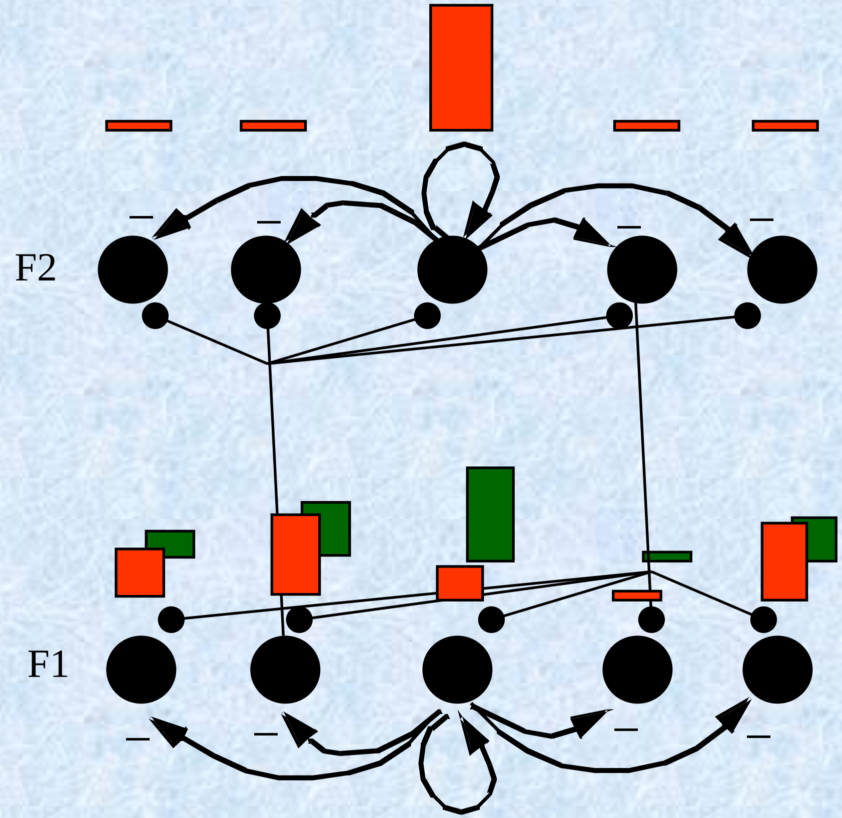
Signal Flow

Two directions:

Feedforward (bottom-up) activates the category nodes in F2 by activity in F1

Feedback (top-down):

- if there is a perfect match we are fine
- if there is a mismatch, then we will corrupt the input pattern

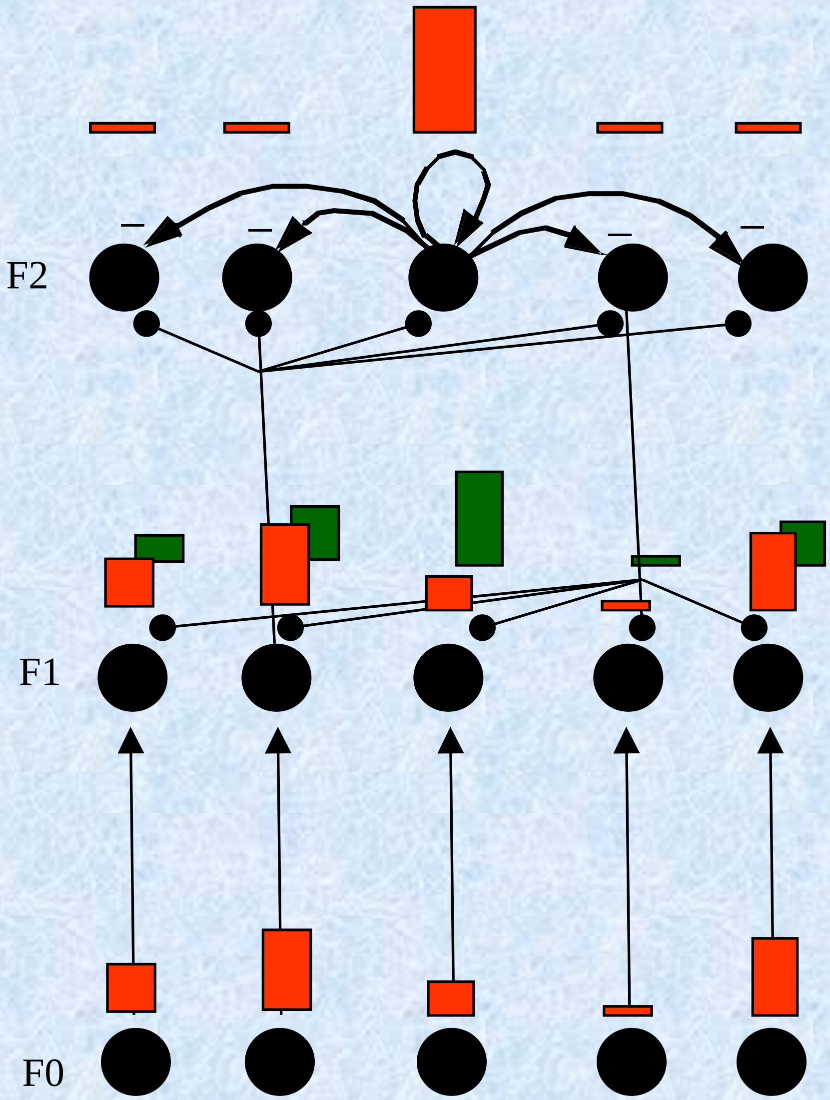


Good case: we kill the input

Bad case: we learn the corrupted pattern

How to Preserve the Original Input?

- Split the matching layer F1 from the input layer F0
- So now F0 is retina, F1 is LGN and F2 is V1
- Still does not prevent us from learning a corrupted mismatched pattern
- Multiple solutions:
  - *Grossberg (1980)* – Suppression of uniform inputs
  - *Carpenter et al* – 2/3 rule
  - SMART – cholinergic modulation



Suppression of Uniform Inputs

In feedforward competitive network if

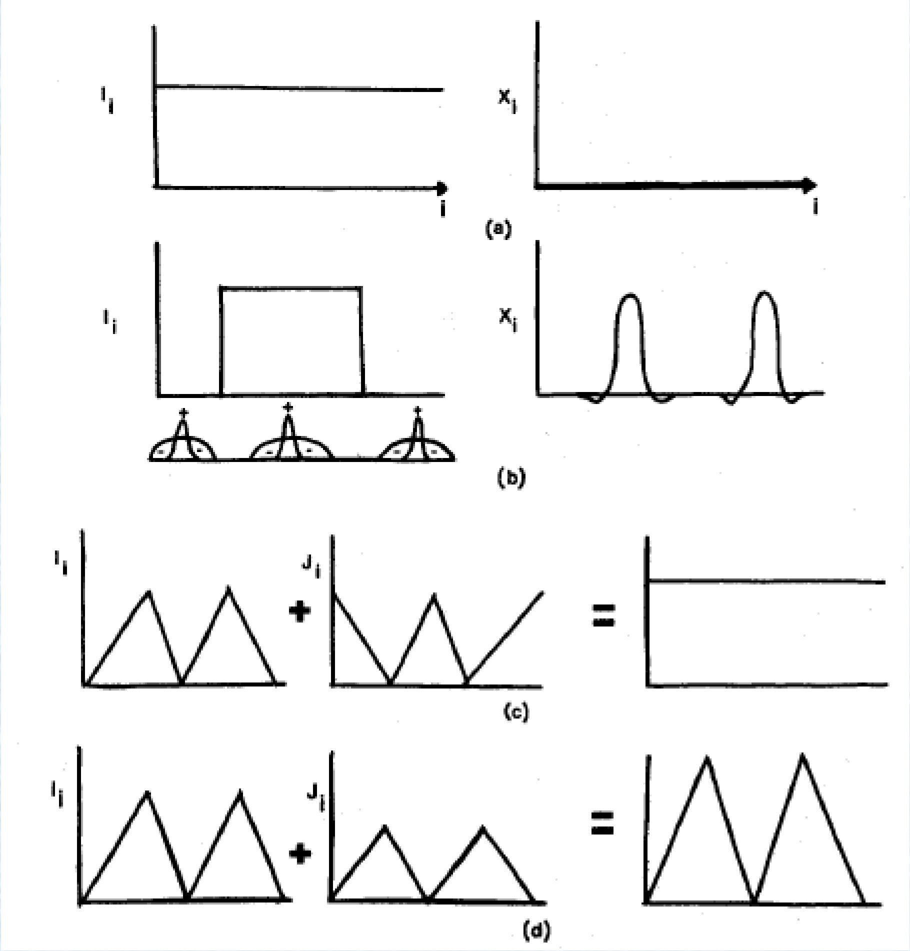
$$\frac{C}{B+C}=\frac{1}{N-1+1}=\frac{1}{N}$$

$$x_i=\frac{NI}{A+I}\left(\frac{I_i}{I}-\frac{1}{N}\right)$$

And if all inputs are uniform ($I_i=I/N$) then all $x_i=0$

(see lecture 6 for details)

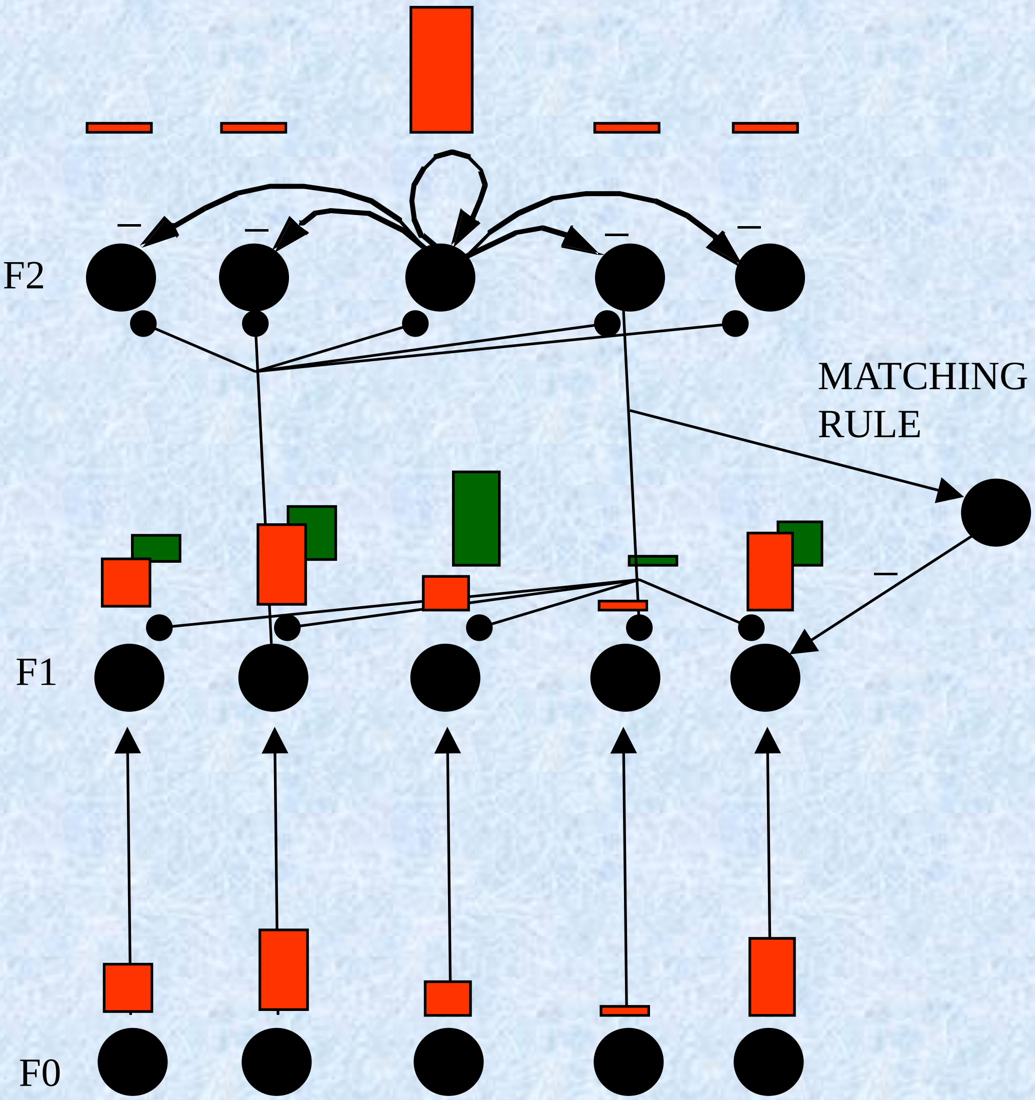
Note 1: feedforward net is different from RCF



Note 2: needs an exact opposite of the input for this to work, but how such a category can be selected in the first place??

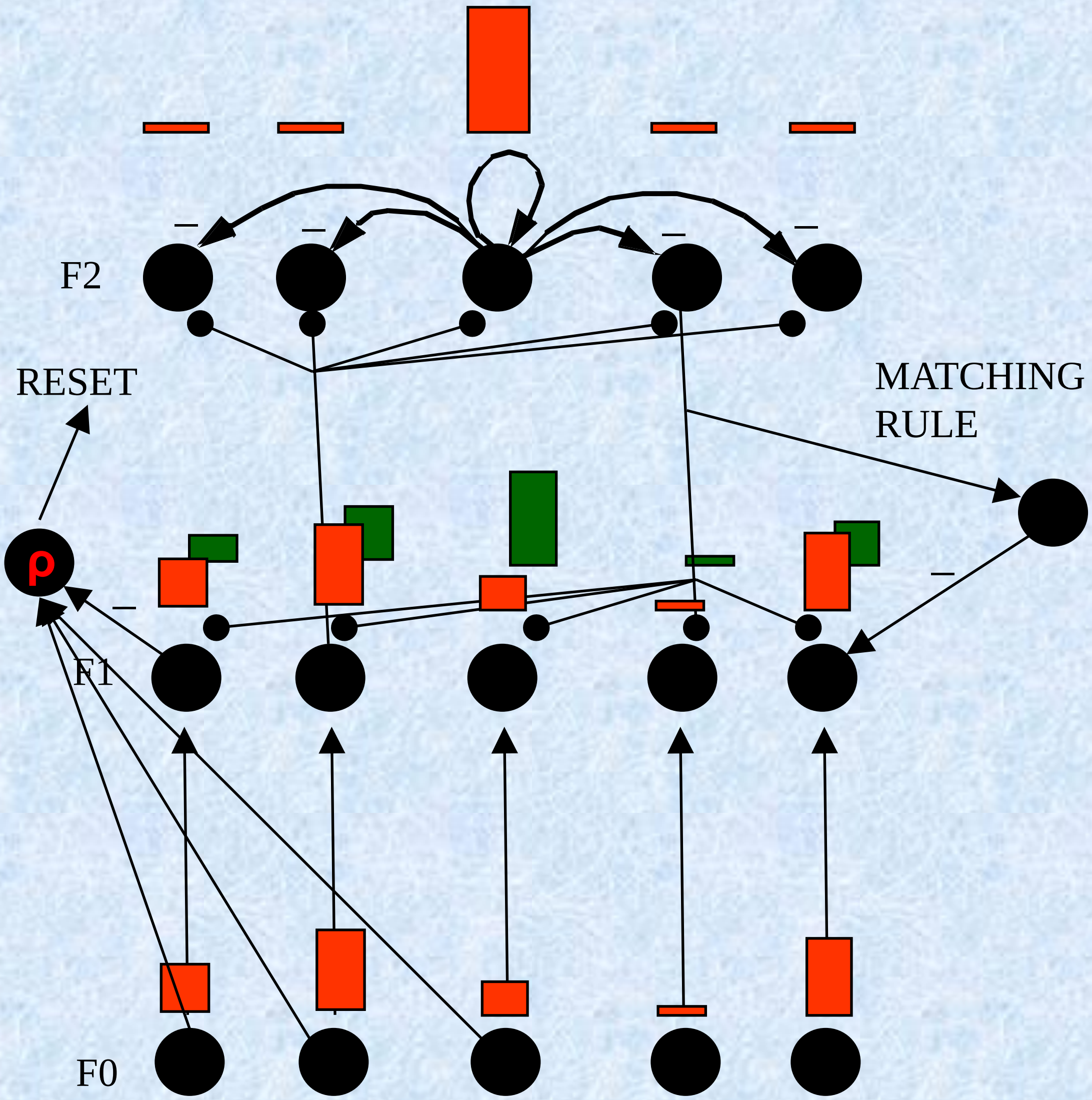
2/3 Rule for Binary Inputs (ART 1)

- When any category in F2 is active, then inhibition is provided to all F1 nodes
- Activity in F1 is the sum of bottom-up and top-down inputs minus inhibition
- Works perfectly for binary input patterns



Resetting The Category

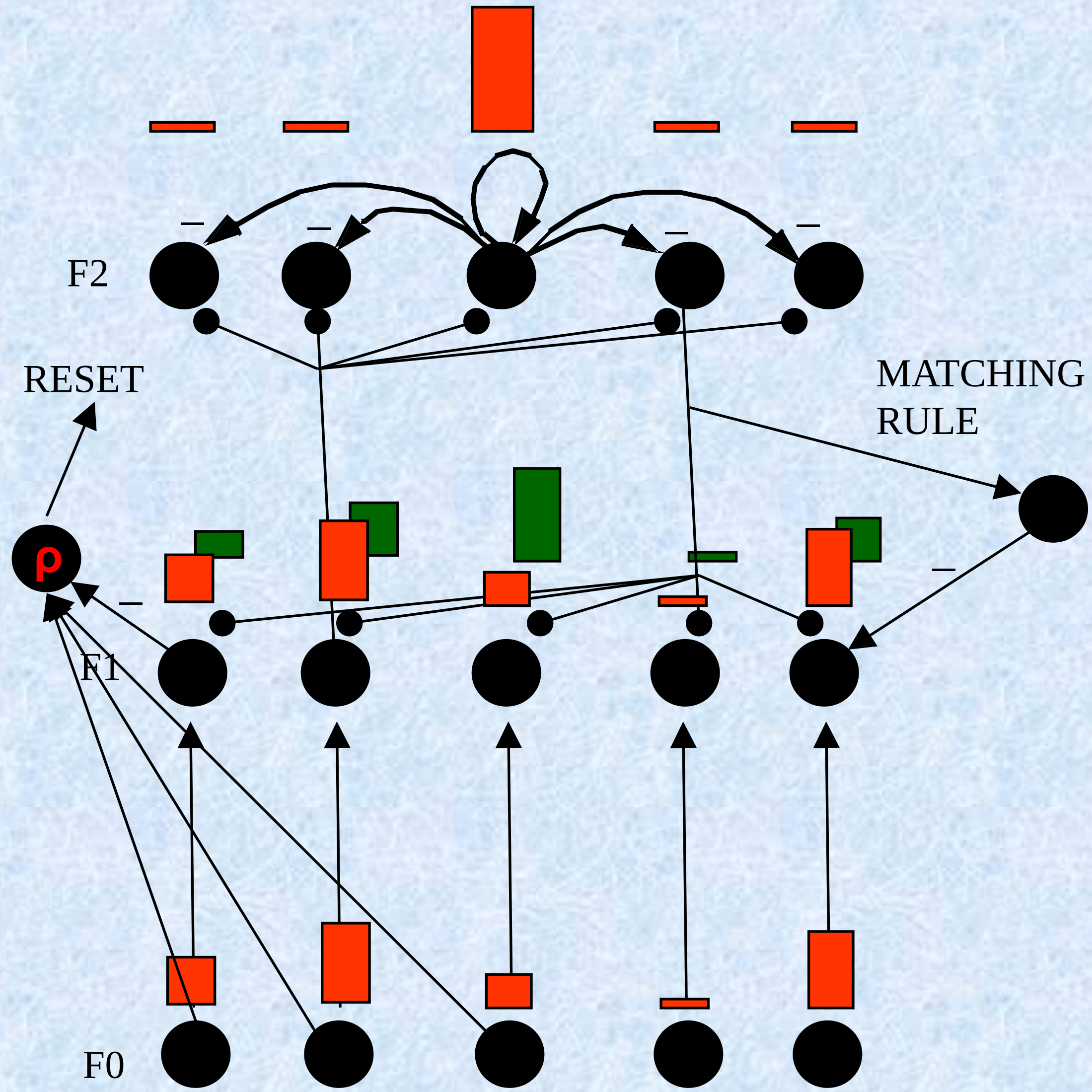
- If we have a good match in F1 we inhibit the reset
- But we do not want reset to be active when there is no input, so it is fed by cumulative input activity
- How good the match should be is determined by a vigilance parameter



Resetting The Category

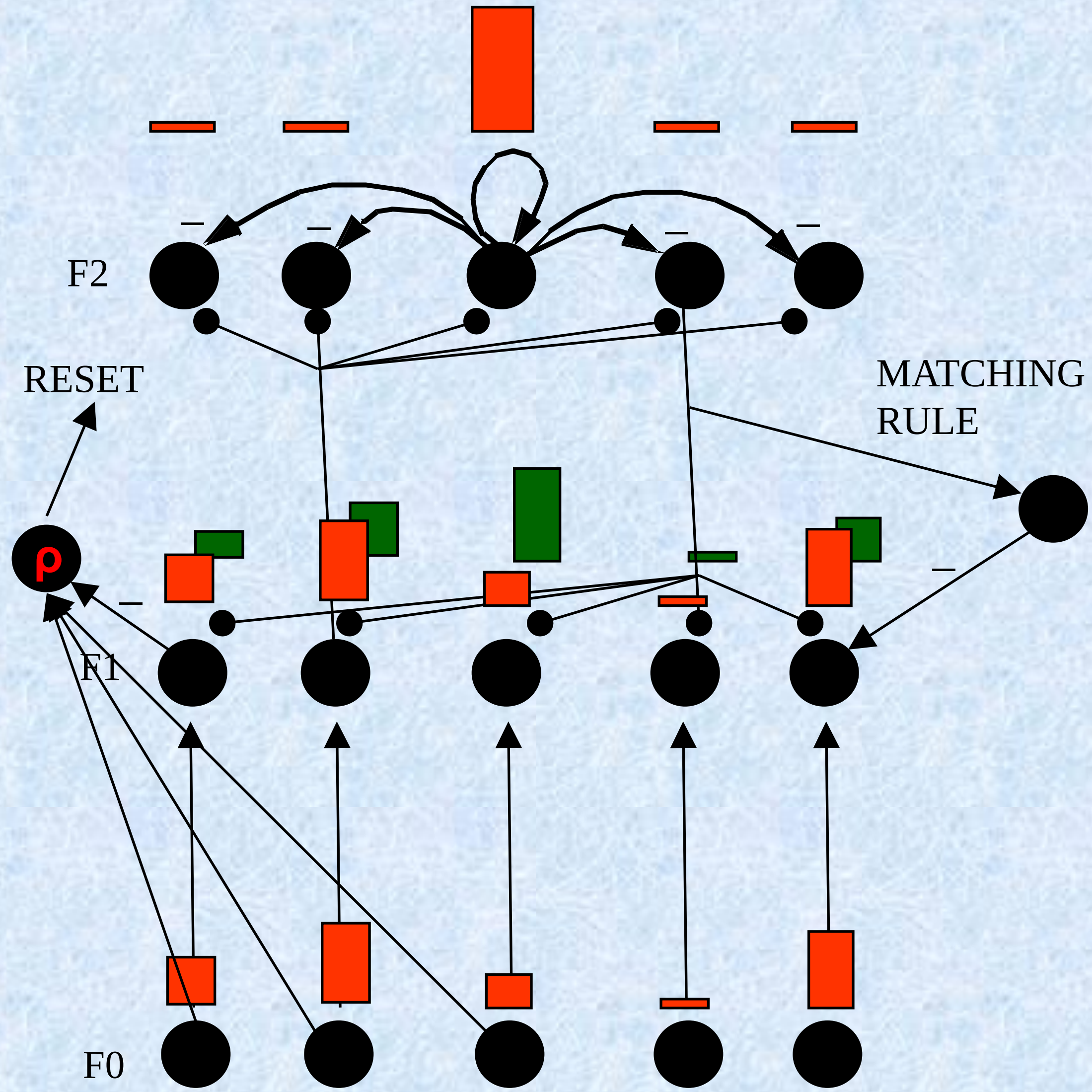
Reset has to be non-specific since we do not know in F1 which category was chosen in F2

Thus input has specific function (each node drives an F1 node) and non-specific function (all of them drive reset)



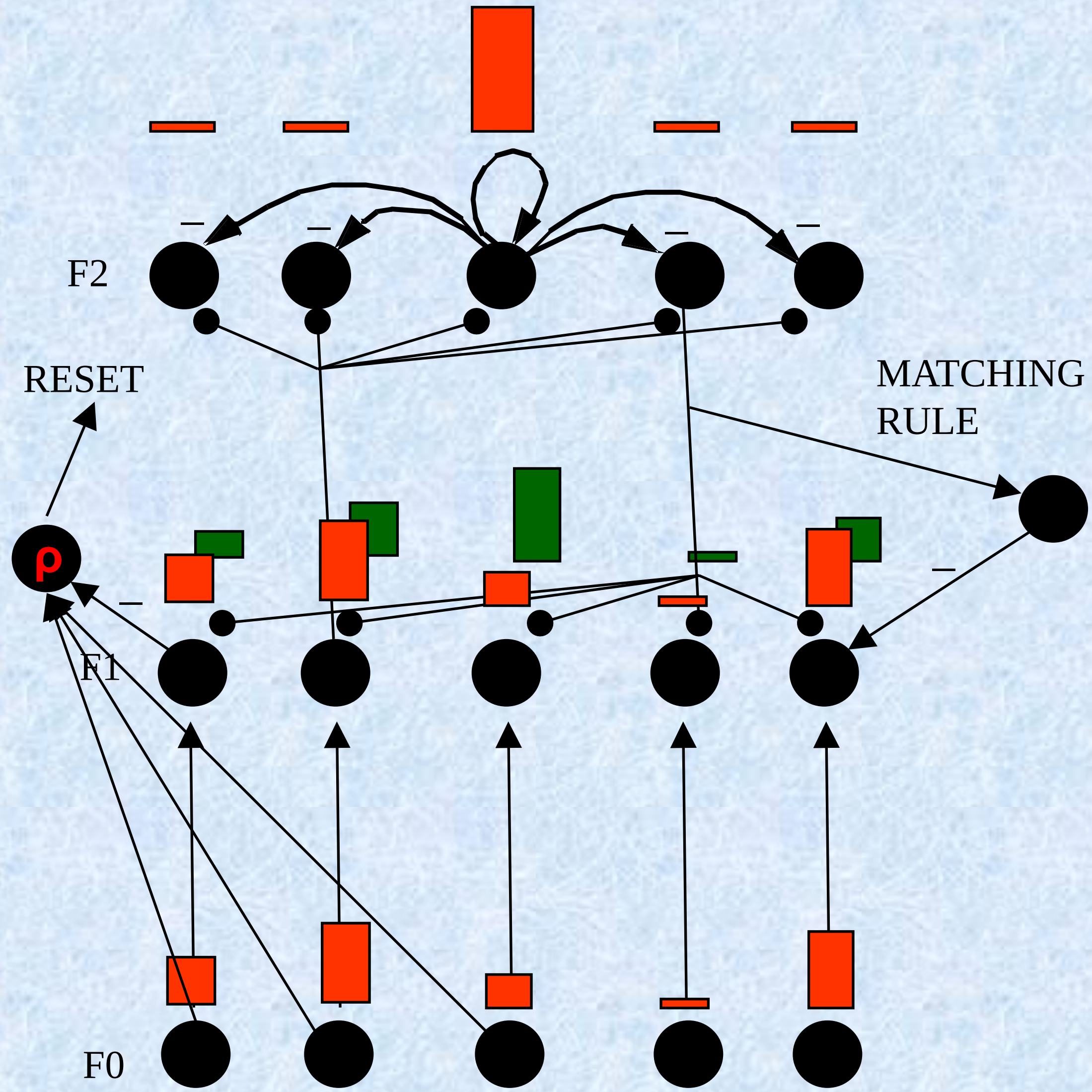
Priming in ART

- The system can be primed by category activation: know what to expect
- In this case the match or reset will happen faster: so the network will have faster response times



Vigilance Control

- Low vigilance leads to coarse categories
- High vigilance leads to fine categories
- Feedback given to a behavioral response can modify vigilance: this is where the reinforcement or supervision comes in



How to Kill a Wrong Winner?

A winner won because its weights were the best match to the pattern

What can prevent it from winning again after reset?

Given that our reset is non-specific?

Use prior activation to prevent it from being active again

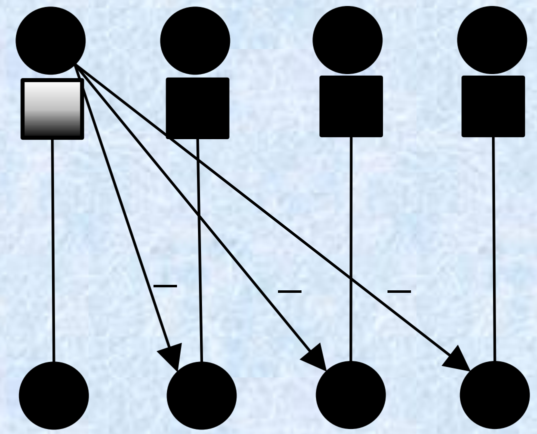
Originally proposed gated dipoles, in later ARTs simplified out

Gated Multipole

Split the F2 RCF into two layers with depletable transmitter connections between them

Initially all connections are at full strength, so the strongest match to the input inhibits the others and wins

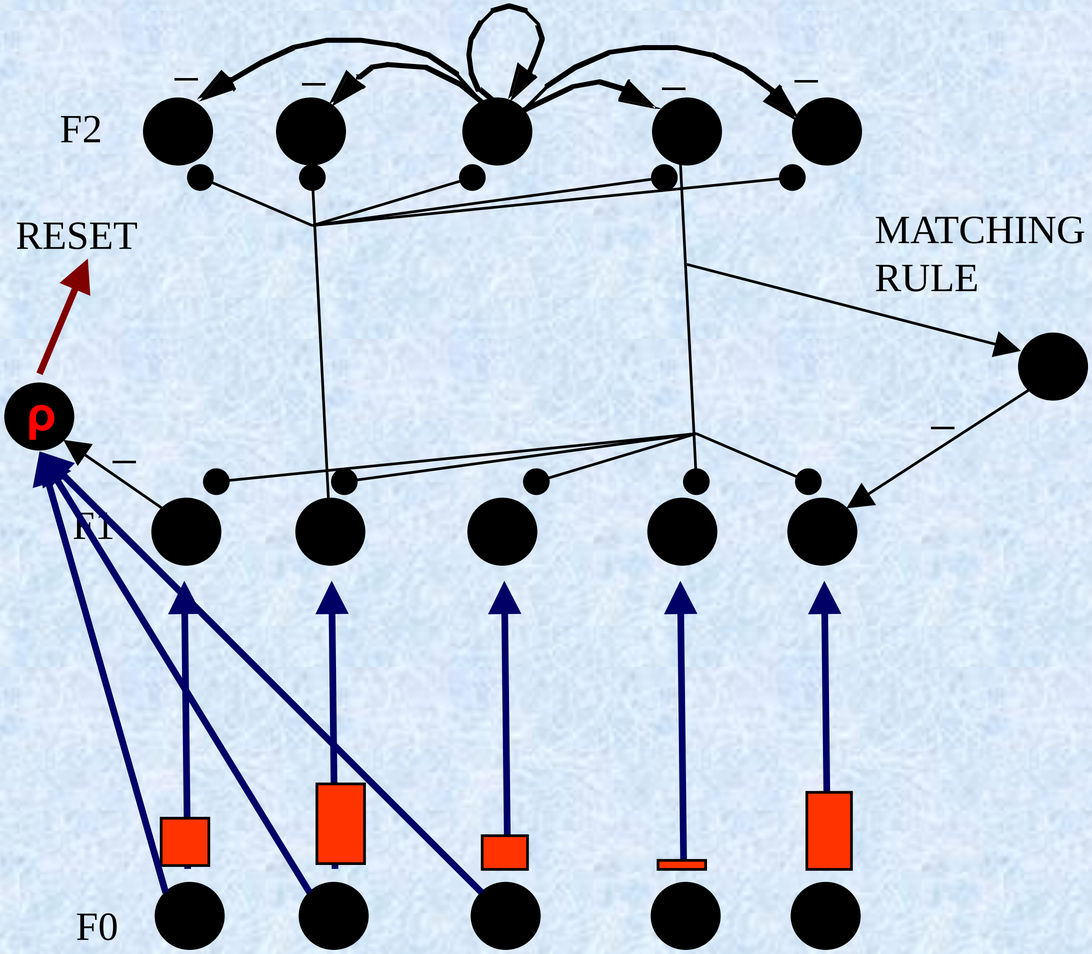
While the match is evaluated in F1 the transmitter get depleted for the winner, but not for all others



After the reset, the strongest active cell at the bottom cannot activate the top cell to the full extent, so the next strongest match wins

ART Cycle

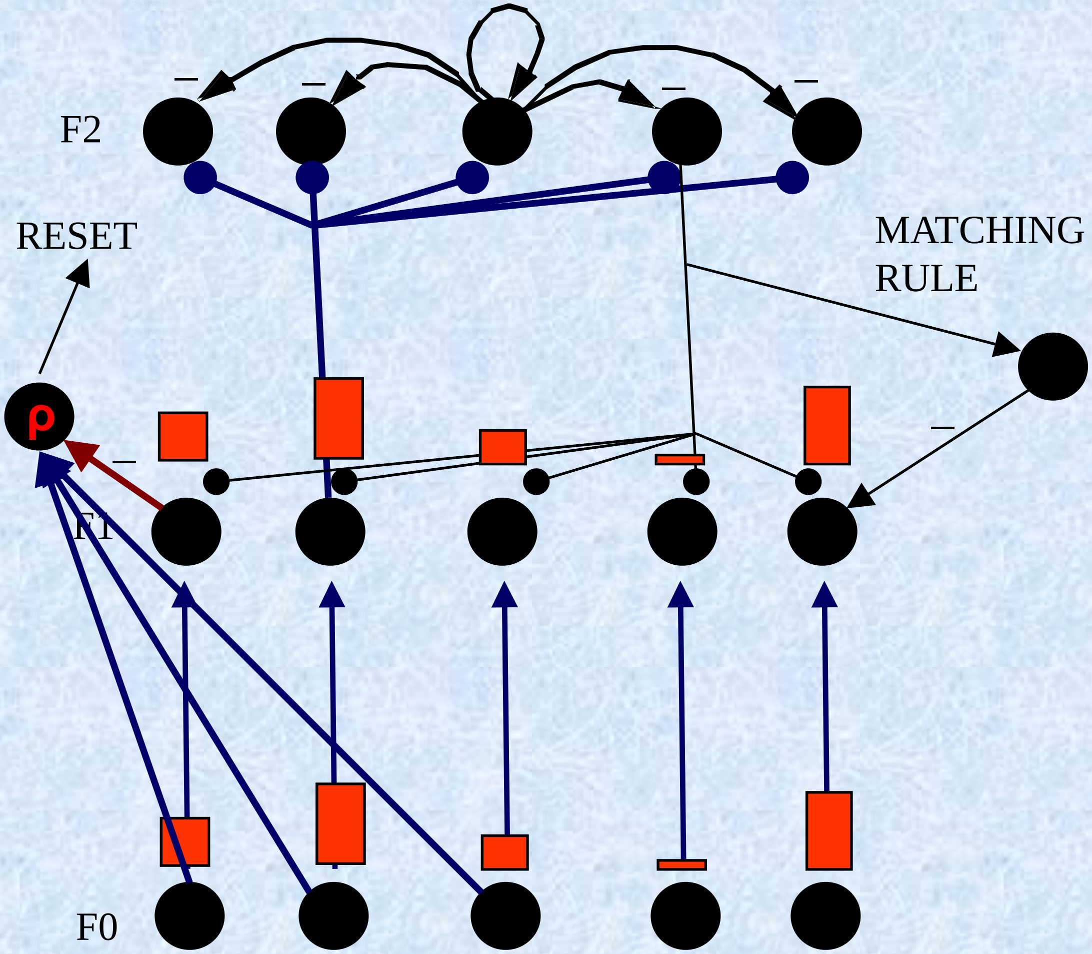
Step 1: input is provided



ART Cycle

Step 1: input is provided

Step 2: F1 is activated

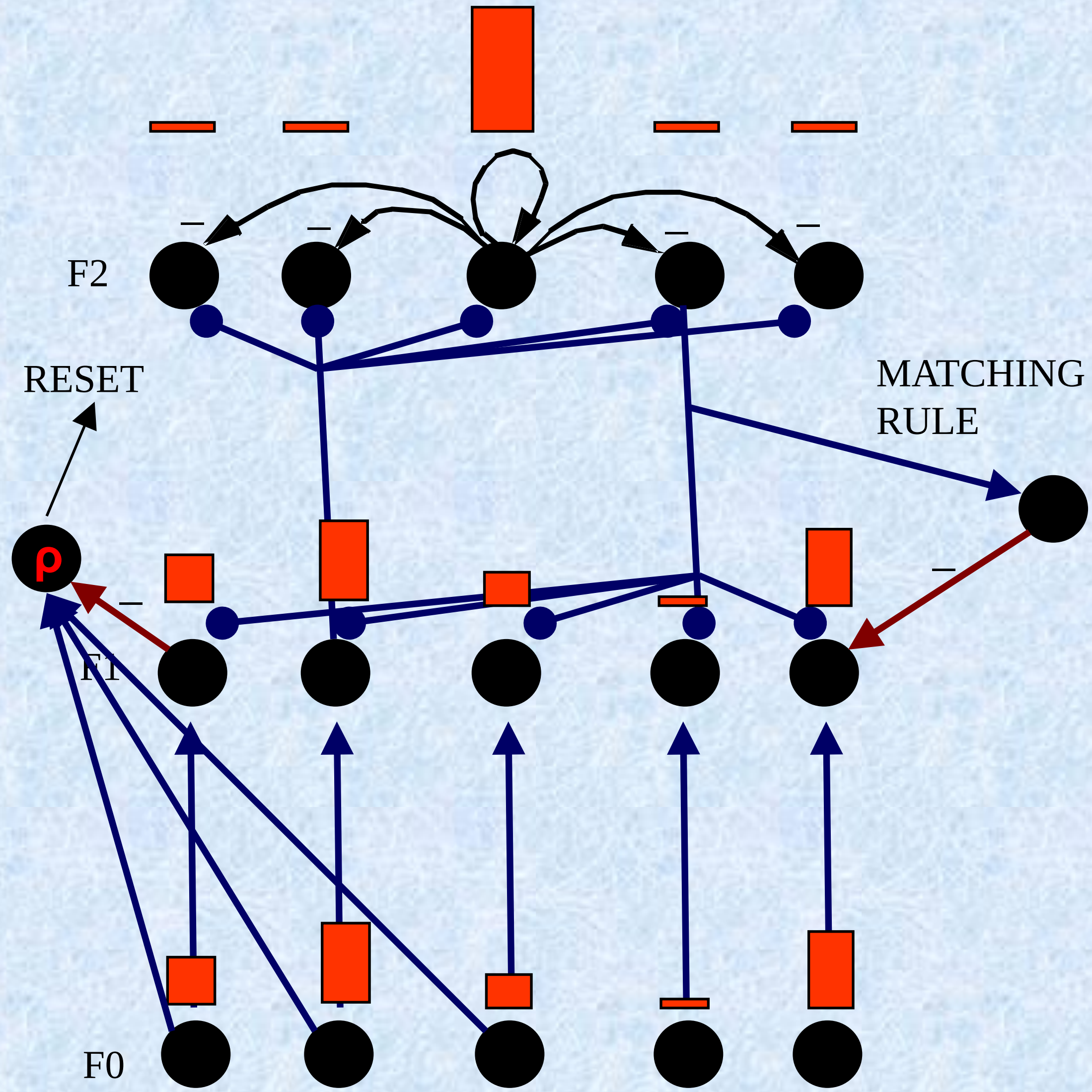


ART Cycle

Step 1: input is provided

Step 2: F1 is activated

Step 3: F2 category is selected



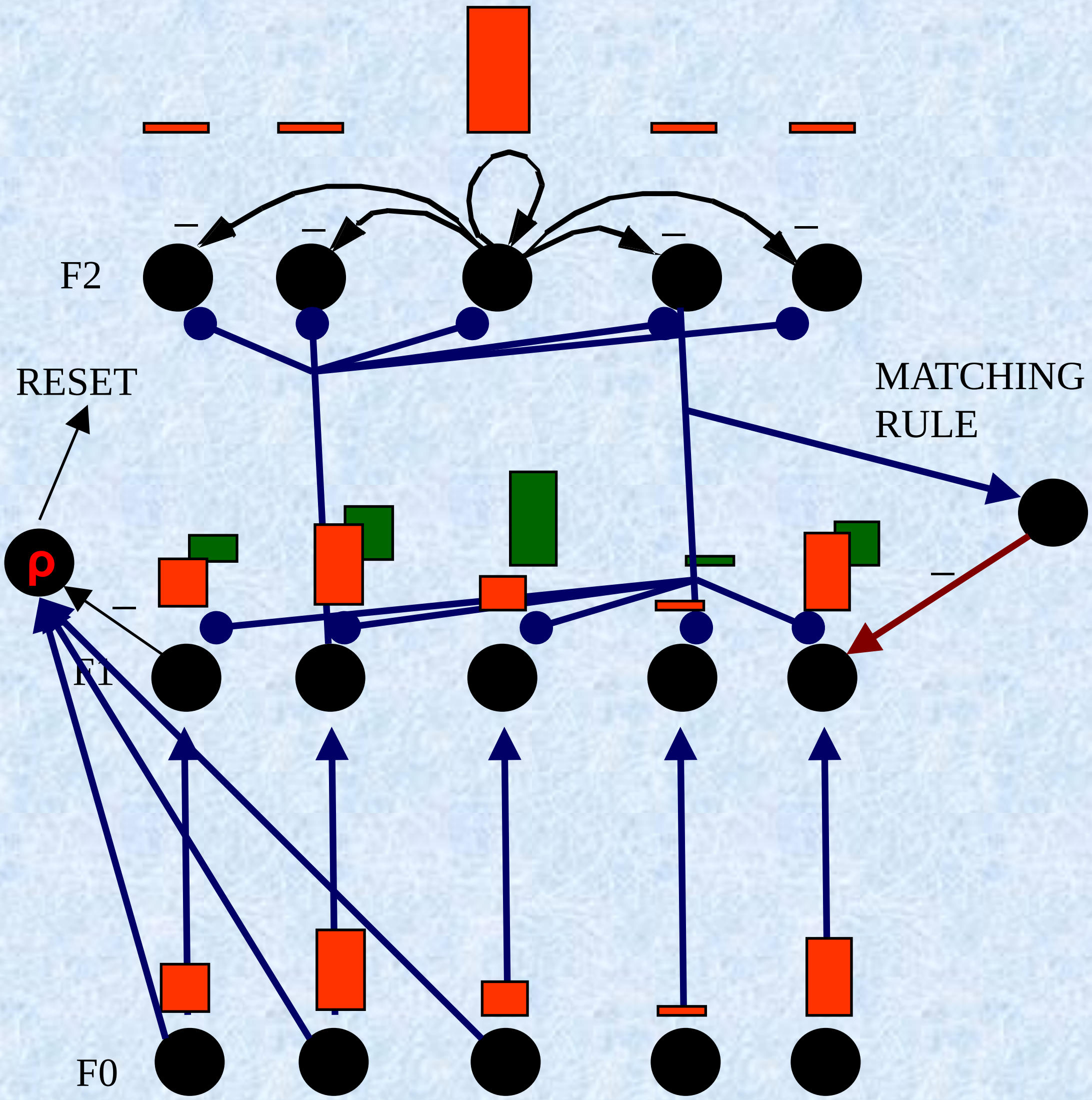
ART Cycle

Step 1: input is provided

Step 2: F1 is activated

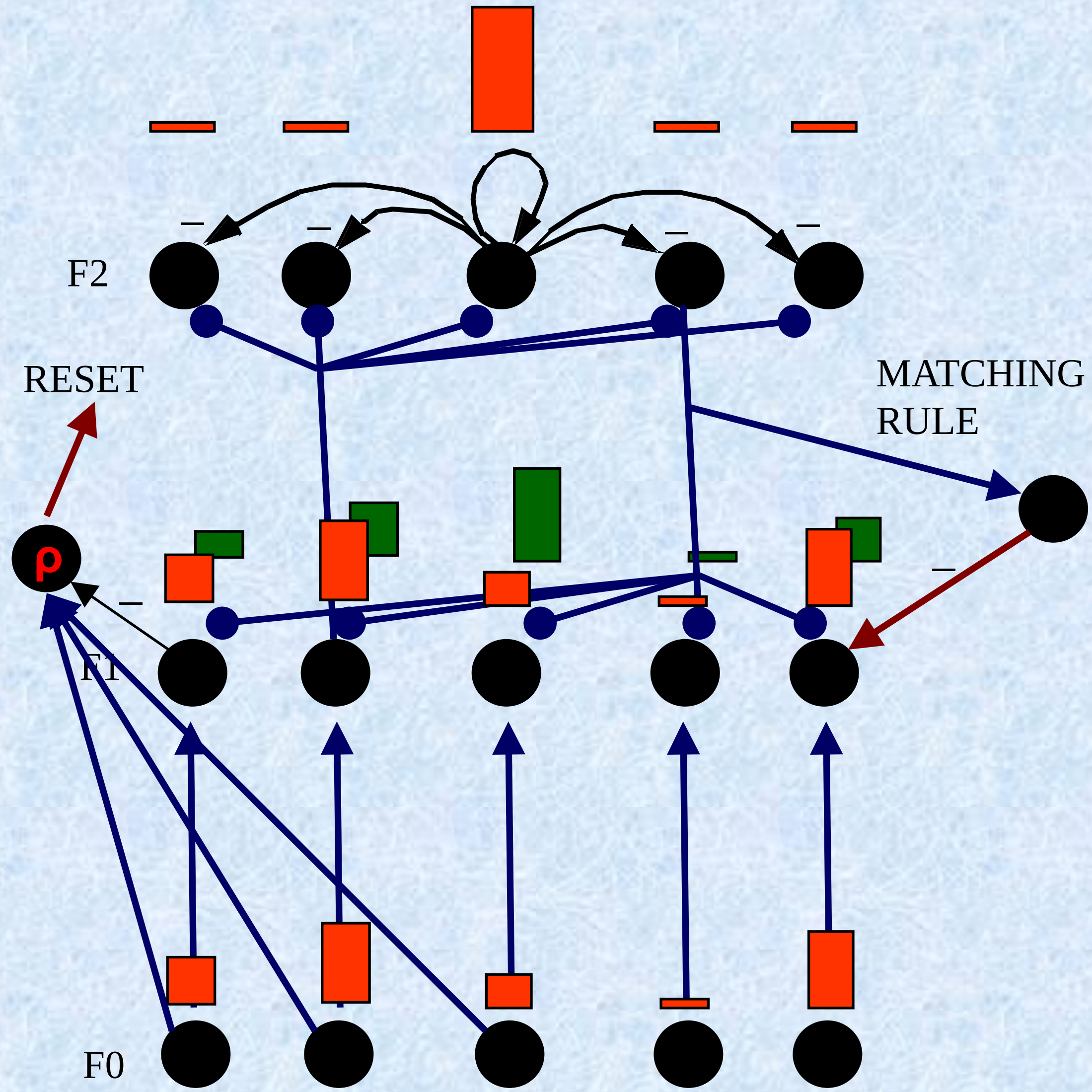
Step 3: F2 category is selected

Step 4: match is determined



ART Cycle

- Step 1: input is provided
- Step 2: F1 is activated
- Step 3: F2 category is selected
- Step 4: match is determined
- Step 5: reset



ART Cycle

Step 1: input is provided

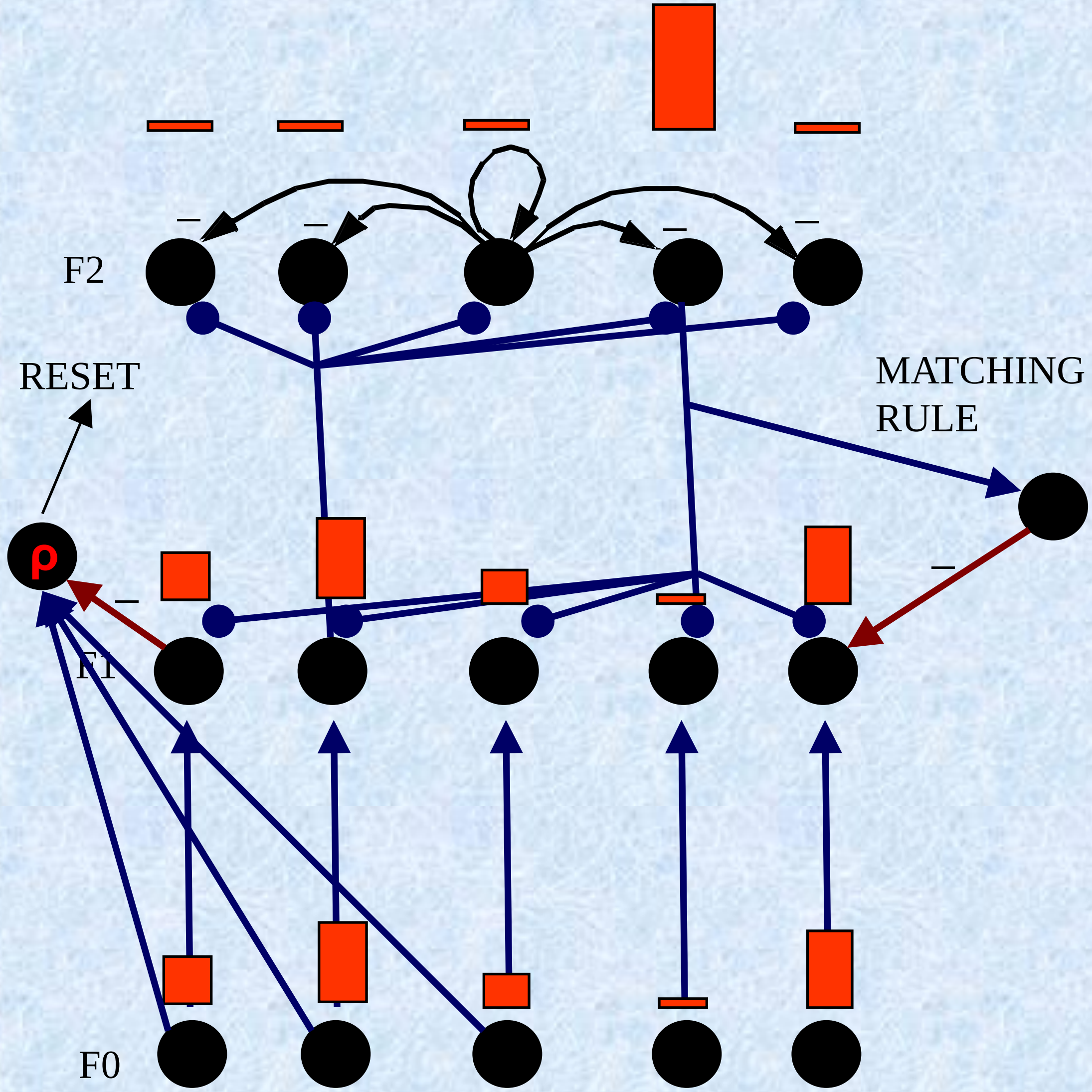
Step 2: F1 is activated

Step 3: F2 category is selected

Step 4: match is determined

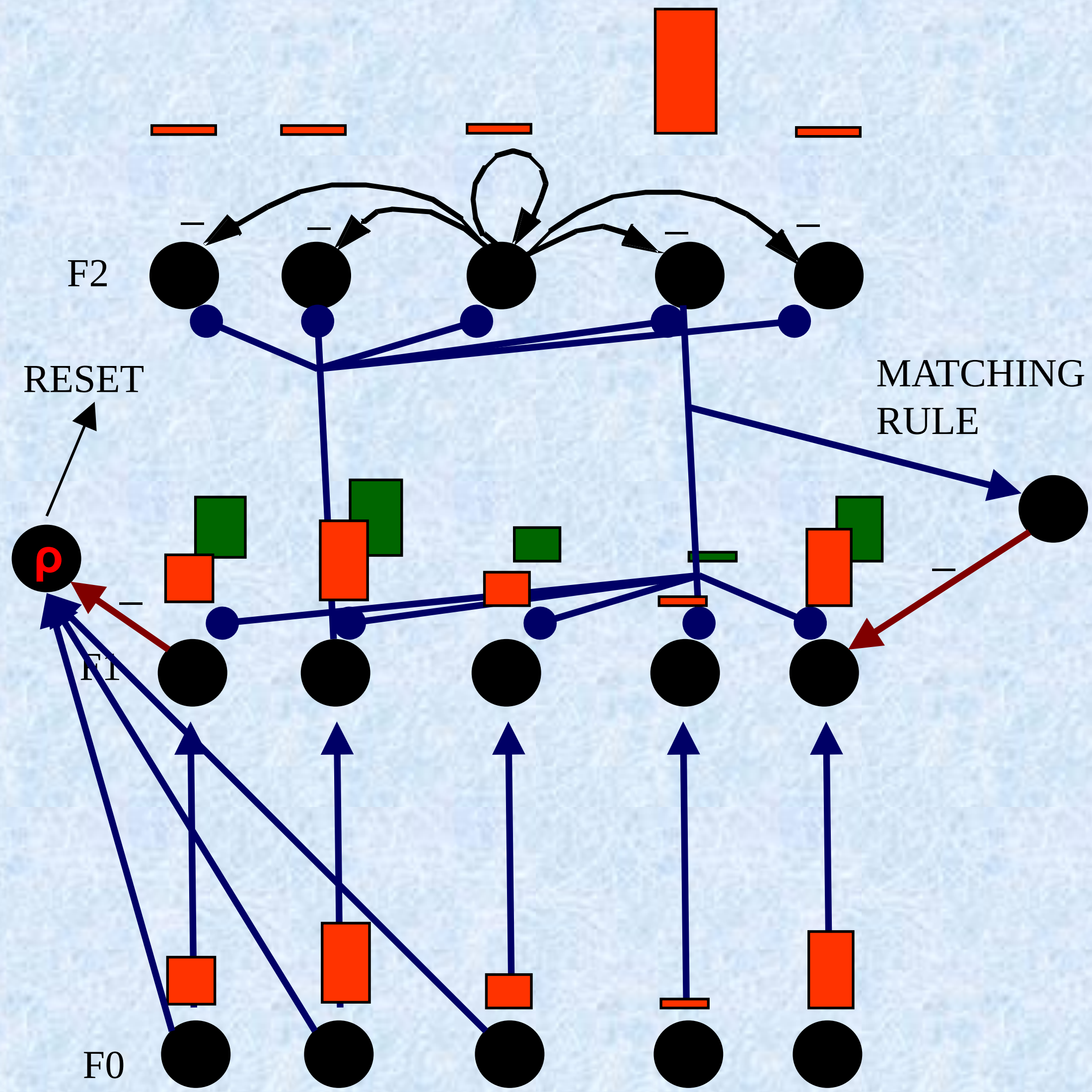
Step 5: reset

Step 3a: new F2 category is selected



ART Cycle

- Step 1: input is provided
- Step 2: F1 is activated
- Step 3: F2 category is selected
- Step 4: match is determined
- Step 5: reset
- Step 3a: new F2 category is selected
- Step 4a: match is determined



We Got Resonance, Now What?

- System enters stable state that will persist as long as the input stays on
- Weights are adapted following instar/outstar rules
- The core idea is that a resonance between F1 and F2 differentiates between insignificant transients and significant states
- Note that the paper talks about signal amplification during resonance, but this is not quite true due to normalizing properties of RCFs
- The core advantage of ART is active self-regulation of the learning process

Timing Issues

Note that timing in the system is crucial: implementing it in realistic neurons like SMART did required a lot of attention to axonal delays, synaptic time constants, etc

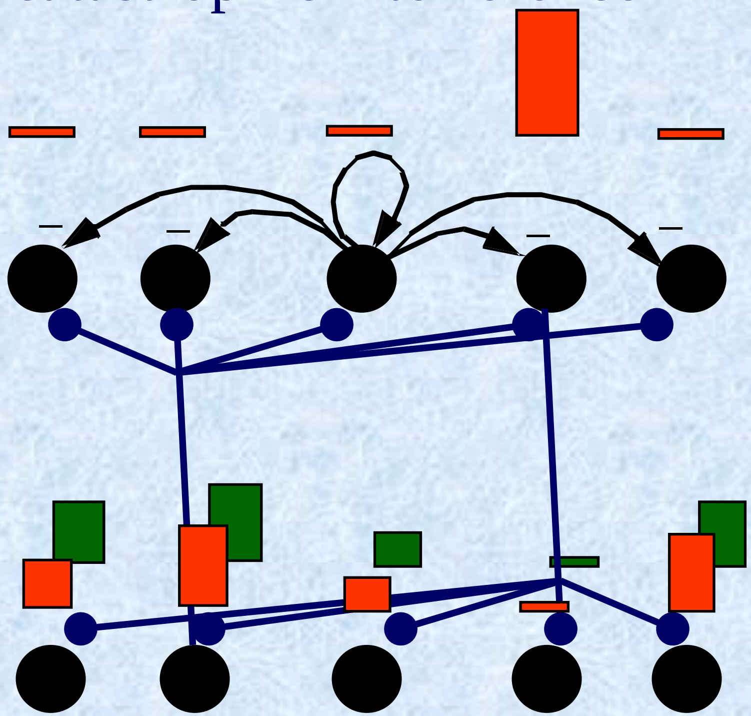
Furthermore, resonance state shall last longer then the search cycle for learning to be effective, hard to achieve in dynamic world with changing inputs

Alternative solution in SMART is to suppress learning during search cycle through cholinergic modulation

Finally, a depletion rates shall be picked carefully to last long enough to prevent repetitive wins, but not long enough to allow for next pattern correct classification

Representation Issue

WTA in F2 is not realistic, the brain has distributed representations, but having distributed representations leads to catastrophic interference



$$\dot{w}_{ij} = \eta x_i y_j - \alpha y_j w_{ij}$$

$$\dot{w}_{ji} = \eta x_i y_j - \alpha y_j w_{ji}$$

Jesse Palma currently works on SMART extension with sparse distributed representations of categories

Next Time

Jesse Palma will present the SMART model

Optional Reading:

Grossberg, S., and Versace, M. (2008). Spikes, synchrony, and attentive learning by laminar thalamocortical circuits. *Brain Research*. 1218C, 278-312

Oscillations and synchrony on the cellular level

Readings:

- Ermentrout, GB, and Kopell, N. (1998). Fine structure of neural spiking and synchronization in the presence of conduction delays. Proc. Nat. Acad. Sci. USA 95: 1259-1264.