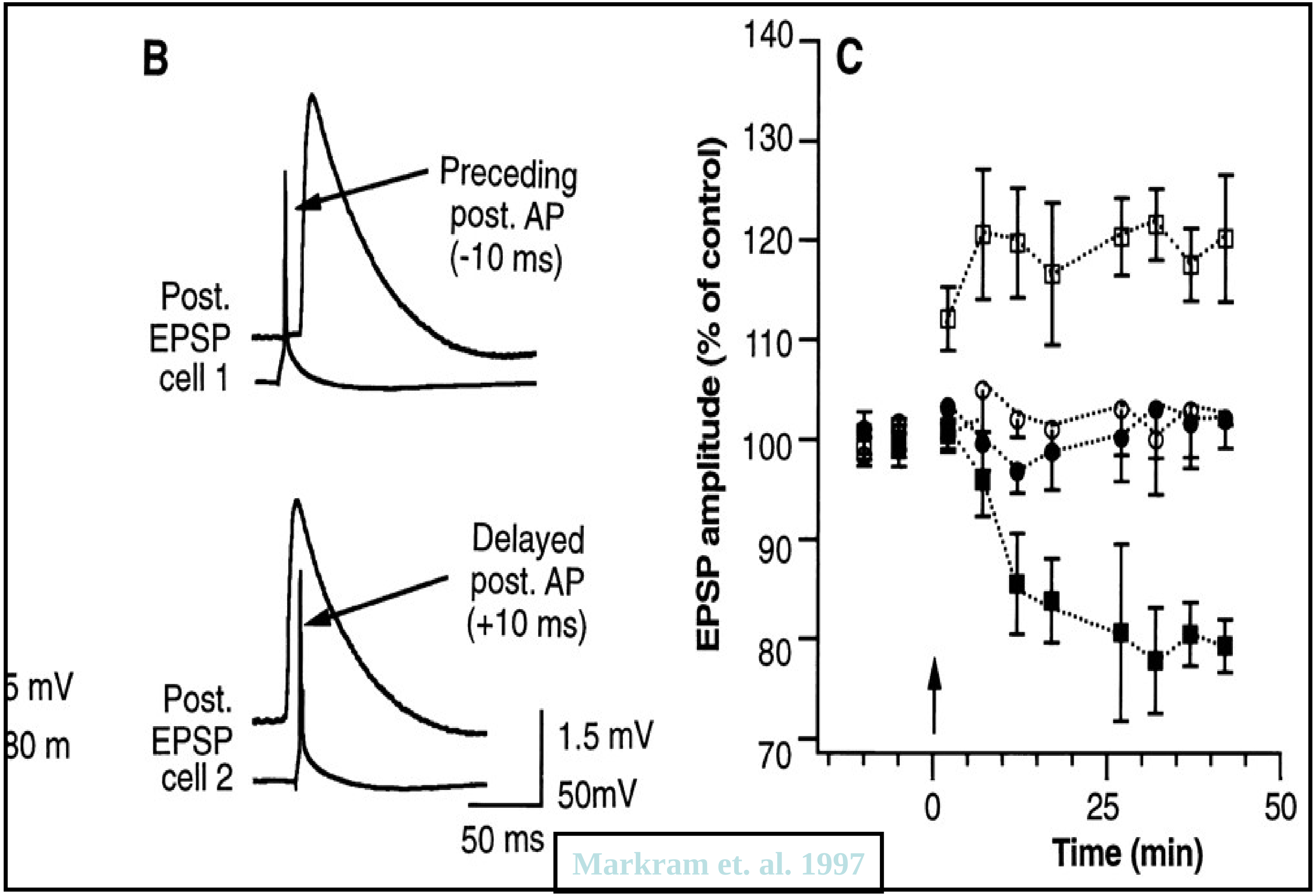
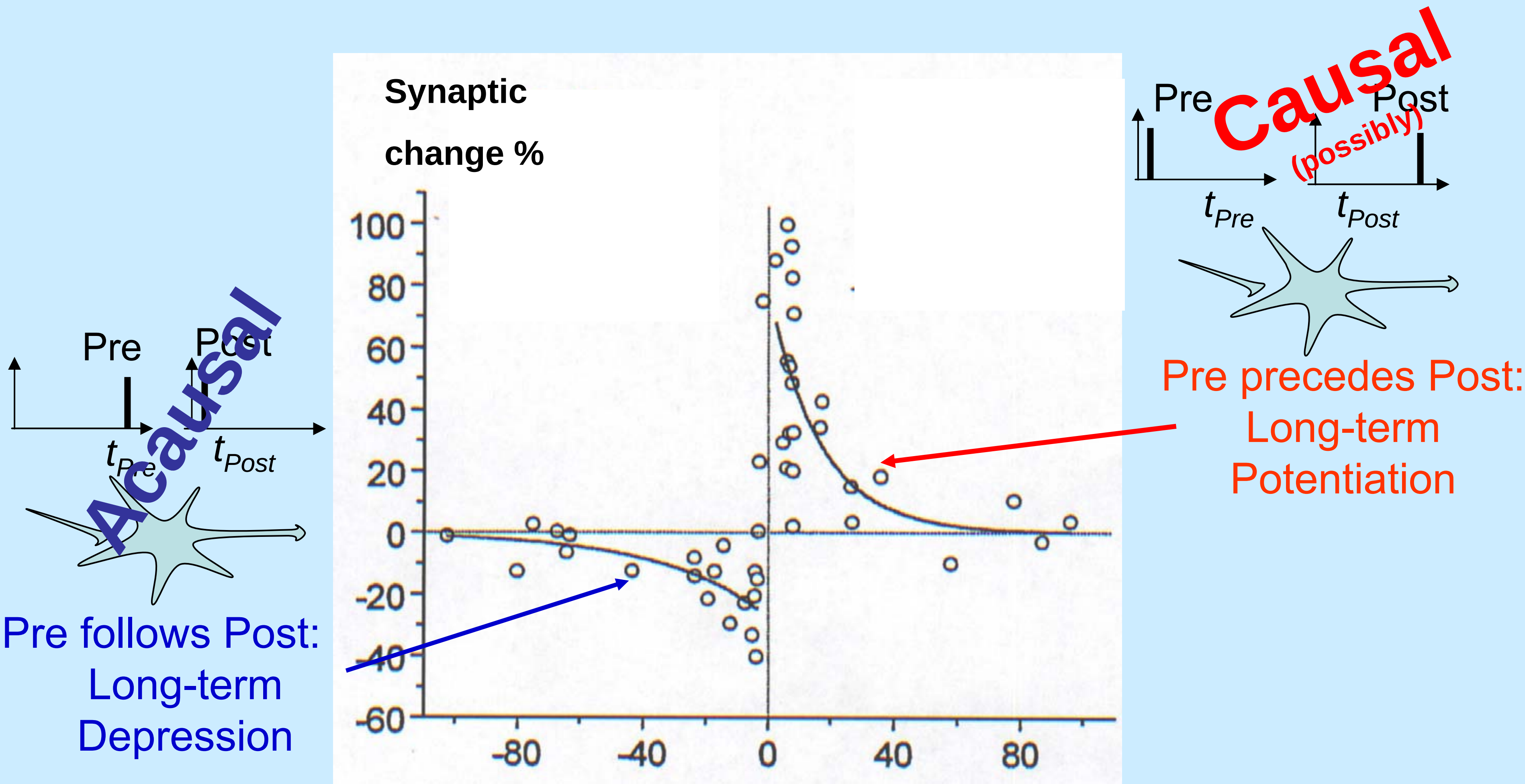
Differential Hebbian Learning – Introducing Temporal Asymmetry

Spike timing dependent plasticity - STDP



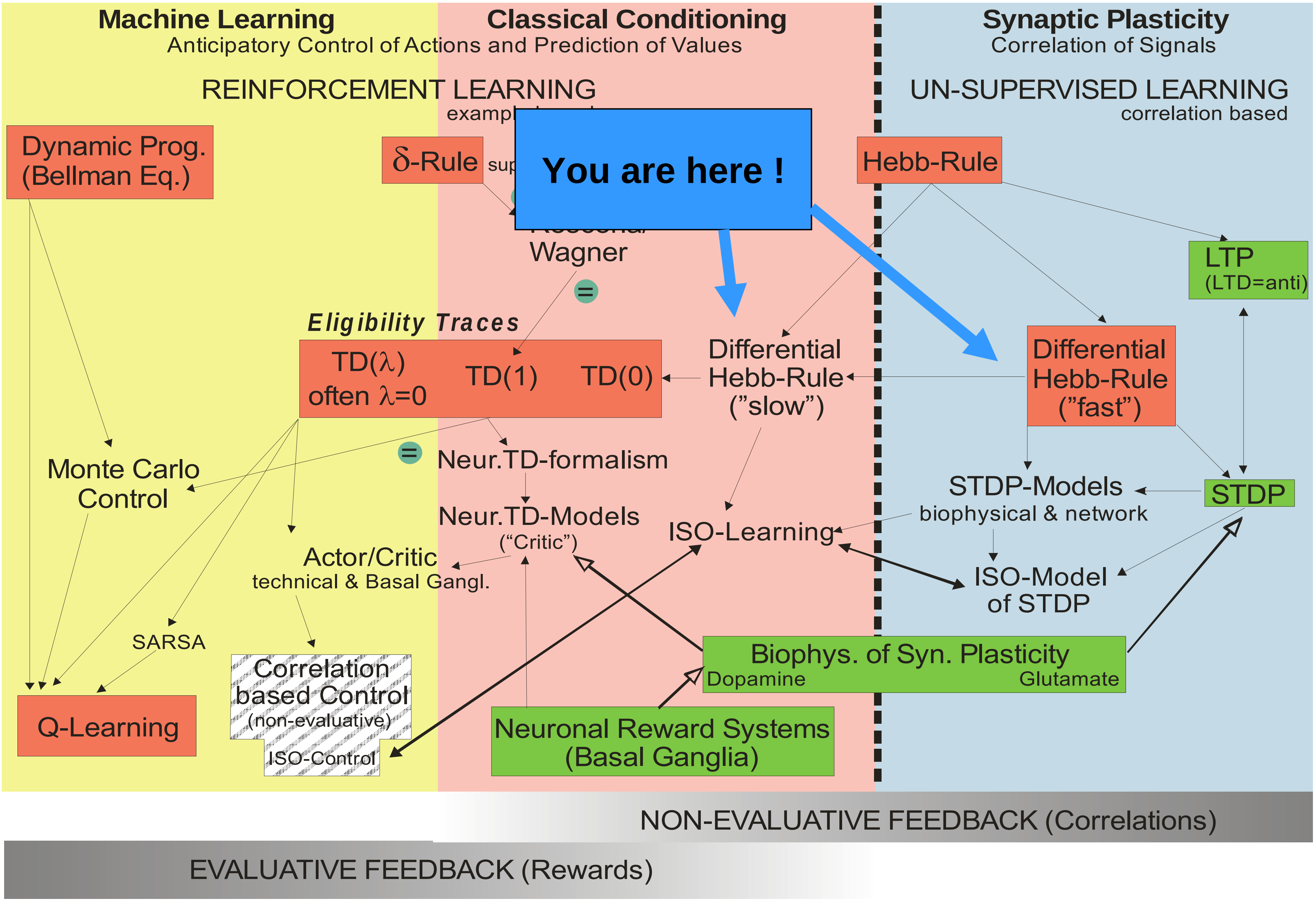
Markram et. al. 1997

Spike Timing Dependent Plasticity: Temporal Hebbian Learning



Weight-change curve
(Bi&Poo, 2001)

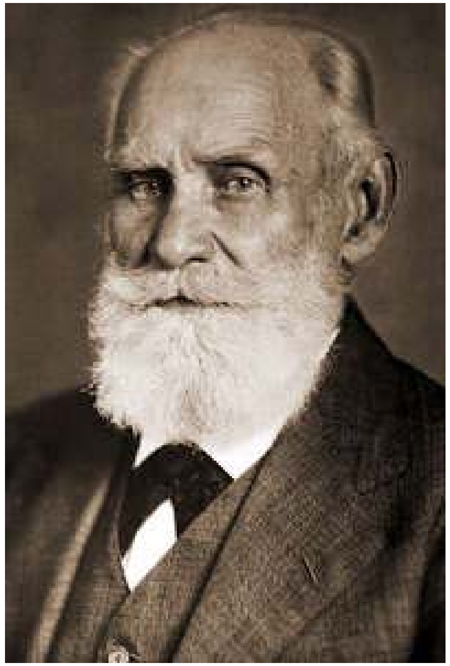
Overview over different methods



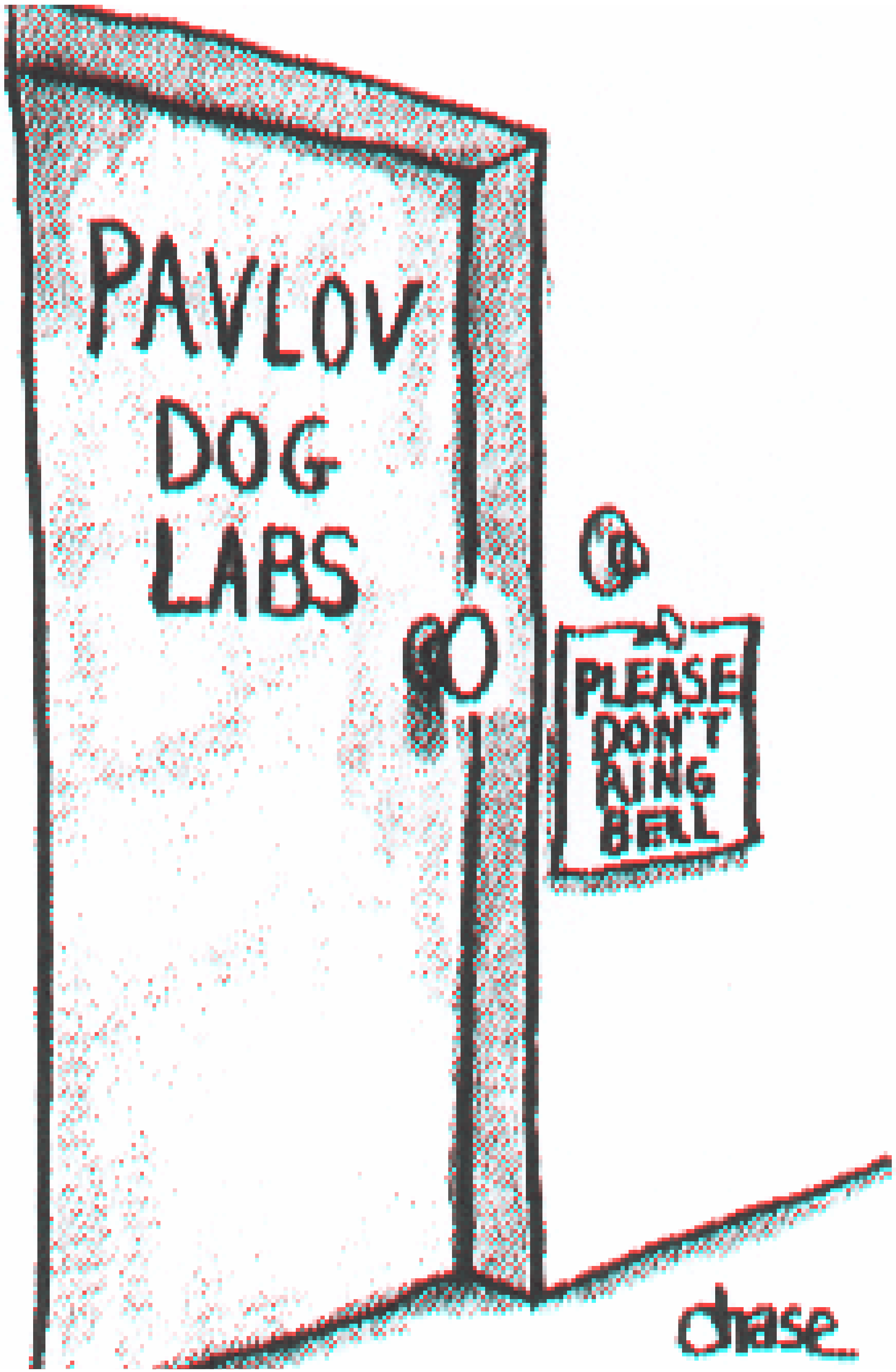
History of the Concept of Temporally Asymmetrical Learning: Classical Conditioning

CS

CR



I. Pawlow


PAVLOV
DOG
LABS
PLEASE
DON'T
RING
BELL
chase

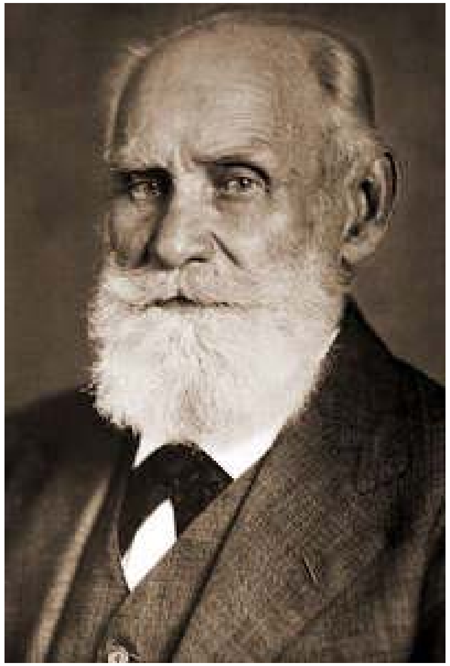
History of the Concept of Temporally Asymmetrical Learning: Classical Conditioning

Correlating two stimuli which are shifted with respect to each other in time.

Pavlov's Dog: "Bell *comes earlier* than Food"

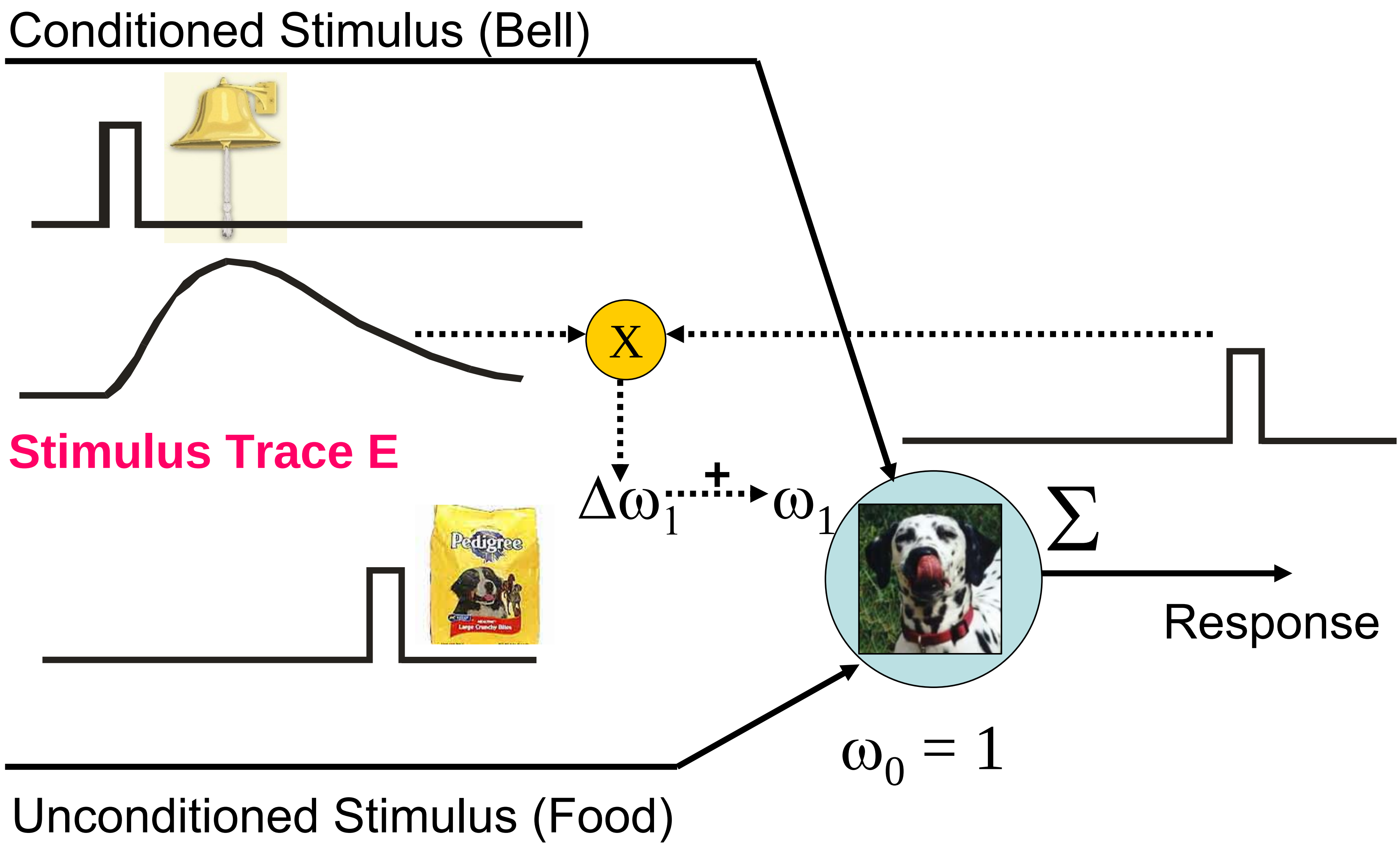
This requires to remember the stimuli in the system.

Eligibility Trace: A synapse remains "eligible" for modification for some time *after* it was active (Hull 1938, then a still abstract concept).



I. Pawlow

Classical Conditioning: Eligibility Traces



The first stimulus needs to be "remembered" in the system

History of the Concept of Temporally Asymmetrical Learning: Classical Conditioning

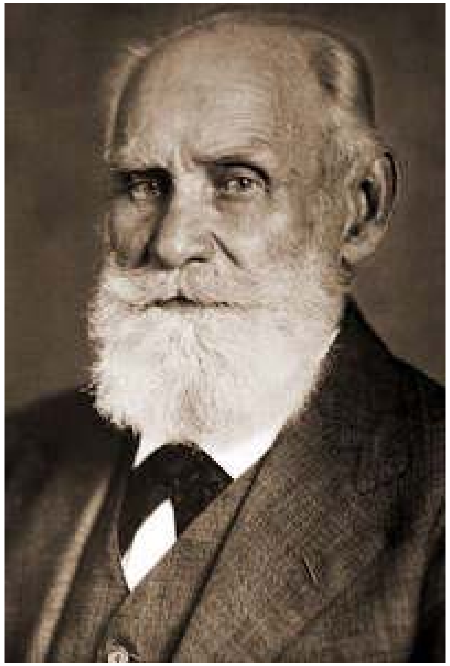
Eligibility Traces

Note: There are vastly different time-scales for (Pavlov's) behavioural experiments:

Typically up to 4 seconds

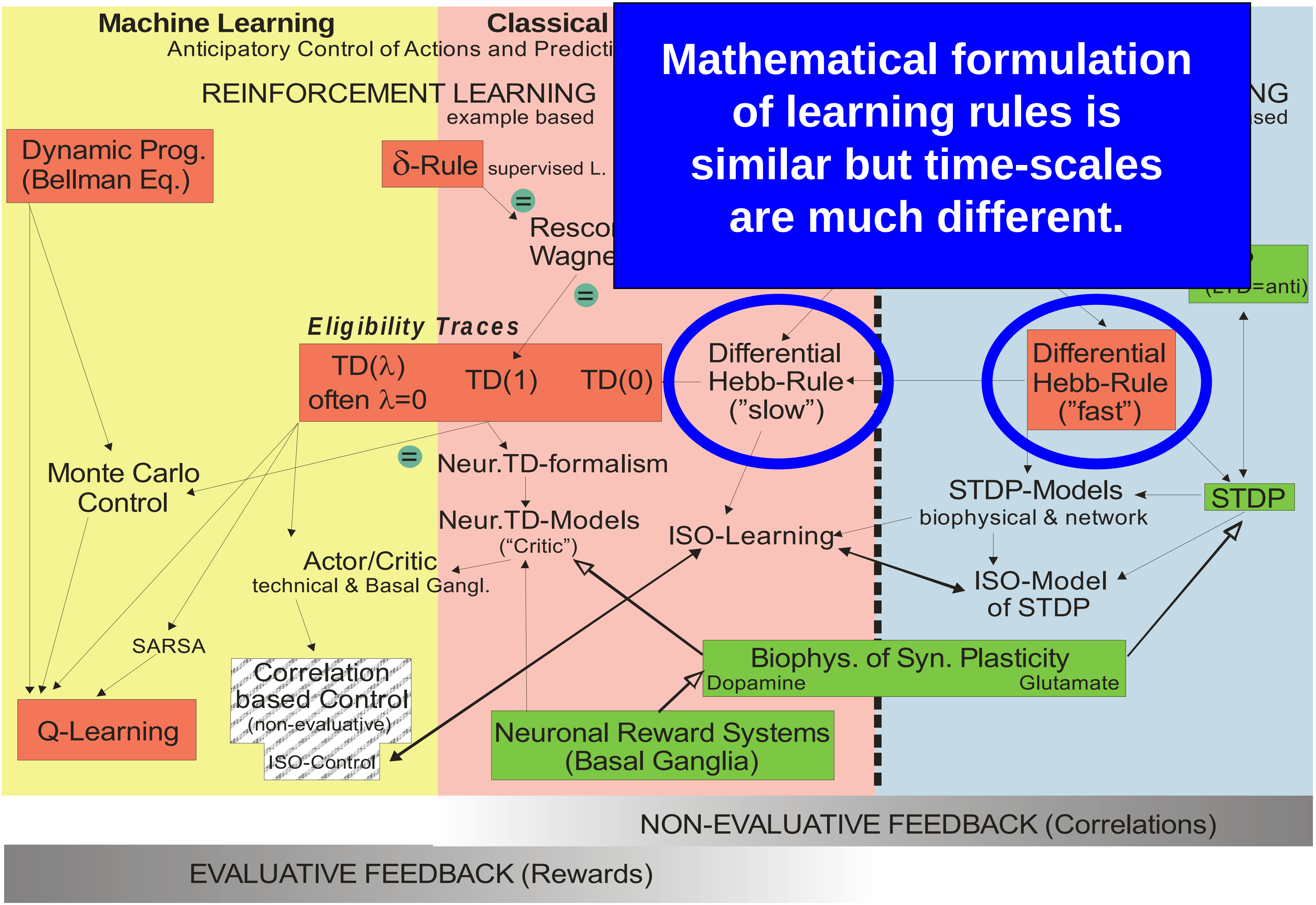
as compared to STDP at neurons:

Typically 40-60 milliseconds (max.)



I. Pawlow

Overview over different methods

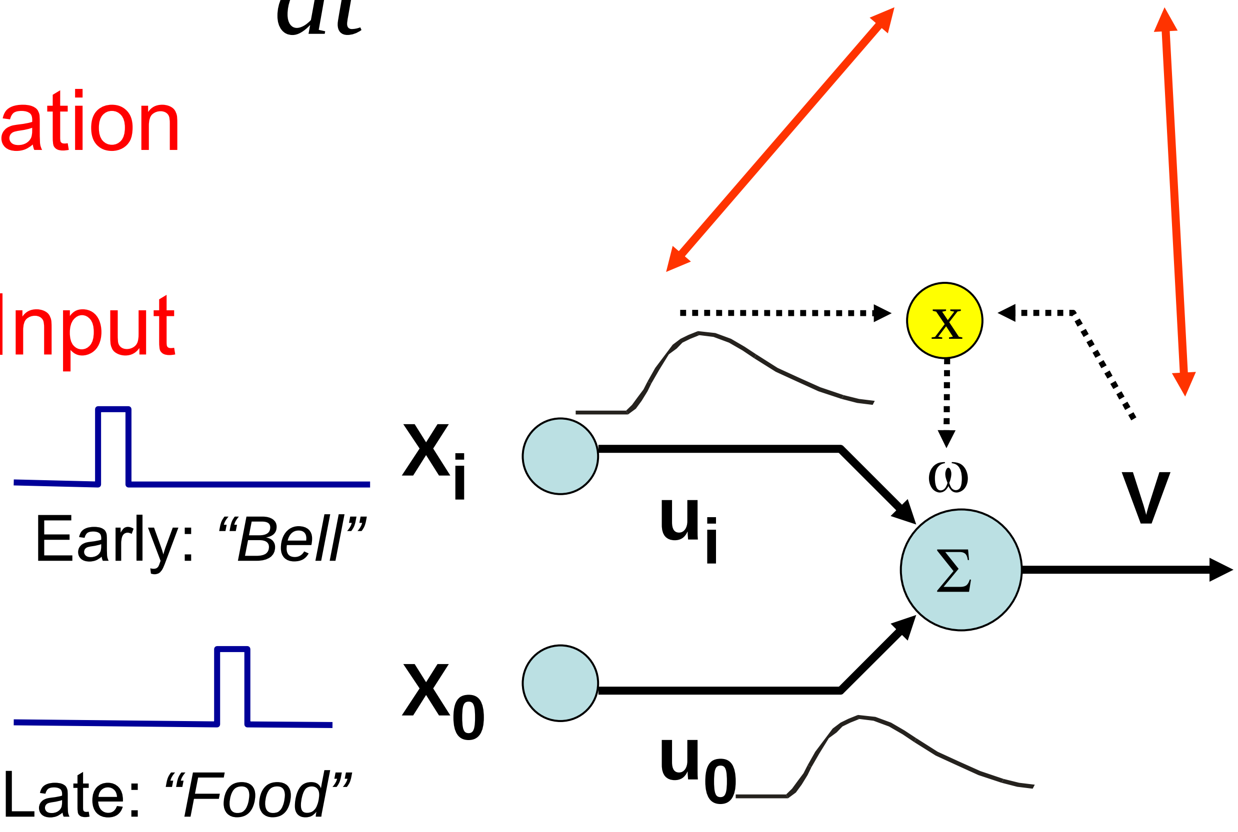


Mathematical formulation of learning rules is similar but time-scales are much different.

Differential Hebb Learning Rule

$$\frac{d}{dt}\omega_i(t) = \mu\, u_i(t)\ V'(t)$$

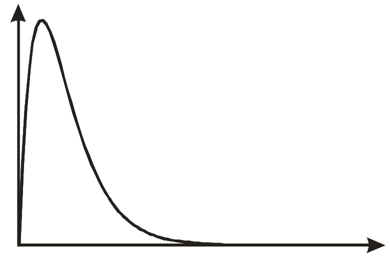
Simpler Notation
x = Input
u = Traced Input



Defining the Trace

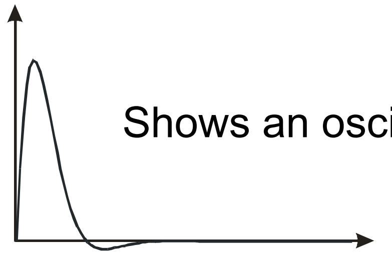
In general there are many ways to do this, but usually one chooses a trace that looks biologically realistic and allows for some analytical calculations, too.

$$h(t) = \begin{cases} h_k(t) & t \geq 0 \\ 0 & t < 0 \end{cases}$$



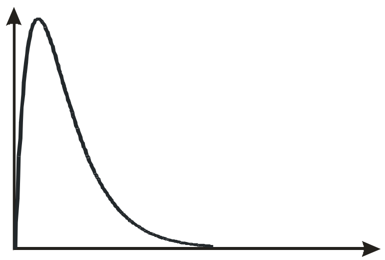
EPSP-like functions:

α-function: $h_k(t) = te^{-at}$



Shows an oscillation.

Dampened Sine wave: $h_k(t) = \frac{1}{b}\sin(bt)\ e^{-at}$



Double exp.: $h_k(t) = \frac{1}{\delta}(e^{-at} - e^{-bt})$

This one is most easy to handle analytically and, thus, often used.

Defining the Traced Input u

Convolution:

$$u(x) = \int_{-\infty}^{\infty} f(u)g(x-u)du = f(x) \circ g(x) = g(x) \circ f(x)$$

Note the differences !

Correlation:

$$w(x) = \int_{-\infty}^{\infty} f(u)g(u-x)du = g(x) * f(x) \neq f(x) * g(x)$$

Convolution used to define the traced input,

Correlation used to calculate weight growth (see below).

Defining the Traced Input u

Convolution:

$$u(x) = \int_{-\infty}^{\infty} f(u)g(x-u)du = f(x) \circ g(x) = g(x) \circ f(x)$$

Specifically (we are dealing with causal functions!):

$$u(t) = \int_{0}^{\infty} x(\tau)h(t-\tau)d\tau$$

If x is a spike train (using the δ-function):

$$x(t) = \sum_{j=0}^{M} \delta(t_j)$$

Then:

$$u(t) = \sum_{j=0}^{M} h(t-t_j)$$

For example:

$$x(t) = \delta(0) \qquad u(t) = h(t)$$

$$x(t) = \delta(T) \qquad u(t) = h(t-T)$$

Differential Hebb Rules – The Basic Rule

General:

Two inputs only. Thus we get for the output:

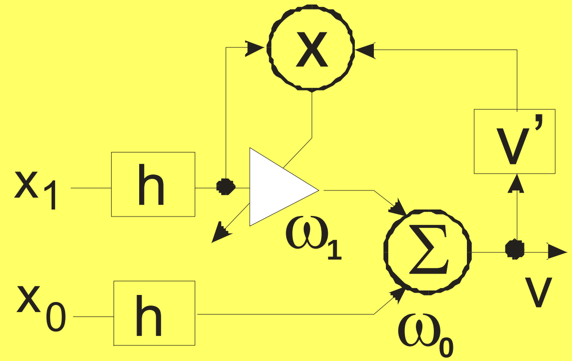
$$v = w_0 u_0 + w_1 u_1$$

One weight unchanging:

w_0=1=const.

Same h for all inputs.

The basic rule: ISO-Learning

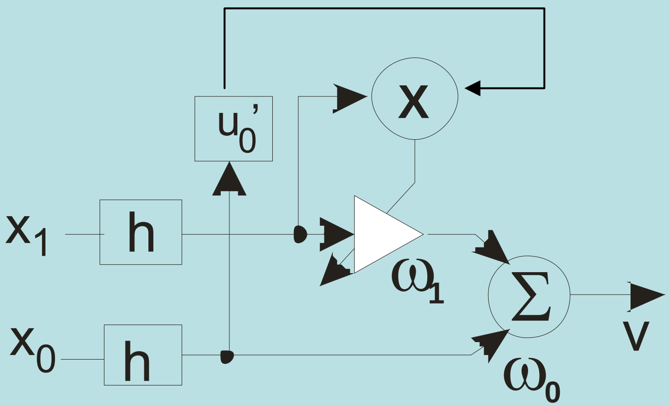


ISO rule $$\frac{dw_1}{dt} = \mu \; u_1 \; v'$$

Isotropic Sequence Order Lng.
(as we can also allow w_0 to change!)

Differential Hebb Rules – More rules (but why?)

ICO - Learning

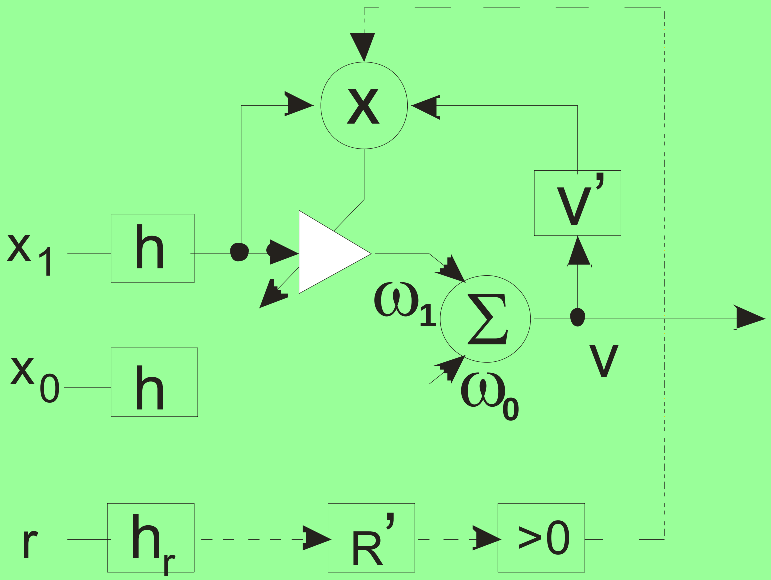


ICO $\frac{dw_1}{dt} = \mu \; u_1 \; u_0'$

Input correlation Learning

(as we take the derivative of the unchanging input u_0)

ISO3 - Learning

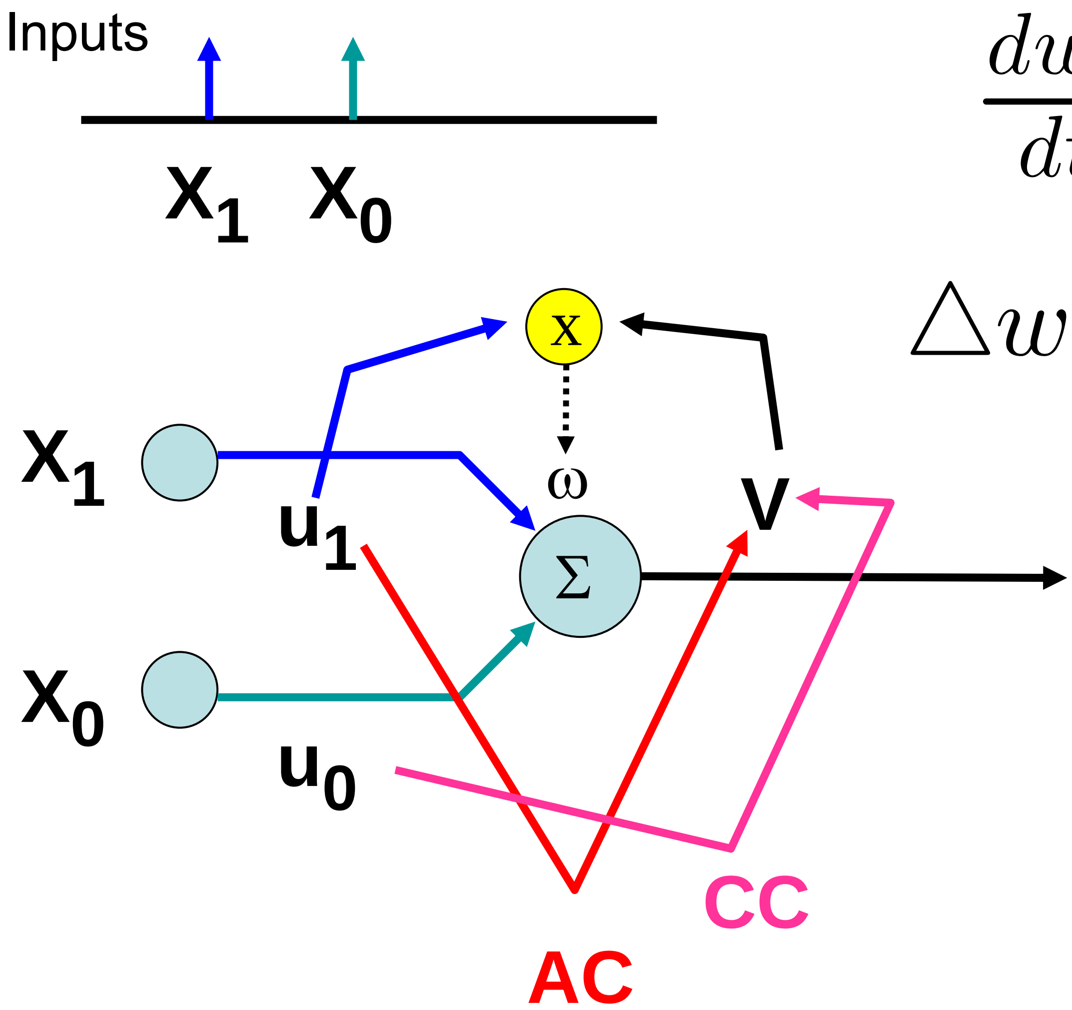


ISO3 $\frac{dw_1}{dt} = \mu \; u_1 \; v' \; R' \Big\rfloor$

Three factor learning

The $\rfloor$ denotes that we are only using positive contributions

Stability Analysis



$$\frac{dw_1}{dt} = \mu u_1 v'$$

$$\Delta w_1(t) = \int_0^\infty \frac{dw_1(t)}{dt} dt$$

$$\Delta w_1(t) = \Delta w_1^{AC}(t) \ + \ \Delta w_1^{CC}(t)$$

Stability Analysis

$$\triangle w_1(t) = \triangle w_1^{AC}(t) \quad + \quad \triangle w_1^{CC}(t)$$

Desired contribution

Undesired contribution

Some problems with these differential equations:

$$\triangle w_1(t) = \int_0^{\infty} \frac{dw_1(t)}{dt} dt$$

1) As we are integrating to ∞ strictly we need to assume that there is no second pulse pair coming in “ever”.

2) Furthermore we should assume that $w_1' \rightarrow 0$ (hence μ small) or we get second order influences, too.

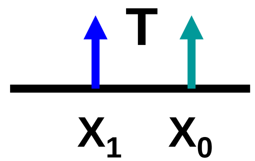
Stability Analysis (ISO)

Under these assumptions we can calculate Δw^{AC} and Δw^{CC} to find out whether the rules are stable or not.

In general we assume two inputs:

$x_1(t) = \delta(t)$ and $x_0(t) = \delta(t - T)$

Inputs



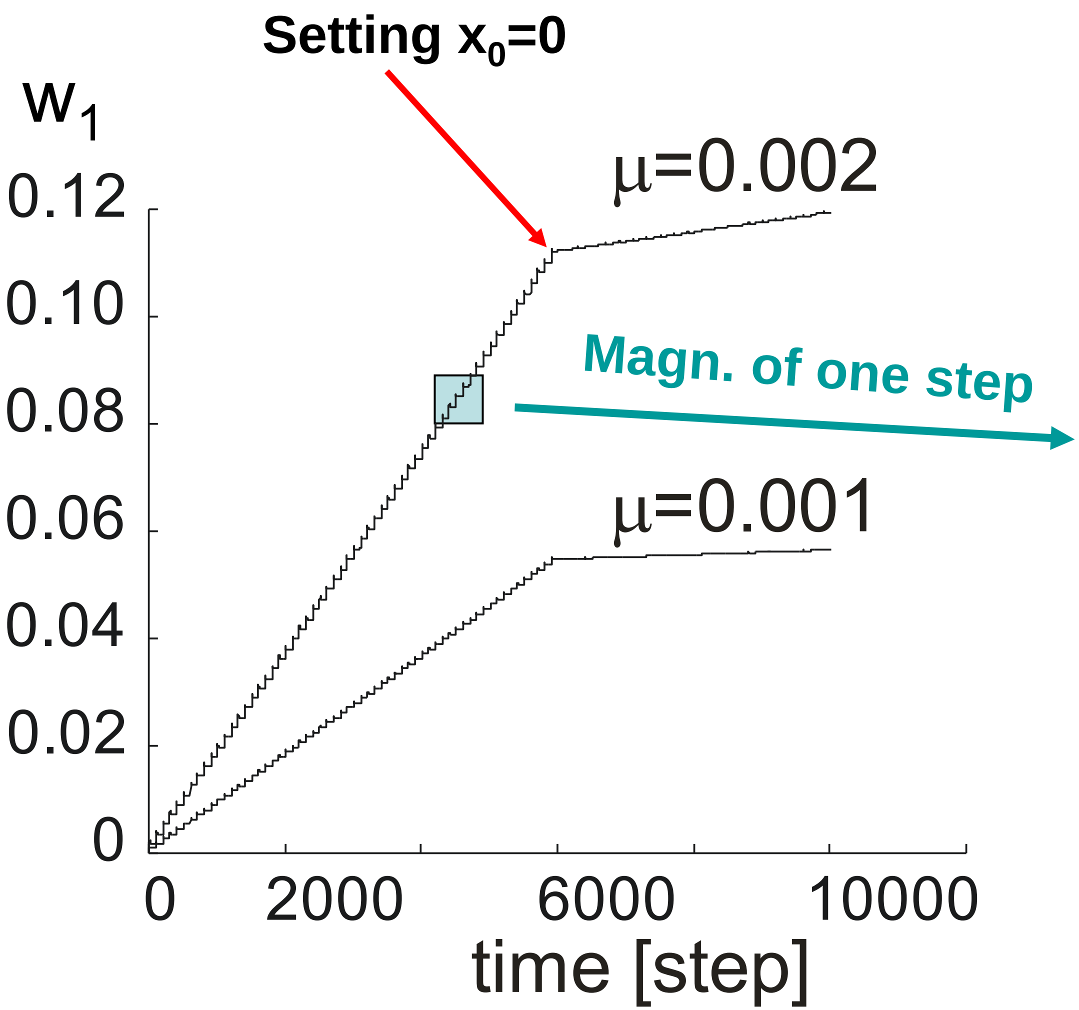
and get for **ISO:** $$\frac{dw_1}{dt} = \mu u_1 v'$$

$$\triangle w_1^{CC} = w_0 \int h(t)h'(t-T)dt \quad = w_0 \frac{1}{2\sigma}\frac{a-b}{a+b} h(t)$$

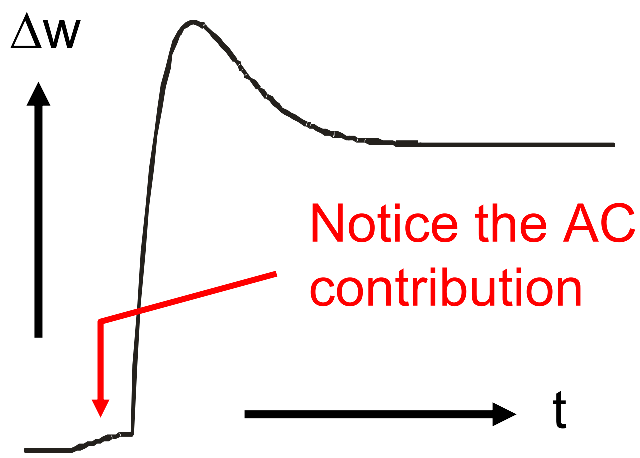
$$\triangle w_1^{AC} = w_1\left(e^{\int h(t)h'(t-T)dt} - 1\right) = w_1\left(e^{\frac{1}{2}h^2(\infty)} - 1\right) = 0$$

ISO is (only) asymptotically stable for $t \rightarrow \infty$

Stability Analysis for pulse pair inputs (ISO)



Single pairing relaxation behavior

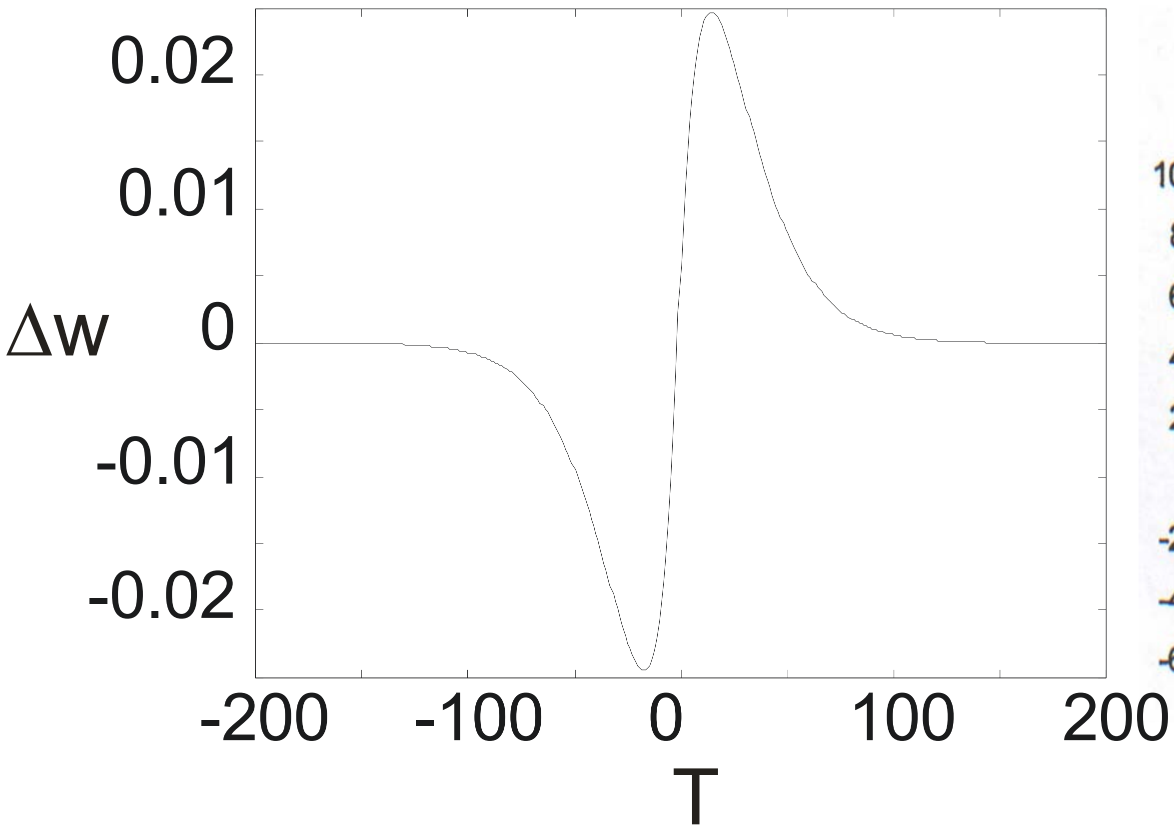


The remaining upward drift is only due to the AC term influence (Instable !)

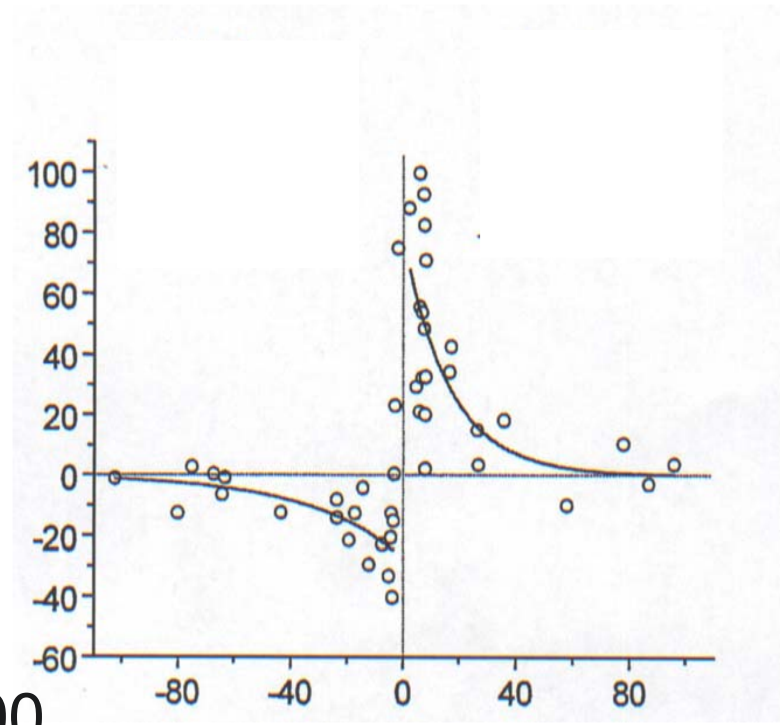
This shows that early arrival of a new pulse pair might easily fall into a not fully relaxed system. (Instable !)

Learning Window (weight change curve)

ISO: Weight change curve



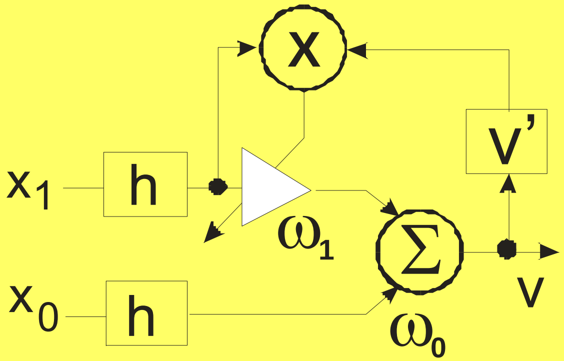
Compare to STDP



The weight change curve plots Δw in dependence on the pulse pairing distance T in steps, where we define $T>0$ if the x_1 signal arrives before x_0 and $T<0$ else.

Stability Analysis: Compare ISO with ICO

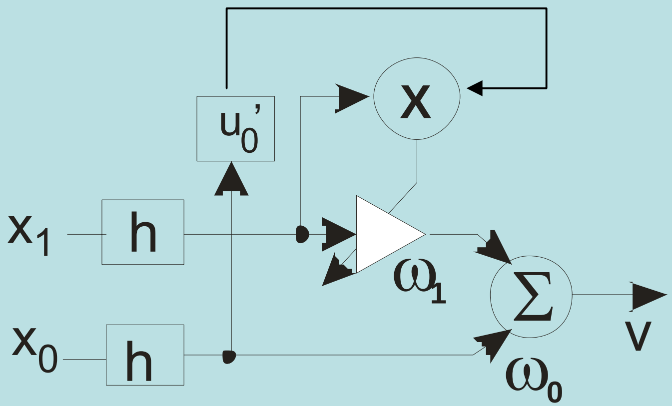
The basic rule: ISO-Learning



ISO rule $\frac{dw_1}{dt} = \mu \;\; u_1 \;\; v'$

Notice the difference

ICO - Learning



ICO $\frac{dw_1}{dt} = \mu \;\; u_1 \;\; u_0'$

Input correlation Learning

(as we take the derivative of the unchanging input u_0)

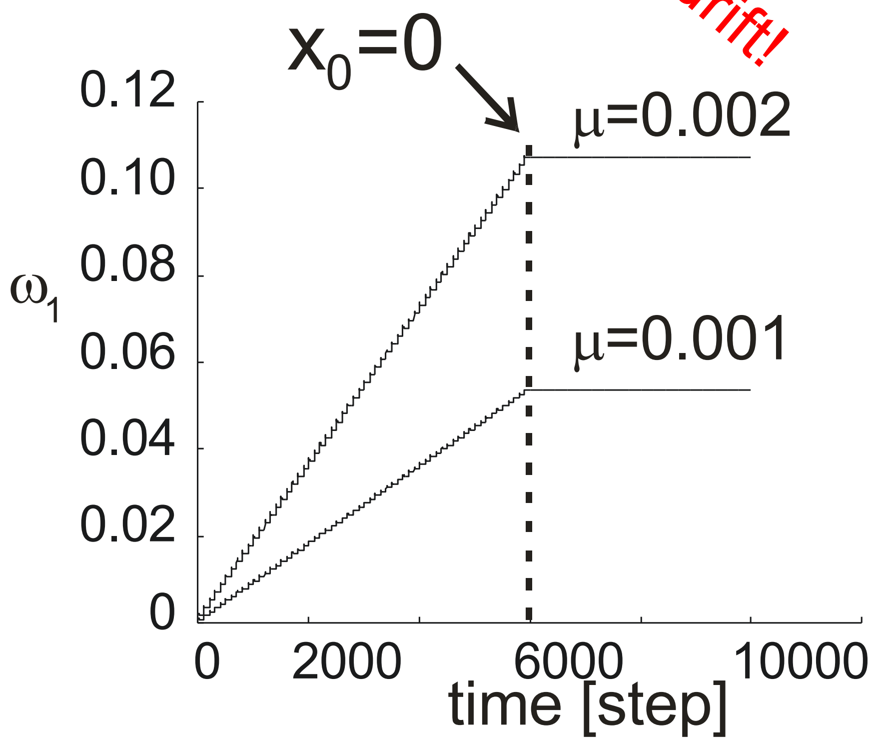
Stability Analysis: ICO

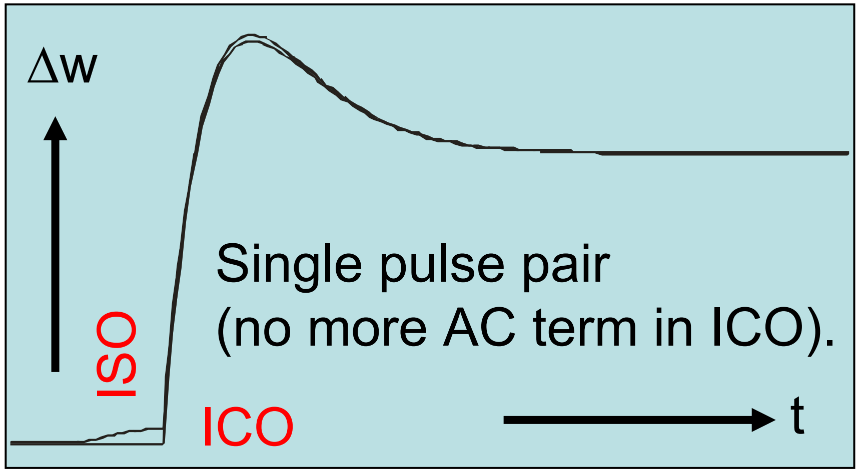
Same as for ISO!

$$\Delta w_1^{CC} = w_0 \int h(t)h'(t-T)dt \quad = w_0 \frac{1}{2\sigma}\frac{a-b}{a+b} h(t)$$

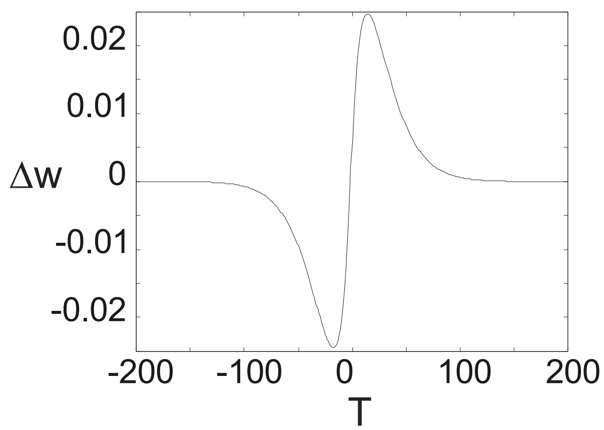
$$\Delta w_1^{AC} \equiv 0$$

Fully stable !
No more drift!



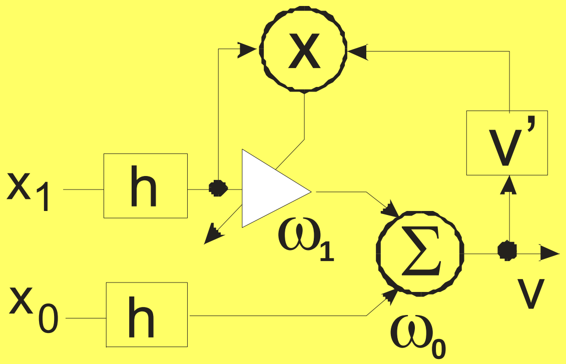


ICO: Weight change curve
(same as for ISO)



Stability Analysis: More comparisons

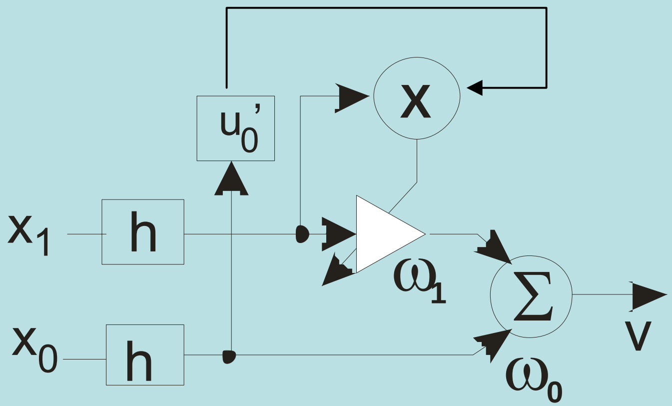
The basic rule: ISO-Learning



ISO rule $$\frac{dw_1}{dt} = \mu \; u_1 \; v'$$

Conjoint learning-control-signal (same for all inputs !)

ICO - Learning



ICO $$\frac{dw_1}{dt} = \mu \; u_1 \; u_0'$$

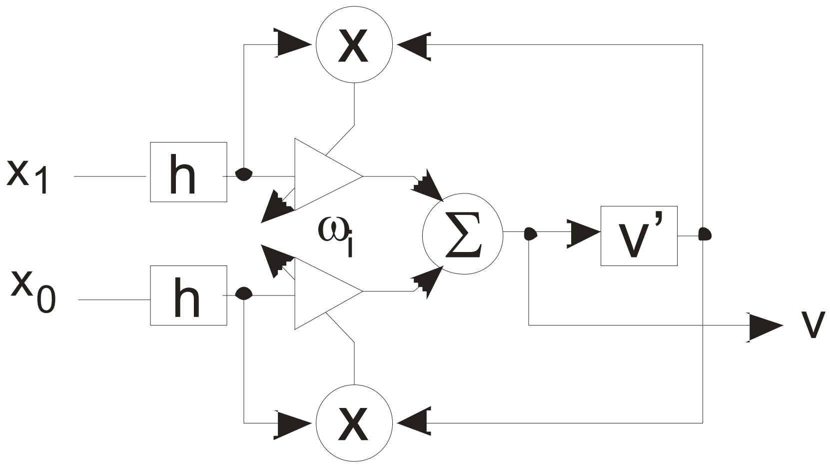
Single input as designated learning-control-signal.

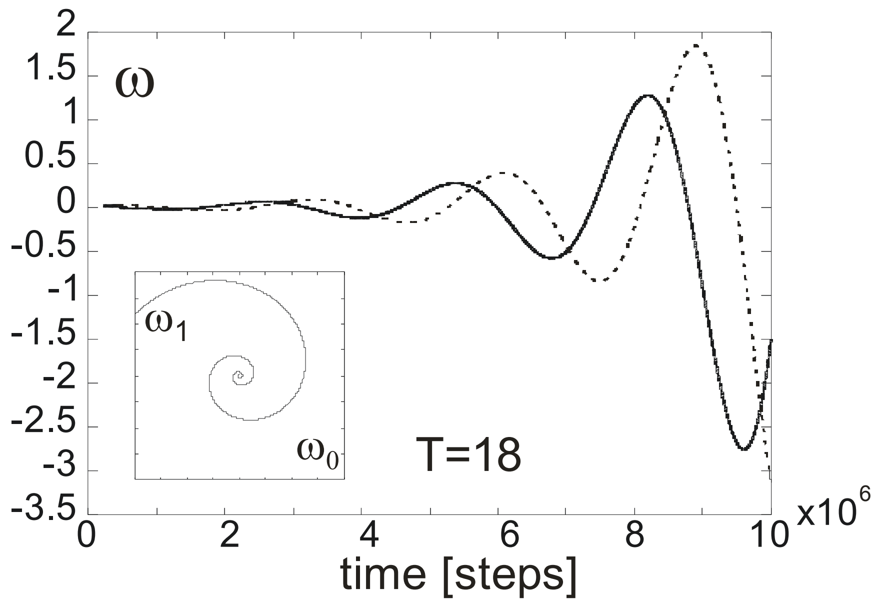
Makes ICO a heterosynaptic rule of questionable biological realism.

Stability Analysis: More comparisons

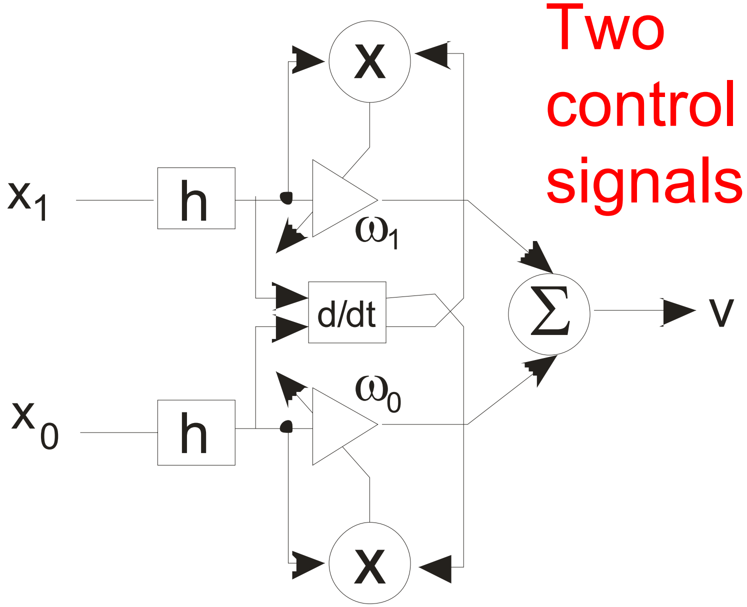
This difference is especially visible when wanting to symmetrize the rules (both weights can change!).

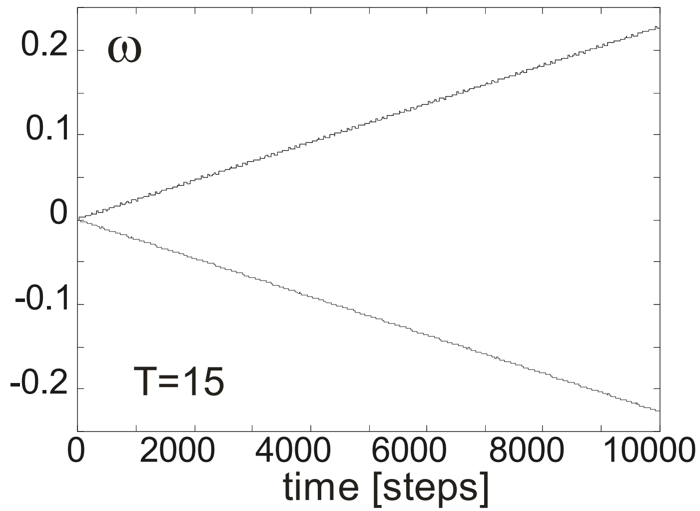
ISO-Sym One control signal !



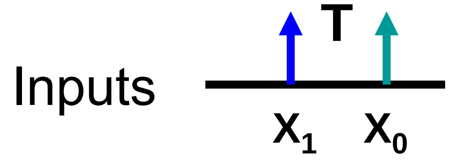


ICO-Sym Two control signals !



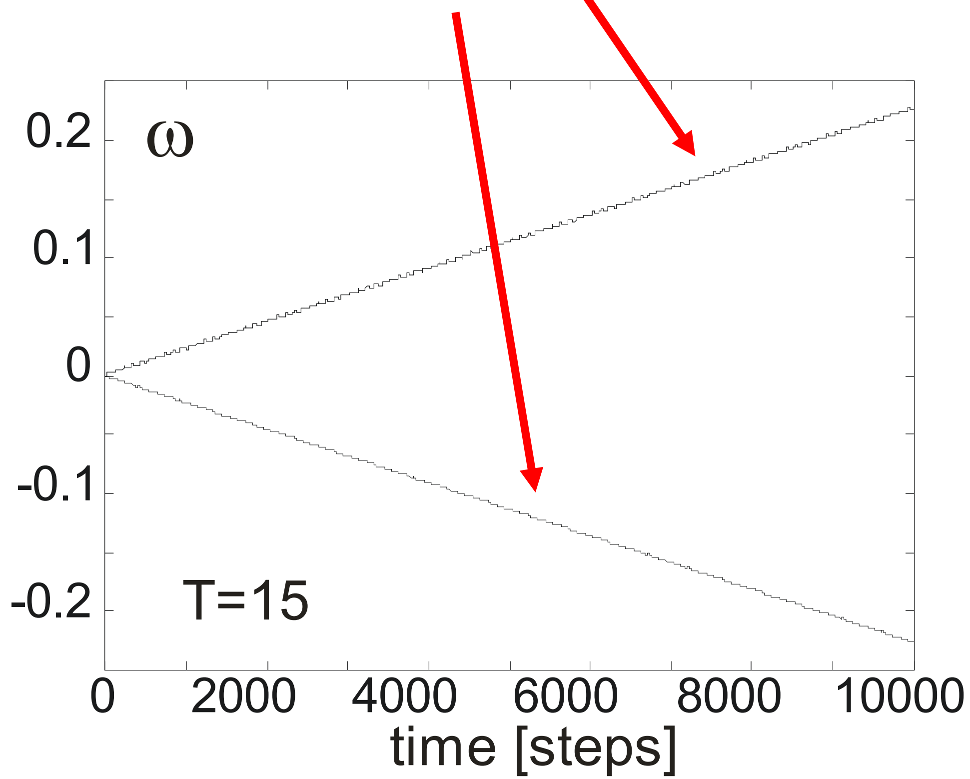


The Effects of Symmetry



Synapse w_1 grows because x_1 is before x_0.

Synapse w_0 shrinks because x_0 is after x_1.

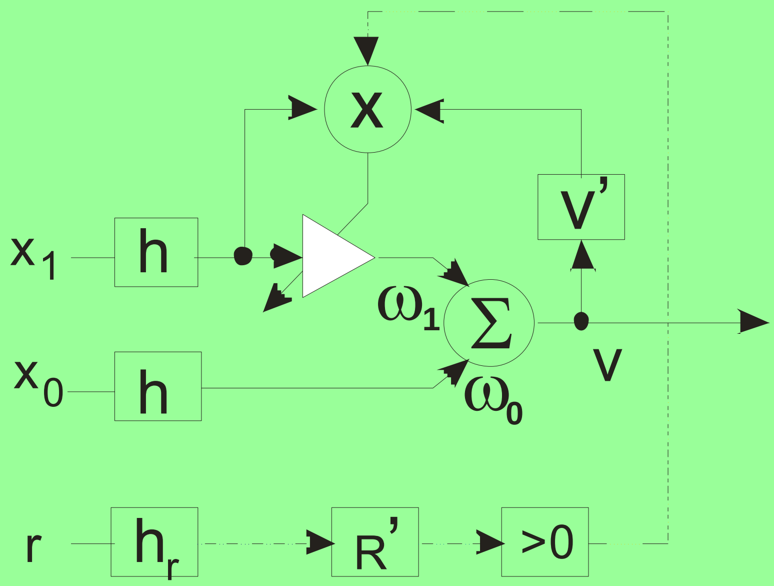


ICO-sym is truly symmetrical, but needs two control signals.

ISO-sym behaves in a difficult and unstable oscillatory way.

ISO3: uses – like ISO – a single learning-control-signal

ISO3 - Learning



ISO3 $\frac{dw_1}{dt} = \mu \ u_1 \ v' \ R'\Big|$

Idea: The system should learn ONLY at that *moment in time* when there was a "relevant" event r !

We use a shorter trace for r, as it should remain rather restricted in time.

Same filter function h but parameters a_r and b_r.

We also define T_r as the interval between x_1 and r. Many times T_r=T, hence r occurs together with x_0.

Stability Analysis: ISO3

$$\triangle w_1^{CC} = w_0 \int h(t)h'(t-T)h'_r(t-T_r)dt$$

$$\triangle w_1^{AC} = w_1 \int h(t)h'(t)h'_r(t-T_r)dt$$

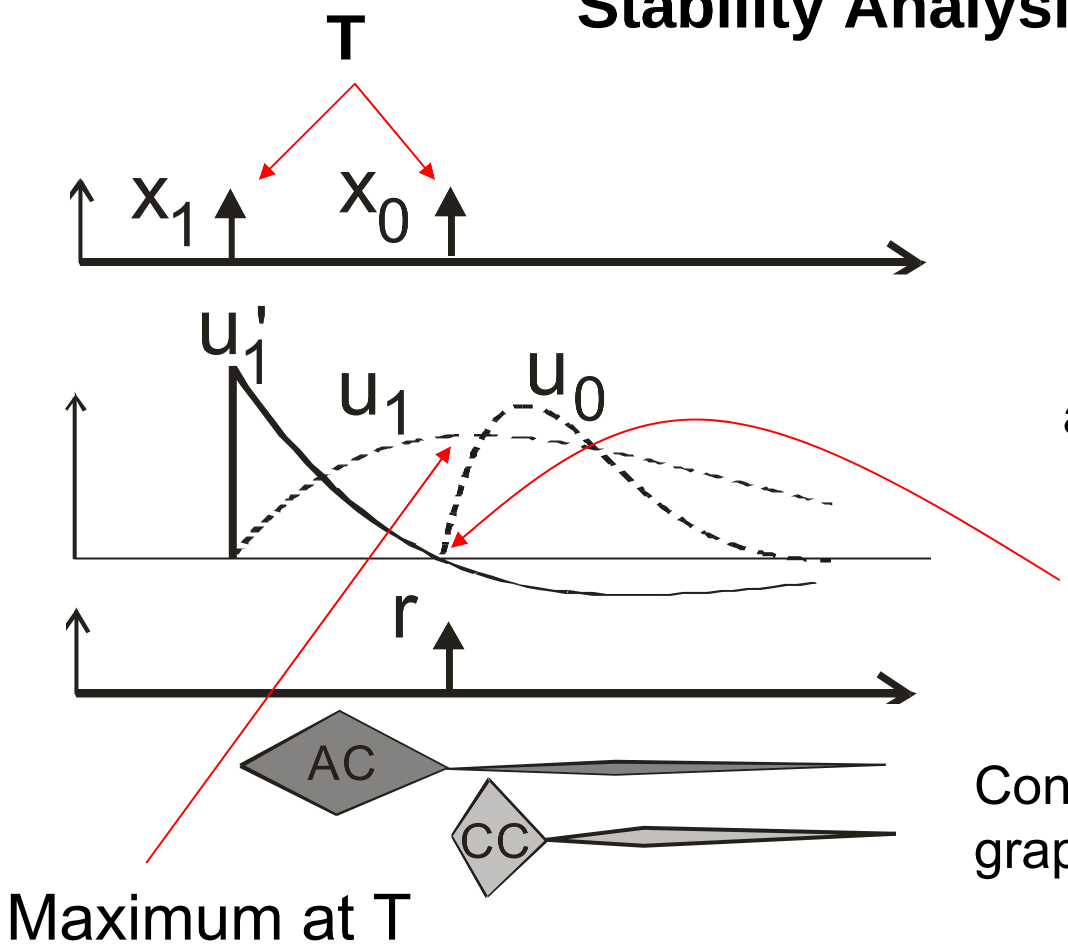
Observations:

1) Cannot be solved anymore!
2) AC term is generally NOT equal to zero.
3) Not even asymptotic convergence can be generally assured.

So what have we gained ?

One can show that for T_r=T the AC term vanishes if v has its maximum at T.

Stability Analysis: ISO3, graphical proof



$$v'(t) = u_1'(t), \quad t < T$$

as x_0 has not yet happened

$$\lim_{t \to T_-} v'(t) = 0$$

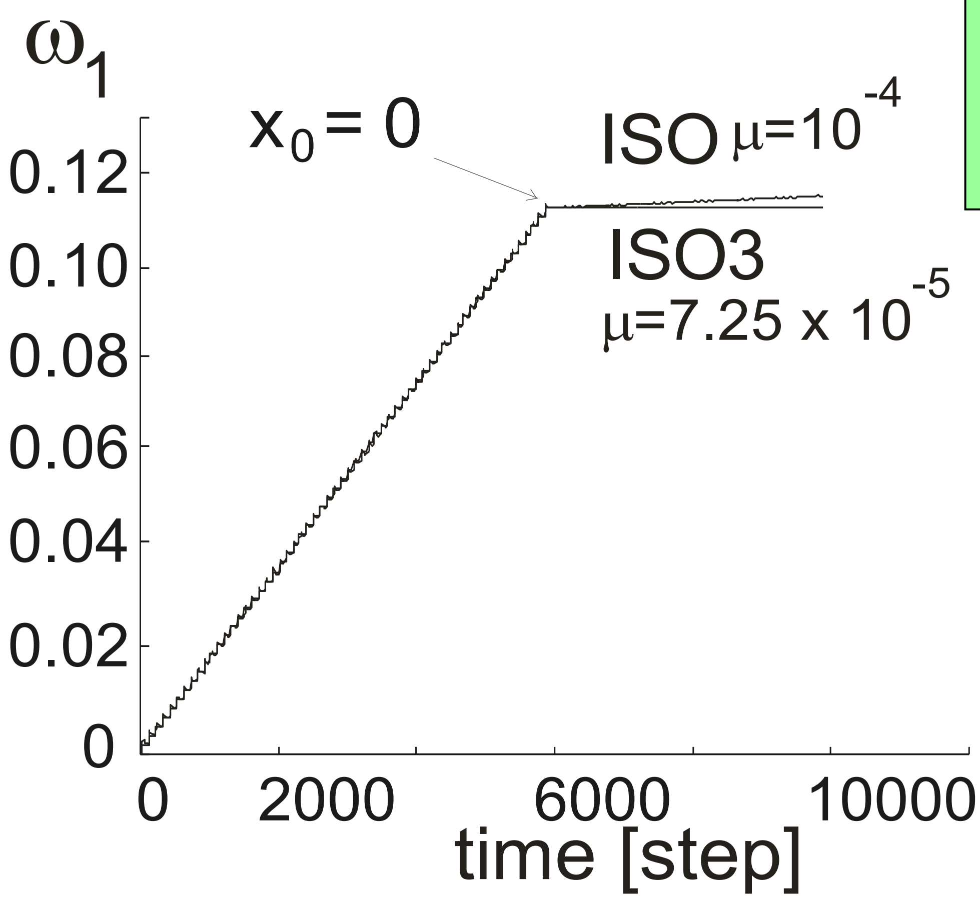
Contributions of AC and CC graphically depicted

If we restrict learning to the moment when x_0 occurs then we do not have any AC contribution.

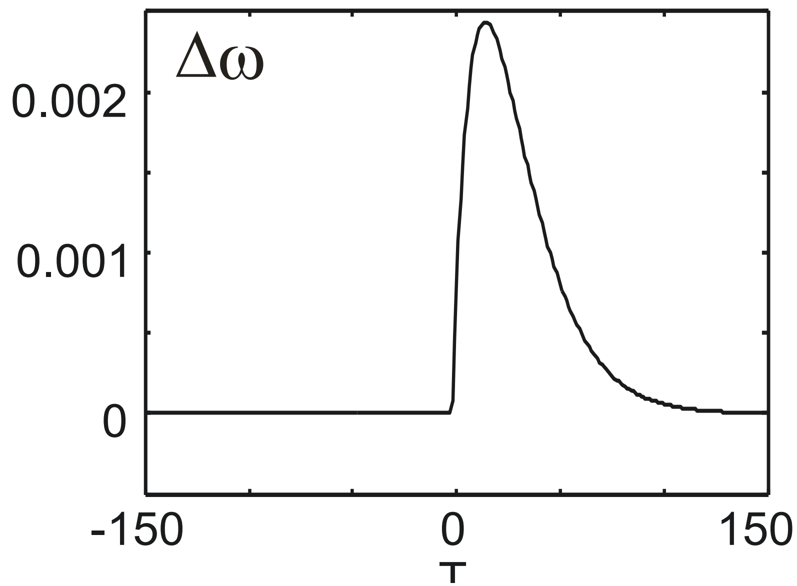
!! A questionable assumption: argmax(u_1) = T !!

Stability Analysis: ISO3

No more upwards drift for ISO3

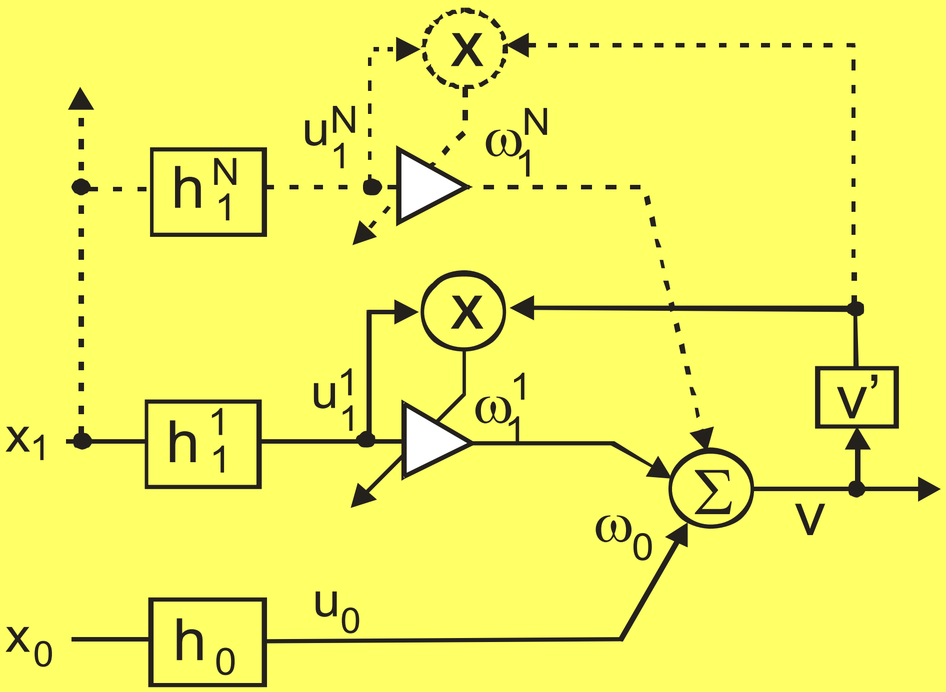


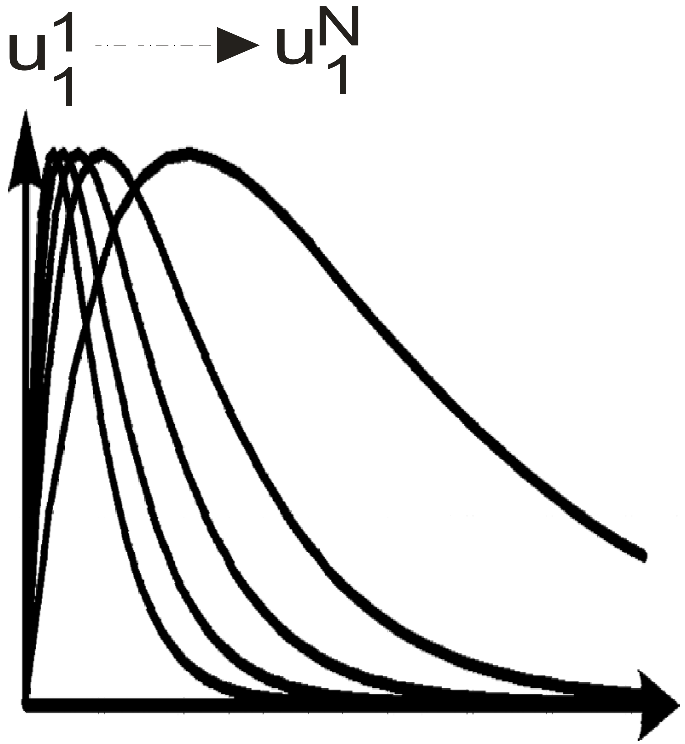
Weight change curve
(no more STDP!)



A General Problem: T is usually unknown and variable

Introducing a **filter bank:**
(example ISO)





Spreading out the earlier input over time!

Remember: "A questionable assumption: argmax(u_1) = T"

Stability Analysis: ISO3 with a filter bank

With a filter bank we get for the output: $v = w_0 u_0 + \sum_{j=0}^{N} w_1^j u_1^j$

Original Rule was:

$$\frac{dw_1}{dt} = \mu \;\; u_1 \;\; v' \;\; R' \Big\rfloor$$

Single weights develop now as:

$$\frac{1}{\mu} \triangle w_1^k = \int w_0 \, u_1^k u_0' R' \rfloor \; + \; \int u_1^k \sum_{j=1}^{N} w_1^j (u_1^j)' R' \rfloor$$

CC AC

With delta-function inputs at t=0 and t=T we get:

$$\frac{1}{\mu} \triangle w_1^k = w_0 u'(0) u_1^k(T) \; + \; \underbrace{\left(\sum_j w_1^j u_1^j(T) \right)'} u_1^k(T)$$

It is possible to prove that this term becomes zero

as a consequence of the learning!