# Reinforcement learning: computational modeling for reward systems

Jaeseung Jeong, Ph.D
Department of Bio and Brain Engineering,
KAIST

# What is Reinforcement learning?

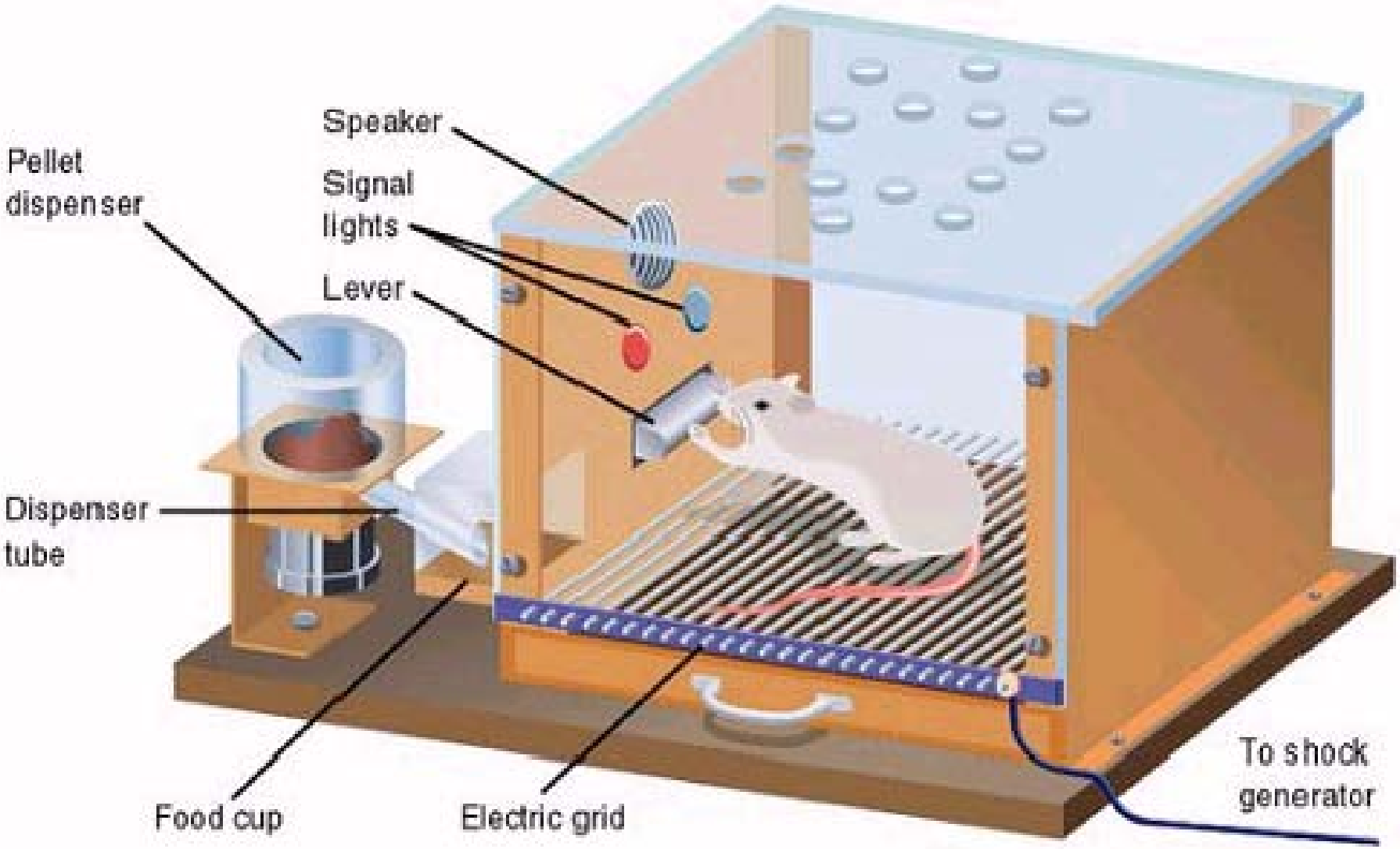
- **Reinforcement learning (RL)** is learning by interacting with an environment.
- An RL agent learns from the consequences of its actions, rather than from being explicitly taught and it selects its actions on basis of its past experiences (exploitation) and also by new choices (exploration), which is essentially **trial and error** learning.
- Exploration and exploitation

# Reward is important in RL

- 'The reinforcement signal' that the RL-agent receives is a numerical reward, which encodes the success of an action's outcome, and the agent seeks to learn to select actions that maximize the accumulated reward over time.

- The use of the term 'reward' is used sometimes in a neutral fashion and does not imply any *pleasure*, *hedonic impact* or other psychological interpretations.

# High sensitivity to reward



# The Algorithmic level (Machine-Learning perspective)

- Markov Decision Problems (MDP): The agent can visit a finite number of *states* and in visiting a state, a numerical *reward* will be collected, where negative numbers may represent punishments.
- Each state has a changeable *value* attached to it. From every state there are subsequent states that can be reached by means of *actions*.
- The value of a given state is defined by the *averaged future reward* which can be accumulated by selecting actions from this particular state. Actions are selected according to a policy which can also change.
- The goal of an RL algorithm is to select actions that maximize the expected cumulative reward (the *return*) of the agent.

# The Prediction Problem

- RL is used to learn the value function for the policy followed.
- At the end of learning, this value function describes for every visited state ‘how much future reward we can expect’ when performing actions starting at this state.

# Control problem

- By interacting with the environment, we wish to find a policy which maximizes the reward when traveling through state space.

- At the end, we have obtained 'an *optimal policy*' which allows for action planning and optimal control.

- Since this is really a predictive type of control, solving the control problem would seem to require a solution to the prediction problem as well.

# How to determine the optimal value function

- (1) If we know the state transition function function T(s,a,s'), which describes the transition probability in going from state s to s' when performing action a, and (2) if we know the reward function r(s,a), which determines how much reward is obtained at a state,

then algorithms can be which are called '*model based algorithms*.'

# How to determine the optimal policy

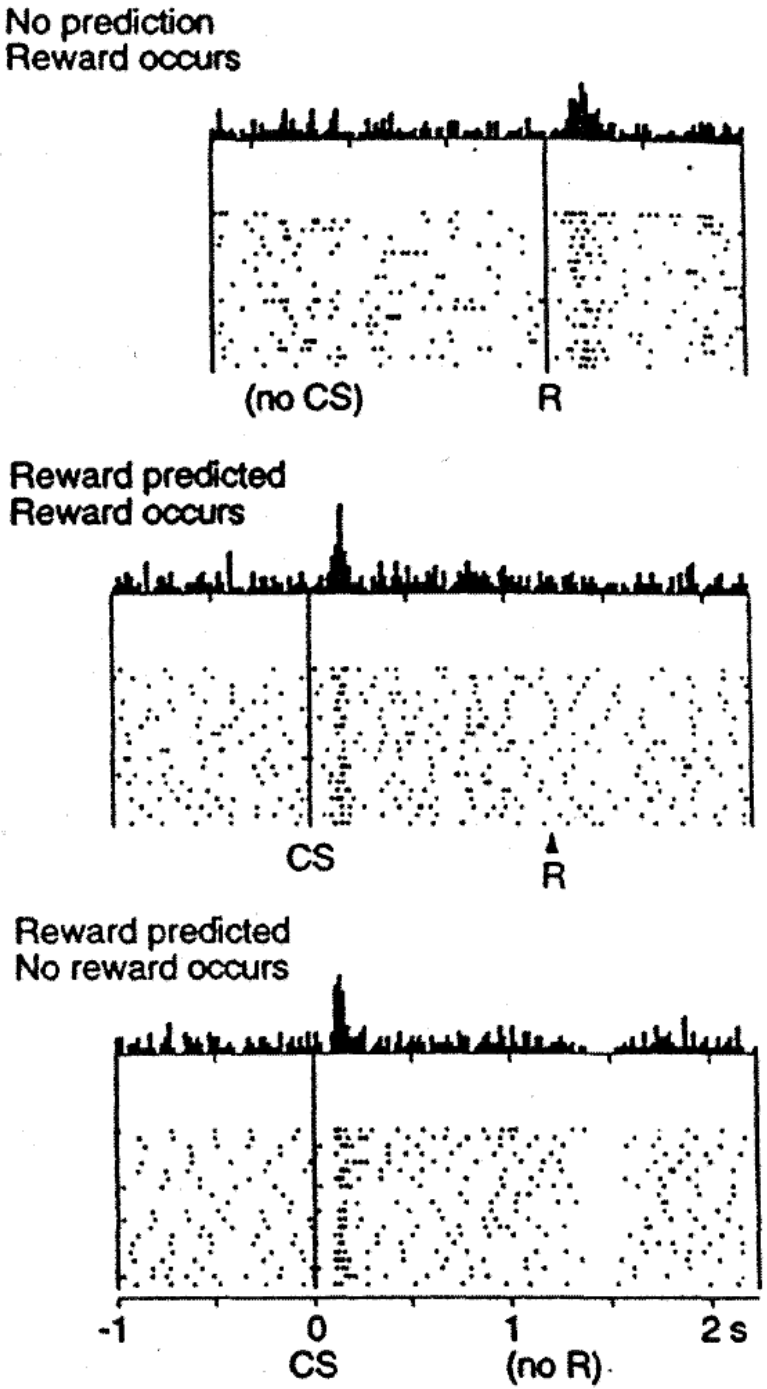
- If the model (T and r) of the process is not known in advance, then we are truly in the domain of RL, where by an adaptive process the optimal value function and/or the optimal policy will have to be learned. The most influential algorithms, which will be described below, are:

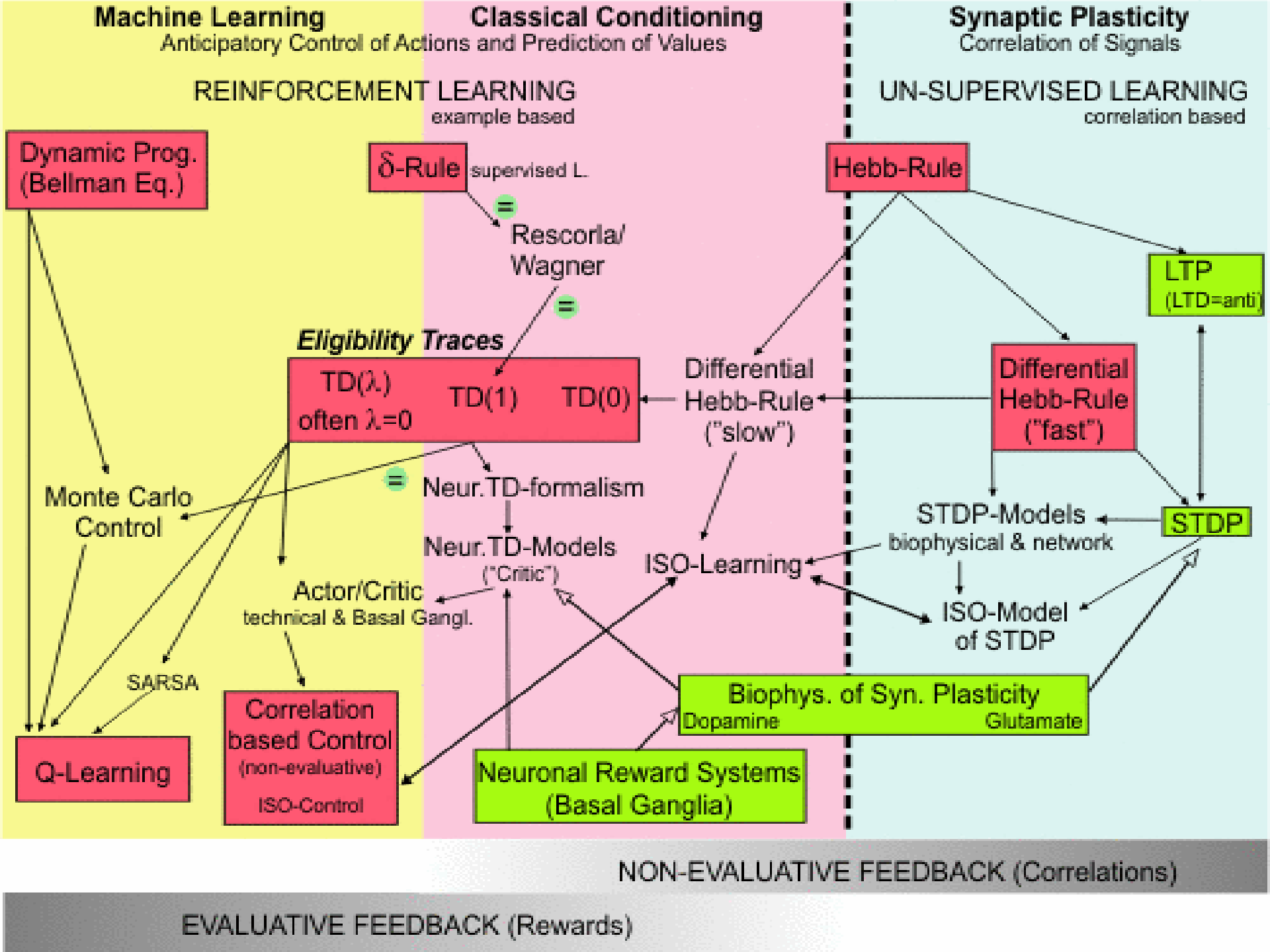
- **Temporal Difference Learning**: by itself used for value function learning,
- **Adaptive Actor-Critics**: an adaptive policy iteration algorithm, which approximates the model of the value function by TD where the TD error is used for both the actor and critic.
- **Q-learning**: a unifying algorithm which allows for simultaneous value function and policy optimization.

# The mechanistic level (Neuronal Perspective)

- The machine learning perspective deals with states, values and actions, etc., whereas the neuronal perspective tries to obtain neuronal signals related to reward-expectation or **prediction-error**.

# Sensitive dependence on unexpected incentives




Machine Learning
Anticipatory Control of Actions and Prediction of Values
Classical Conditioning
Synaptic Plasticity
Correlation of Signals
REINFORCEMENT LEARNING
example based
UN-SUPERVISED LEARNING
correlation based
Dynamic Prog.
(Bellman Eq.)
δ-Rule
supervised L.
=
Rescorla/
Wagner
=
Hebb-Rule
LTP
(LTD=anti)
Eligibility Traces
TD(λ)
often λ=0
TD(1)
TD(0)
Differential
Hebb-Rule
("slow")
Differential
Hebb-Rule
("fast")
=
Neur.TD-formalism
Monte Carlo
Control
STDP-Models
biophysical & network
STDP
Neur.TD-Models
("Critic")
ISO-Learning
Actor/Critic
technical & Basal Gangl.
ISO-Model
of STDP
SARSA
Biophys. of Syn. Plasticity
Dopamine
Glutamate
Correlation
based Control
(non-evaluative)
ISO-Control
Q-Learning
Neuronal Reward Systems
(Basal Ganglia)
NON-EVALUATIVE FEEDBACK (Correlations)
EVALUATIVE FEEDBACK (Rewards)

# Supervised or unsupervised?

- The difference should be discussed between evaluative and non-evaluative feedback, where we associated evaluative feedback to [[supervised learning]] (feedback from a teacher) and rightfully stated that the environment does not produce any evaluation.

- Feedback that arrives from the environment at the sensors of a creature can only be non-evaluative.

- Any evaluation, in this case, must be performed only internally by the animal itself. Because animals don't receive evaluative feedback, RL would appear to be an example of unsupervised learning.

# Temporal difference (TD) methods

- TD methods need to predict future rewards. In order to achieve this, TD learning use value functions V(t) which assign values to states and then calculates the change of those values by ways of a temporal derivative.

- As a consequence, these methods are related to methods of correlation based, **differential Hebbian learning**, where a synaptic weight changes by the correlation between its input signals with the derivative of its output.

# Temporal difference (TD) methods

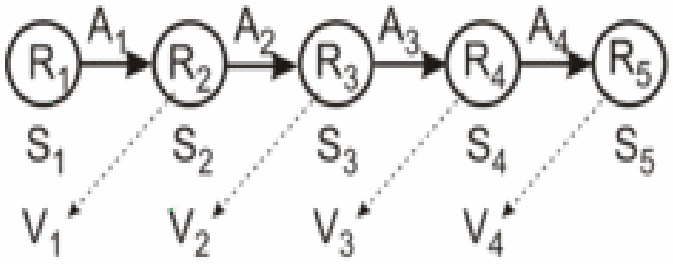
- Sutton and Barto's 1981 paper, however, really also described a differential Hebbian learning rule.

- Differential Hebbian rules moved back into the focus of interest only after 1997, when they had been related to spike-timing dependent plasticity (Markram et al 1997).

- In this new context, Gerstner et al rediscovered these rules in 1996 (Gerstner et al 1996) and they had been applied to RL control problems some years later by Porr and coworkers (Porr and Wörgötter 2003, 2006).

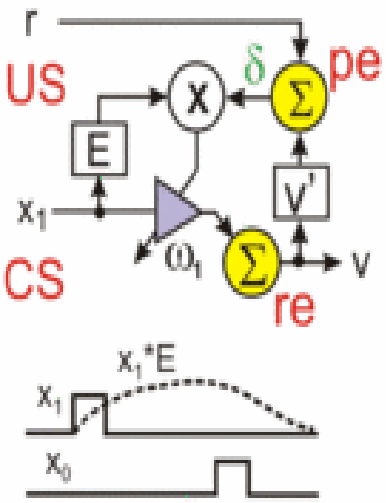
A Machine Learning Perspective

TD-learning

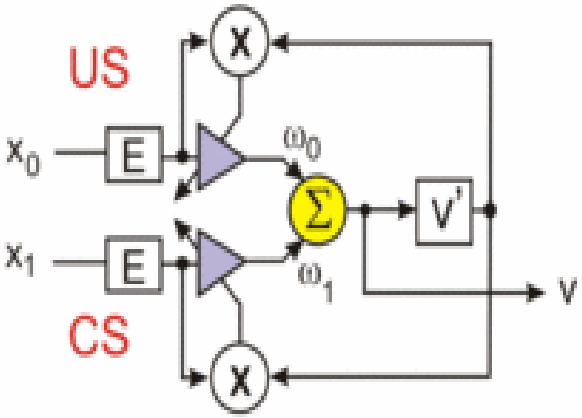



Neuronal Perspective B

TD-learning




ISO-Learning



# What is TD model?

- We assume that a neuron v can approximately predict a reward r , then we should at every time step t, t+1 find that $v(t) \cong R(t)$ and $v(t+1) \cong R(t+1)$.

- Since this is only approximately true (until convergence) we can in the same way define the error by:

$\delta = r(t+1)+v(t+1)-v(t) = r(t+1)-v'$ (neglecting discount factors here for brevity).

# TD model

- Thus we can update weight w 1 with: δω 1 =[x 1 ∗E]δ , where x 1 ∗E is a convolutionwith the filter kernel E and denotes the fact that inputx 1 needs to be remembered for some time in the system. The function E is usually a first order low pass response and is also known as eligibility trace. Because the errorδ occurs later than x 1 , the correlation of δx 1 would be zero without this type of memory.

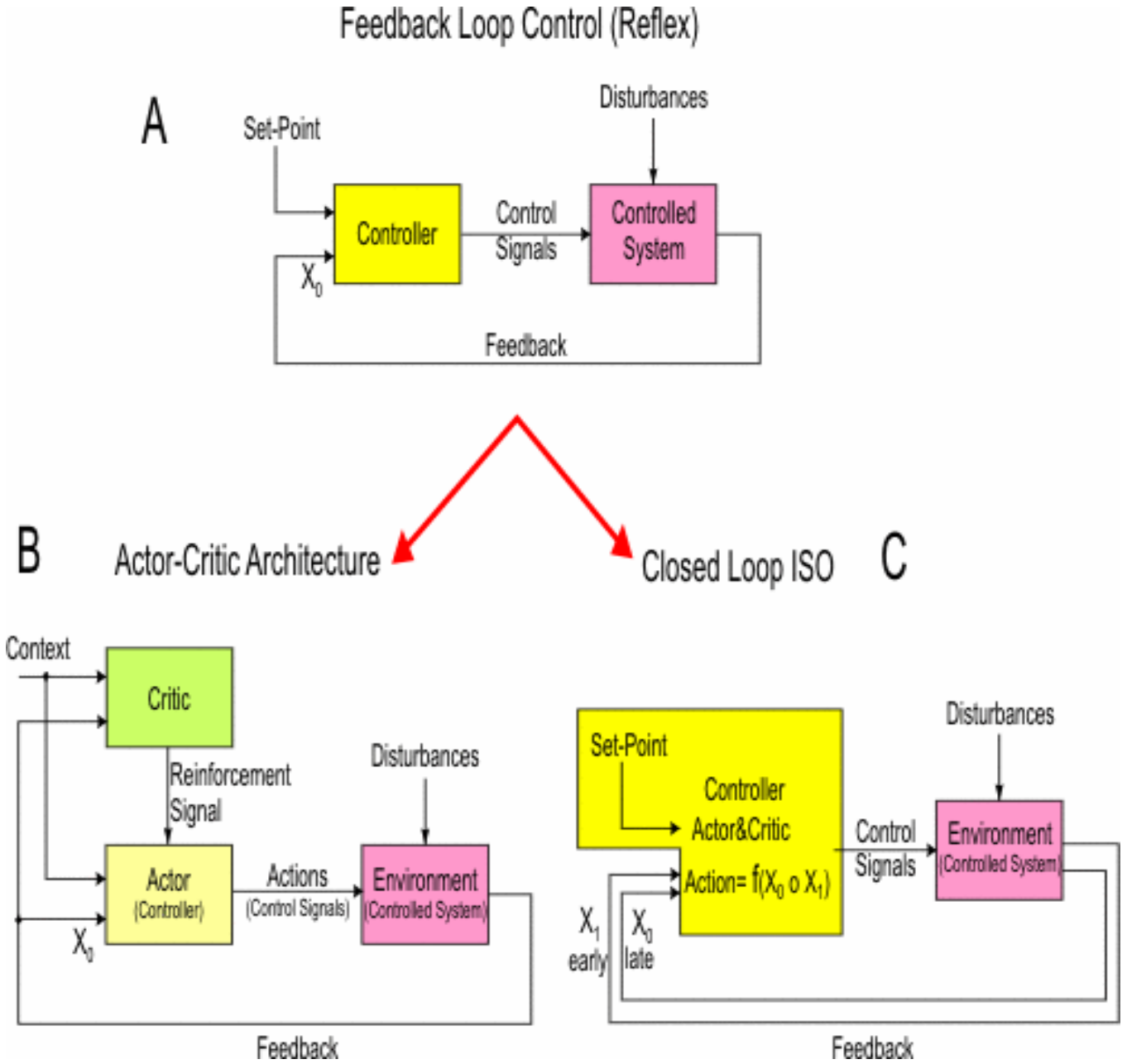
# TD model

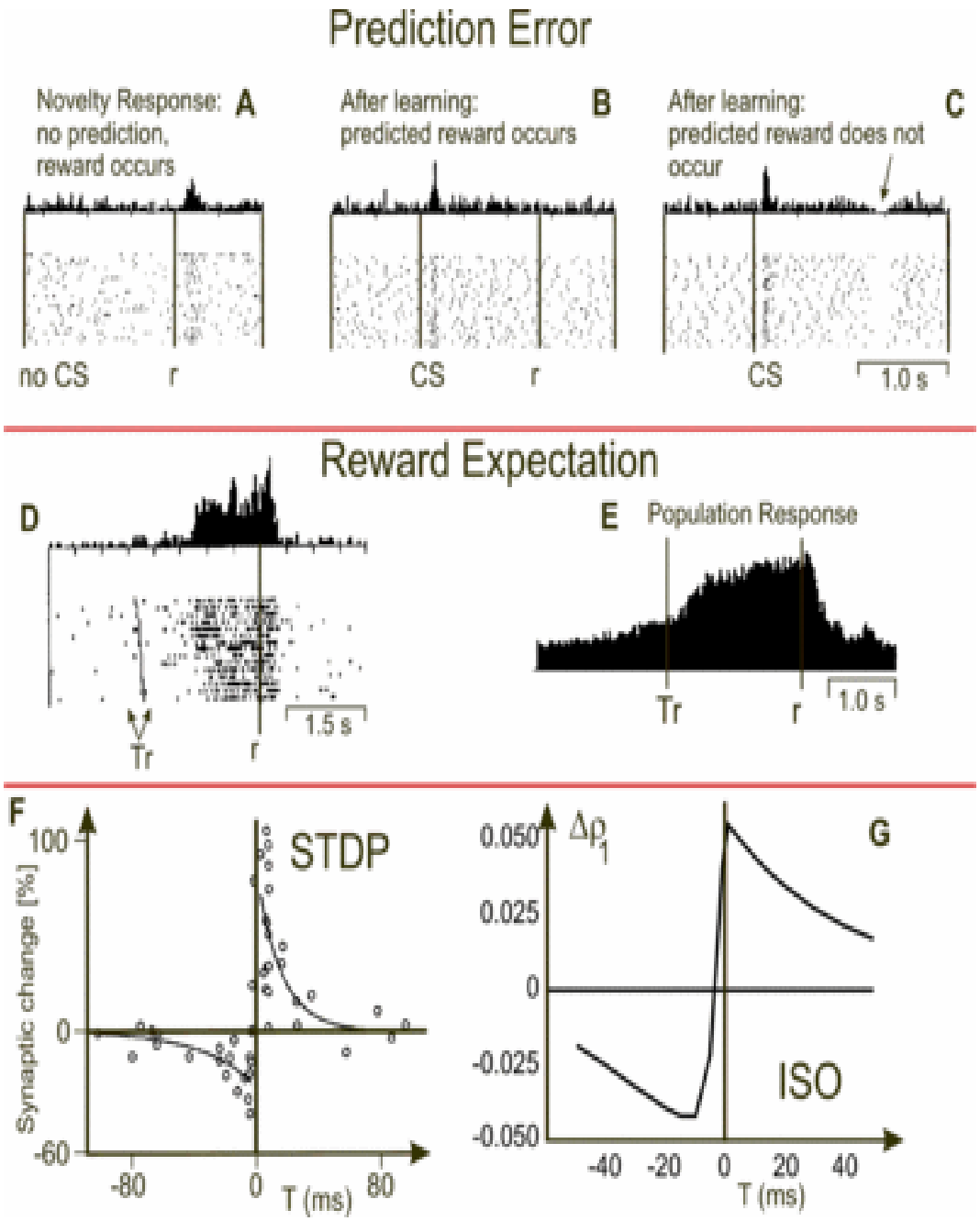
- Figure 2B shows the basic TD-rule for a neuron with one predictive CS-input x 1 and a reward liner , the US. When combining this with a delay line which splits x 1 into many inputs, each delayed with respect to each other by a unit delay (serial compound representation) this algorithm emulates the backward-TD(1) procedure.

# Neuronal control model

- **Actor-Critic Architectures**: Figure 3A shows a conventional feedback control system. In neuronal terms this is a reflex-loop. A controller provides control signals to a system, which is influenced by disturbances. Feedback allows the controller to adjusts it signals. In addition, a set-point is defined which the control loop tries to approximate. Part B shows how to extend this into an Actor-Critic architecture (Witten 1977, Barto et al. 1983, Sutton 1984, Barto 1995).

- The Critic produces evaluative, reinforcement feedback for the Actor by observing the consequences of its actions. The Critic takes the form of a TD-error, which gives an indication if things have gone better or worse than expected with the preceding action. If the TD-error is positive the tendency to select this action should be strengthened or else, lessened. Thus, Actor and Critic are adaptive through reinforcement learning.


Feedback Loop Control (Reflex)
A
Set-Point
Disturbances
Controller
Control Signals
Controlled System
$X_0$
Feedback
B
Actor-Critic Architecture
C
Closed Loop ISO
Context
Critic
Reinforcement Signal
Disturbances
Actor (Controller)
Actions (Control Signals)
Environment (Controlled System)
$X_0$
Feedback
Set-Point
Controller Actor&Critic
Action= $f(X_0 \circ X_1)$
Control Signals
Disturbances
Environment (Controlled System)
$X_1$ early
$X_0$ late
Feedback

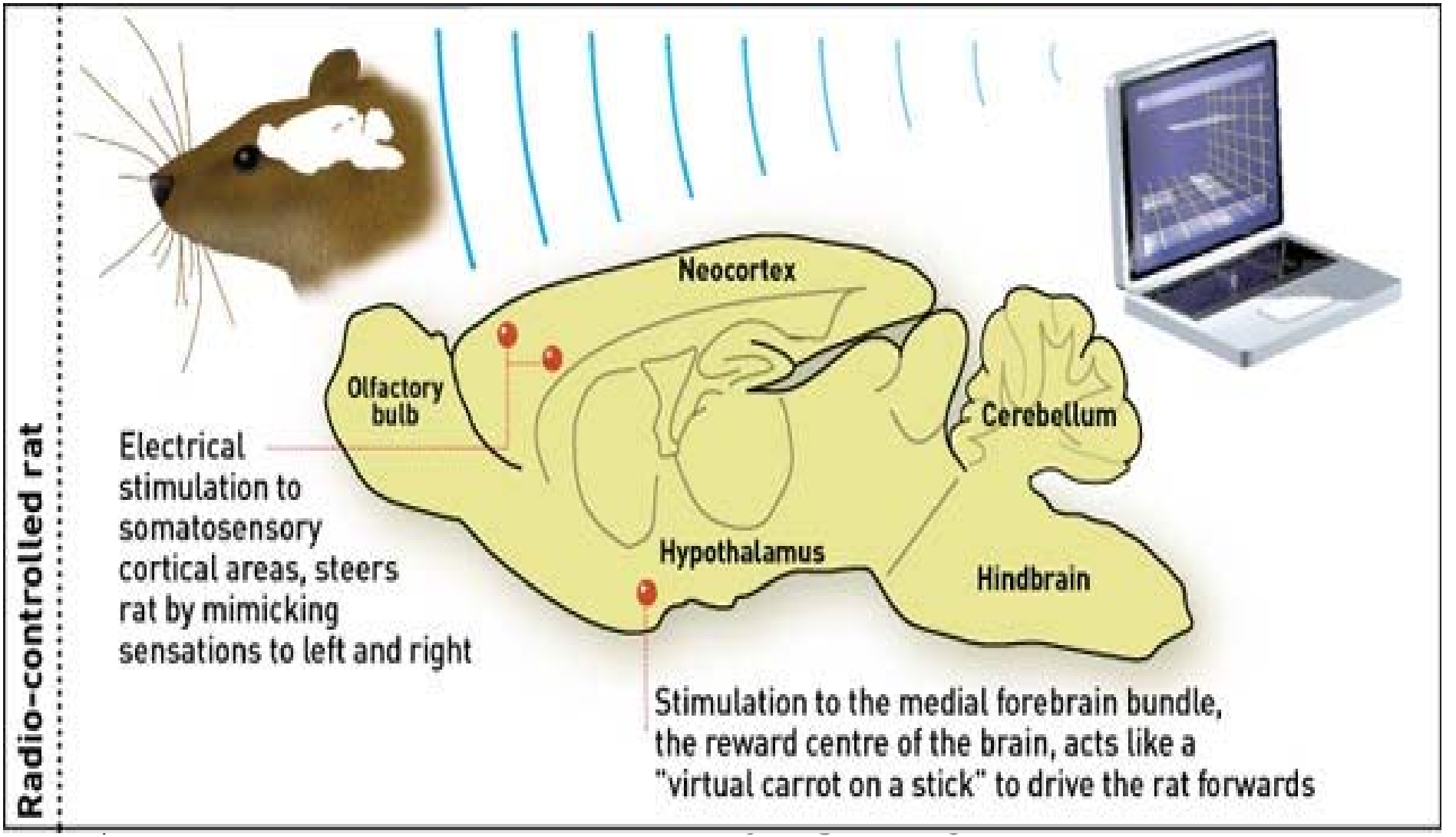

Prediction Error
Novelty Response:
no prediction,
reward occurs
A
After learning:
predicted reward occurs
B
After learning:
predicted reward does not
occur
C
no CS
r
CS
r
CS
1.0 s
Reward Expectation
D
Tr
r
1.5 s
E
Population Response
Tr
r
1.0 s
F
STDP
Synaptic change [%]
100
0
-60
-80
0
80
T (ms)
G
$\Delta\rho_1$
0.050
0.025
0
-0.025
-0.050
ISO
-40
-20
0
20
40
T (ms)

- Reinforcement learning is also reflected at the level of neuronal sub-systems or even at the level of single neurons. In general theDopaminergic system of the brain is held responsible for RL. Responses from dopaminergic neurons have been recorded in the Substantia Nigrapars compacta (SNc) and the Ventral Tegmental Area (VTA) where some reflect the *prediction error* $\delta$ of TD-learning (see Figure 3B *pe*). Neurons in the Striatum, orbitofrontal cortex and Amygdala seem to encode reward expectation (for a review see Reward Signals, Schultz 2002, see Figure 3B *re*). These neurons have been discovered mostly in conjunction with appetitive (food-related) rewards. Figure 4 shows some examples of prediction error- as well as reward expectation neurons.

- However, only few dopaminergic neurons produce error signals that comply with the demands of reinforcement learning. Most dopaminergic cells seem to be tuned to arousal, novelty, attention or even intention and possibly other driving forces for animal behavior. Furthermore the TD-rule reflects a well-defined mathematical formalism that demands precise timing and duration of theδ error, which cannot be guaranteed in the basal ganglia or the limbic system (Redgrave et al. 1999).

- Consequently, it might be difficult to calculate predictions of future rewards. For that reason alternative mechanisms have been proposed which either do not rely on explicit predictions (derivatives) but rather on a Hebbian association between reward and CS (O'Reilley et al. 2007), or which use the DA signal just as a switch which times learning after salient stimuli (Redgrave and Gurney 2007, Porr and Wörgötter2007). Hence the concept of derivatives and therefore predictions has been questioned in the basal ganglia and the limbic system and alternative more simpler mechanisms have been proposed which reflect the actual neuronal structure and measured signals.

- Differential Hebbian learning (e.g. ISO-rule) seem to be to some degree compatible with novel findings on spike-timing dependent synaptic plasticity (STDP, Markram et al 1997). In this type of plasticity, synapses potentiate (become stronger) when the presynaptic input is followed by post-synaptic spiking activity, while else they are depressed (become weaker). The multiplicative (correlative) properties necessary to emulate a Hebb rule can be traced back to second messenger chains, which phosphorilate AMPA receptors and the required *differential* aspect appears to arise from the sensitivity of real synaptic plasticity to Calcium *gradients* (Lindskog et.al. 2006). Figure 4 shows two examples of weight change curves (often called *learning window*) from a real neuron and from a differential Hebbian learning rule emulated to be compatible with some basic biophysical characteristics (Saudargiene et al 2004).




that took place inside the dashed enclosure. **b,** Route taken by a rat guided over a three-dimensional obstacle course. The animal was instructed to climb a vertical ladder, cross a narrow ledge, descend a flight of steps, pass through a hoop and descend a steep (70°) ramp. Two rounds of high-density MFB stimulation were required to guide the rat successfully down the ramp, demonstrating the motivational qualities of MFB stimulation.

- **Exploration-Exploitation Dilemma**
- RL-agents need to explore their environment in order to assess its reward structure. After some exploration the agent might have found a set of apparently rewarding actions. However, how can the agent be sure that the found actions where actually the best? Hence, when should an agent continue to explore or else, when should it just exploit its existing knowledge? Mostly heuristic strategies are employed for example *annealing-like* procedures, where the naive agent starts with exploration and its exploration-drive gradually diminishes over time, turning it more towards exploitation. The annealing rate, however depends also on the structure of the world and especially also on the graining of the state space and cannot be decided without guided guessing. Recently Singh et al, 2002 have developed more efficient solutions for the exploration/exploitation dilemma.