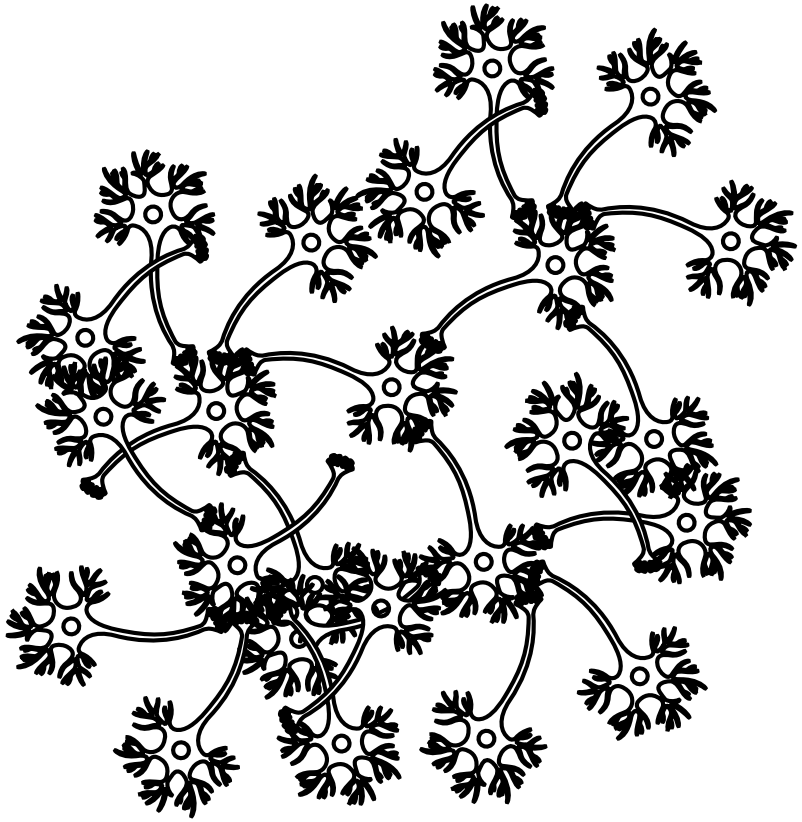
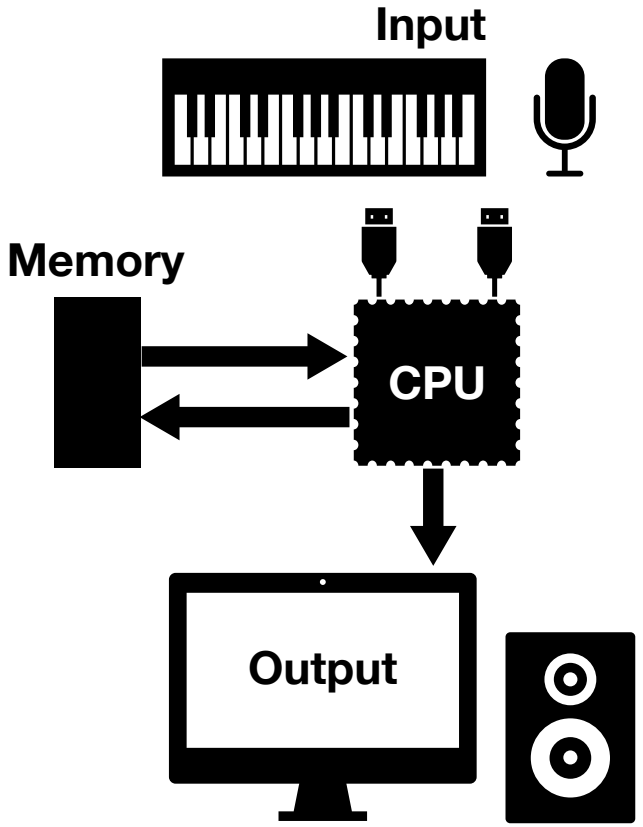
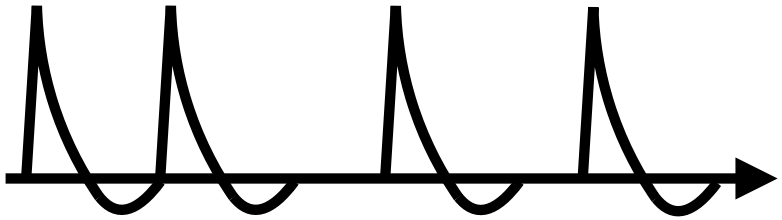
Three Perspectives on
Neuromorphic Computing

Algorithm Designer

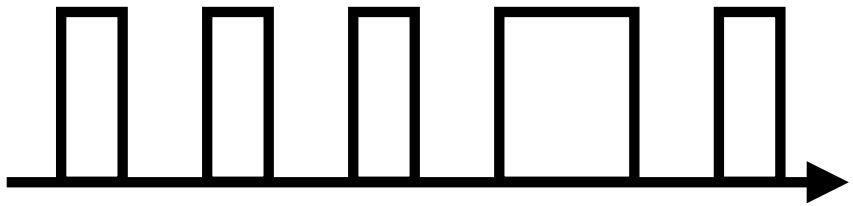
NOEL







Neuromorphic Architecture



Von-Neuman Architecture

NOEL

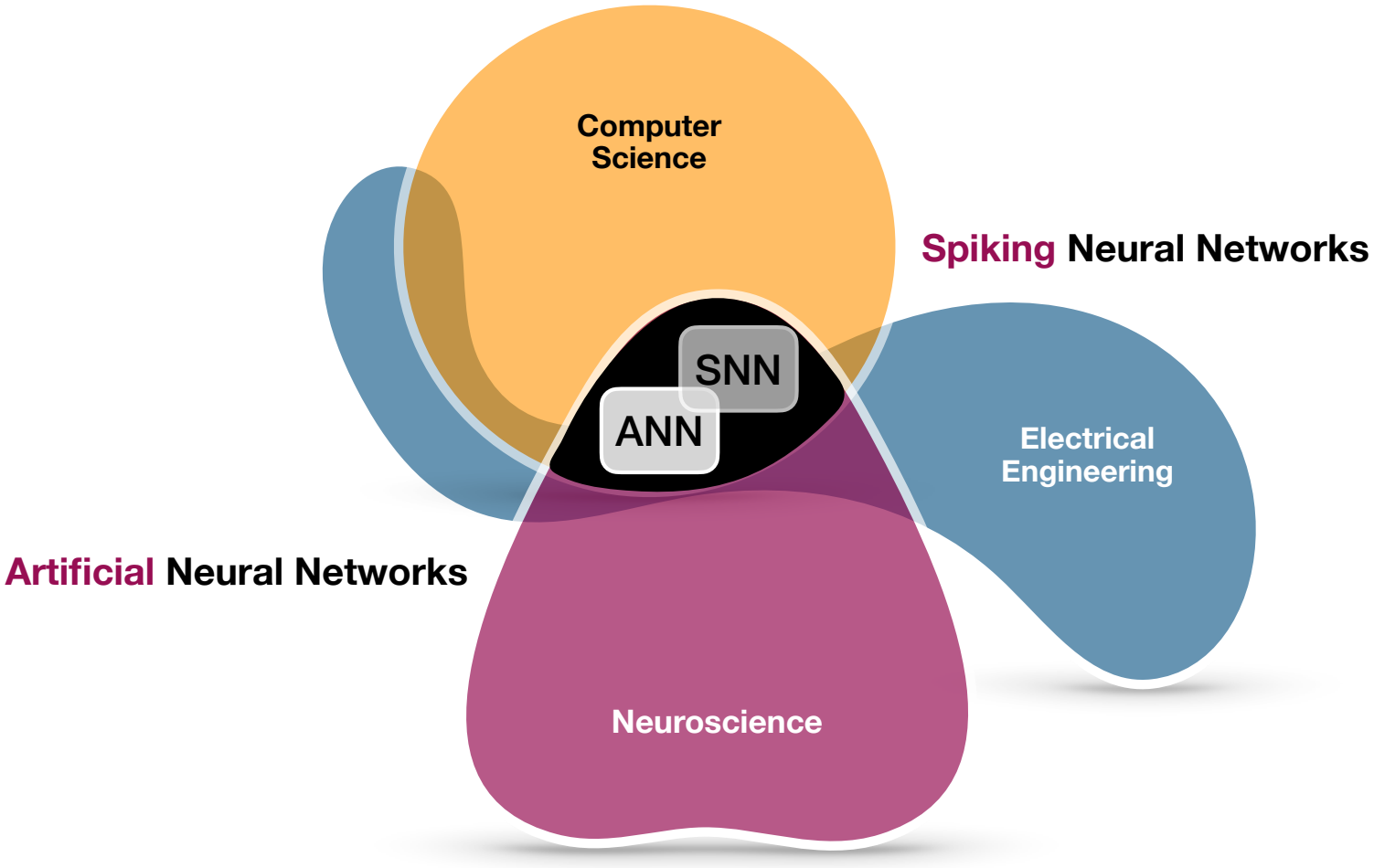
Can Programming Be Liberated from the von Neumann Style? A Functional Style and Its Algebra of Programs

John Backus
IBM Research Laboratory, San Jose

"It is an **intellectual bottleneck** that has kept us tied to **word-at-a-time thinking** instead of encouraging us to think in terms of the **larger conceptual units**"

NOEL

Neuromorphic Engineering



NOEL

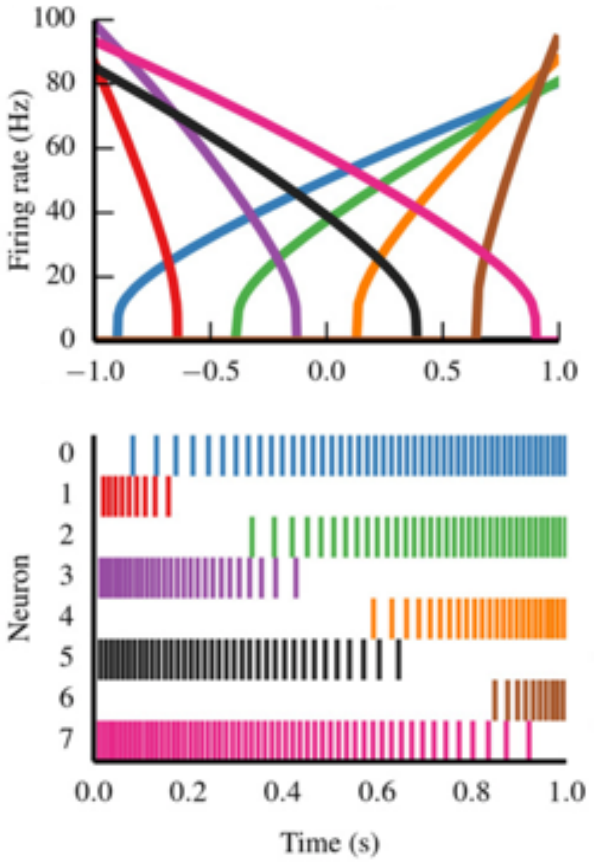
Neural Engineering Framework

Computing with Spiking Neural Networks

Neural Engineering Framework

Computing with Spiking Neural Networks

Representation

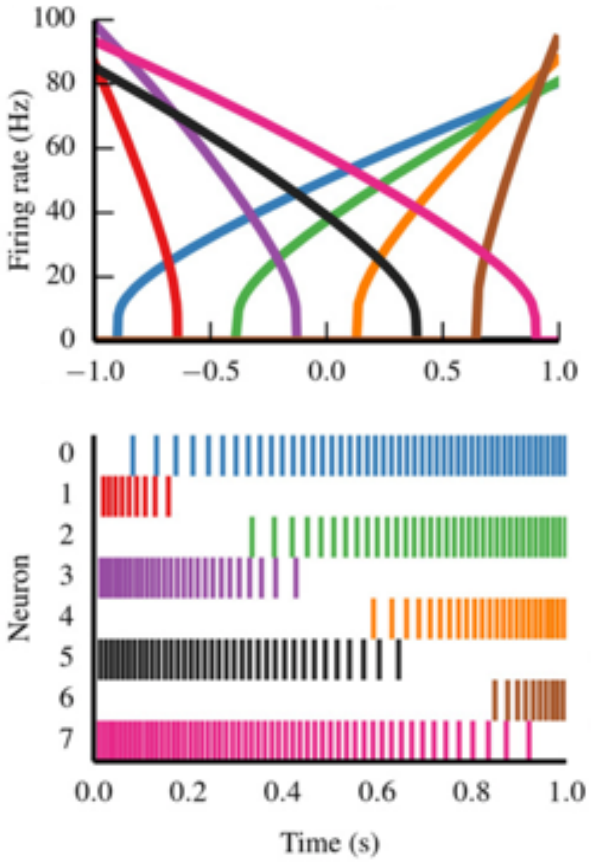


NOEL

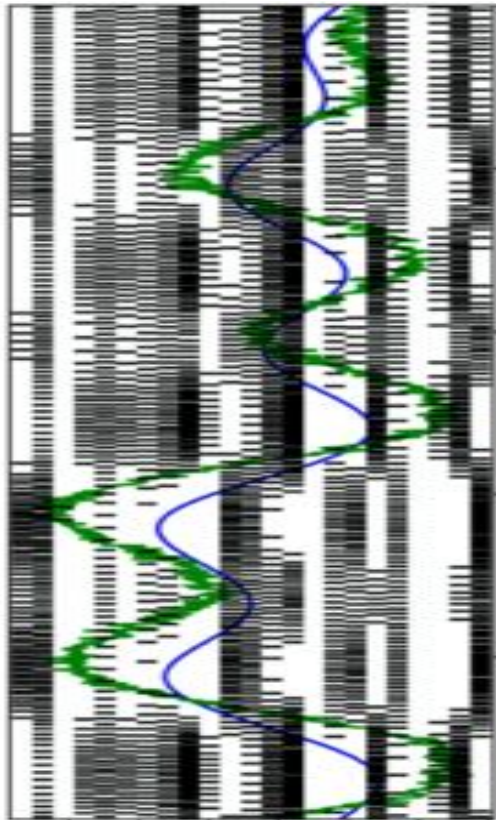
Neural Engineering Framework

Computing with Spiking Neural Networks

Representation



Transformation

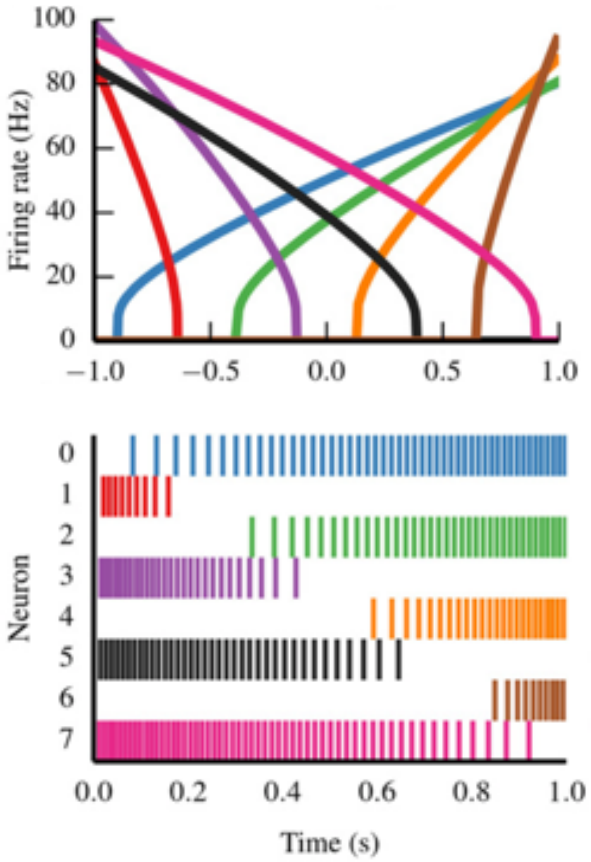


NOEL

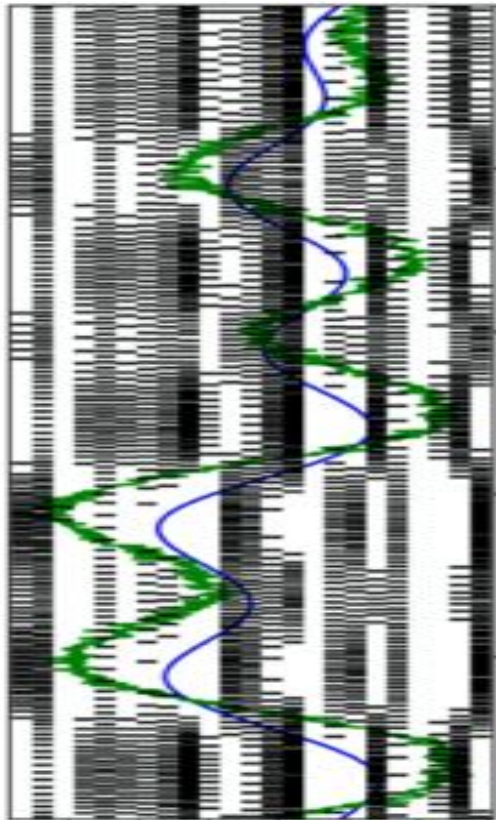
Neural Engineering Framework

Computing with Spiking Neural Networks

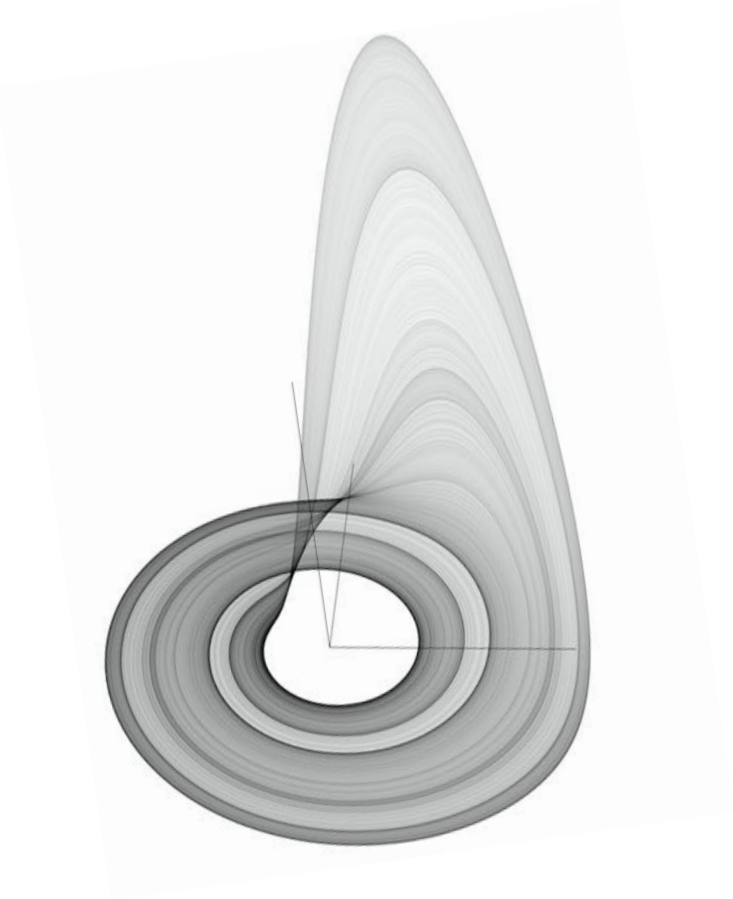
Representation



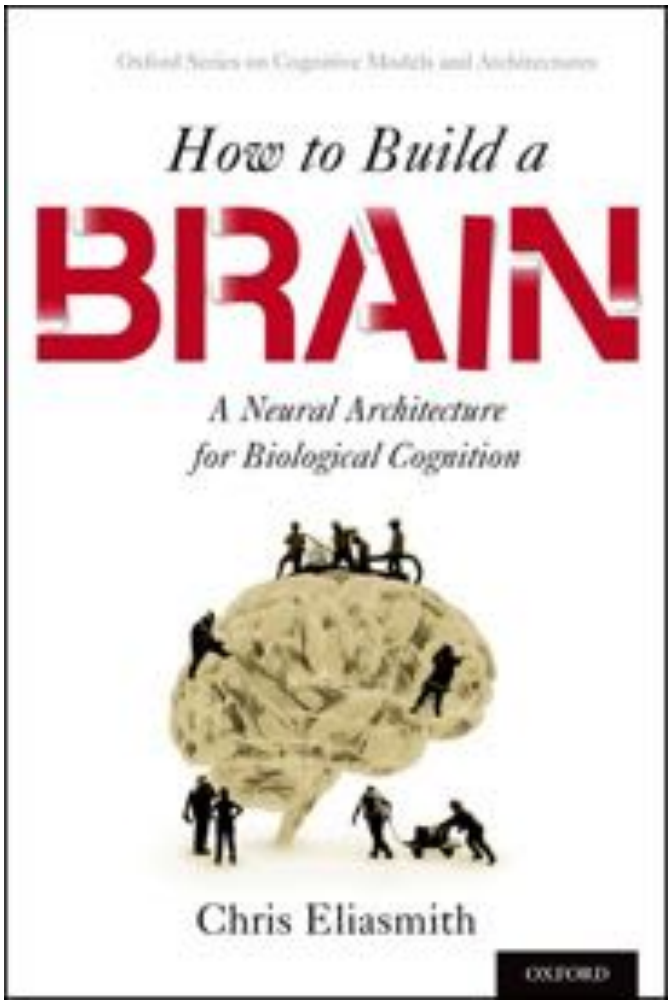
Transformation



Dynamics

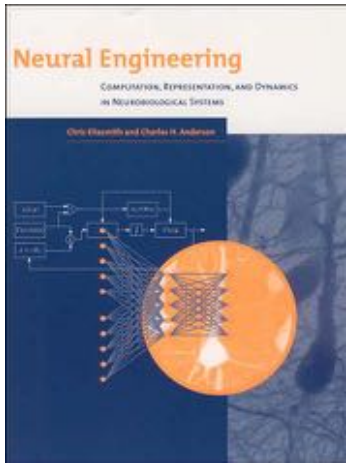


The Neural Engineering Framework (NEF)



Throwing out most of the cortex's complicated **anatomy** and **physiology**.

Identified a **minimal set of neuronal properties** - linear exponentially decaying **synaptic responses** and heterogeneous nonlinear **somatic responses** - sufficiently powerful to **approximate** any linear or nonlinear **dynamical** system.

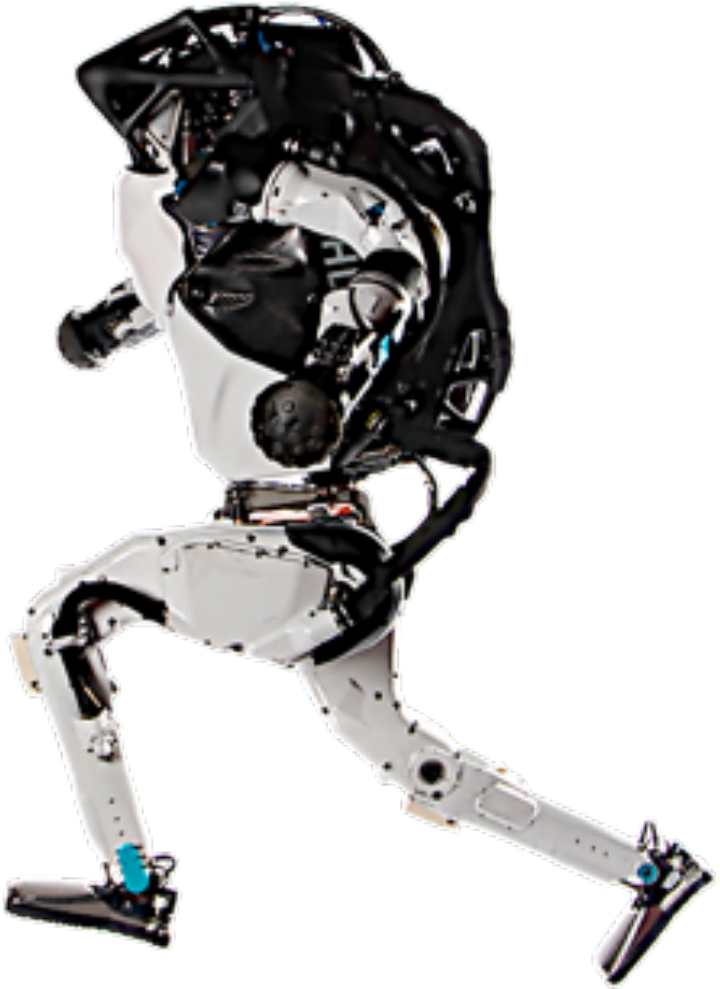


NOEL

SPOT By Boston Dynamics

Main Characteristic

Control System / Feedback Loops

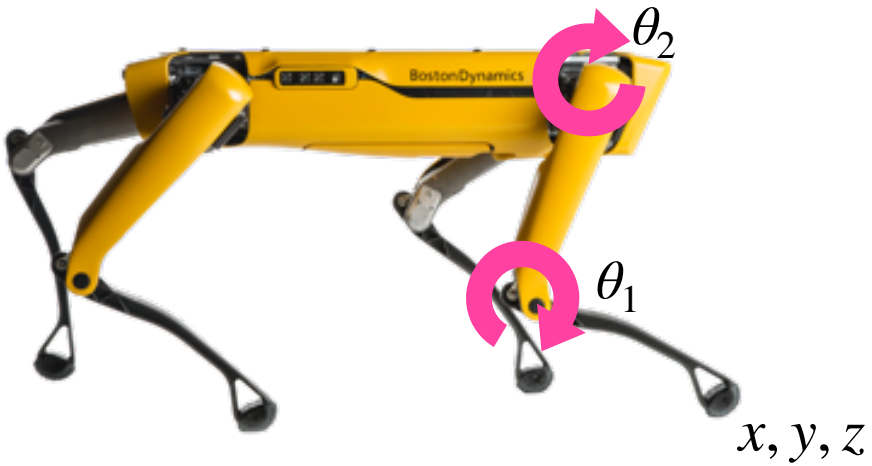


NOEL

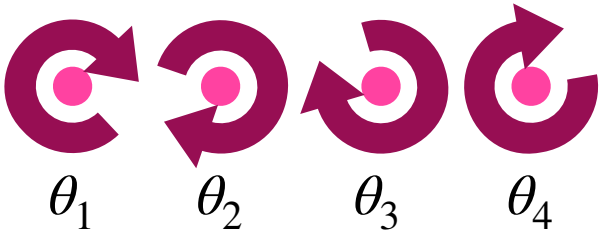
Robotics Fundamentals

Forward Transformations

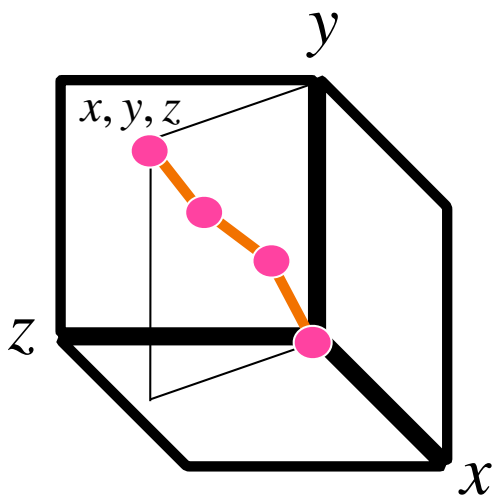
Robotic arm/leg



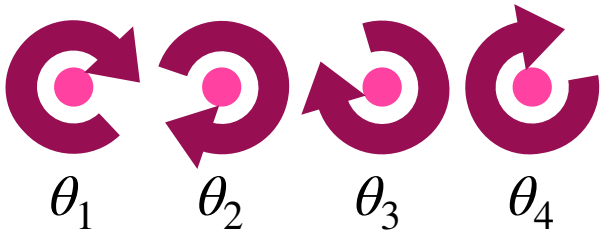
Joint-angle space

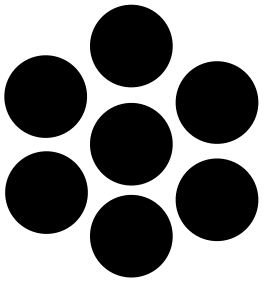


Operational (task) space

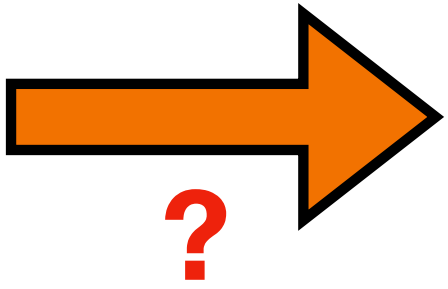


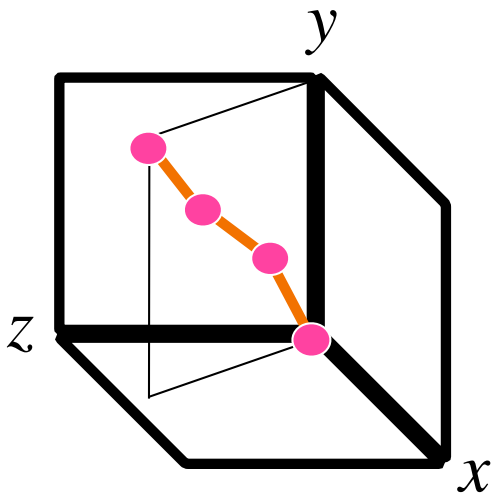
Neuromorphic Robotic Control with Spiking Neural Networks

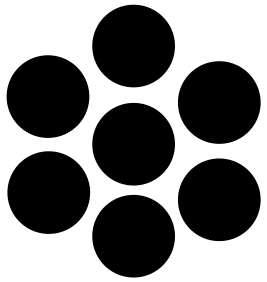




Neuron Ensemble







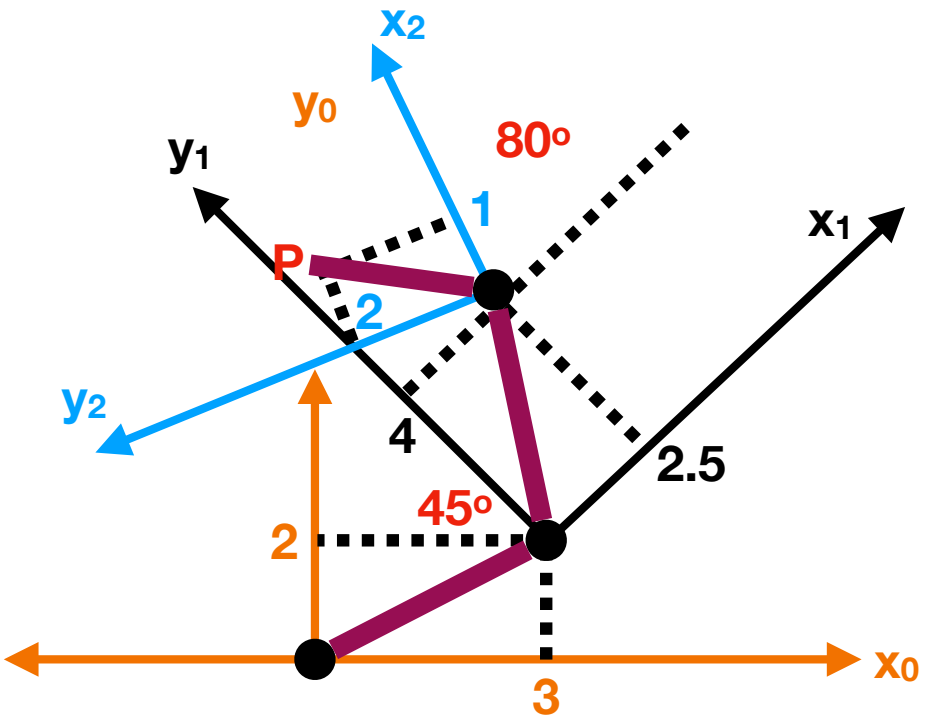
Neuron Ensemble

NOEL

Robotics Fundamentals

Forward Transformations

Yuval Zaidel



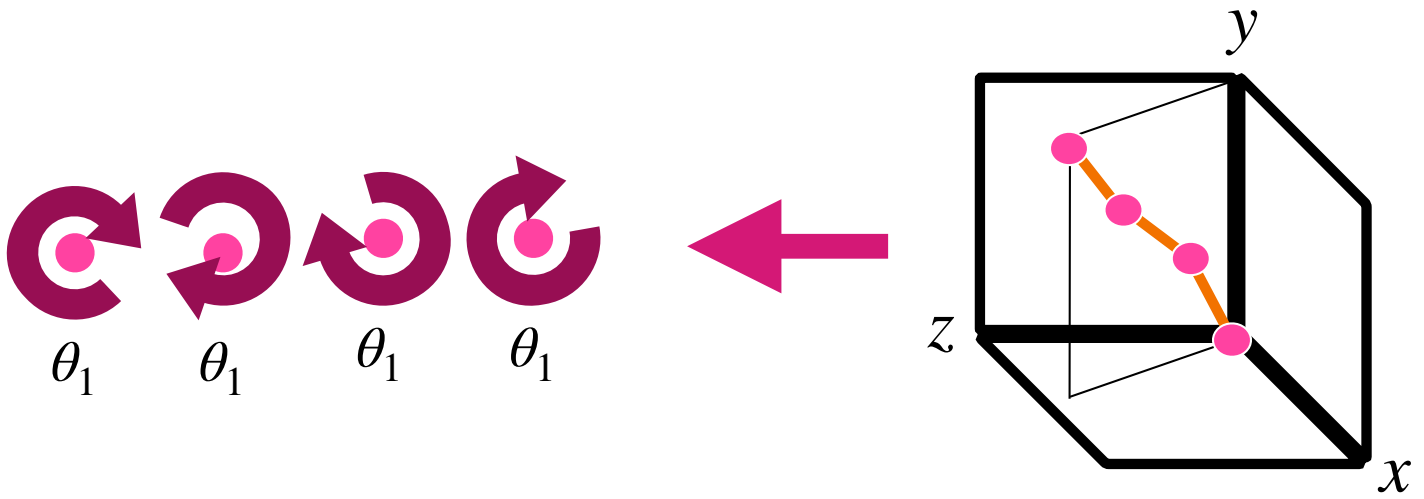
$$ {}^{2}_{1}T = \begin{pmatrix} cos(80) & -sin(80) & 2.5 \\ sin(80) & cos(80) & 4 \\ 0 & 0 & 1 \end{pmatrix} \quad {}^{1}_{0}T = \begin{pmatrix} cos(45) & -sin(45) & d_3 \\ sin(45) & cos(45) & d_2 \\ 0 & 0 & 1 \end{pmatrix} $$

$$ P_0 = {}^{1}_{0}T\,{}^{2}_{1}T \cdot p_2 = {}^{2}_{0}T \cdot p_2 = \begin{pmatrix} cos(45) & -sin(45) & d_3 \\ sin(45) & cos(45) & d_2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} cos(80) & -sin(80) & 2.5 \\ sin(80) & cos(80) & 4 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} -0.273 \\ 6.268 \\ 1 \end{pmatrix} $$

Robotics Fundamentals

Inverse Kinematics

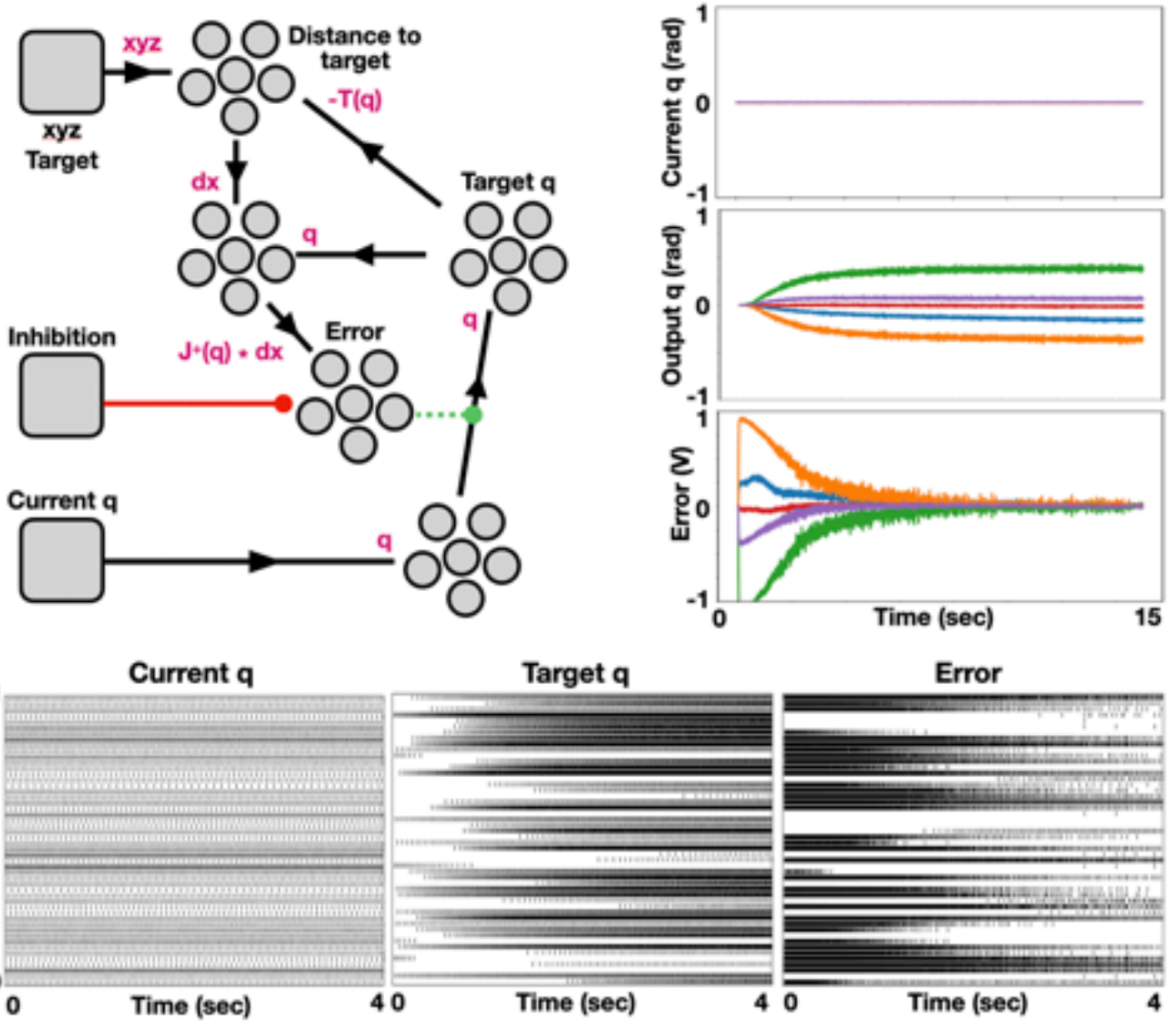
Yuval Zaidel



Math is more complicated
Need to optimise (multiple ways to achieve one goal)

Inverse Kinematics

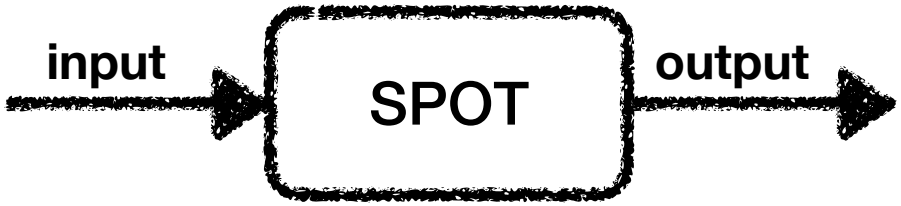
Yuval Zaidel



Main Characteristic

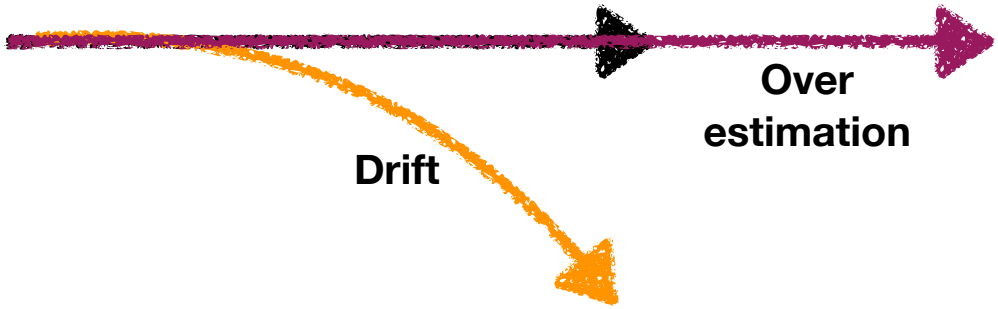
Control System / Feedback Loops

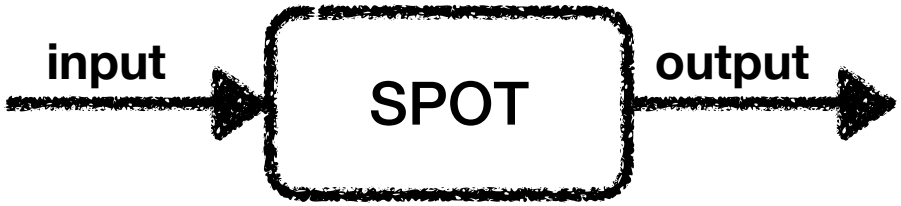
Calculate time based on movement velocity and distance from goal



Main Characteristic

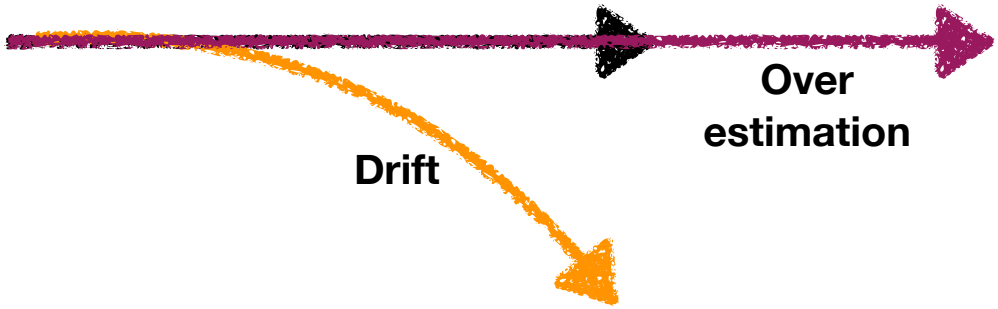
Control System / Feedback Loops

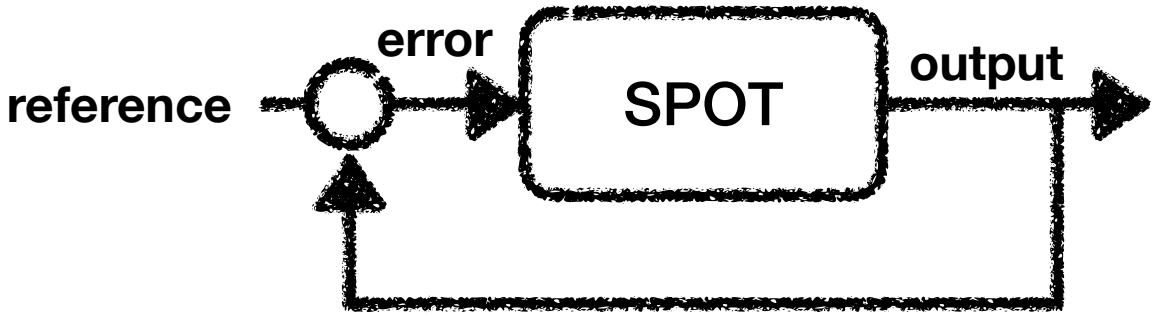




Main Characteristic

Control System / Feedback Loops

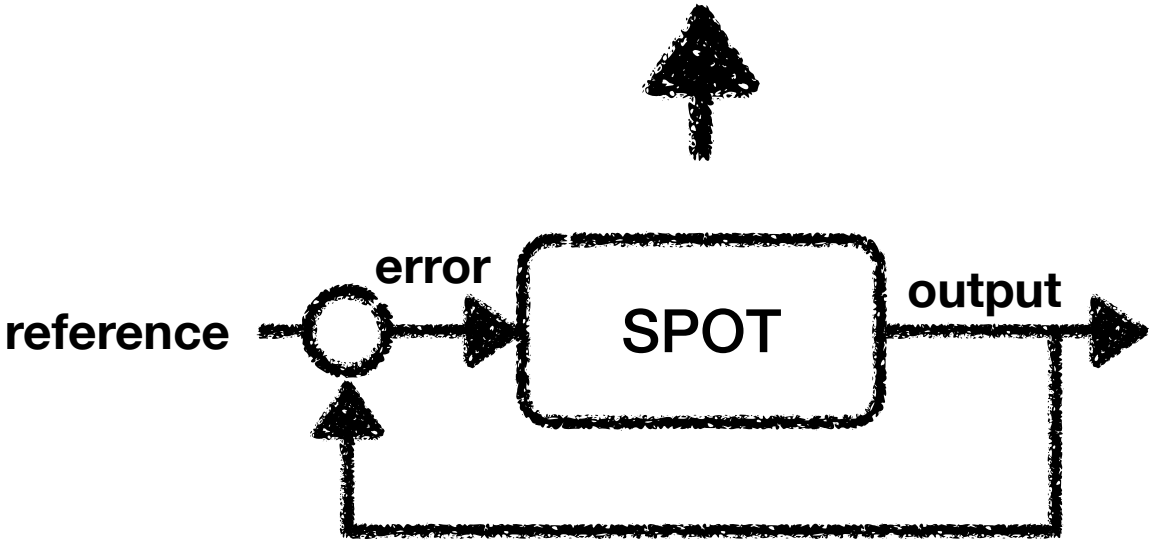




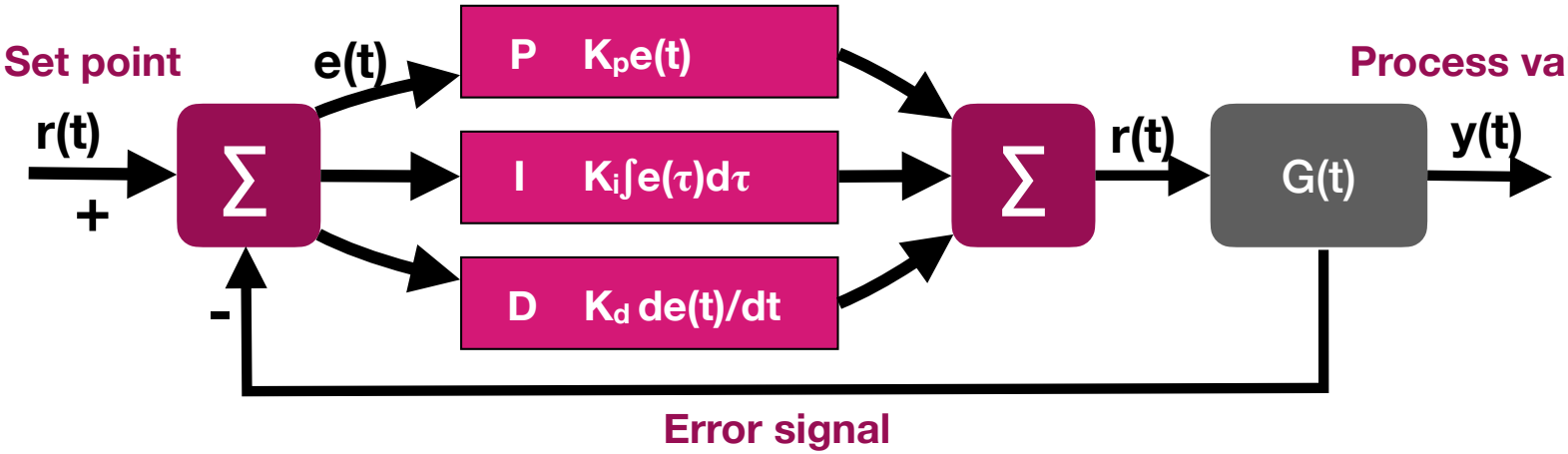
Main Characteristic

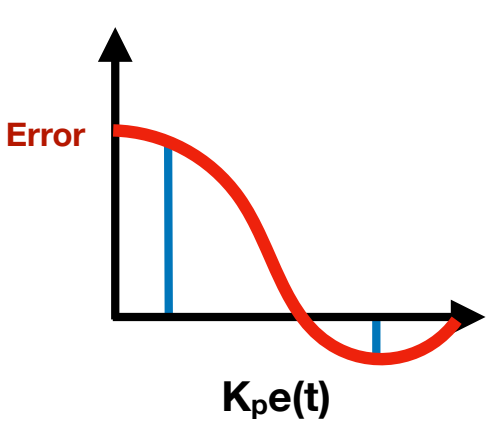
Control System / Feedback Loops

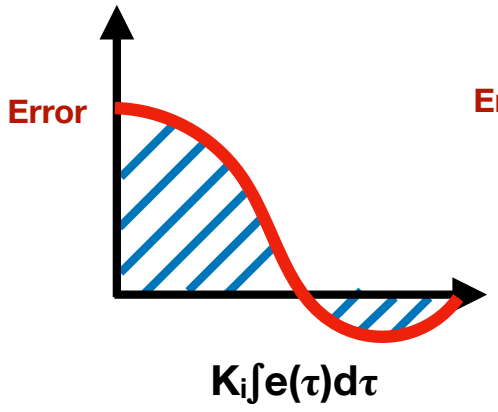
Integration of **sensing** and **computation**

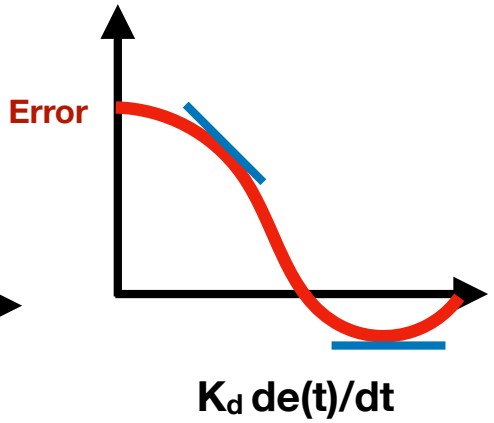


A Proportional–Integral–Derivative (PID) Controller

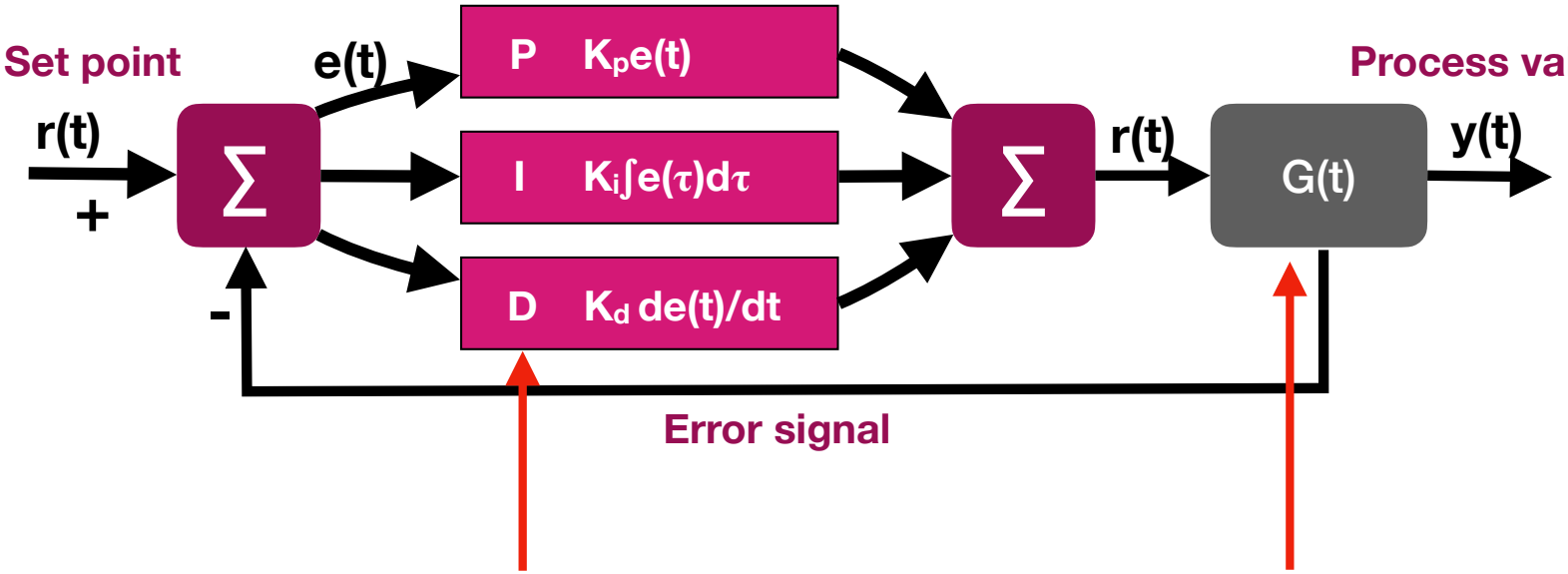






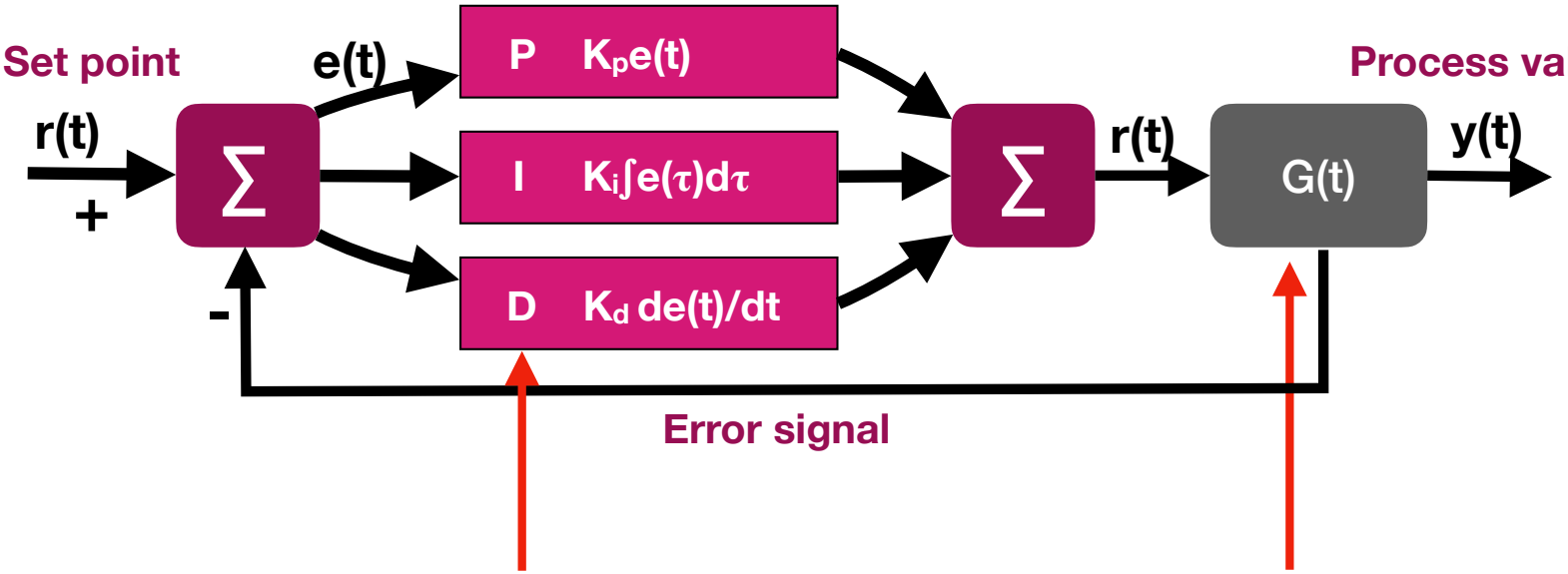


A Proportional–Integral–Derivative (PID) Controller



One specific operating condition: ideal for industrial robotics
What about supporting ***several*** conditions?

A Proportional–Integral–Derivative (PID) Controller



One specific operating condition: ideal for industrial robotics

What about supporting ***several*** conditions?

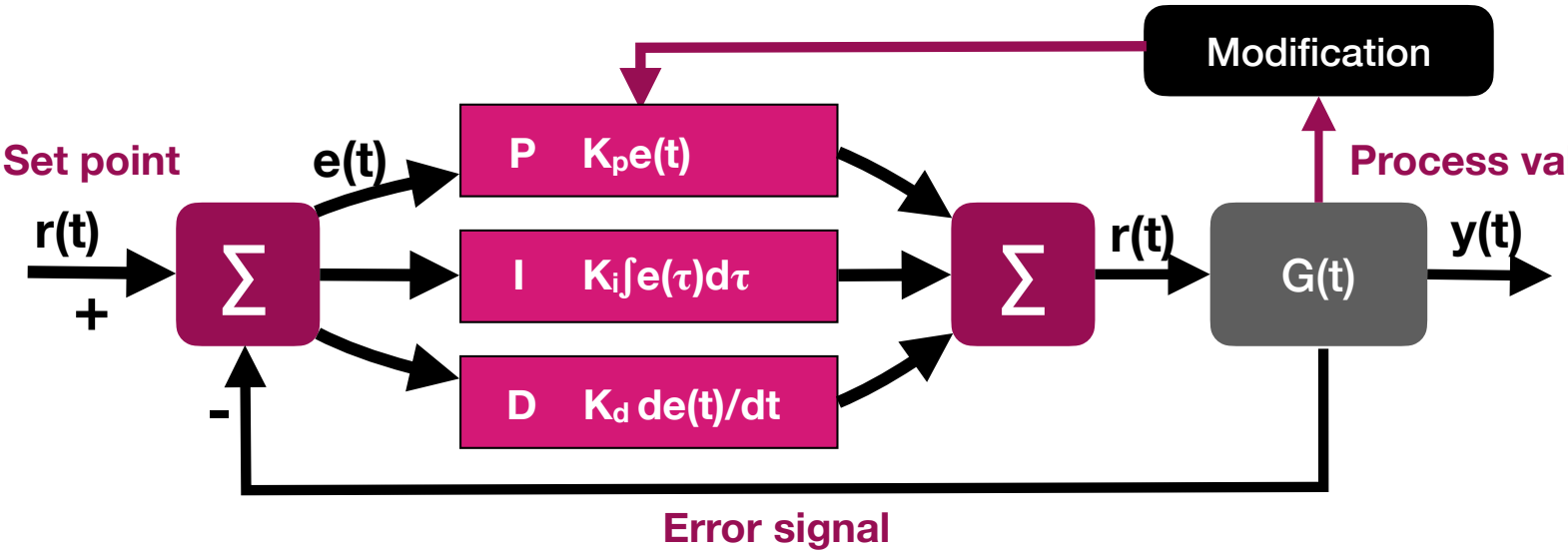
Design nonlinear controller
Hard, sometimes impossible

Build linear robust controller
Drop in performance

Limit operation

PID
Robust / Adaptive control

A Proportional–Integral–Derivative (PID) Controller



Robust (e.g. LPV)

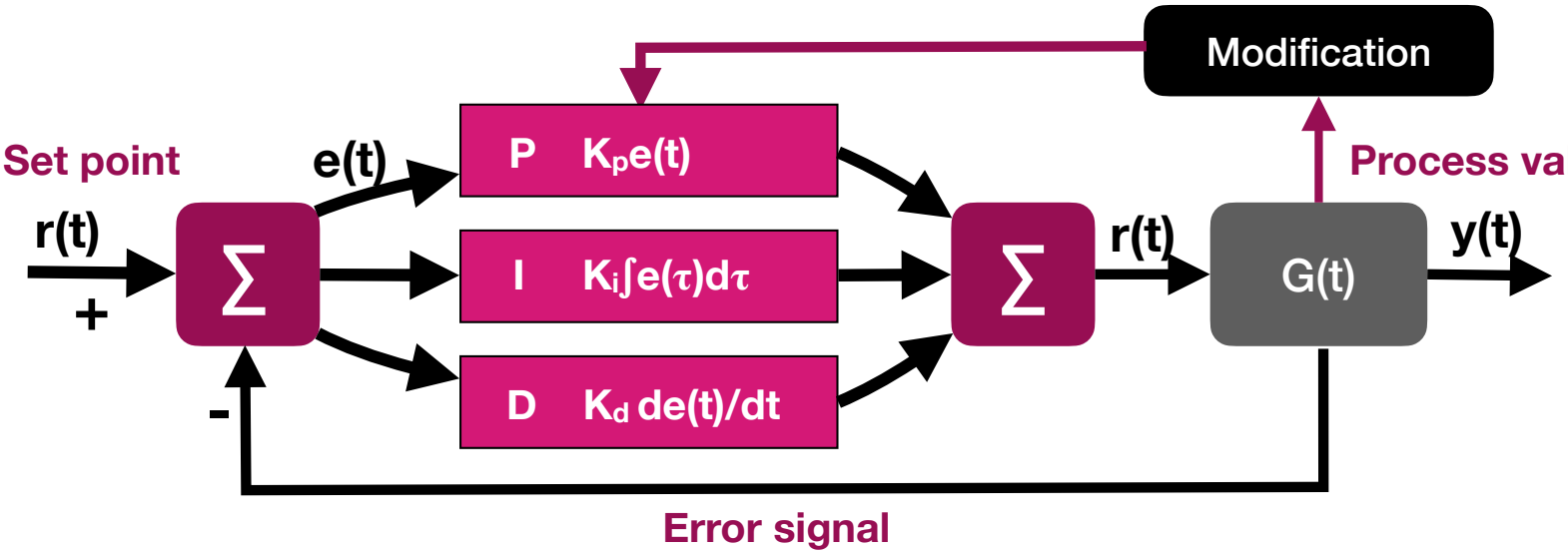
Priori knowledge and not efficient

Highly dimensional parameter space

Memory-performance tradeoff

Smooth transitioning

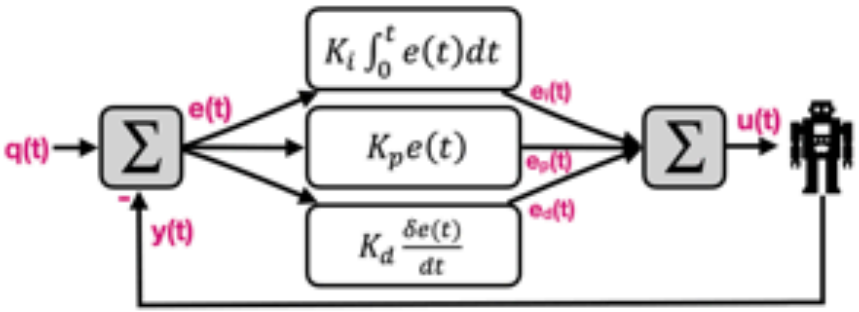
A Proportional–Integral–Derivative (PID) Controller

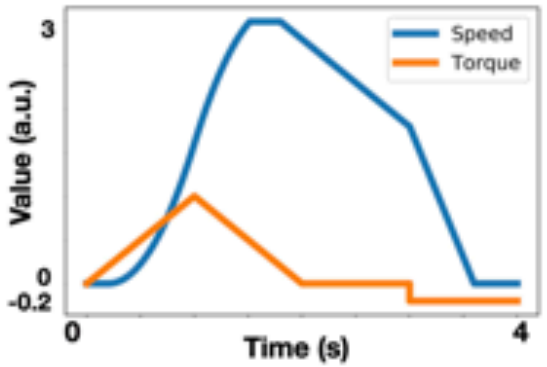


Robust (e.g. LPV) → **Adaptive PID control**

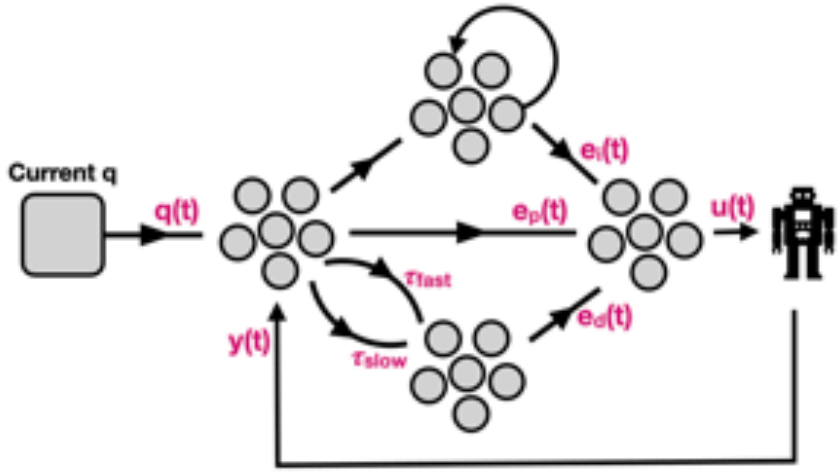
Priori knowledge and not efficient

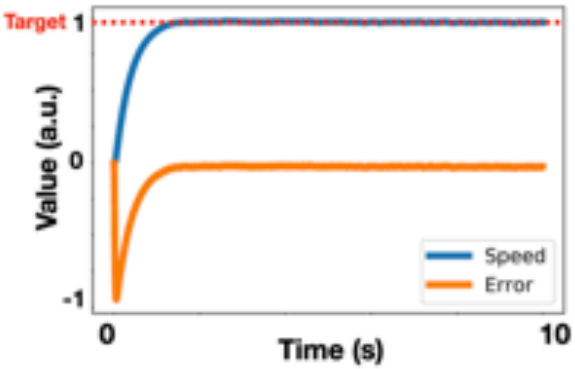
- Highly dimensional parameter space
- Memory-performance tradeoff
- Smooth transitioning

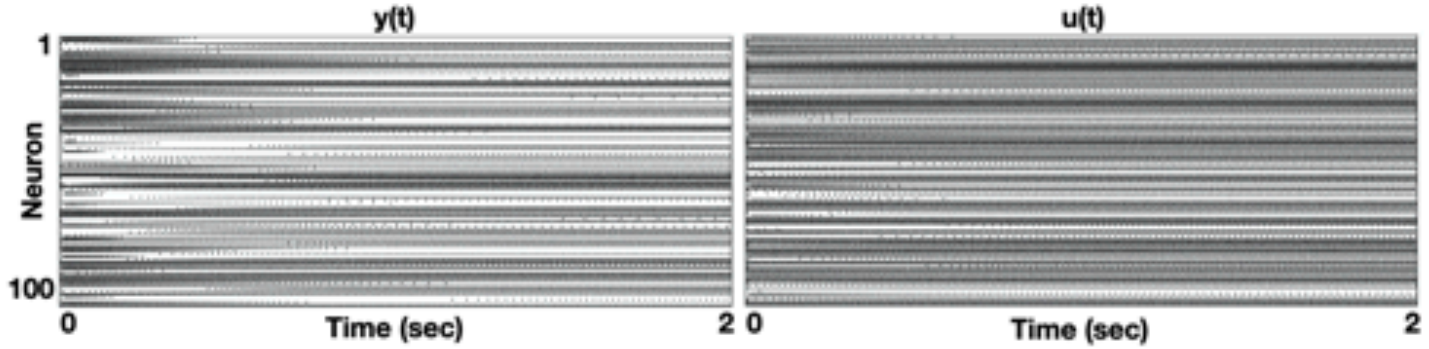




Albert Shalumov

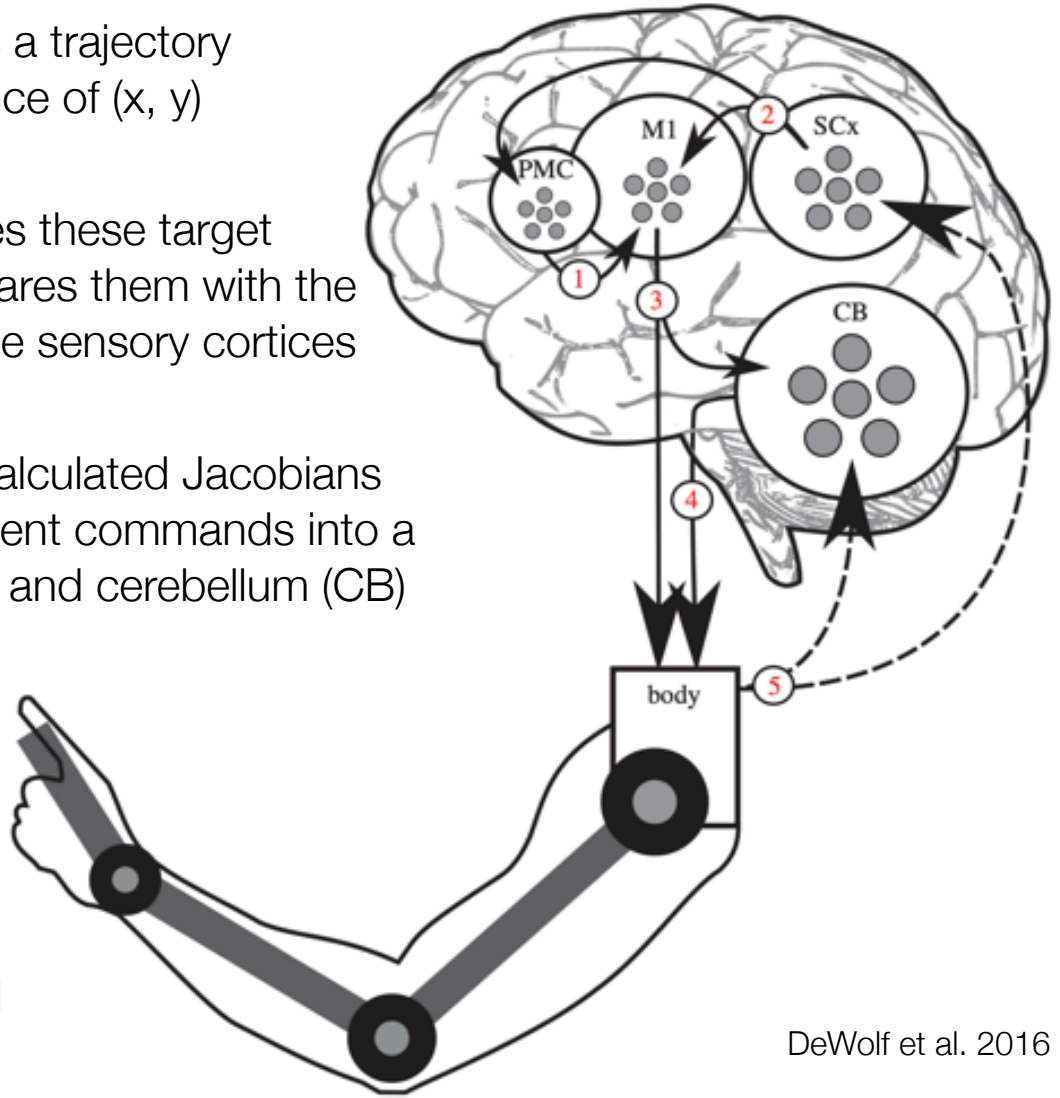






Adaptive Control

- The premotor cortex (PMC) generates a trajectory for the system to follow with a sequence of (x, y) coordinates.
- The primary motor cortex (M1) receives these target positions (1) from the PMC and compares them with the current system state, received from the sensory cortices (SCx), through (2).
- M1 combines this signal with locally calculated Jacobians to transform the desired hand movement commands into a low-level signal that is sent to the arm and cerebellum (CB) along (3).
- The CB projects an adaptive signal to the body along (4) that compensates for velocity and movement errors. **Visual and proprioceptive feedback** projects from the body along (5) to the CB and SCx.



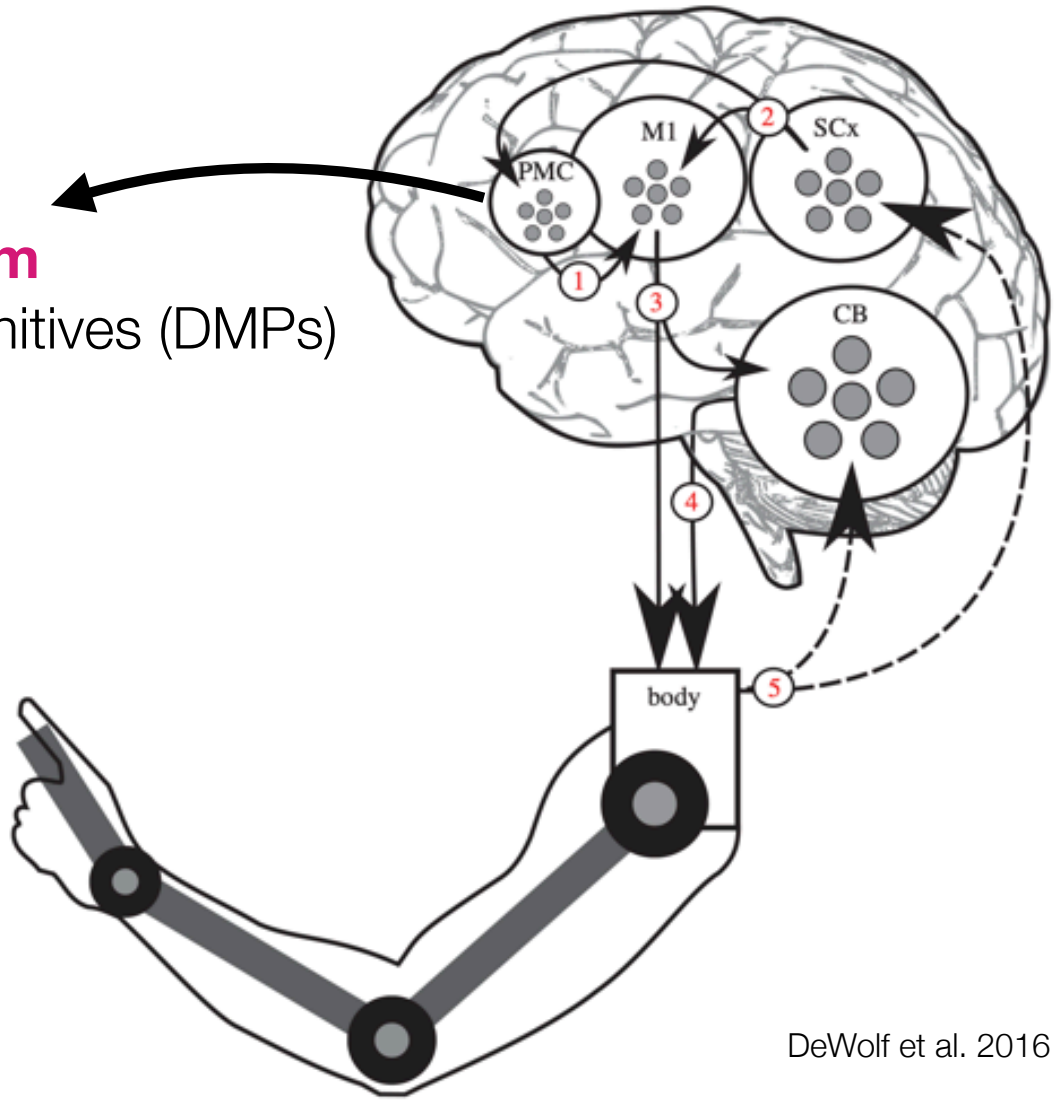
DeWolf et al. 2016

Adaptive Control

PMC:

Trajectory generation system

Using Dynamic Movement Primitives (DMPs)



DeWolf et al. 2016

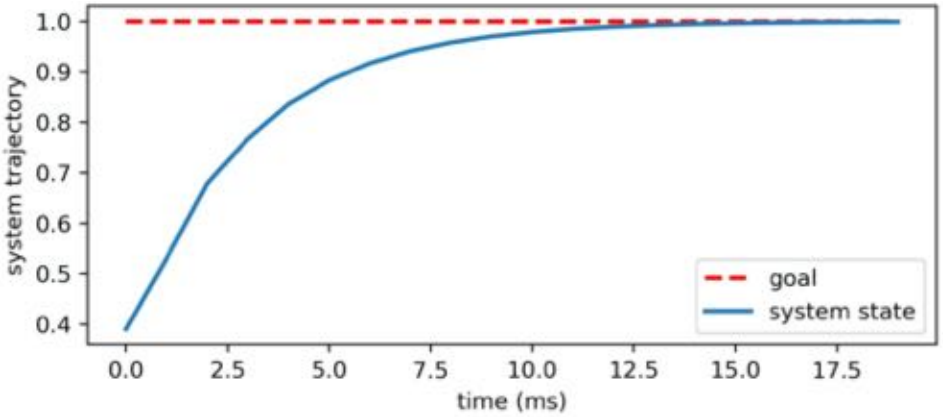
Dynamic movement primitives (DMPs)

Discret DMP

$$\ddot{y} = \alpha_y(\beta_y(g - y) - \dot{y})$$

Gain terms — Goal — System state

PD controller



Dynamic movement primitives (DMPs)

Discret DMP

$$\ddot{y} = \alpha_y(\beta_y(g - y) - \dot{y}) + f$$

Gain terms: α_y, β_y — Goal: g — System state: y, $\dot{y}$ — Modifying Force: f

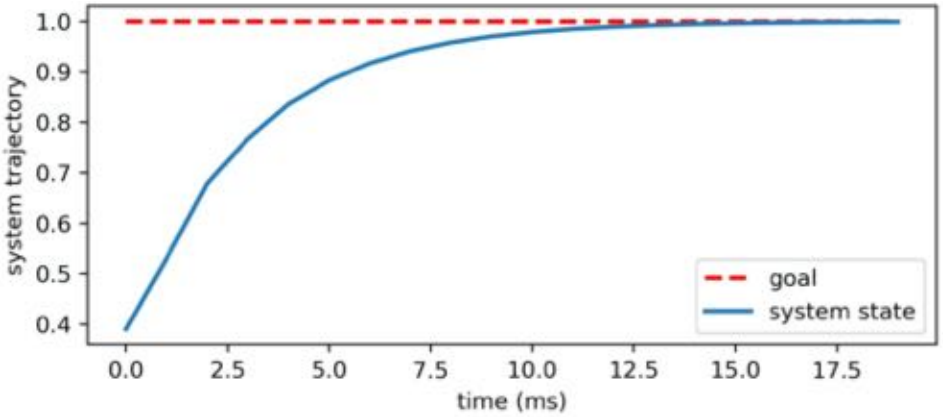
PD controller | **Non-linear term**

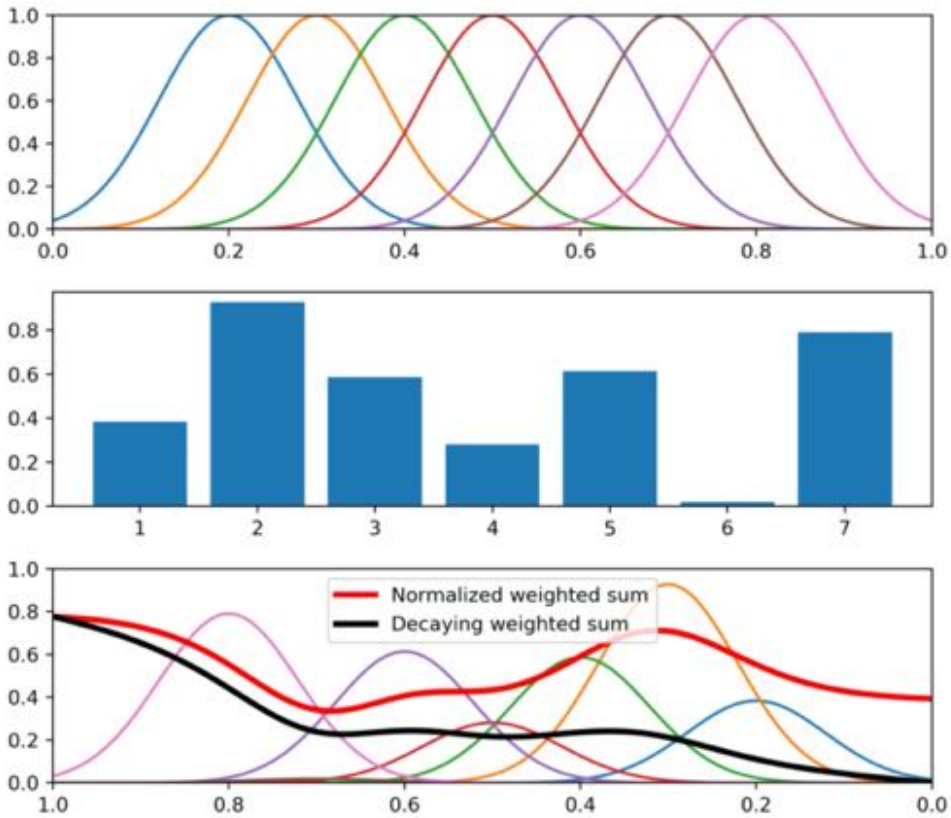
$$f(x, g) = \frac{\sum_{i=1}^{N} \Psi_i \cdot w_i}{\sum_{i=1}^{N} \Psi_i} \cdot x(g - y)$$

Canonical system
Decaying from 1 to 0

$$\Psi_i = exp(-h_i \cdot (x - c_i)^2)$$

Gaussian cantered at c_i with h_i variance





Dynamic movement primitives (DMPs)

Discret DMP

$$\sum_t \Psi_i(t)(f_d(t) - w_i(x(t)(g-y)))^2 \longleftarrow f(x,g) = \frac{\sum_{i=1}^N \Psi_i \cdot w_i}{\sum_{i=1}^N \Psi_i} \cdot x(g-y)$$

Canonical system
Decaying from 1 to 0

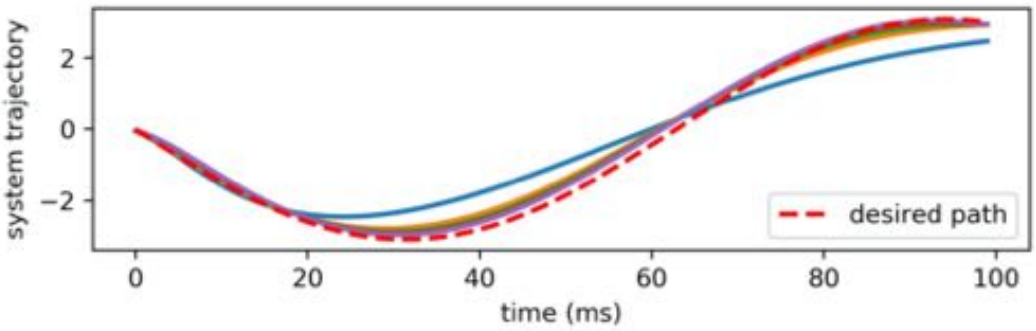
$$f_d = \ddot{y}_d - \alpha_y(\beta_y(g-y) - \dot{y})$$

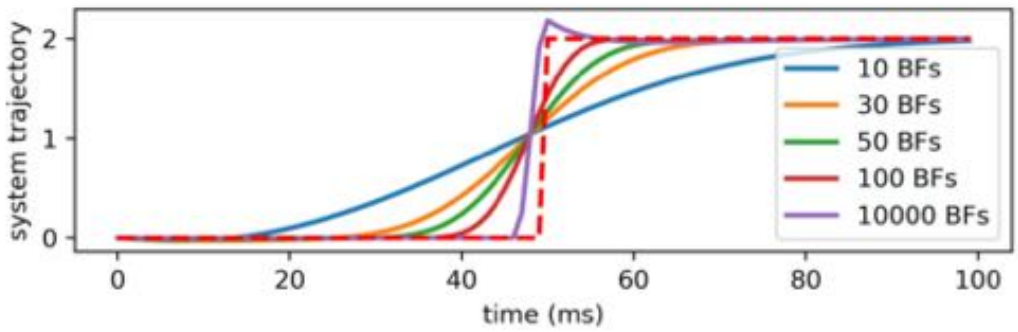
Desired force to trace desired trajectory y_d

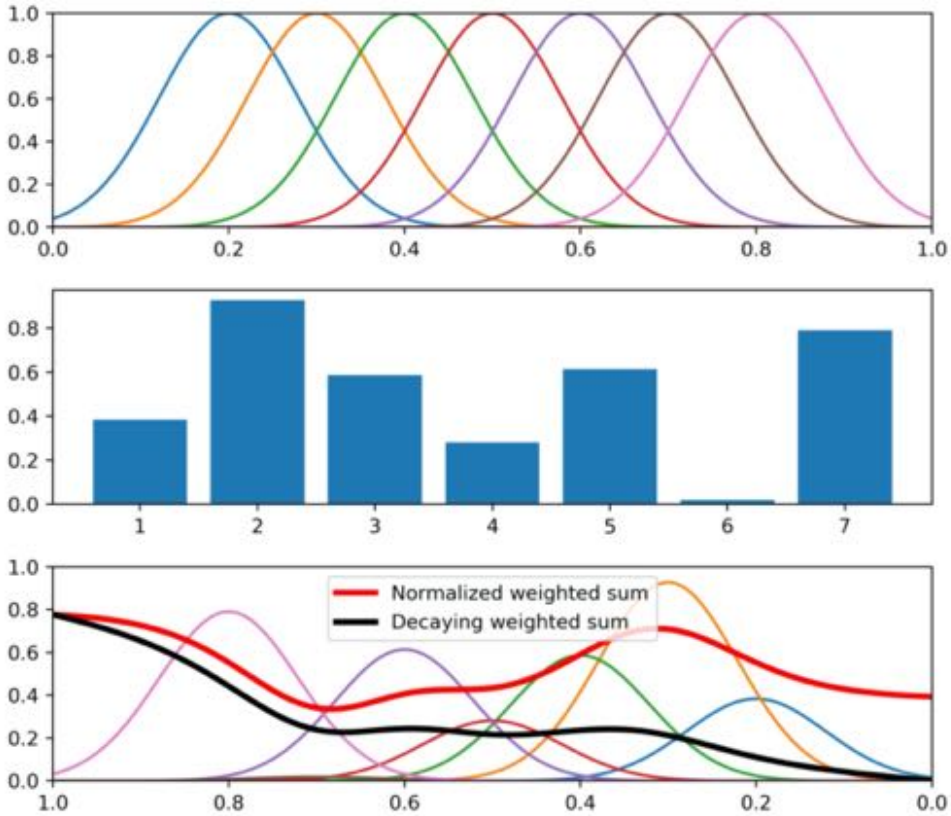
$$\Psi_i = exp(-h_i \cdot (x - c_i)^2$$

Gaussian cantered at c_i with h_i variance

Optimizing for a desired path



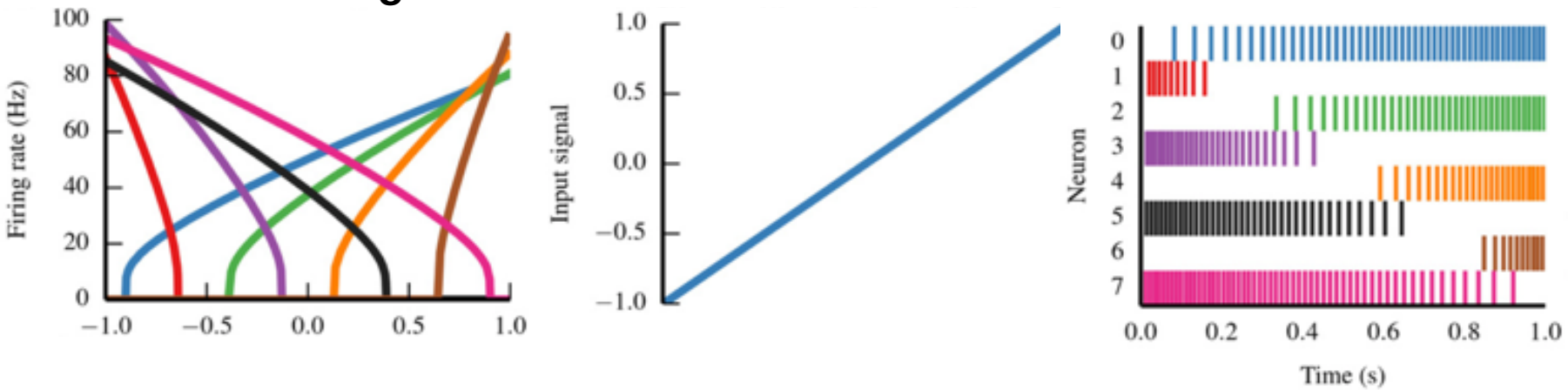




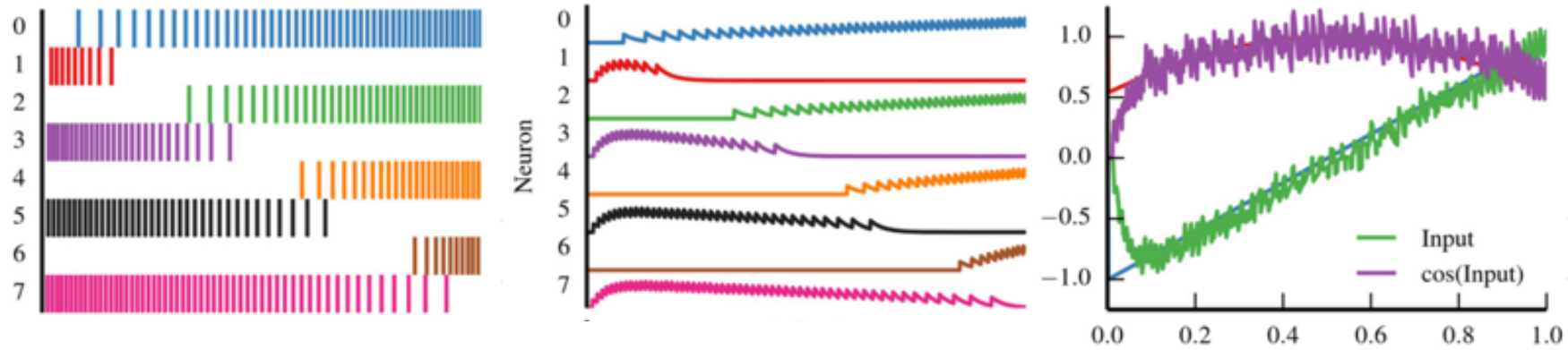
Dynamic movement primitives (DMPs)

In the Neural Engineering Framework

Encoding →



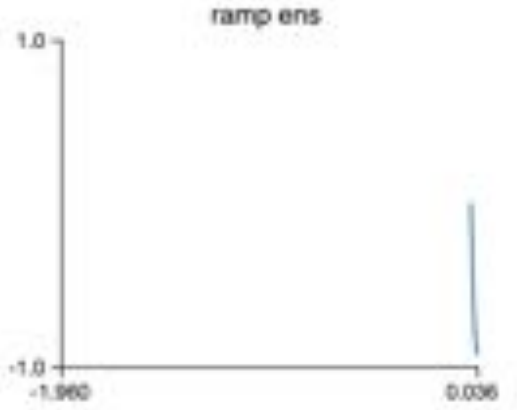
Decoding →

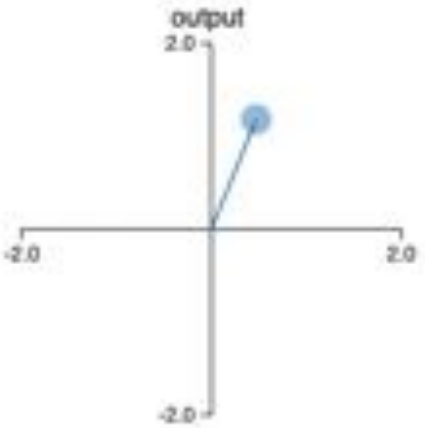


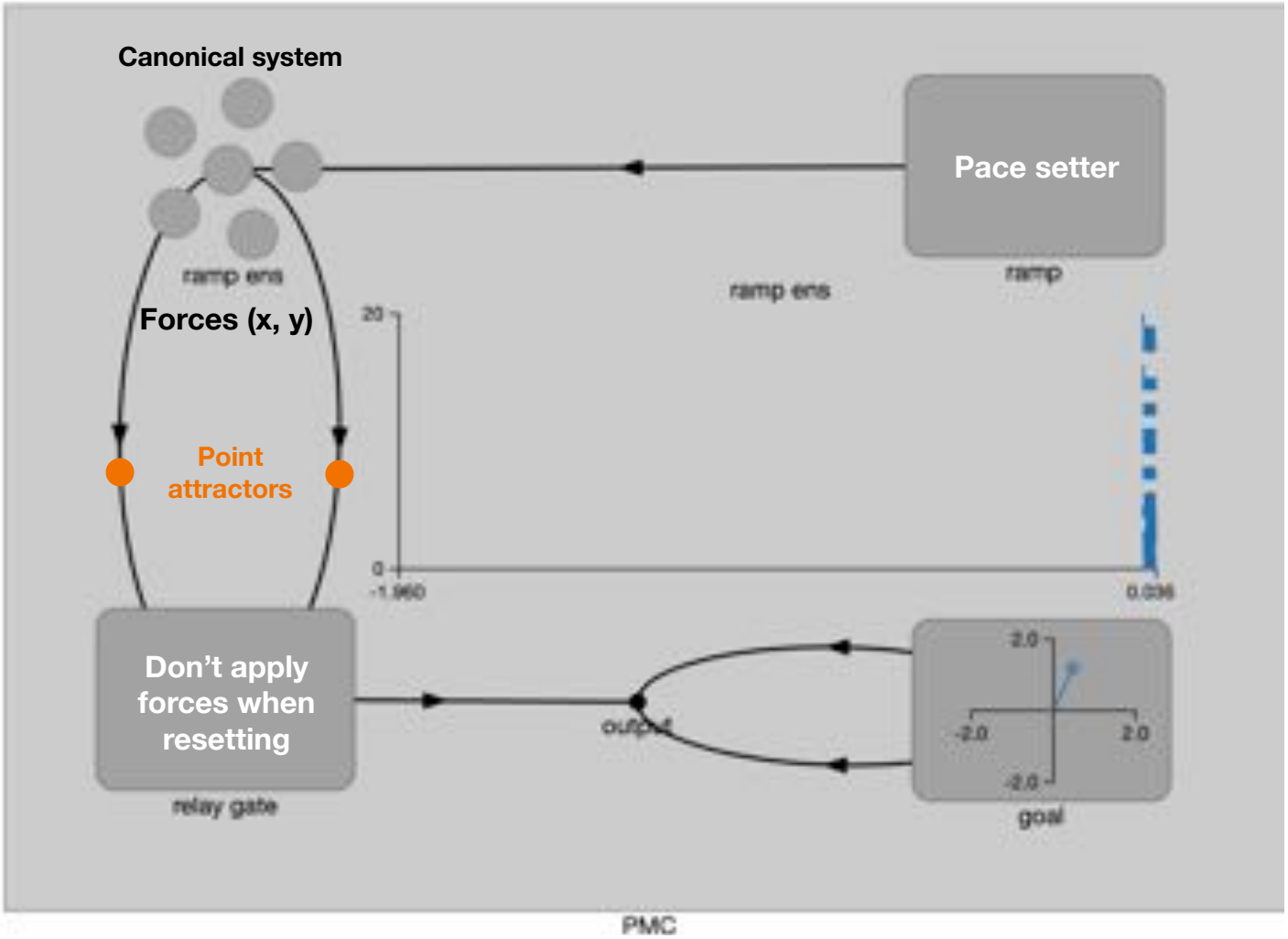
NOEL accenture intel בית חולים אלי״ן (ע״ר) מרכז לשיקום ילדים ונוער אלין

Dynamic movement primitives (DMPs)

In Spiking Neurons







Operation Space Control

$$u = -J \cdot f_x$$

Control in joint space — **Force in operational space**

$$f_x = M_x \cdot \ddot{x}$$ **Acceleration to force**

Zero velocity @ target

$$\ddot{x} = k_p(x - x_t) + k_v(\dot{x} - \cancel{\dot{x}_t})$$ **PD controller for acceleration**

Already in joint space

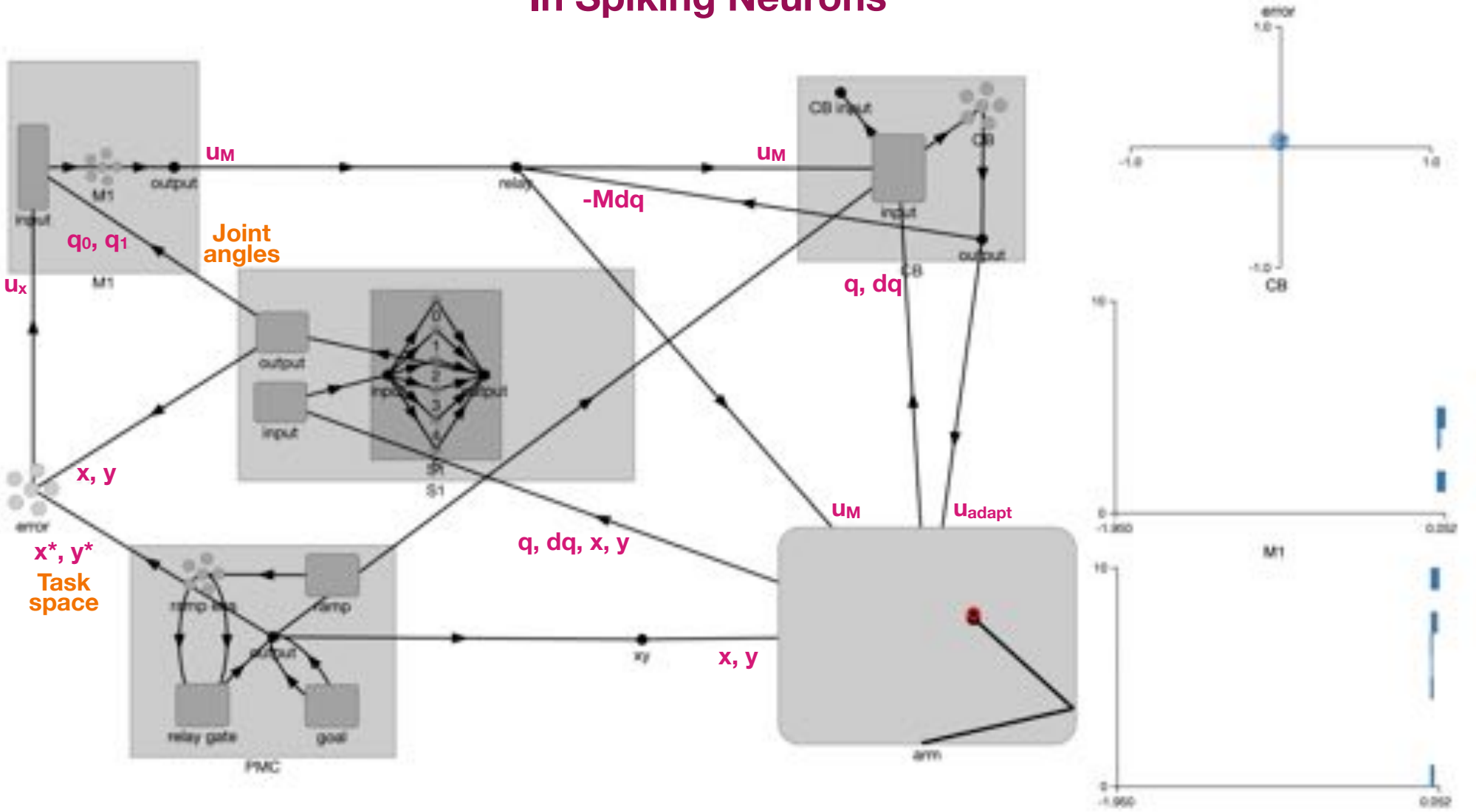
$$u = -J \cdot M_x \cdot K_p(x - x_t) - M \cdot K_v \cdot \dot{x} - u_{bias}$$

External perturbation

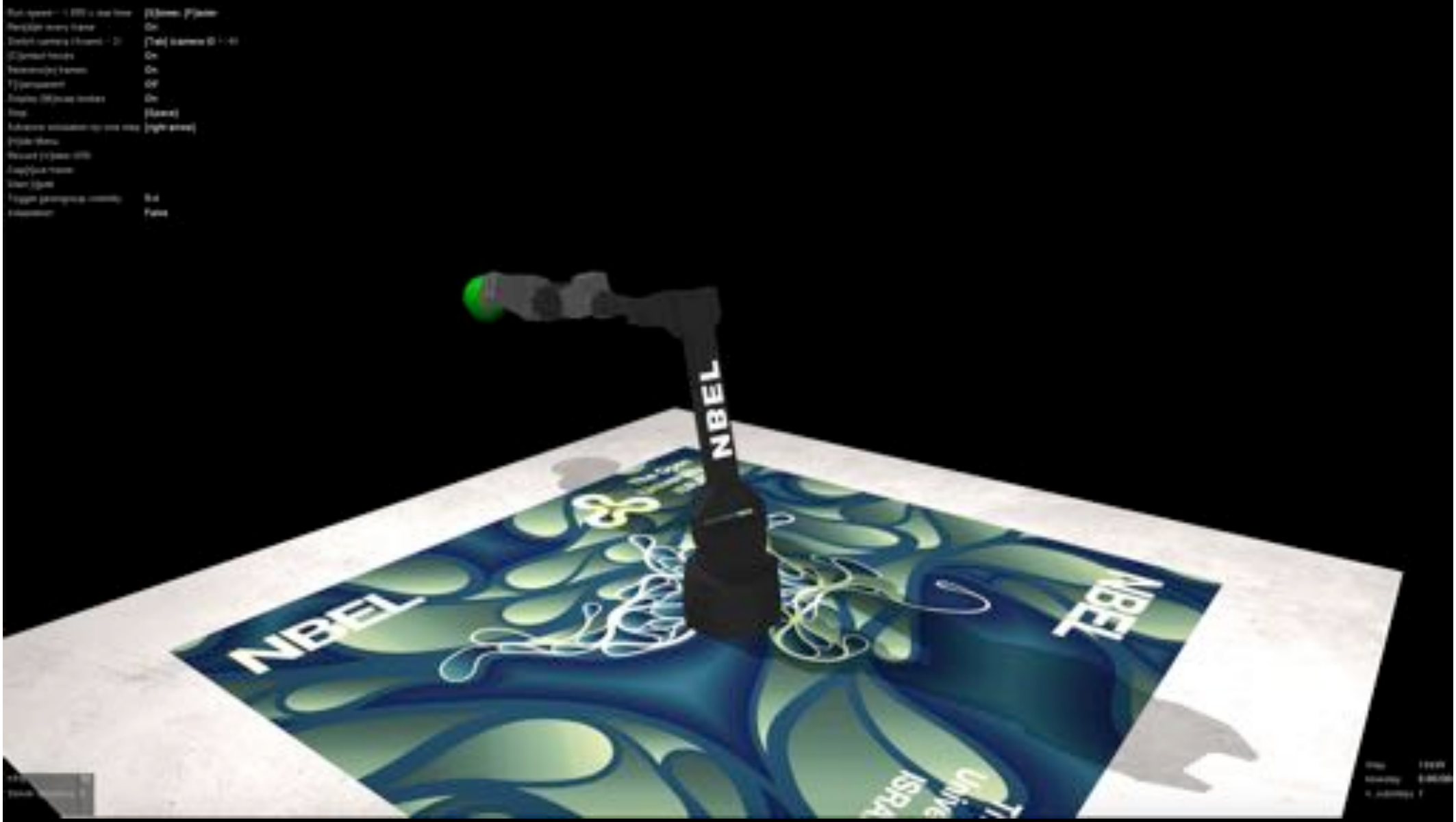
NOEL

Dynamic movement primitives (DMPs)

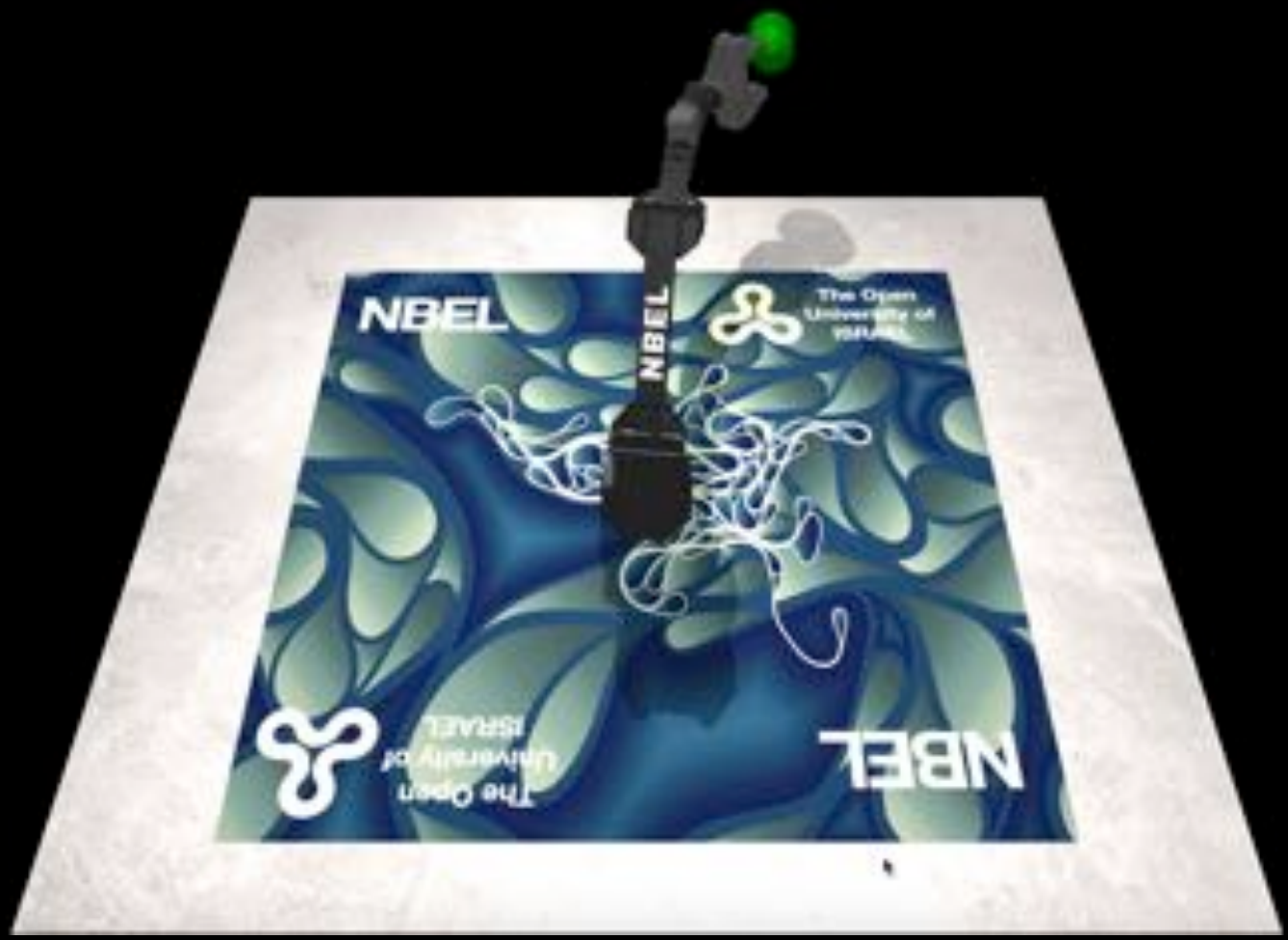
In Spiking Neurons



In Simulation
No External Force

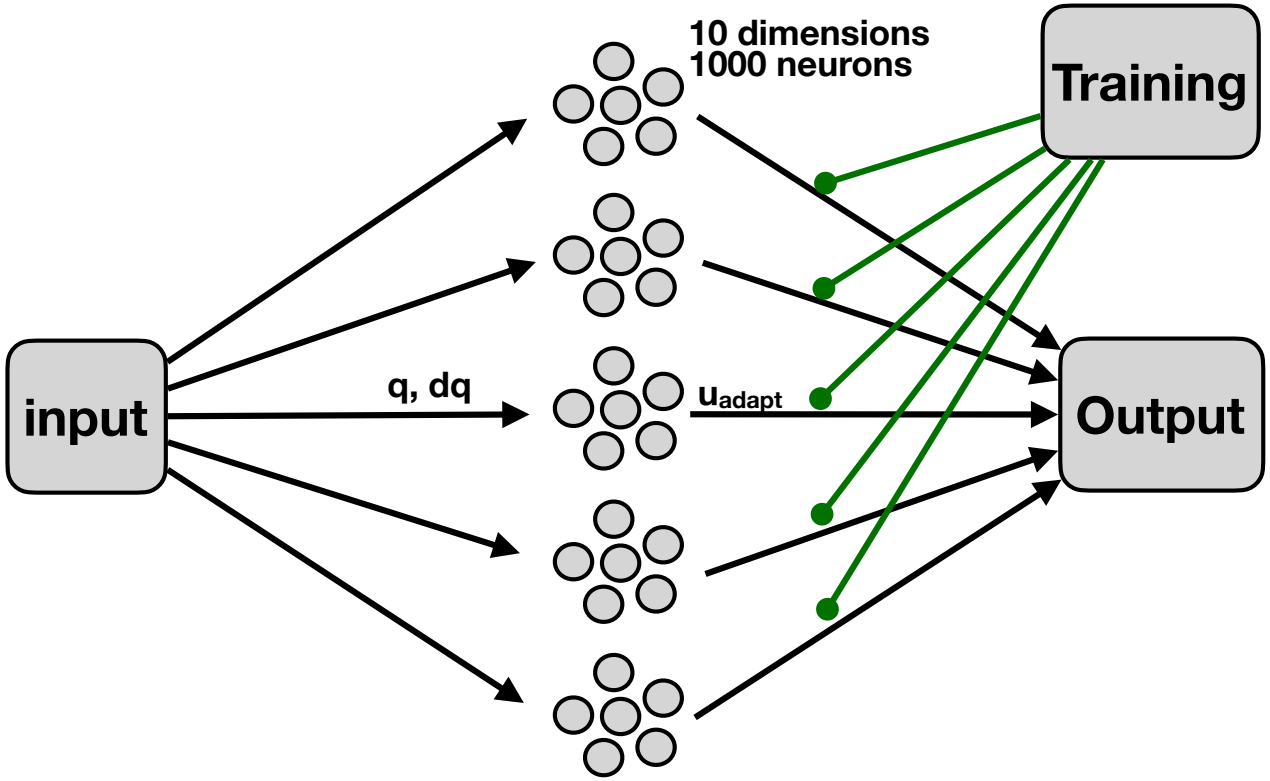


In Simulation
With External Force



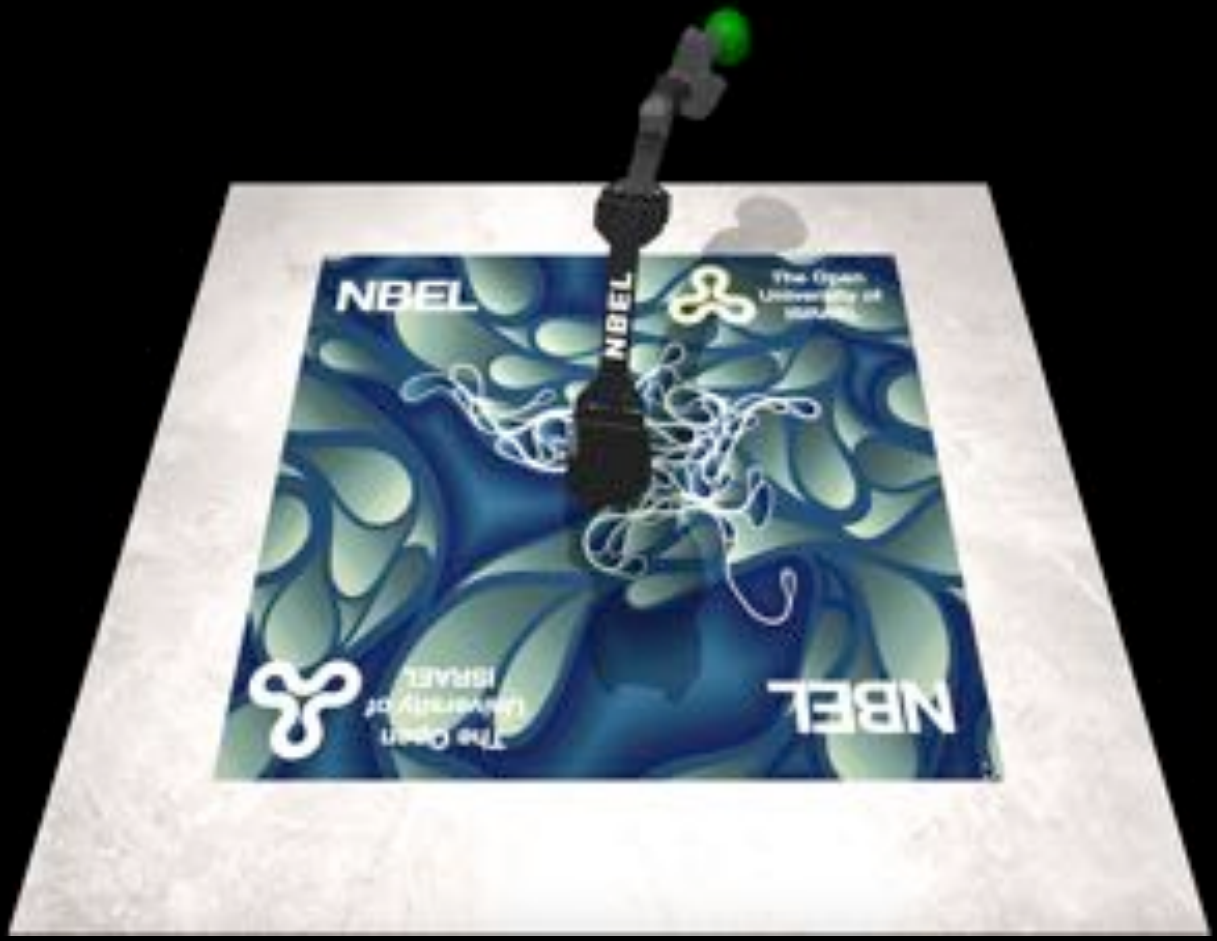
On-line Learning

Adaptive control is held neuromorphically

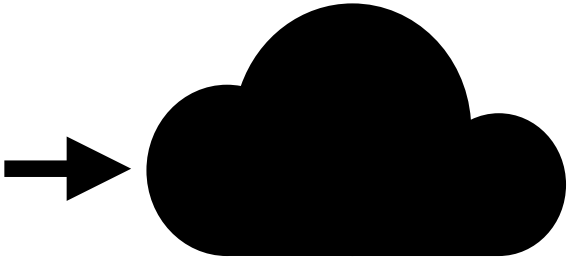


Representing values in 10 dimensions is not trivial. Tuning curves has to be (very) carefully defined. Values should be scaled experimentally.

In Simulation
With External Force and Adaptation



Controllers



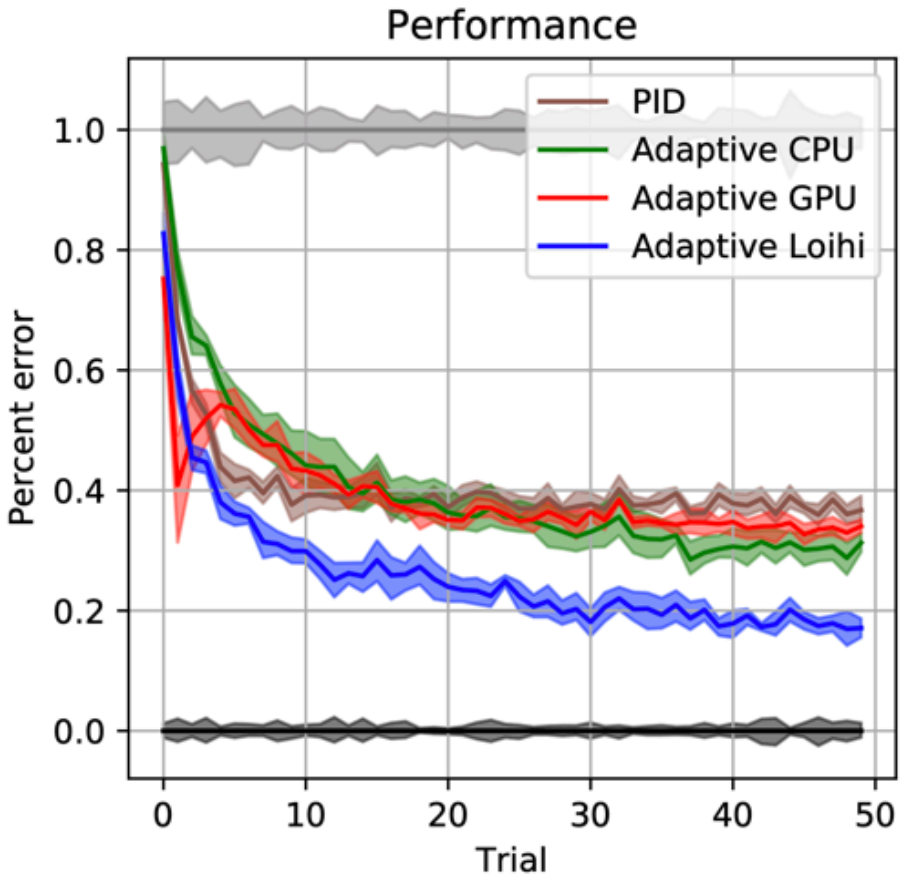
NVIDIA Xavier
Running Simulated Spiking Neural Network

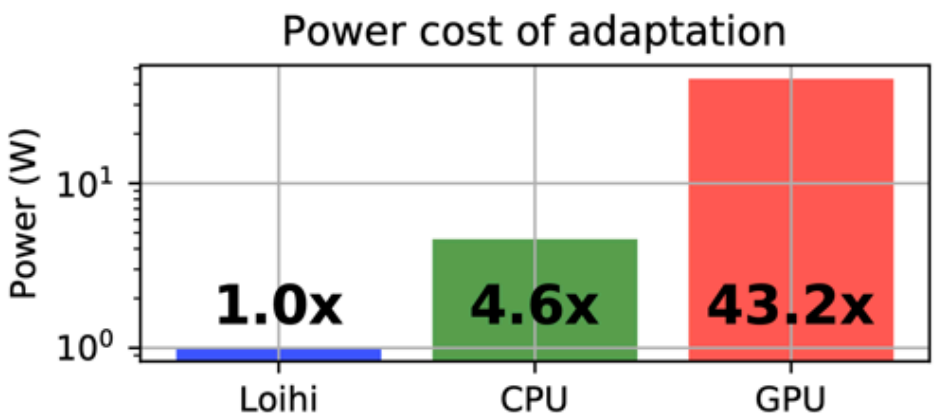
INRC Cloud
Running Spiking Neural Network

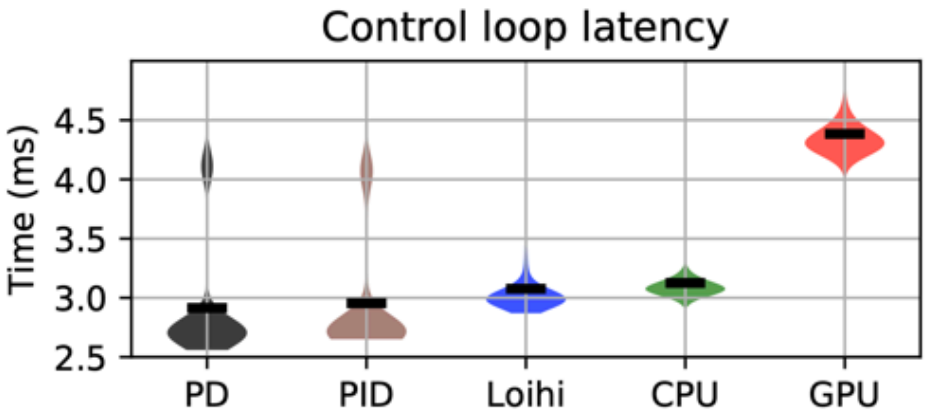
Nahuku-32
Running Embedded Spiking Neural Network

NOEL

Adaptive Control







DeWolf et al. 2020

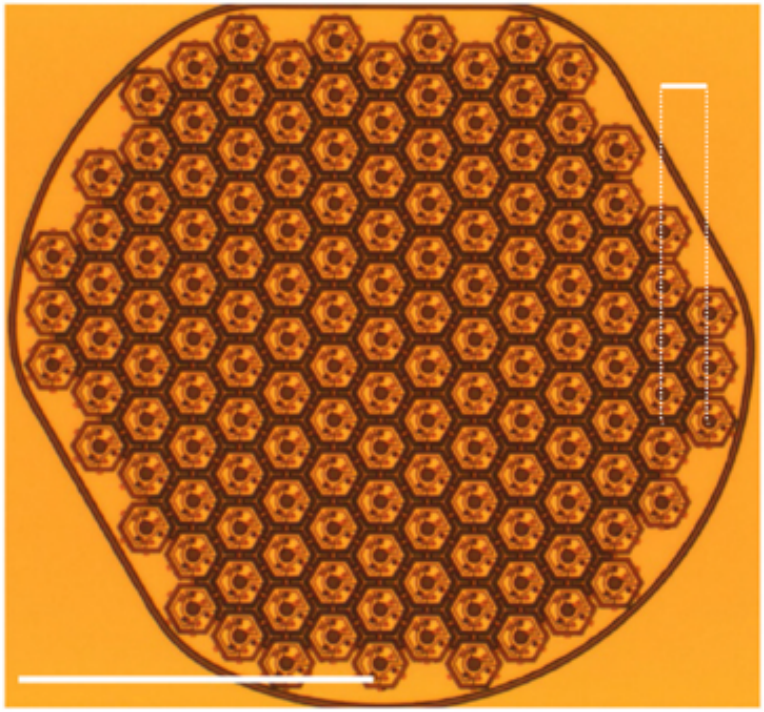
Is it all about computing power?

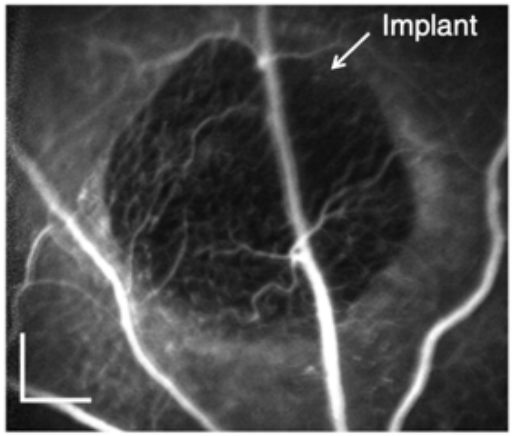
Restoring the senses

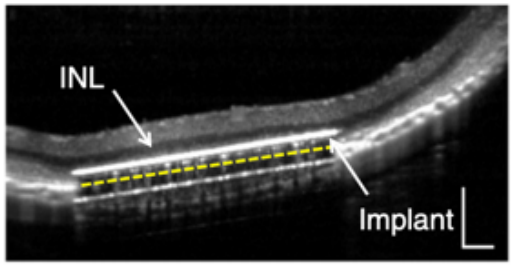
cochlear implant

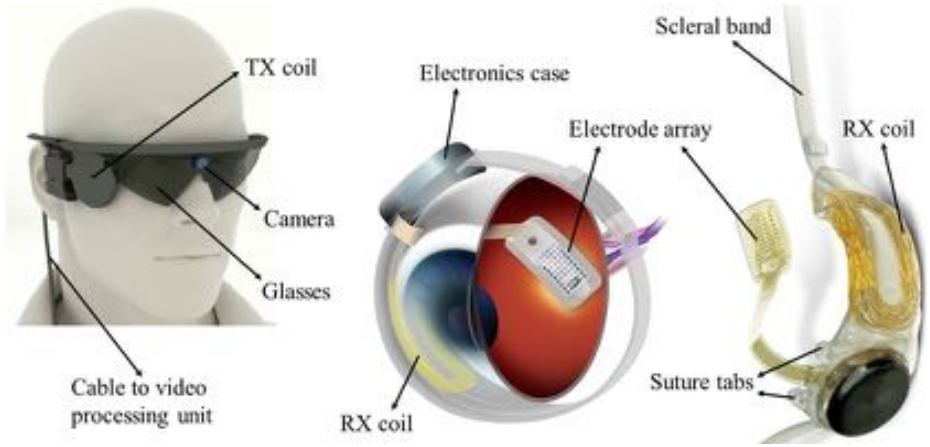
cleft palate

Restoring the senses

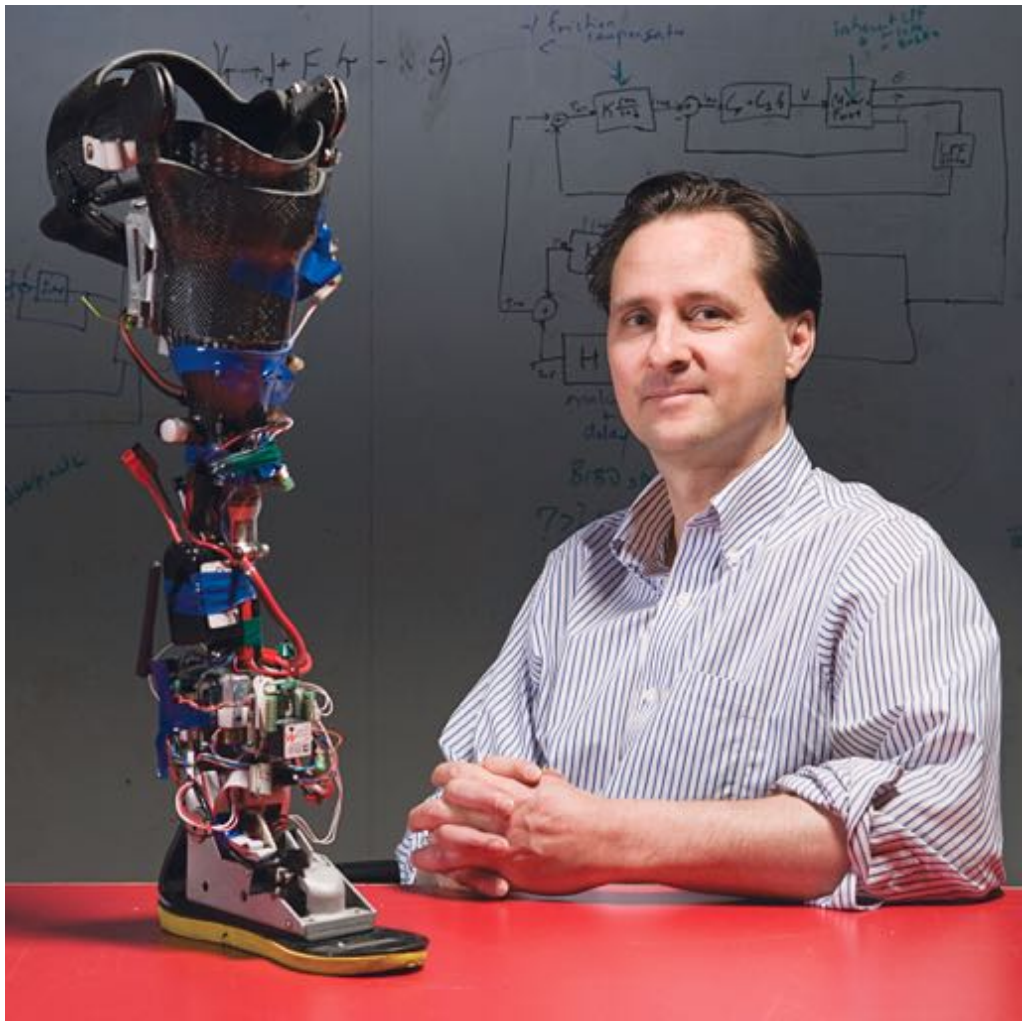


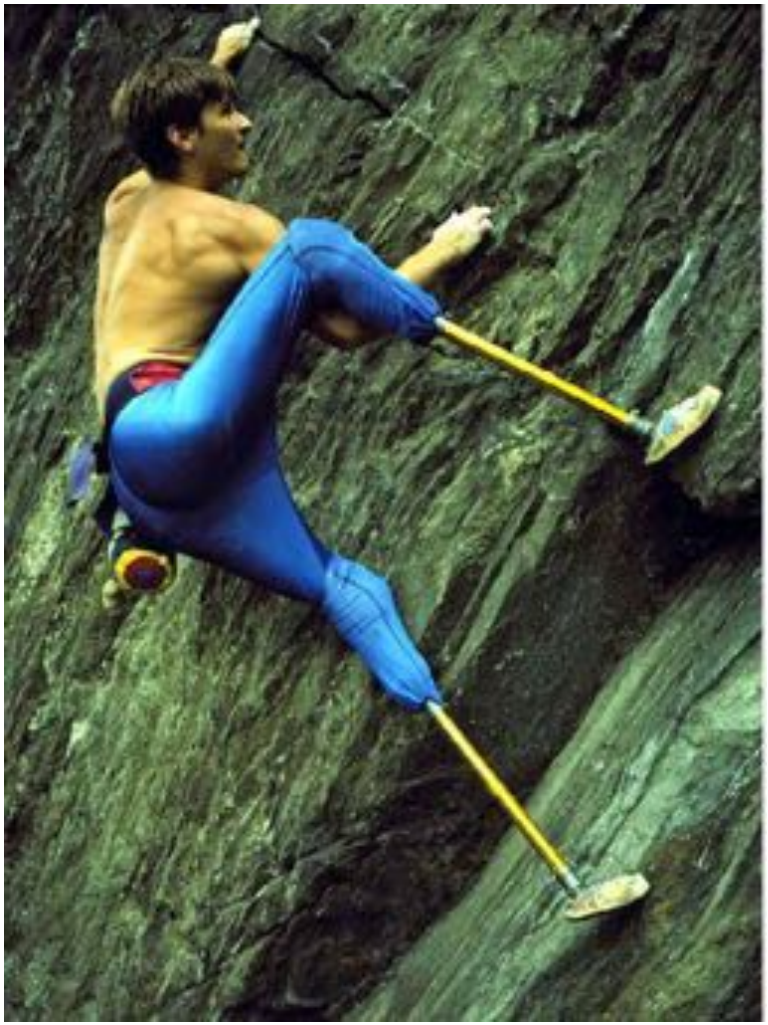






Restoring the senses





NOEL

NEL
NEURO-BIOMORPHIC ENGINEERING LAB

Partners

XILINX
ALL PROGRAMMABLE™

NVIDIA

intel

accenture

האוניברסיטה הפתוחה
THE OPEN UNIVERSITY OF ISRAEL
الجامعة المفتوحة

אלין
בית חולים אלי"ן (ע"ר)
מרכז לשיקום ילדים ונוער

Collaborators:

Prof. Yaakov Nahmias, Hebrew University
Prof. Tamar Weiss, University of Haifa
Prof. Efraim Jaul, Herzog Medical Center
Dr. Oded Meiron, Herzog Medical Center
Dr. Michal Rivlin, Weizmann Institute of Science
Arie Melamed, Alyn Hospital