

**SYDE 556/750**

**Simulating Neurobiological Systems**  
**Lecture 13: Conclusion**

Chris Eliasmith

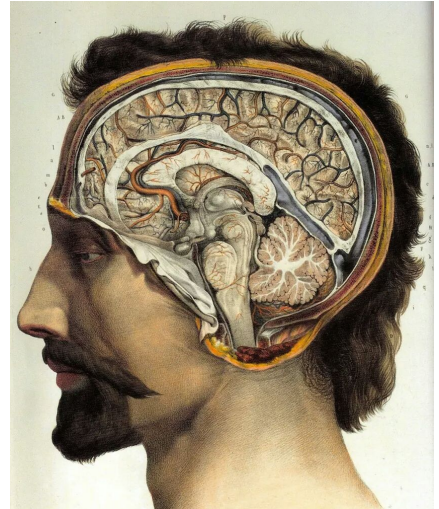
Dec 4, 2023

- ▶ Slide design: Andreas Stöckel
- ▶ Content: Terry Stewart, Andreas Stöckel, Chris Eliasmith



UNIVERSITY OF  
**WATERLOO**

FACULTY OF  
ENGINEERING



# Goal of This Course

## Building Large-Scale Brain Models

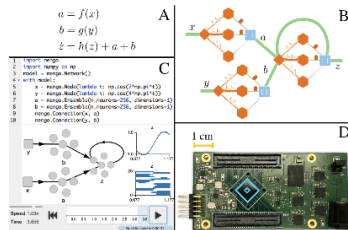
Why?



Understand how Brains  
Work



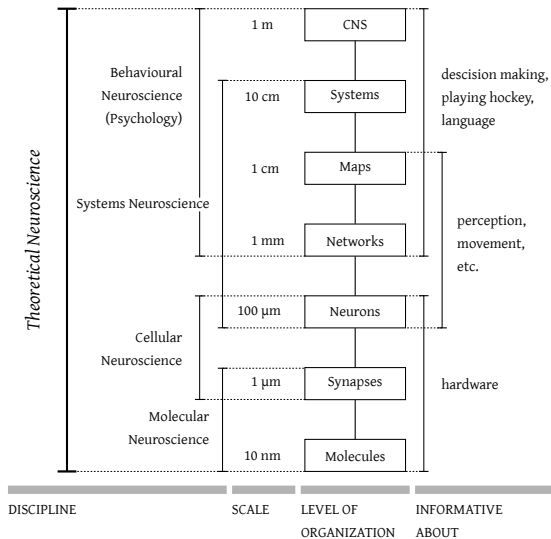
Build Better AI Systems



Program Neuromorphic  
Hardware

**Image Sources.** Left: "A chimpanzee brain at the Science Museum London", from Wikimedia. Centre: "Robot at a campus faire in São Paulo" from Wikimedia. Right: The Braindrop Neuromorphic hardware system, from "Braindrop: A Mixed-Signal Neuromorphic Architecture With a Dynamical Systems-Based Programming Model", Neckar et al., 2019.

# Theoretical Neuroscience



# The Brain – Some Statistics

- ▶ **Weight:**  
2 kg (2% of the body weight)
- ▶ **Power consumption:**  
20 W (25% of the body's total power consumption)
- ▶ **Surface area:**  
1500 cm<sup>2</sup> to 2000 cm<sup>2</sup> (roughly four A4/letter pages of paper)
- ▶ **Number of neurons:**  
100 billion ( $10^{11}$ , 150 000 mm<sup>-2</sup>)
- ▶ **Number of synapses:**  
100 trillion ( $10^{14}$ , about 1000 per neuron)

# Neuromorphic Hardware

**Goal:** Brain-inspired hardware; lower power consumption; stream processing

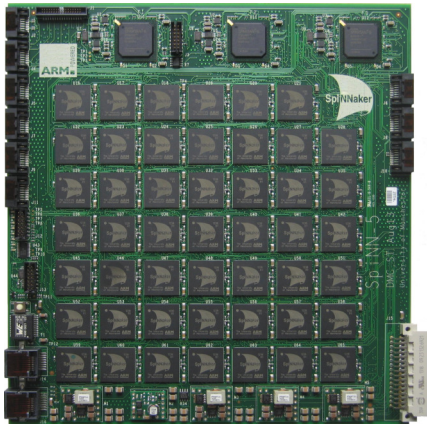
## Digital

- ▶ Specialised digital hardware for simulating spiking neural networks
- ▶ Trivial weight-spike multiplication
- ▶ Often asynchronous (no central clock)
- ⊕ Deterministic
- Higher power consumption than analogue

## Analogue/Mixed Signal

- ▶ Neuron models in analogue hardware (capacitors, resistors, ...)
- ▶ Digital interconnect and programming (weights, neuron parameters)
- ⊖ Not deterministic
- ⊖ Hard to program
- ⊕ Very low power consumption

# Neuromorphic Hardware – SpiNNaker



- ▶ Manchester/Dresden collaboration; HBP
- ▶ Fully digital
- ▶ 18 ARM968 processors @ 180 MHz per chip
- ▶ 1000 current-based LIF neurons per core
- ▶ Toroidal, asynchronous interconnect mesh
- ▶ Up to  $\approx 10^9$  neurons in one system
- + Easy to program, (outdated) Nengo interface
- + Public access via HBP
- Not very power efficient (version from 2013)
- High setup times

# Neuromorphic Hardware – Loihi

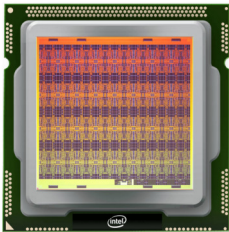
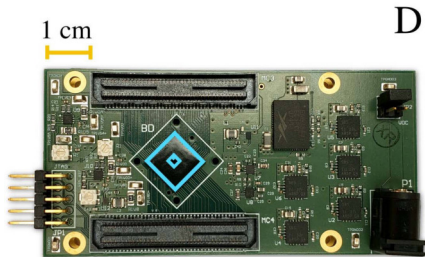


Image Sources. Intel Marketing Material

- ▶ Developed by Intel
- ▶ Digital, fully asynchronous architecture
- ▶ Circuits accelerating individual spiking neurons
- + Extremely low power consumption
- + Nengo Interface
- Proprietary/no low level programming without signing an NDA

# Neuromorphic Hardware – BrainDrop

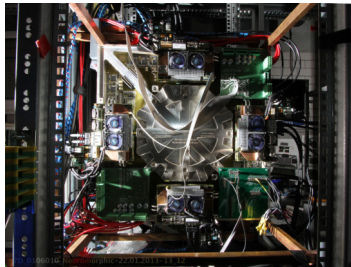
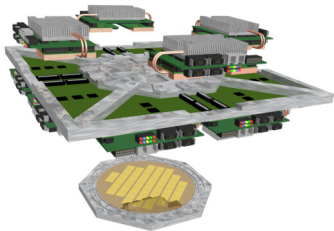


**Image Sources.** Braindrop: A Mixed-Signal Neuromorphic Architecture With a Dynamical Systems-Based Programming Model, Neekar et al. 2019

- ▶ Research project at Stanford
- ▶ Mixed signal, analogue neurons, synapse arrays
- ▶ Exploits process noise for diverse neural tuning
- ▶ Optimized for NEF networks
- ⊕ Extremely low power consumption
- ⊕ Nengo Interface
- Availability/Documentation?
- ⊖ Small networks only



# Neuromorphic Hardware – BrainScaleS



- ▶ Research Project in Heidelberg; HBP
- ▶ Mixed signal; above realtime
- ▶ Wafer-scale system;  $384 \times 256$  neurons

- ⊕ Low power consumption
- ⊕ Public access via HBP
- ⊖ Complex; relatively low precision

# Review: What Did We Learn? (I)

## ► The Neural Engineering Framework

- Theory for theoretical neuroscience ( **bridging laws** )
- *Principle 1:*  
Populations of neurons represent values  $x$
- *Principle 2:*  
Connections compute functions  $f$
- *Principle 3:*  
Values are states in a dynamical system
- Model of *how* the brain computes. Is it wrong? Of course! But hopefully useful!

# Review: What Did We Learn? (II)

## ► Cognitive Architectures

- *Jackendoff's Challenges:*  
How to explain language in neural networks?
- *Vector Symbolic Algebras:*  
Compressing symbolic information into vectors,  
circular convolution, word embeddings
- *Semantic Pointer Architecture:*  
Combination of four ideas – *What* the brain computes.
  - VSAs
  - NEF
  - Deep Semantics: compression, decompression
  - Architecture: Basal Ganglia/Thalamus/Cortex Loop

# Review: What Did We Learn? (III)

## ► **Methods & Techniques**

- *The Delay Network:*  
Efficiently compress past history into a vector; optimal recurrent update rule
- *Spatial Semantic Pointers:*  
Bio-inspired representation of continuous spaces; e.g., maps, probabilities
- *Machine Learning:*  
Unsupervised, supervised learning, gradient descent, least squares, nonnegative least squares, delta learning rule (PES)
- *Signal Processing:*  
Fourier, Laplace transformation; computing optimal filters
- *Dimensionality Reduction:*  
PCA/SVD; function bases; Hebbian learning/Oja learning rule

## Review: What Did We Learn? (IV)

### ► “Meta-Level Skills”

- *Solving Problems by Building a Signal Flow Graph:*  
Applications to Hardware design, differentiable computing
- *Programming with Python/Numpy*
- *Building Neural Networks using Nengo:*  
Can be applied to neuromorphic hardware (see above), cognitive modeling, machine learning

# Summary

## ► **Party tricks *or* How to impress your mom:**

- *Did you know you can't keep your eyes still in the dark?*

They're controlled by the nuclei prepositus hypoglossi (NPH), part of the brainstem. I built one of those.

- *Did you know people are building hardware that works like the brain?*

It's called neuromorphic hardware, and uses spikes to communicate like the brain. I built some neural networks that can run on that hardware.

- *I know how the world's largest brain model works.*

In fact, I built some of the parts. Let me tell you about the working memory task.

Thank You!