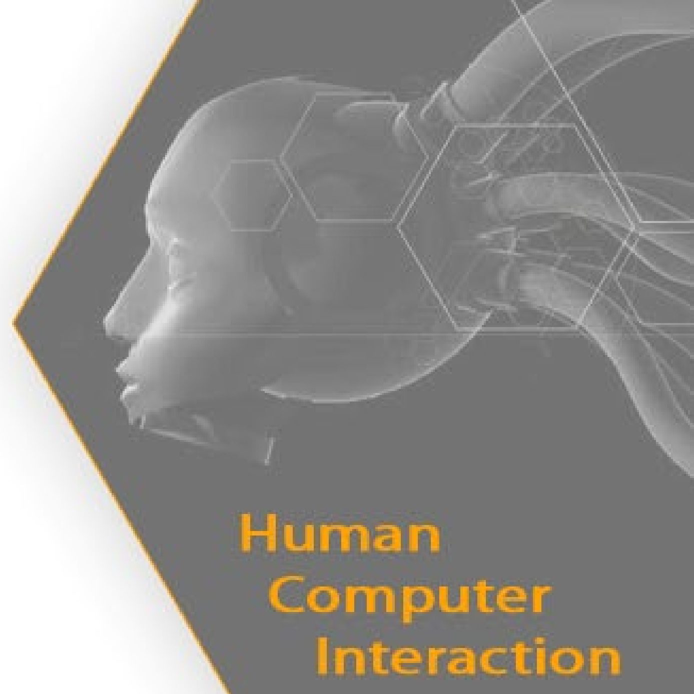


Sapience: A Brain-Inspired Cognitive Architecture

Tyler Streeter

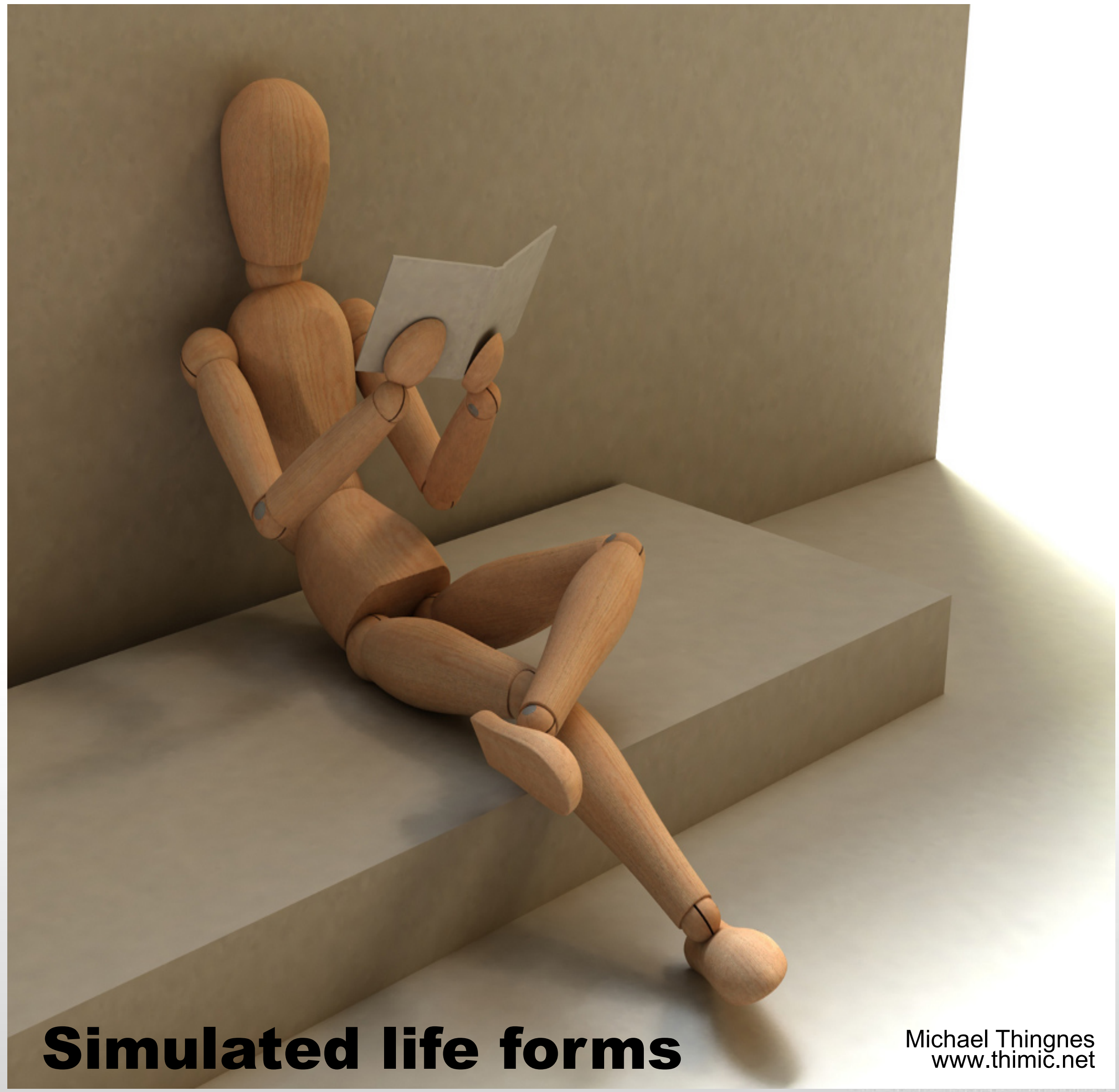
PhD Preliminary Proposal
June 11, 2009

IOWA STATE UNIVERSITY
OF SCIENCE AND TECHNOLOGY

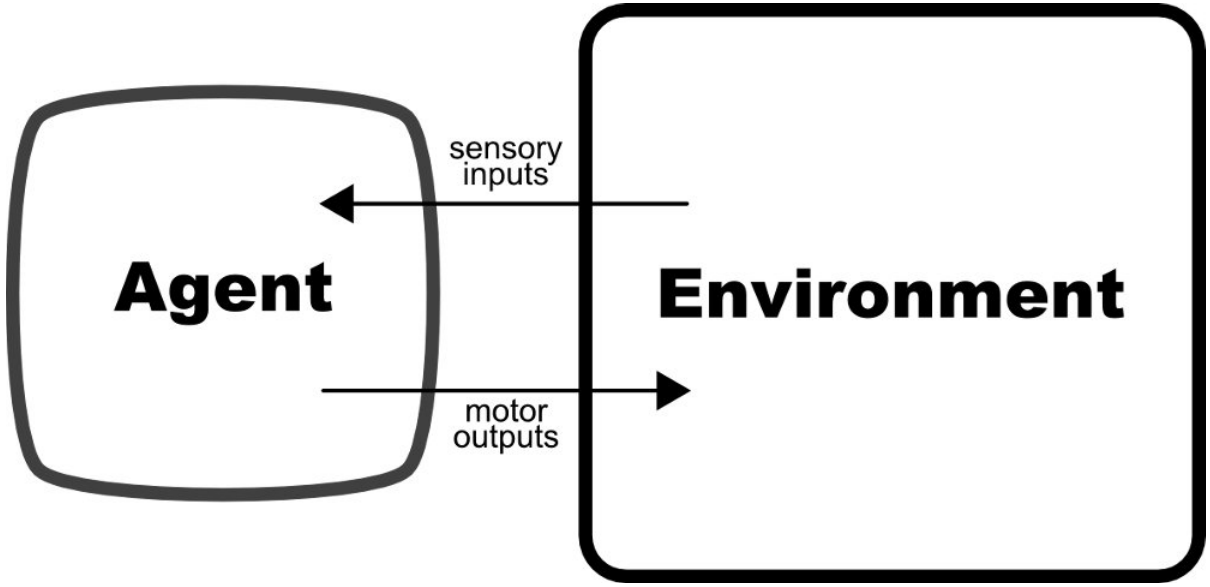
Outline

- Motivation
- Research strategy
- System objectives
- Architecture overview
- Detailed component descriptions
- Completed work
- Proposed experiments, timeline

Motivation: Software Brains for...



Motivation



- Software brain as a black box (agent approach)
- Adaptable (machine learning algorithms)
- Useful (achieves human-provided goals)
- Autonomous goal selection (curiosity)
- Arbitrary body (sensor & motor) configurations

System Objectives

- What should our software brain *do*?
- Achieve human-provided goals
- Define goals with theoretical reinforcement learning
  - Positive reinforcement = reward = good
  - Negative reinforcement = punishment = bad
  - Try to maximize positive reinforcement
- Human programmer defines goals, gives rewards for achieving them
- System implicitly achieves goals by maximizing rewards

System Objectives

Divide into two learning objectives and rewards:

- **Objective 1** Achieve external goals
  - External rewards given by programmer
  - Needs a good world model
- **Objective 2** Achieve internal curiosity goals
  - Internal rewards proportional to improvements to the world model
  - Autonomous goal selection
  - Helps achieve Objective 1

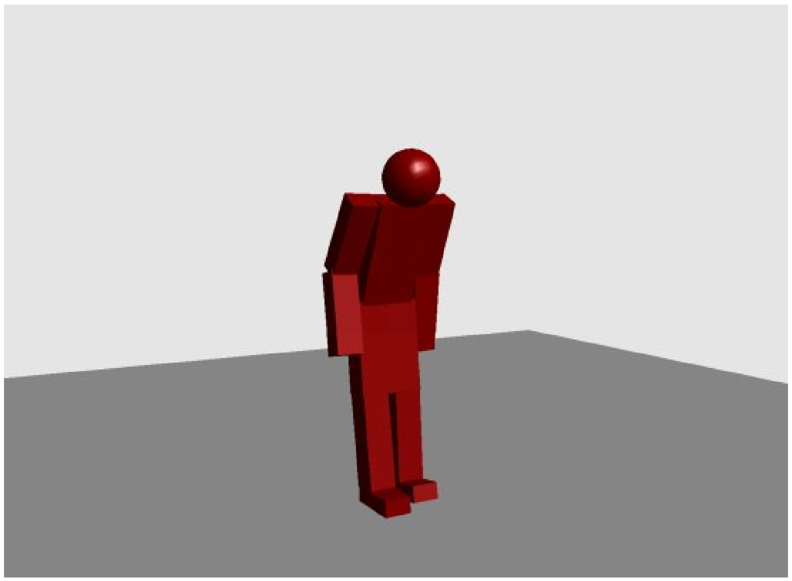
Organizing Principle: The Brain

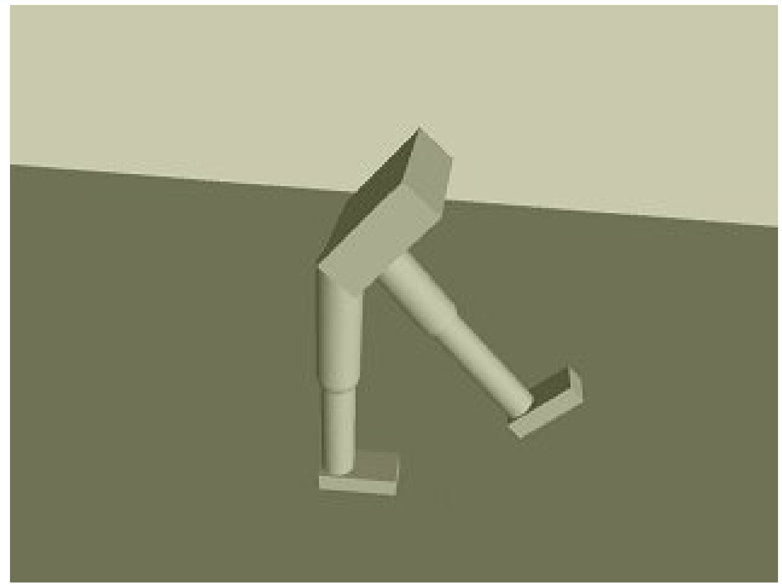
- Mammalian brain already achieves our objectives
  - Learned world model (sensory and motor cortex)
  - Learning context-dependent action selection (basal ganglia)
- Use abstract brain organization to guide architecture design
- What was evolution “trying” to design?
- Each major brain structure provides unique computational benefit to the animal
- …which machine learning algorithms solve similar problems? (feature extraction, temporal pattern representation, credit assignment, supervised learning, short-term memory storage and retrieval, …)

Research Strategy

- Design abstract architecture of interacting components
  - Define computational needs
  - Find the current best practical algorithms to fulfill each need
- Implement components in software
- Test individual pieces in isolation
  - Unsupervised learning (density estimation, pattern classification)
  - Sequential prediction (temporal pattern learning)
  - Reinforcement learning (classical conditioning, toy problems)
- Integrate components into a single system
- Simulated test environments, bodies, real-time probe tools
- Measure overall progress toward objectives (external rewards, model improvements)

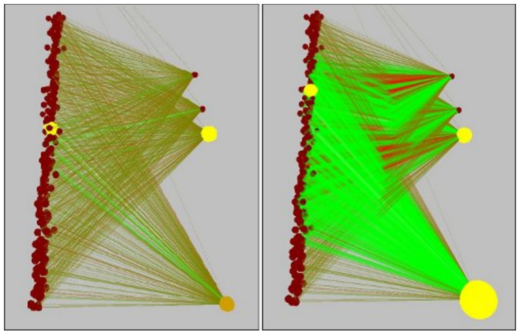
Previous Work





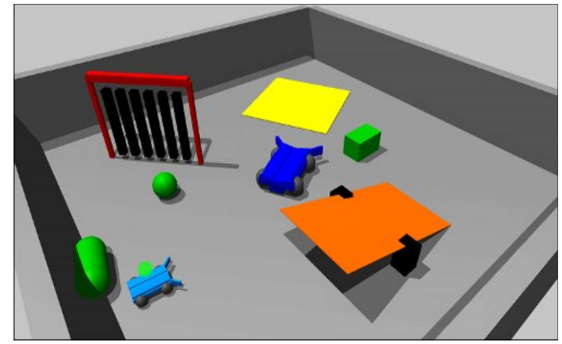
- Artificial evolution of neural network motor controllers
- Complex multi-dimensional control with little human feedback
- Video: standing http://video.google.com/videoplay?docid=-2510462304066175045
- Video: jumping http://video.google.com/videoplay?docid=1002062030982551847
- Video: walking http://video.google.com/videoplay?docid=-1150508620047972951

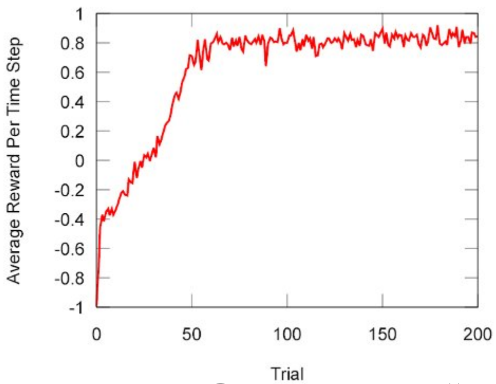
Previous Work

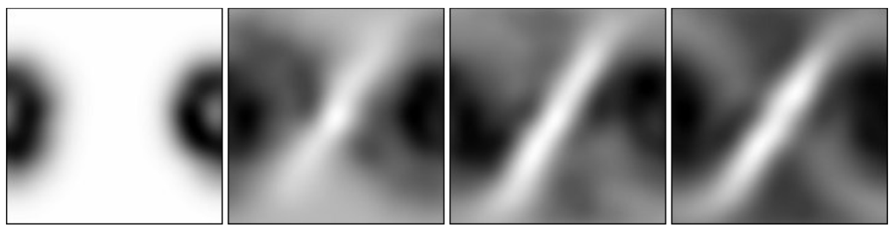


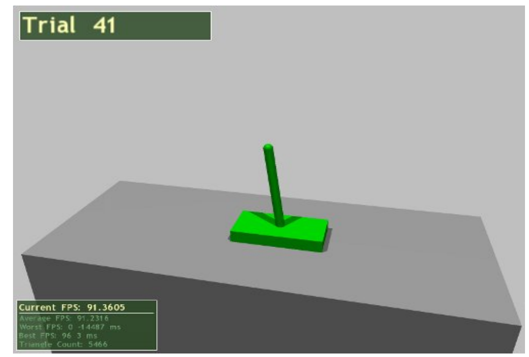
Verve

General Purpose Agents









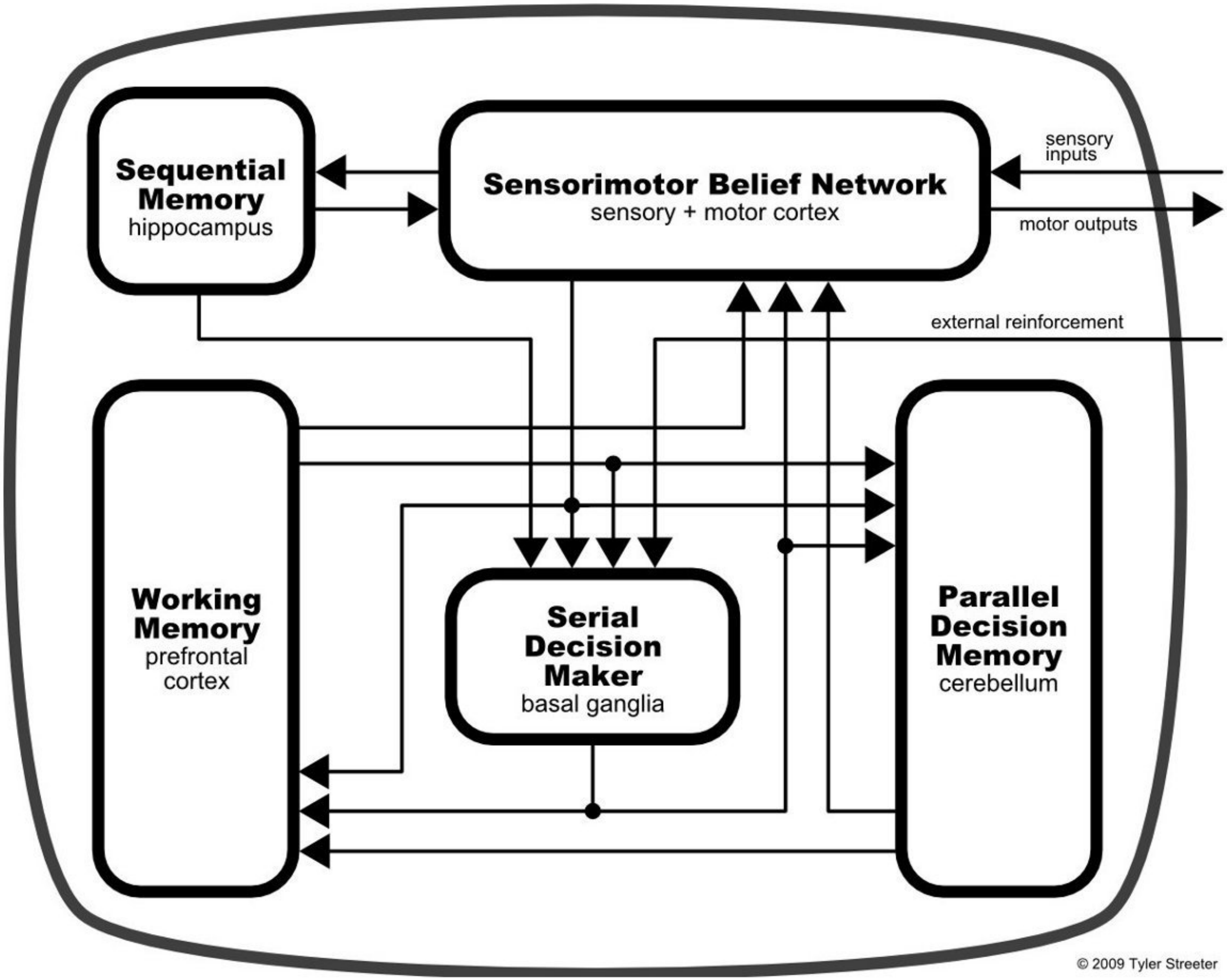
- MS thesis: “Verve” reinforcement learning architecture and implementation
- Same general motivation: to create a software brain
- More heuristic vs. information theoretic methods used here
- Limited to low-dimensional sensors, discrete actions
- Video: pole balancing learned from simple reinforcements
  http://video.google.com/videoplay?docid=8226600171334714429

Sapience Architecture

- Real-valued input/output arrays
- External and internal (curiosity) reinforcement mechanisms
- 5 internal components
  - **Sensorimotor Belief Network:** internal model of the world, inspired by sensory and motor cortex
  - **Sequential Memory:** sequential predictions, inspired by hippocampus
  - **Serial Decision Maker:** choose actions based on reinforcement, inspired by basal ganglia
  - **Parallel Decision Memory:** automates well-learned actions, inspired by cerebellum
  - **Working Memory:** extends action set w/ short-term memory, inspired by prefrontal cortex

Sapience Cognitive Architecture

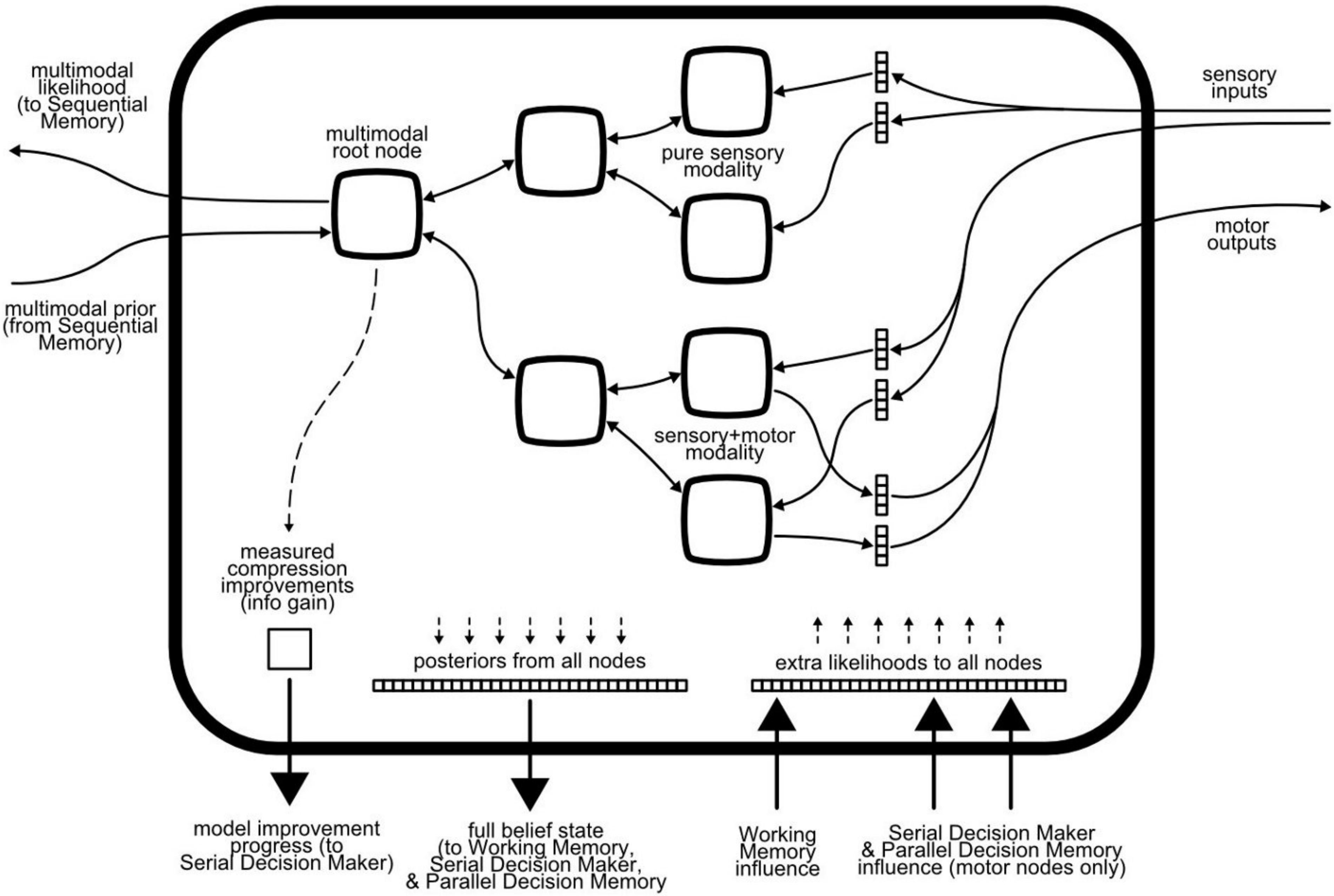
high-level organization



Sensorimotor Belief Network

- Brain inspiration: sensory and motor cortex
- Receives sensory input data, produces motor control outputs
- Probabilistic model of the external world (Bayesian inference)
- Learned symbolic representation of data “causes”
- Computes model improvements (used for curiosity rewards)
- Provides a “context representation” for decision making components
- Can be influenced/biased by other components

Sensorimotor Belief Network



Sensorimotor Belief Network

Bayesian Inference

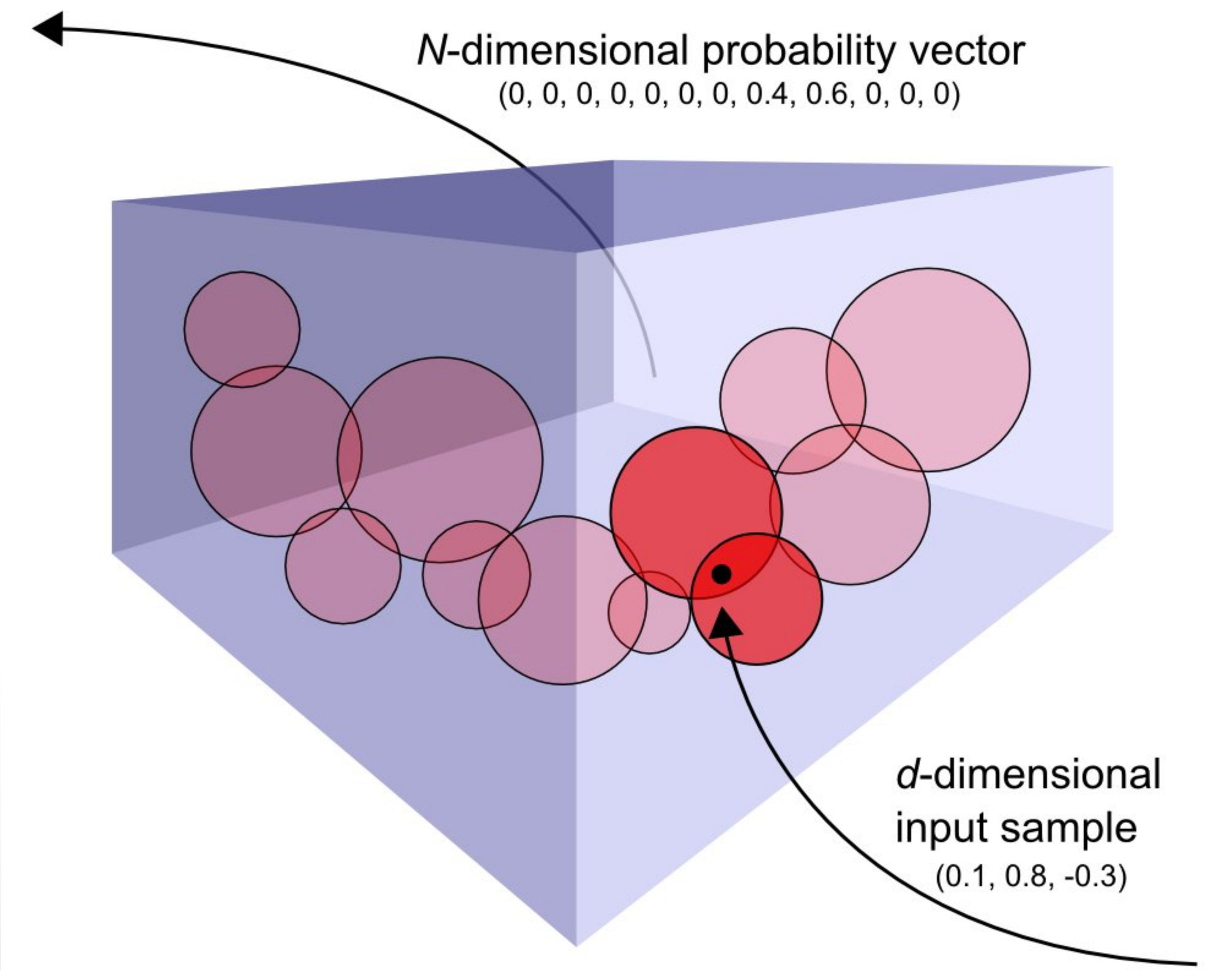
$$P\left(C|E\right) = \frac{P(C)\,P(E|C)}{P(E)}$$

$$posterior \propto prior \times likelihood$$

- *C*: discrete "class" variable (set of causes/hypotheses)
- *E*: continuous "evidence" vector variable (data samples)
- *P(C|E)*: posterior probability of each class being the cause of the given evidence/data
- *P(C)*: prior probability of each class for *any* given data sample
- *P(E|C)*: probability of seeing the current data sample assuming a certain value for *C* (aka the "likelihood")
- *P(E)*: prior probability of seeing the data sample (ignored, used only for normalization)

Sensorimotor Belief Network

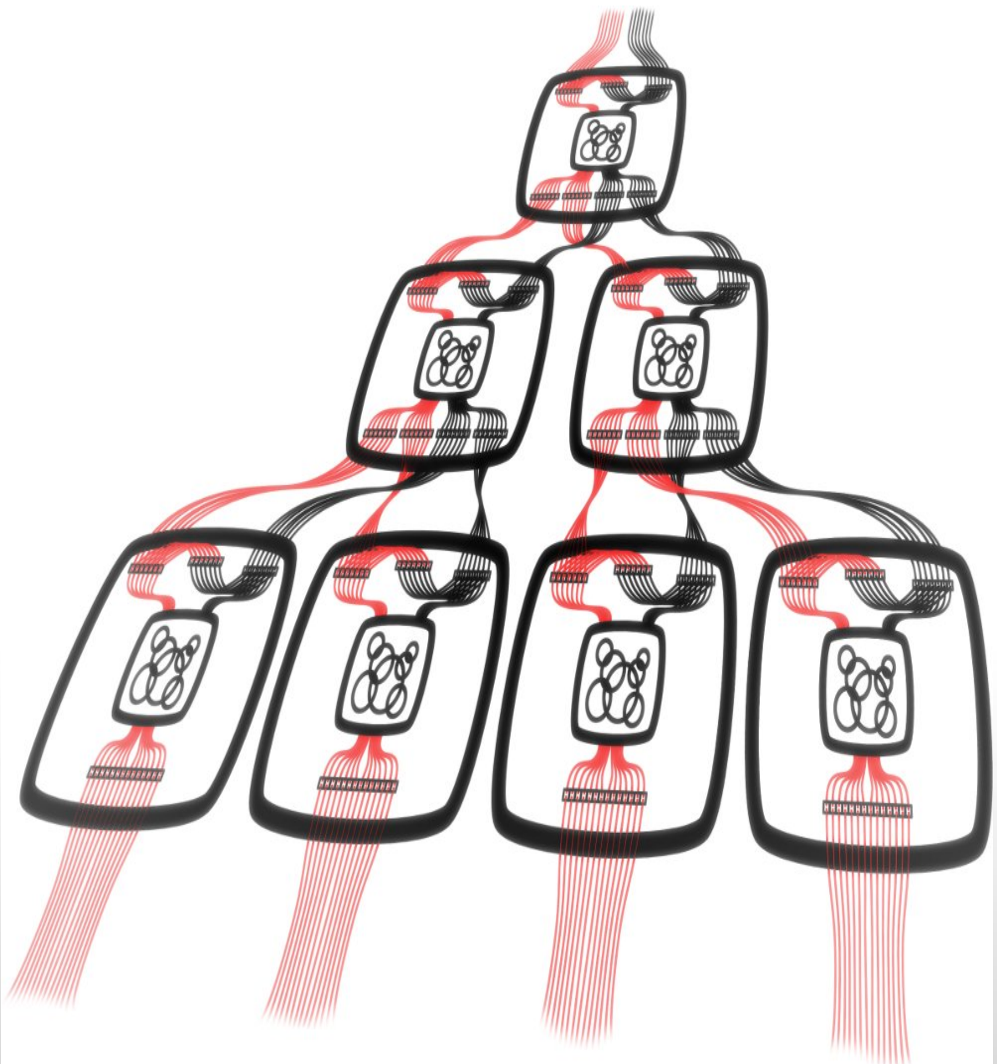
Symbolic Representation



- Represent data points in Euclidean space
- Classify with *d*-dimensional Gaussian kernels (kernel mixture model)
- kMER algorithm learns kernel center, radius (unsupervised, infomax-based)
- For each sample, compute PMF over kernels/classes/causes/hypotheses

Sensorimotor Belief Network

Bayesian Network



- Curse of dimensionality: machine learning gets harder for high-dimensional data
- Subdivide data space into small-dimensional subsets
- Combine results with Bayesian network (distributed Bayesian inference)
- Demo: natural images

Sensorimotor Belief Network

Model Improvements, Curiosity

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

- Measure the “divergence” (in bits/nats) between the prior and posterior distributions
- D_{KL}(posterior||prior) = info gain from new data = improvement to the world model
- Total model improvement: average info gain over all nodes in Bayesian hierarchy
- Use model improvement as internal curiosity reward

Sensorimotor Belief Network

Generating Motor Outputs

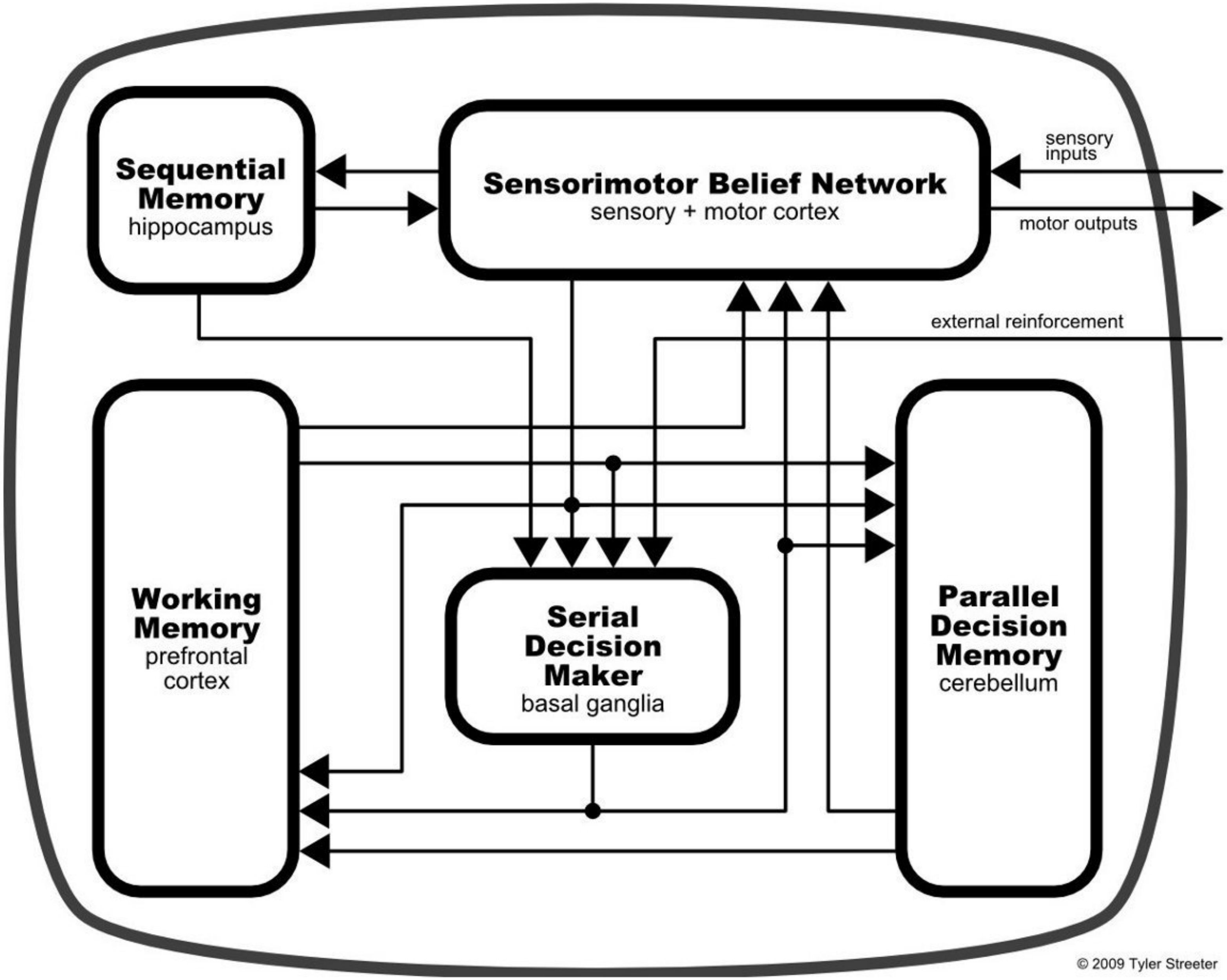
- In hierararchical Bayesian network, top-down priors represent predictions for level below
- Lowest level: priors are raw data predictions
- For motor modalities, use these “predictions” as motor control signals
- Example
  - Proprioceptive inputs: joint angle, stiffness
  - Motor outputs: desired joint angle, desired stiffness
  - Low-level spring-like servo controllers compute actual forces

Sensorimotor Belief Network

Bootstrapped Motor Learning

- Must ensure a thorough initial sampling of motor space
- Otherwise, degenerate learned motor representation
- Reflex system: simple random neural network (sensors to effectors)
- Initially full reflex-based control, smoothly transition to voluntary control

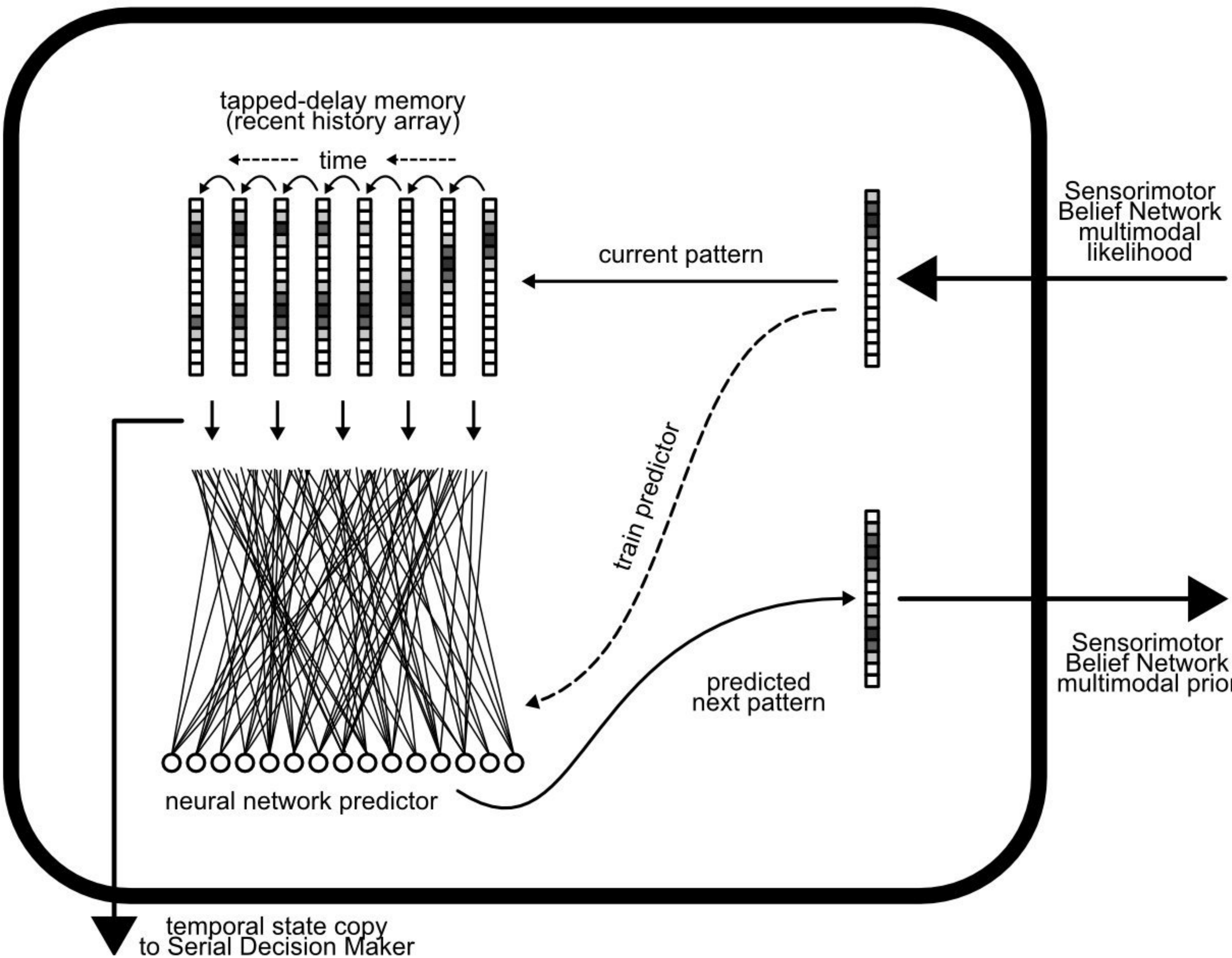
Sapience Cognitive Architecture
high-level organization



Sequential Memory

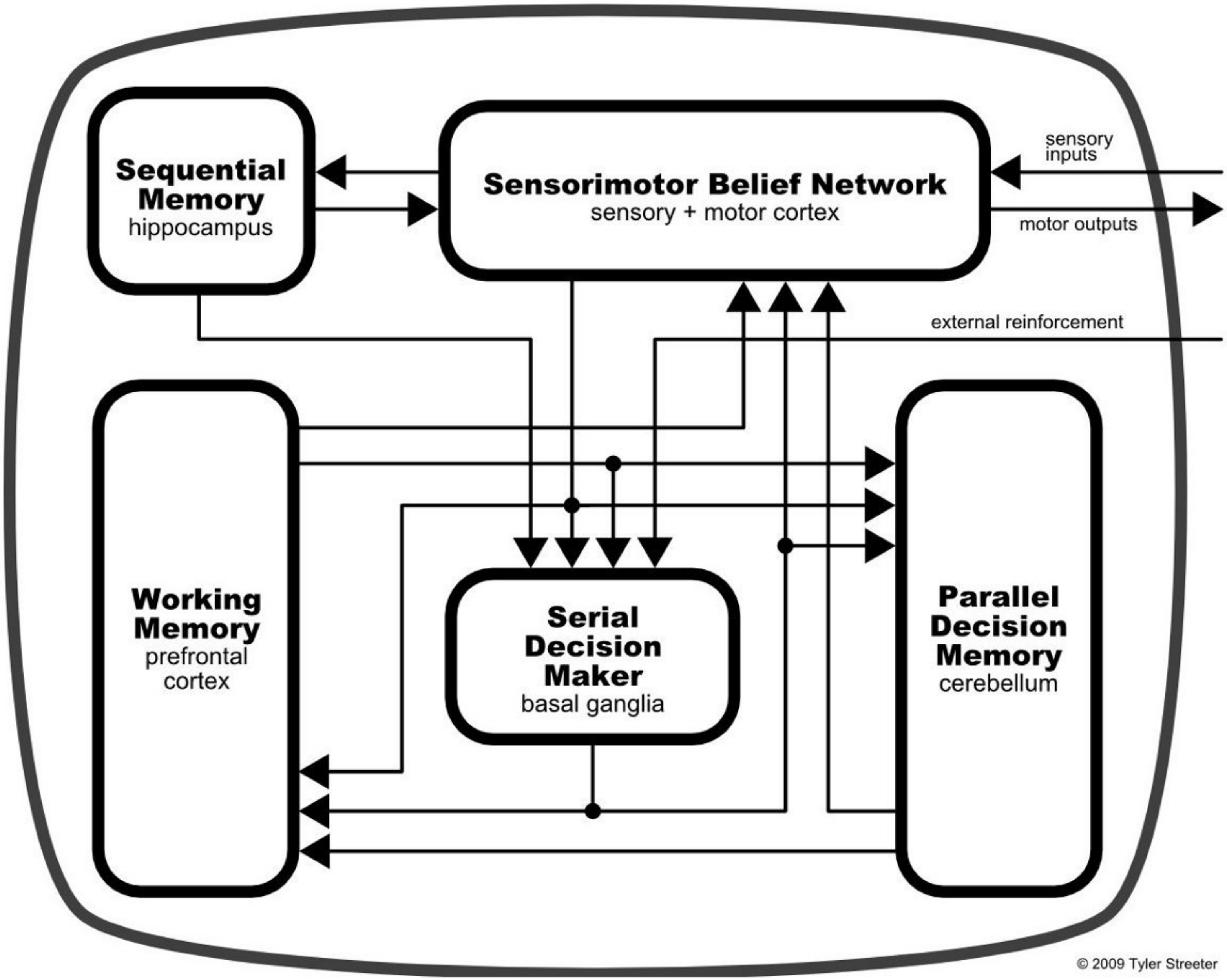
- Brain inspiration: hippocampus
- Makes sequential predictions
- "Dynamic reconstruction" – learning to model and predict complex temporal signals
- Models short-term memory trace with tapped delay line array
- Supervised learning neural network predictor
- Provides prior distribution to multimodal root node in Bayesian network

Sequential Memory



Sapience Cognitive Architecture

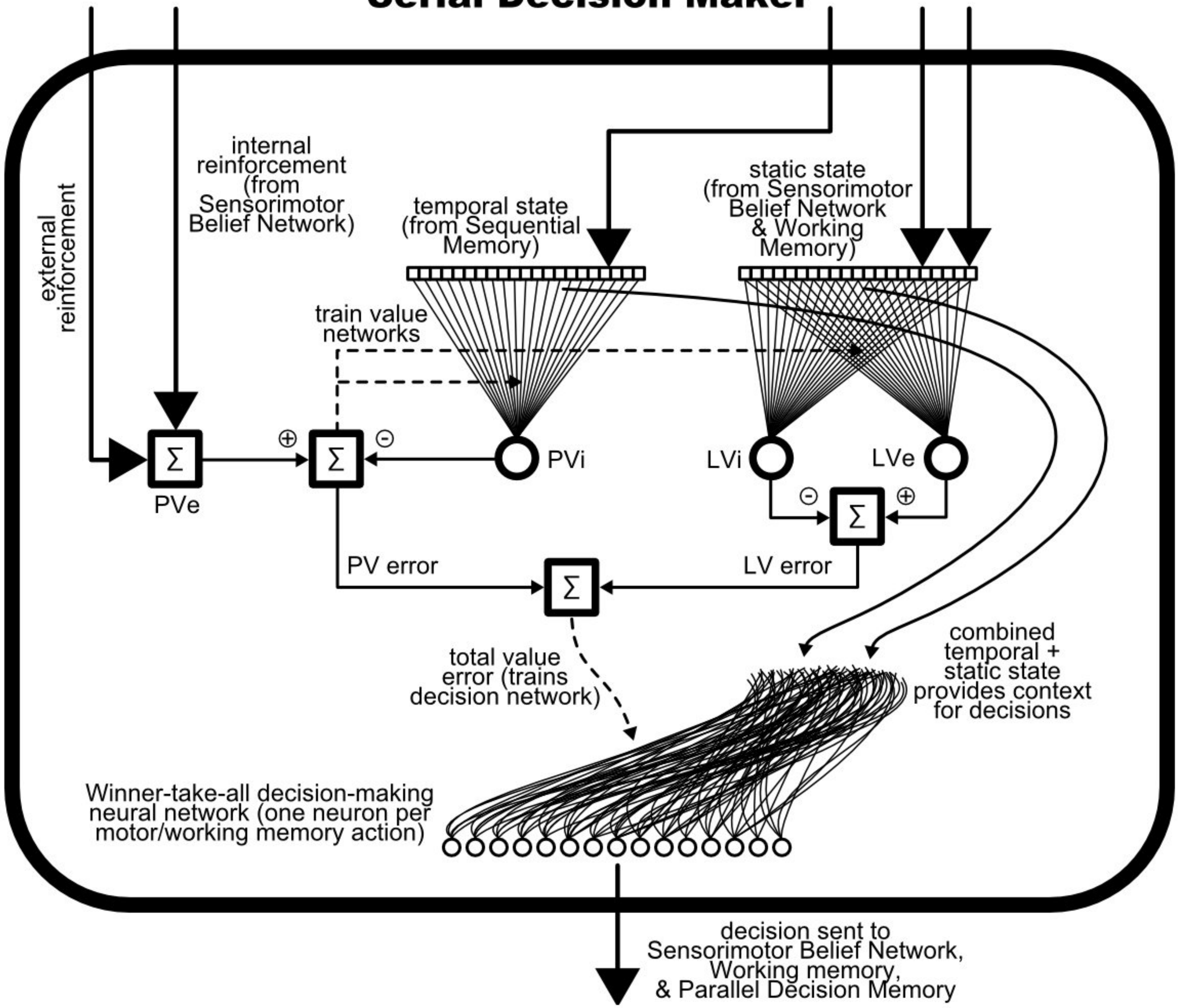
high-level organization



Serial Decision Maker

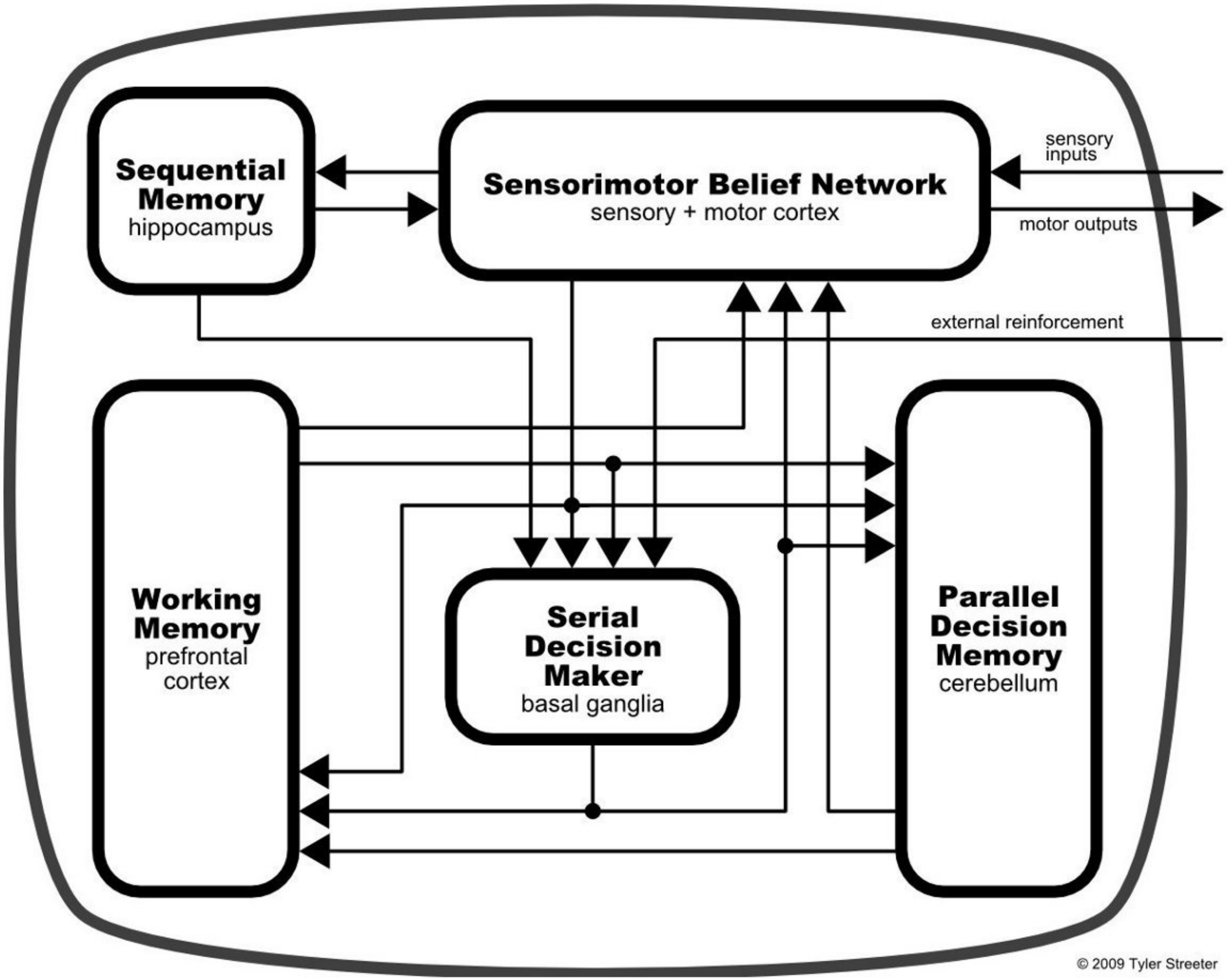
- Brain inspiration: basal ganglia
- Learns context-dependent value
- Learns context-dependent actions
- Chooses from motor actions, working memory actions
- Uses PVLV (primary value learned value) model of midbrain dopamine activity
- “Dopamine” signal reinforces action selection

Serial Decision Maker



Sapience Cognitive Architecture

high-level organization

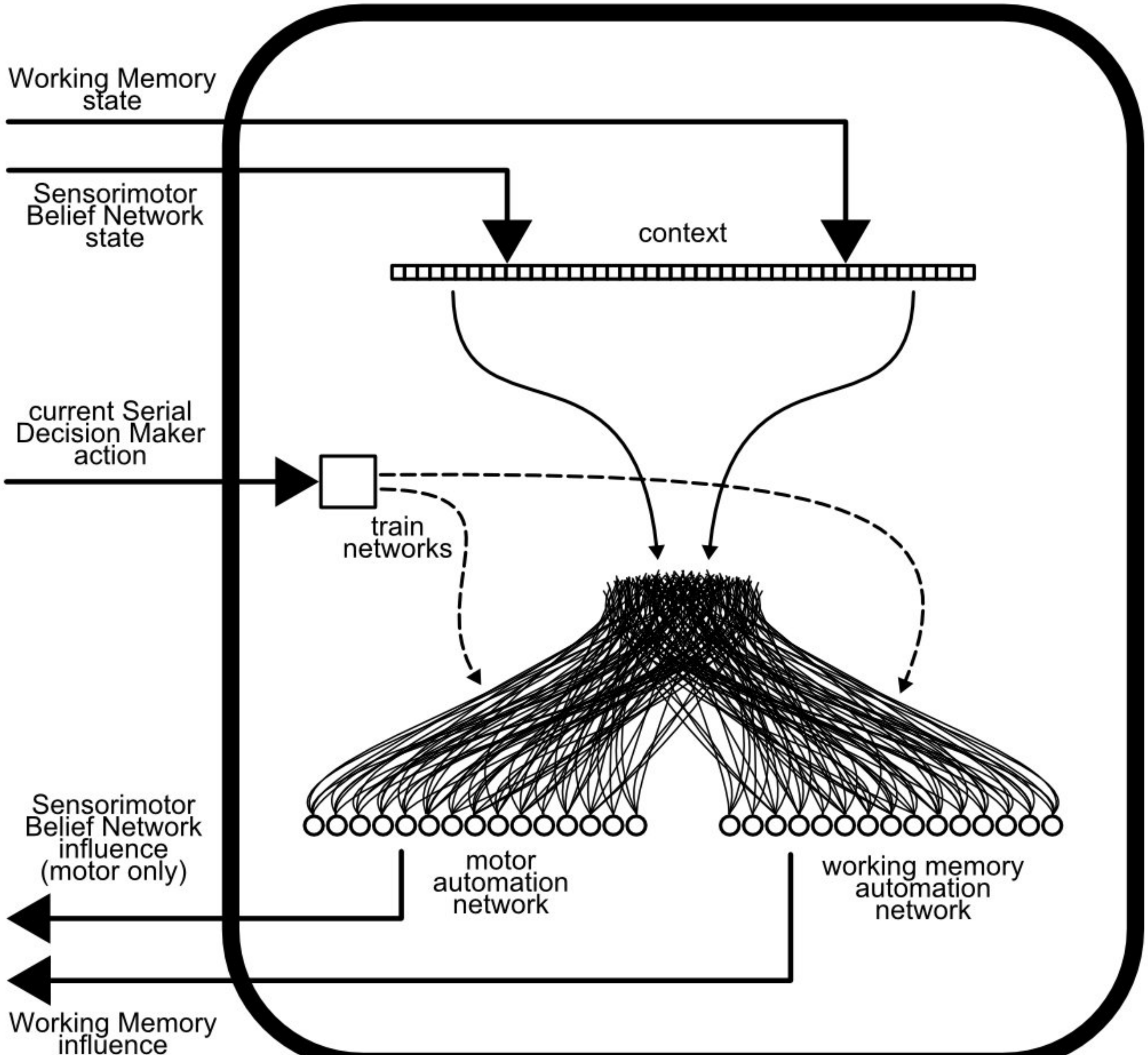


Parallel Decision Memory

- Brain inspiration: cerebellum
- Essentially same inputs/outputs as Serial Decision Maker
- Automates Serial Decision Maker's actions, freeing it to focus on novel tasks
- Storage area for well-learned decisions
- Supervised learning neural networks
- Parallel outputs drive multiple targets simultaneously
- Video: computational model of cerebellum learns to automate parallel muscle control

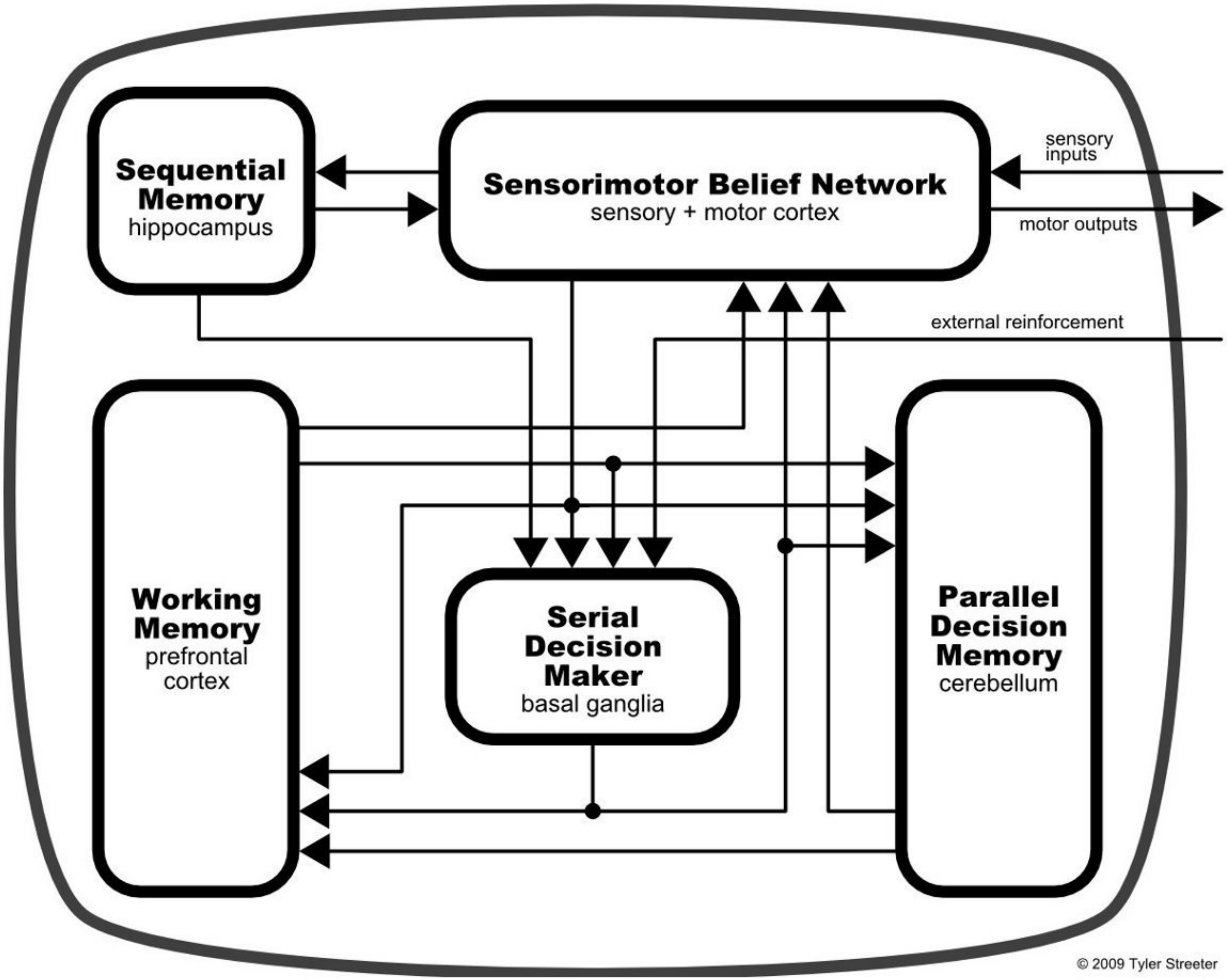
http://video.google.com/videoplay?docid=3602661334569424179

Parallel Decision Memory



Sapience Cognitive Architecture

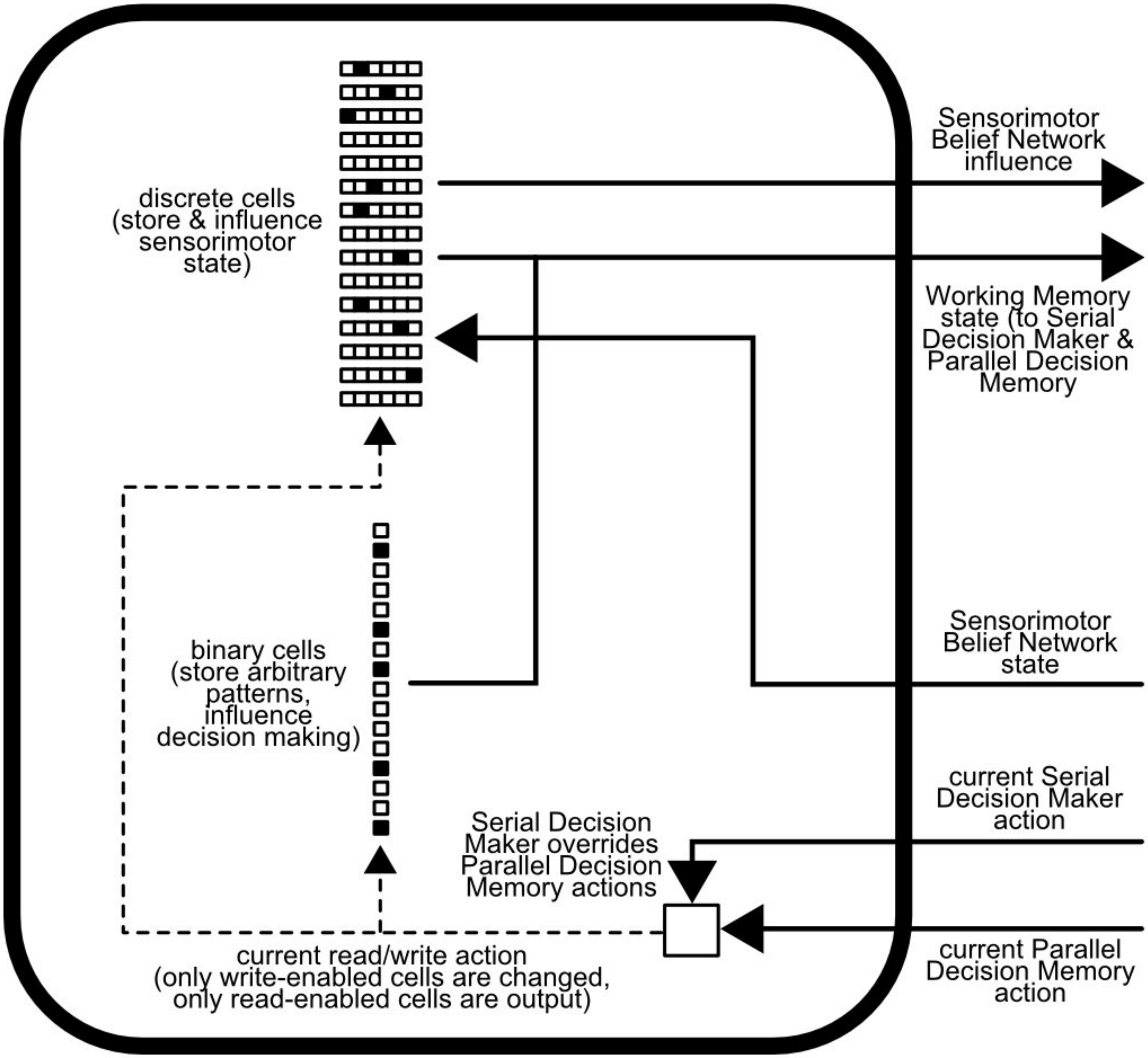
high-level organization



Working Memory

- Brain inspiration: prefrontal cortex
- Temporary data storage
- Discrete set of memory cells
- Working memory actions: read/write
- Read: let contents influence other components
- Write: update contents with new values
- Reinforcement learning of working memory control
- Feedback loop with Serial Decision Maker: powerful mechanism for general program learning

Working Memory

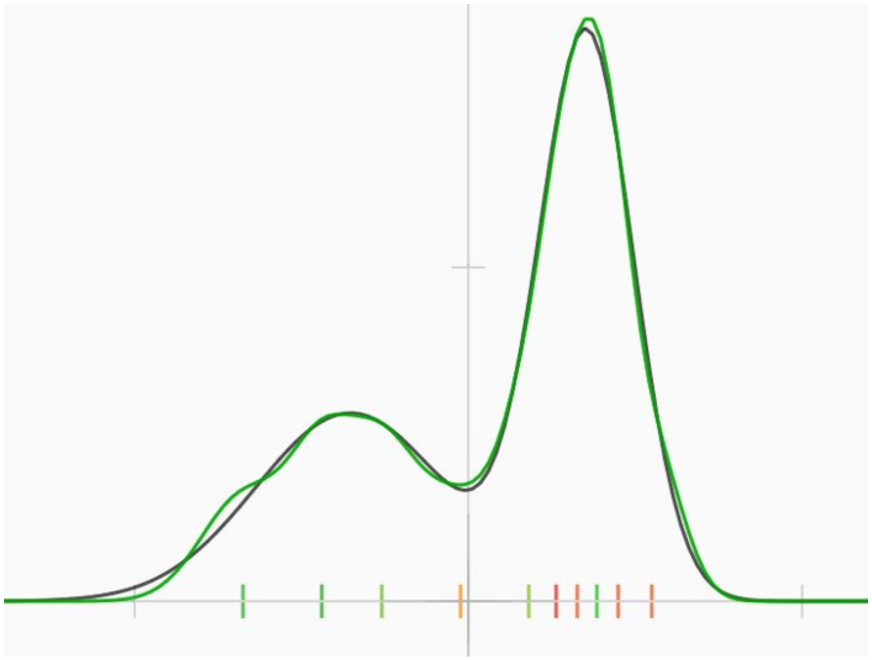


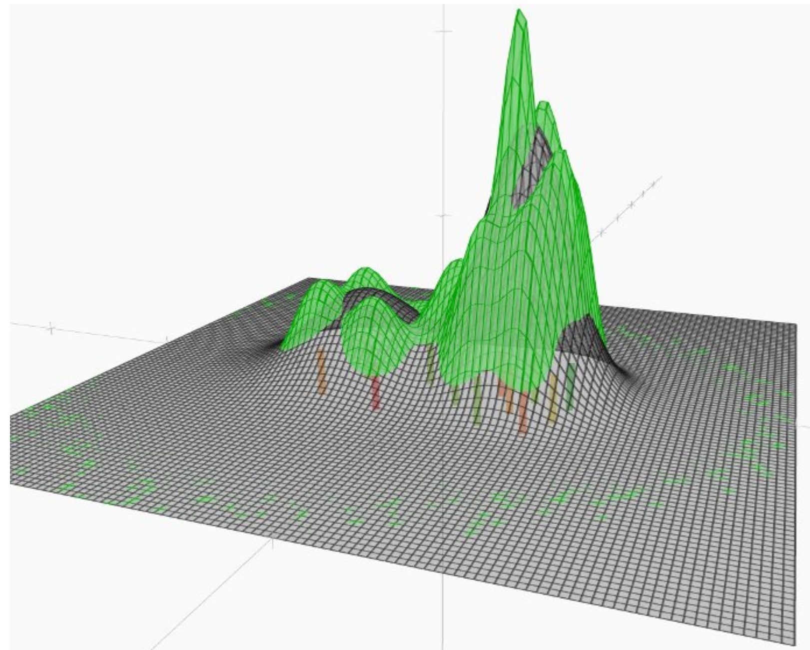
Completed Work

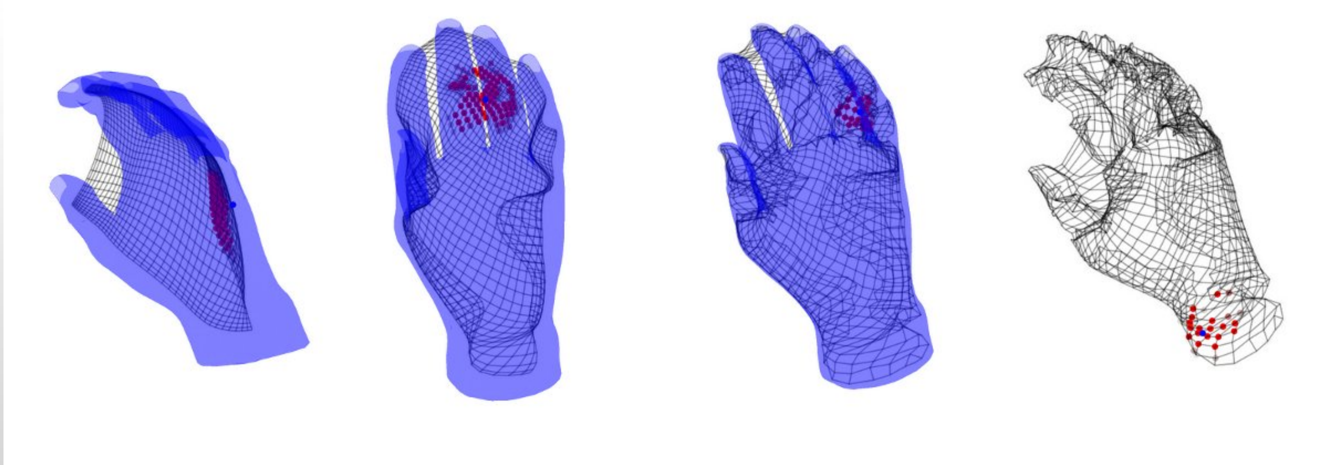
- Complete, implementable architecture design
- Initial architecture implementation
  - C++ library w/ Python bindings for Linux, OS X, Windows
  - Built-in CPU utilization measurements
  - Automatic thread-based parallelization
- Real-time probe tool
- Simulation environment for experimentation
- Over 1 MB of source code (uncompressed text files)

Completed Work

Algorithm Test Programs

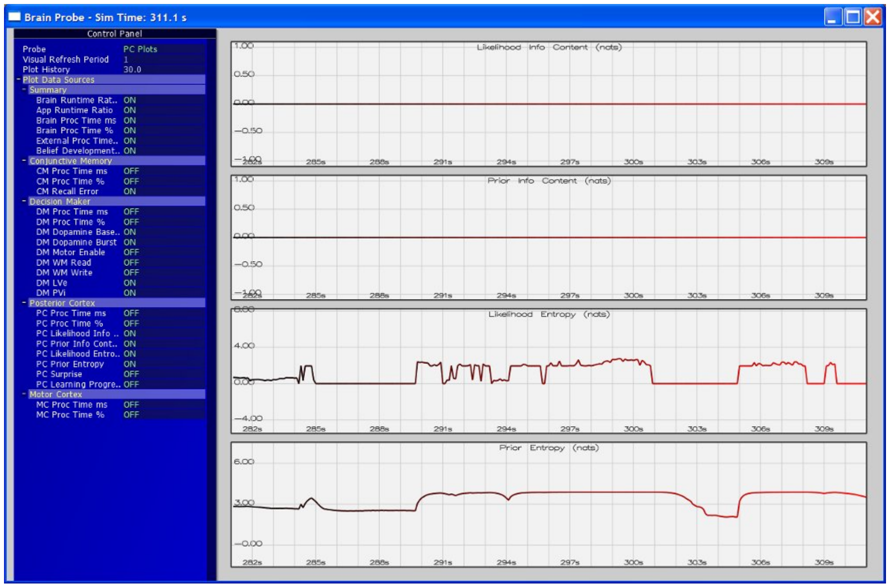


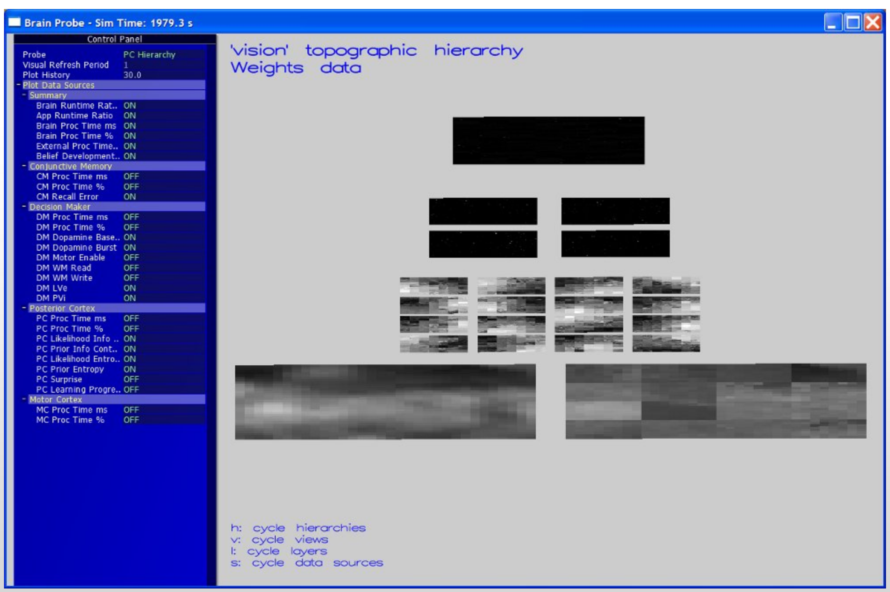




Completed Work

Real-Time Probe

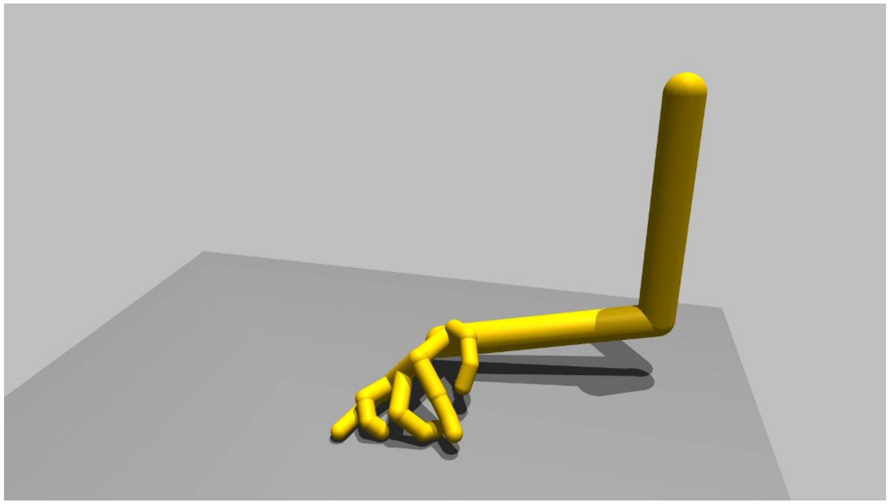


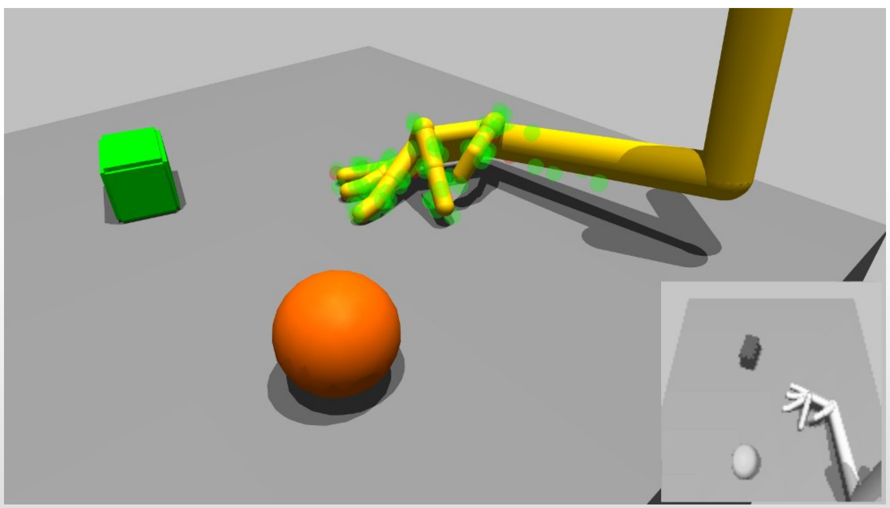


- Probe tool runs in parallel with any application
- Real-time plots of many internal variables (information theoretic values, CPU utilization, etc.)
- Visualization of learned internal representations
- Tight experimental feedback loop – watch internal changes in real-time while interacting with the system

Completed Work

Simulation Environment

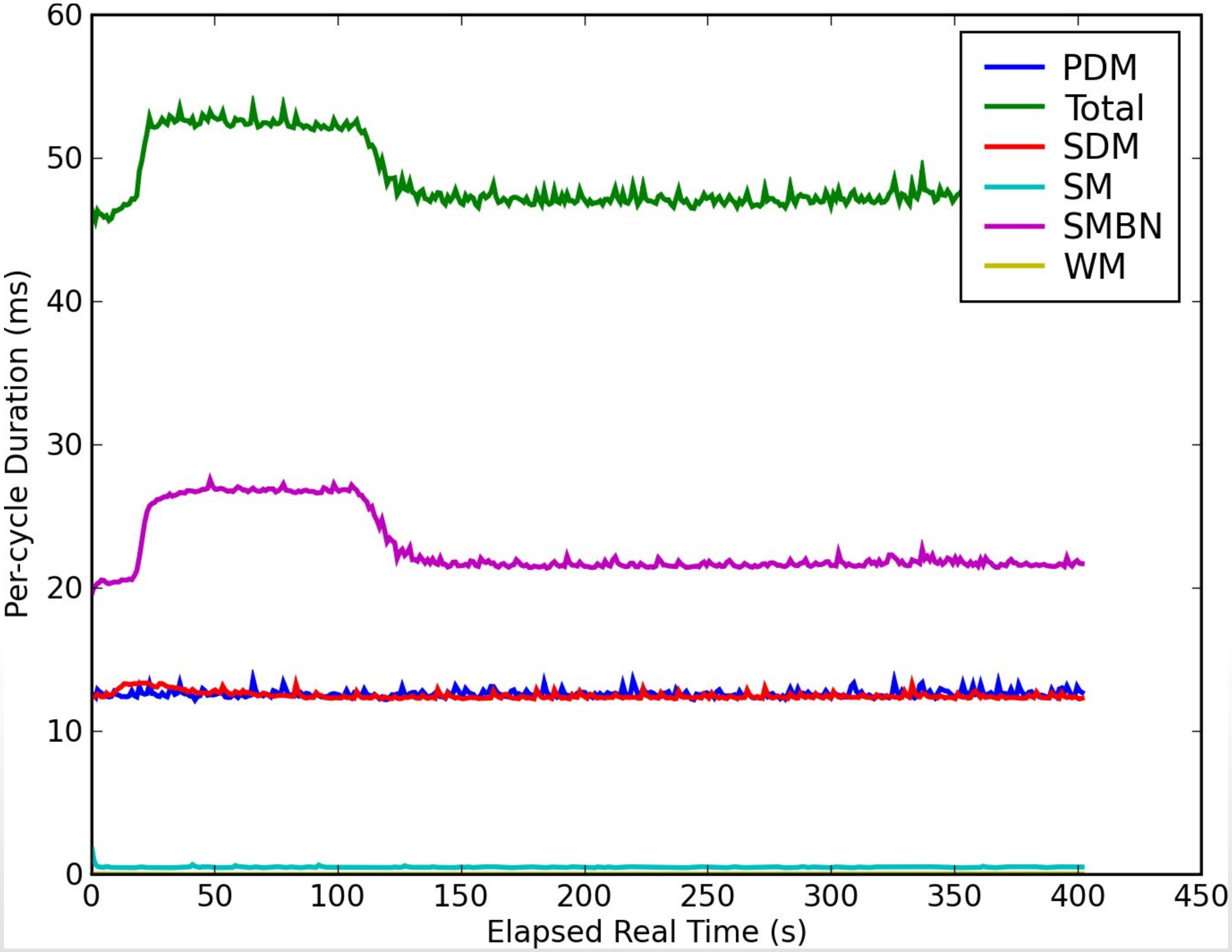




- 24 degree-of-freedom arm
- Shoulder fixed in space
- Proprioceptive sensors (joint angle, stiffness)
- Tactile sensors
- Vision sensor (monocular, monochrome)
- Servo motors (controls desired joint angle, stiffness; 1 per DOF)
- Adjustable sensor resolution (input dimensionality)
- Demo: arm simulation w/ probe

Completed Work

CPU Timing Test



- 32x32 visual inputs
- 42 proprioceptive inputs
- 42 motor outputs
- Target: update @ 10 Hz
- Result: <50ms needed per update
- Each second: 500ms for Sapience, 500ms for simulation

Proposed Experiment 1

Passive Information Gain

- Question: With no active motor control (reflexes only), what is the baseline expected rate of information gain?
- Vision inputs only
- Arm driven by external reflex system
- Plot info gain over time (model improvement rate)
- Many other possible plots:
  - With vs. without hierarchy
  - More vs. fewer kernels
  - Lower vs. higher resolution vision

Proposed Experiment 2

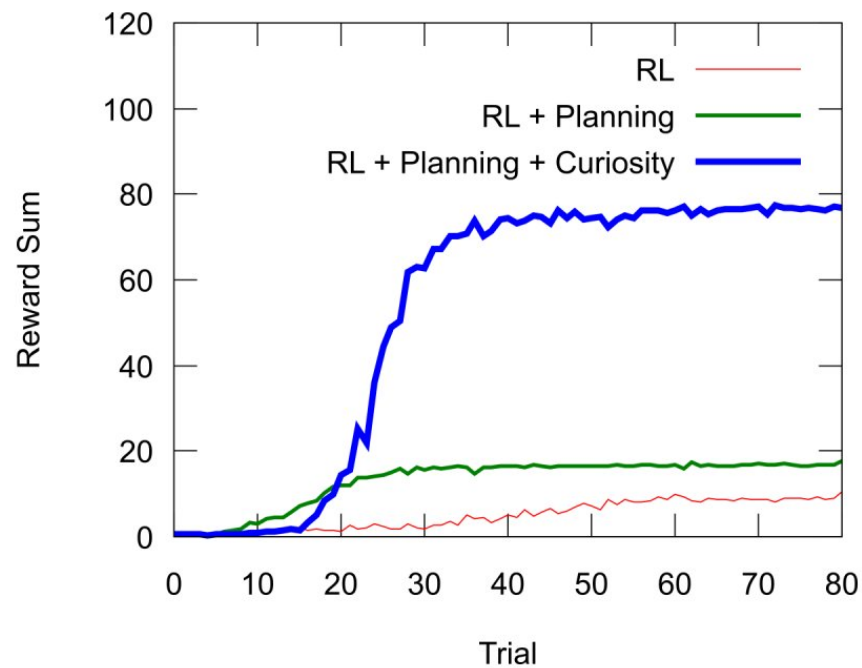
Active Information Gain

- Question: How much does curiosity help? i.e., with active curiosity-driven motor control enabled, by how much does the rate of information gain increase?
- Vision inputs, proprioceptive inputs, motor outputs
- Similar setup as before, but transition from reflex control to full active control
- Reinforcements: internal curiosity rewards for model improvements
- Plot info gain over time (should be greater than passive observation)

Proposed Experiment 3

External Reward Acquisition

- Question: Can the system reliably learn to achieve externally-provided goals in a high-dimensional sensorimotor space?
- Reaching task: target hand positions in space
- Internal curiosity rewards
- External rewards for touching targets
- Measure progress as the reward acquisition rate over time
- Performance with curiosity should be better than without



Prior work: curiosity helps acquire more external rewards