Emergence of Past and Future in Evolving Neural Networks

CSCE 633 Machine Learning (Spring 2015)

Yoonsuck Choe
Department of Computer Science and Engineering
Texas A&M University

* Joint work with Ji Ryang Chung and Jaerock Kwon

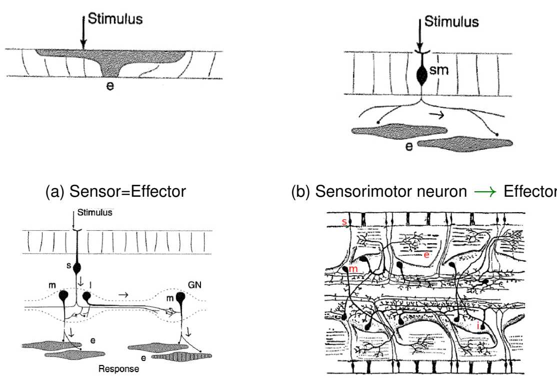
Evolution of Memory and Prediction

Past, Present, and Future:

- Past: memory
- Present: reactive behavior
- Future: prediction, anticipation

→ How did these temporal functions emerge/evolve?

Simple to Complex Brains



(a) Sensor=Effector (b) Sensorimotor neuron → Effector

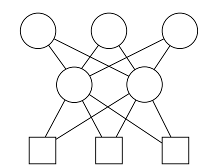
(c) Sensory, Motor, and interneuron (d) Complex circuit

Swanson (2003)

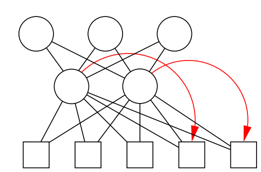
- From reactive to recurrent.
  - Reactive: Input→Output
  - Recurrent: Input modulating on-going internal activity

Time, in the Context of Neural Networks

- Feedforward neural networks:
  Have no memory of past input.

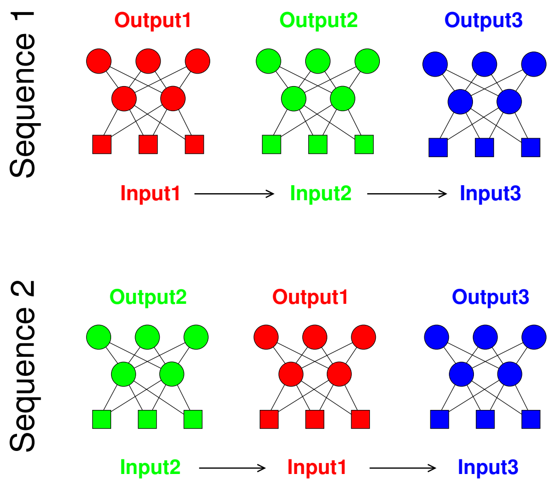


- Recurrent neural networks:
  Have memory of past input.

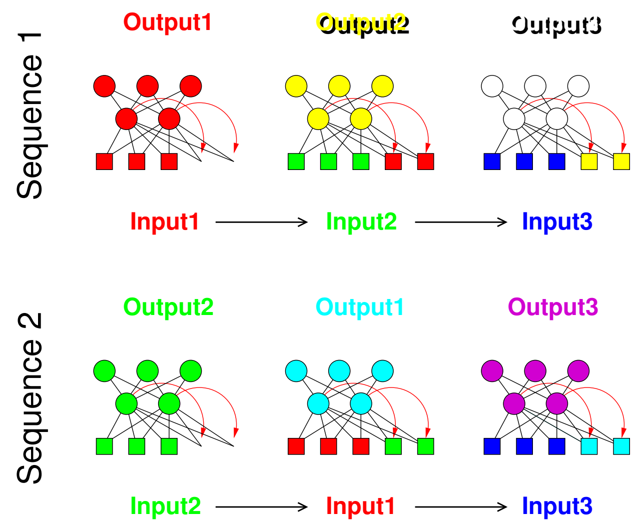


e.g., Elman (1991)

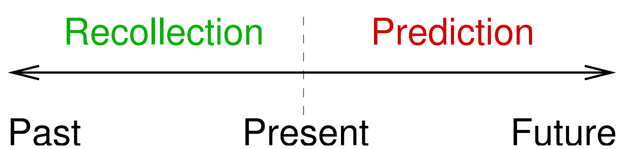
Feedforward Networks



Recurrent Networks



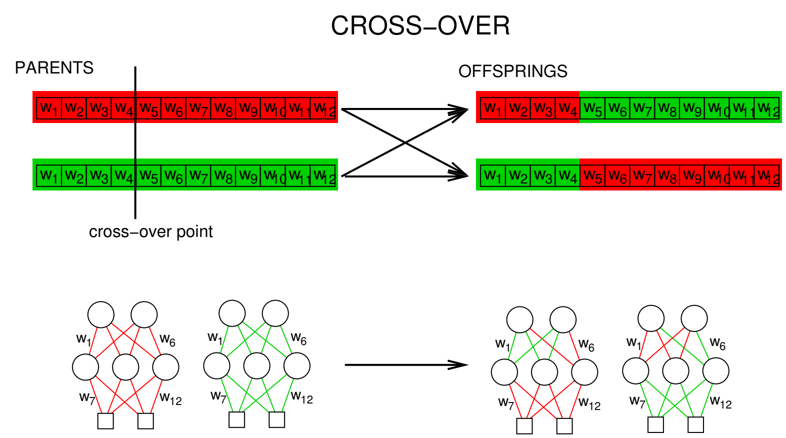
Research Questions



- [Q1] how did **recollection (memory)** evolve?
  - From reactive (present) to recurrent (past).

- [Q2] how did **prediction** evolve?
  - From recurrent (past) to predictive (future).

Approach



- Neuroevolution: evolve neural networks.

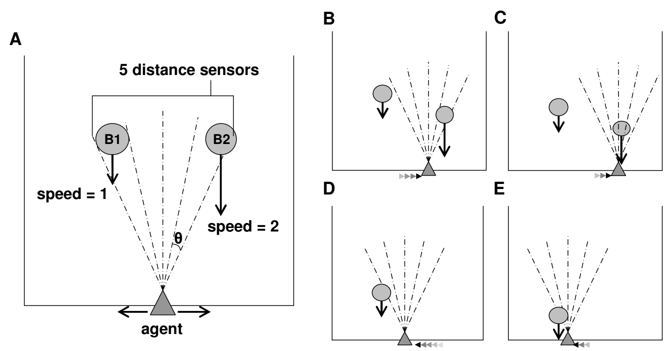
Part I: Recollection

Recollection in Feedforward Networks?

Is it possible for a feedforward network to show memory capacity?

- What would be a minimal augmentation?
- **Idea:** allow **material interaction**, dropping and detecting of external markers.

Memory Task: Catch the Balls



cf. Beer (2000); Ward and Ward (2006)

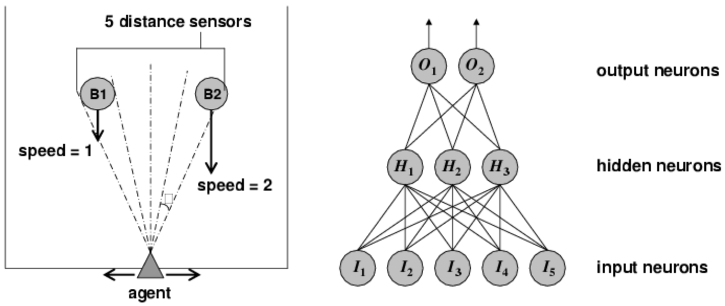
- Agent with range sensors move left/right.
- Must catch both falling balls.
- Memory needed when ball goes out of view.

Three Network Types Compared

Compare three different networks:

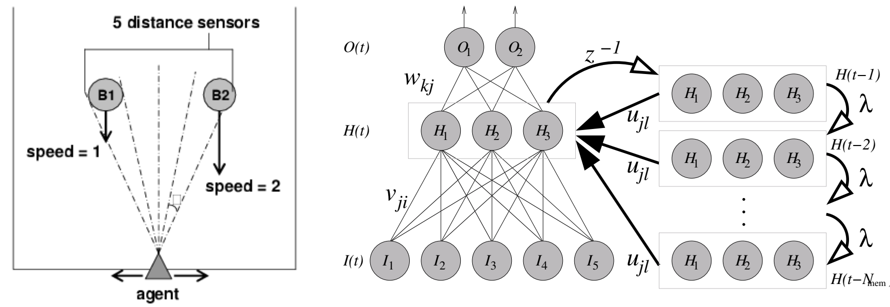
1. Feedforward
2. Recurrent
3. Dropper/Detector (with Feedforward net)

1. Feedforward Network



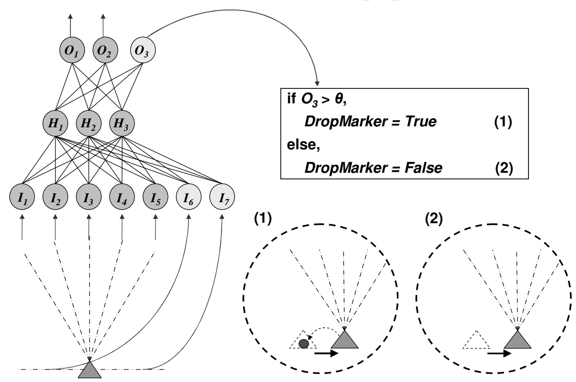
- Standard feedforward network.

2. Recurrent Network



- Standard recurrent network (Elman 1991).

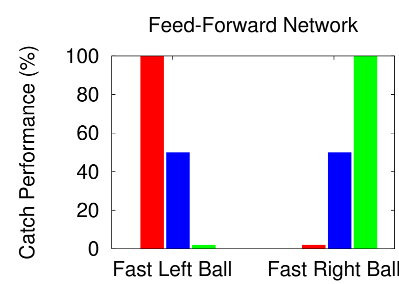
3. Feedfwd Net + Dropper/Detector



Feedforward network plus:

- Extra output to **drop** markers.
- Extra sensors to **detect** the markers.

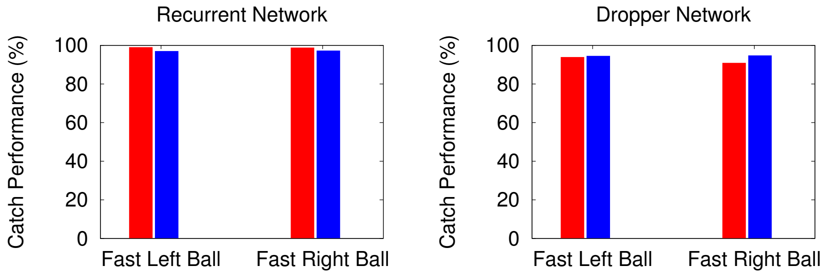
Results: Feedforward



On average, only chance-level performance (50%).

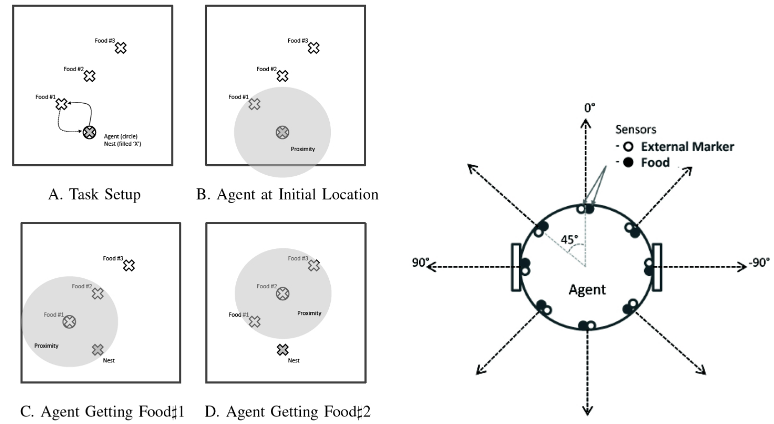
- Always move to the fast ball.
- Randomly pick fast or slow ball and approach it.

Results: Recurrent vs. Dropper



- No difference in performance between dropper/detector net (right) and recurrent network (left).

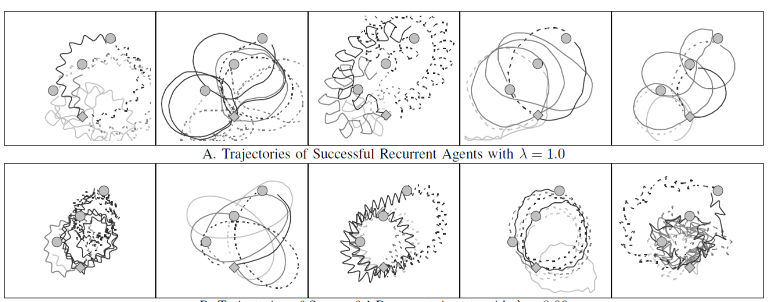
2D Foraging Task



A. Task Setup
B. Agent at Initial Location
C. Agent Getting Food♯1
D. Agent Getting Food♯2

- Simple 2D foraging task that requires memory.
- Simple 2D navigation agent with short-range sensors.

Foraging Behavior: Recurrent Net

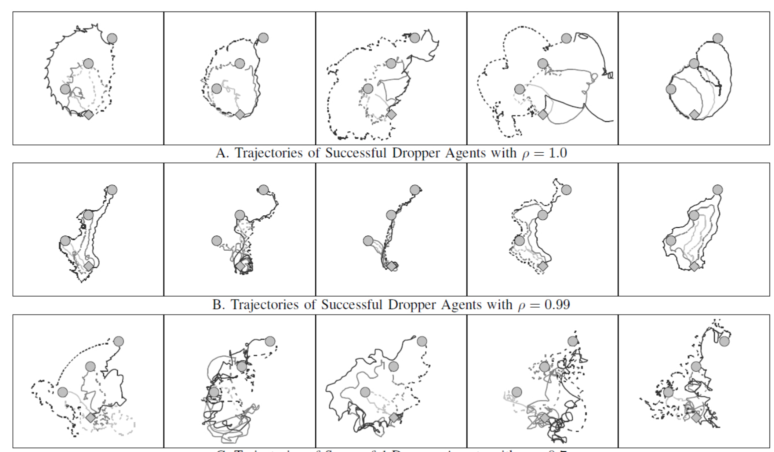


A. Trajectories of Successful Recurrent Agents with $\lambda = 1.0$

B. Trajectories of Successful Recurrent Agents with $\lambda = 0.99$

- λ: memory decay parameter (high = low decay).

Foraging Behavior: Dropper Net



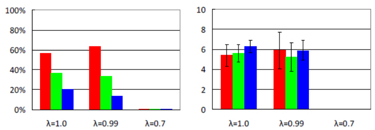
A. Trajectories of Successful Dropper Agents with $\rho = 1.0$

B. Trajectories of Successful Dropper Agents with $\rho = 0.99$

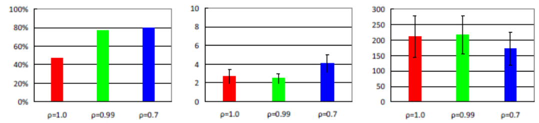
C. Trajectories of Successful Dropper Agents with $\rho = 0.7$

- ρ: dropper evaporation parameter (high = no evaporation)

Foraging Performance



A. Success Rate B. Distance Traveled



A. Success Rate B. Distance Traveled C. Marker Drops

- Left: recurrent network
- Right: dropper network

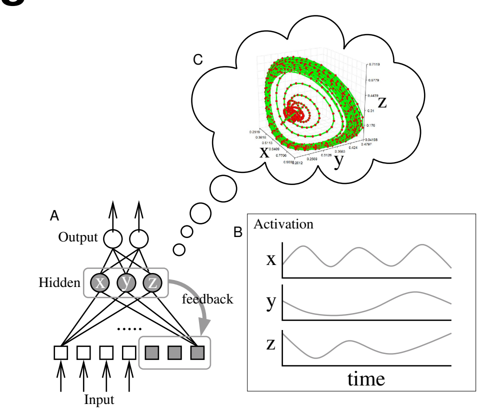
Part I Summary

- Reactive, feedforward networks can exhibit memory-like behavior, when coupled with minimal material interaction.
- Adding sensors and effectors could have been easier than adjusting the neural architecture.
- Transition from external olfactory mechanism to internal memory mechanism?
- Successfully extended to 2D foraging task.

Part II: Prediction

Largely based on Kwon and Choe (2008)

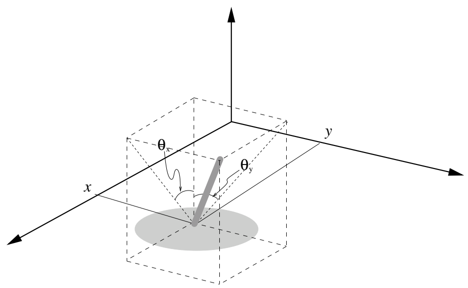
Emergence of Prediction in RNN?



Can prediction emerge in internal state dynamics?

- **Idea:** Test if (1) internal state dynamics is predictable in evolved recurrent nets, and (2) if that correlates with performance.

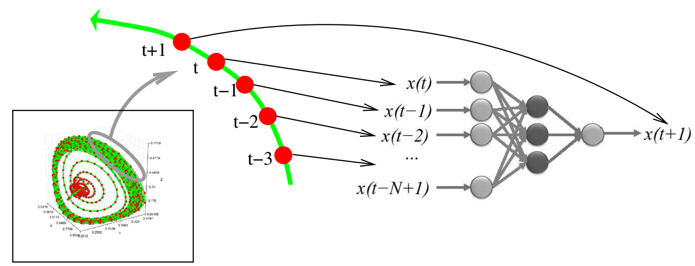
Task: 2D Pole Balancing



Anderson (1989)

- Standard 2D pole balancing problem.
- Keep pole upright, within square bounding region.
- Evolve recurrent neural network controllers.

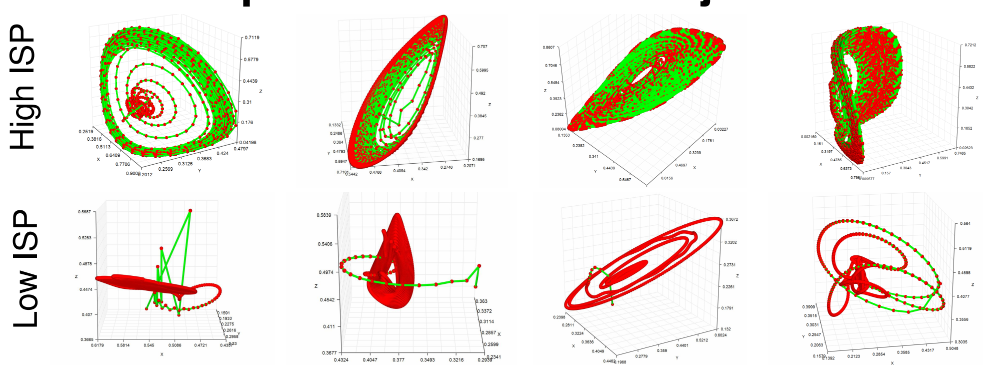
Measuring Predictability



- Train a simple feedforward network to predict the internal state trajectories.
- Measure prediction error made by the network.

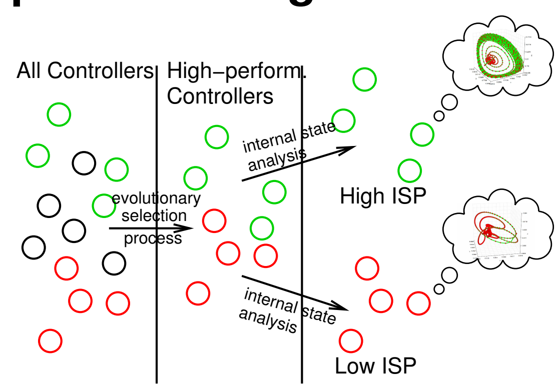
  → High vs. low internal state predictability (ISP)

Example Internal State Trajectories



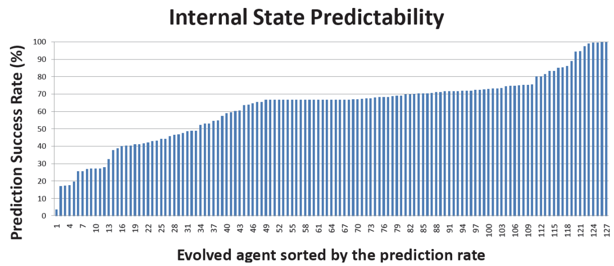
- Typical examples of high (top) and low (bottom) ISP.
- High ISP=predictable, Low ISP=unpredictable.
- Note: Both meet the same performance criterion!

Experiment: High vs. Low ISP



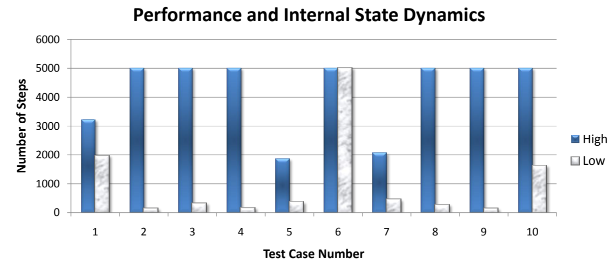
1. Train networks to achieve same performance mark.
2. Analyze internal state predictability (ISP).
3. Select top (High ISP) and bottom (Low ISP) five, and compare their performance in a harder task.

Results: Internal State Predictability (ISP)



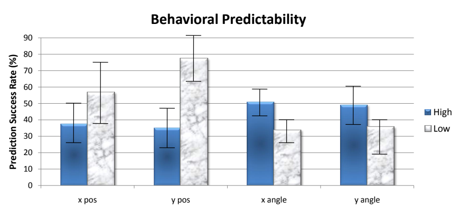
- Trained 130 pole balancing agents.
- Chose top 10 highest ISP agents and bottom 10 lowest ISP.
  - high ISPs: $\mu = 95.61\%$ and $\sigma = 5.55\%$.
  - low ISPs: $\mu = 31.74\%$ and $\sigma = 10.79\%$.

Performance and Int. State Dyn.



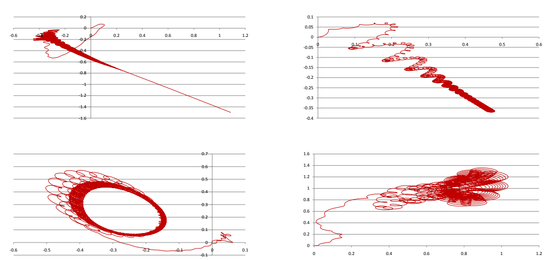
- Made the initial conditions in the 2D pole balancing task harsher.
- Performance of high- and low-ISP groups compared.
- High-ISP group outperforms the low-ISP group in the changed environment.

Behavioral Predictability



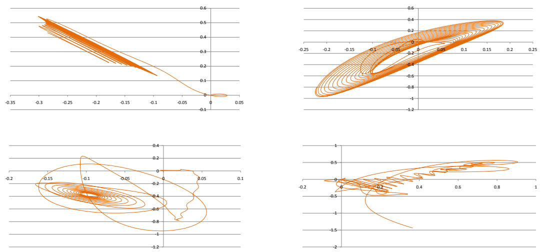
- Success of high-ISP group may simply be due to simpler behavioral trajectory.
- However, predictability in behavioral predictability is no different between high- and low-ISP groups.

Examples of cart x and y position from high ISP



- Behavioral trajectories of x and y positions show complex trajectories.

Examples of cart x and y position from low ISP



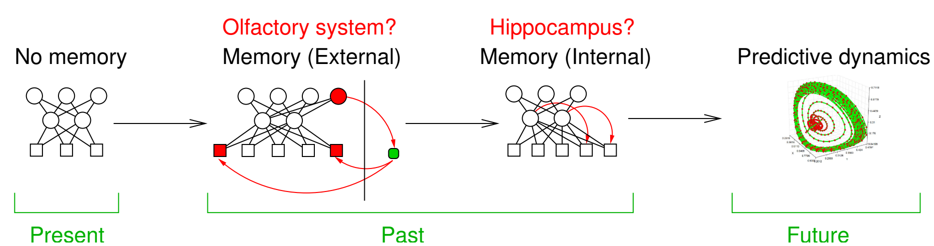
- Behavioral trajectories of x and y positions show complex trajectories.

Part II Summary

- Simulations show potential evolutionary advantage of predictive internal dynamics.
- Predictive internal dynamics could be a precondition for full-blown predictive capability.

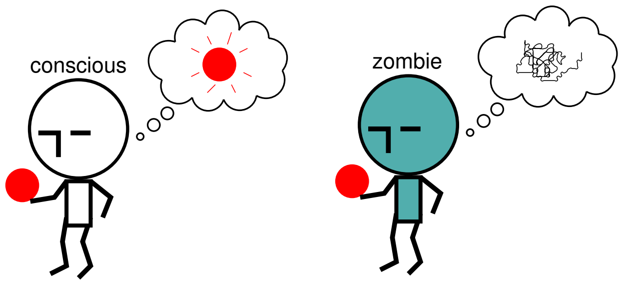
Wrap-Up

Discussion



- From external memory to internalized memory (cf. Rocha 1996).
- Analogous to olfactory vs. hippocampal function?
- Pheromones (external marker) vs. neuromodulators (internal marker)?

Discussion (cont'd) & Future Work



- Implications on the evolution of internal properties invisible to the process evolution.
- **Future work**: (1) actual evolution from dropper/detector net to recurrent net; (2) actual evolution of predictor that can utilize the predictable dynamics.

References

Anderson, C. W. (1989). Learning to control an inverted pendulum using neural networks. *IEEE Control Systems Magazine*, 9:31–37.

Beer, R. D. (2000). Dynamical approaches to cognitive science. *Trends in Cognitive Sciences*, 4:91–99.

Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, 7:195–225.

Kwon, J., and Choe, Y. (2008). Internal state predictability as an evolutionary precursor of self-awareness and agency. In *Proceedings of the Seventh International Conference on Development and Learning*, 109–114. IEEE.

Rocha, L. M. (1996). Eigenbehavior and symbols. *Systems Research*, 13:371–384.

Swanson, L. W. (2003). *Brain Architecture: Understanding the Basic Plan.* Oxford: Oxford University Press.

Ward, R., and Ward, R. (2006). 2006 special issue: Cognitive conflict without explicit conflict monitoring in a dynamical agent. *Neural Networks*, 19(9):1430–1436.

Conclusion

From reactive to contemplative to predictive:

- **Recollection**: External material interaction can be a low-cost intermediate step toward recurrent architecture.
- **Prediction**: Predictable internal state dynamics in recurrent neural nets can have an evolutionary edge, thus prediction can and will evolve.