Automatic Evolution of Multimodal Behavior with Multi-Brain HyperNEAT

JACOB SCHRUM
DEPARTMENT OF MATH AND COMPUTER SCIENCE
SOUTHWESTERN UNIVERSITY
GEORGETOWN, TX 78626 USA
SCHRUM2@SOUTHWESTERN.EDU

JOEL LEHMAN AND SEBASTIAN RISI
CENTER FOR COMPUTER GAMES RESEARCH
IT UNIVERSITY OF COPENHAGEN
COPENHAGEN, DENMARK
{JLEH, SEBR}@ITU.DK

IT University of Copenhagen

1 Multimodal Behavior

- Complex domains require multimodal behavior.
- Multimodal behavior is when agents switch between distinct policies based on environmental context.
- Multi-Brain HyperNEAT (MB-HyperNEAT) is introduced to evolve multimodal behavior.

2 Background

Multi-Brain HyperNEAT combines and extends existing algorithms.

- **Modular Multiobjective NEAT (MM-NEAT [3]):**
  - Direct encoding for neural networks.
  - Networks can have multiple output modules.
  - Each module can represent a different mode of behavior.
  - Module arbitration can be human-specified, or discovered by evolution.
- **Hypercube-based Neuro-Evolution of Augmenting Topologies (HyperNEAT [4]):**
  - Indirect encoding for neural networks.
  - Evolves Compositional Pattern Producing Networks (CPPNs).
  - Paints controller networks in substrate coordinate space (Figure 1a).
- **Situational policy geometry [1]:**
  - HyperNEAT extension that gives agents multiple brains.
  - Brain arbitration depends on human-specified policy.
  - Brains embedded along situation dimension in substrate (Figure 1b).
  - Situation dimension is a restriction that MB-HyperNEAT overcomes.

3 New Approaches Using Multiple Brains

MB-HyperNEAT extends HyperNEAT in three ways. Each idea is inspired by the direct-encoded MM-NEAT approach.

- **Multitask CPPNs:**
  - CPPNs with multiple output modules (Figure 1c).
  - Each module paints a different brain in a different substrate.
  - Human must still designate number and usage of brains.
- **Preference neurons:**
  - Each substrate brain possesses a preference neuron.
  - One brain used per time step: brain with highest preference output.
  - Preference neuron links painted by special CPPN outputs (Figure 1d).
  - Evolution determines when each brain is used.
- **Module mutation:**
  - Structural mutation that adds a module (Figure 2).
  - Evolution determines how many brains an agent has.
  - Brain arbitration still depends on preference neurons.

Module Mutation

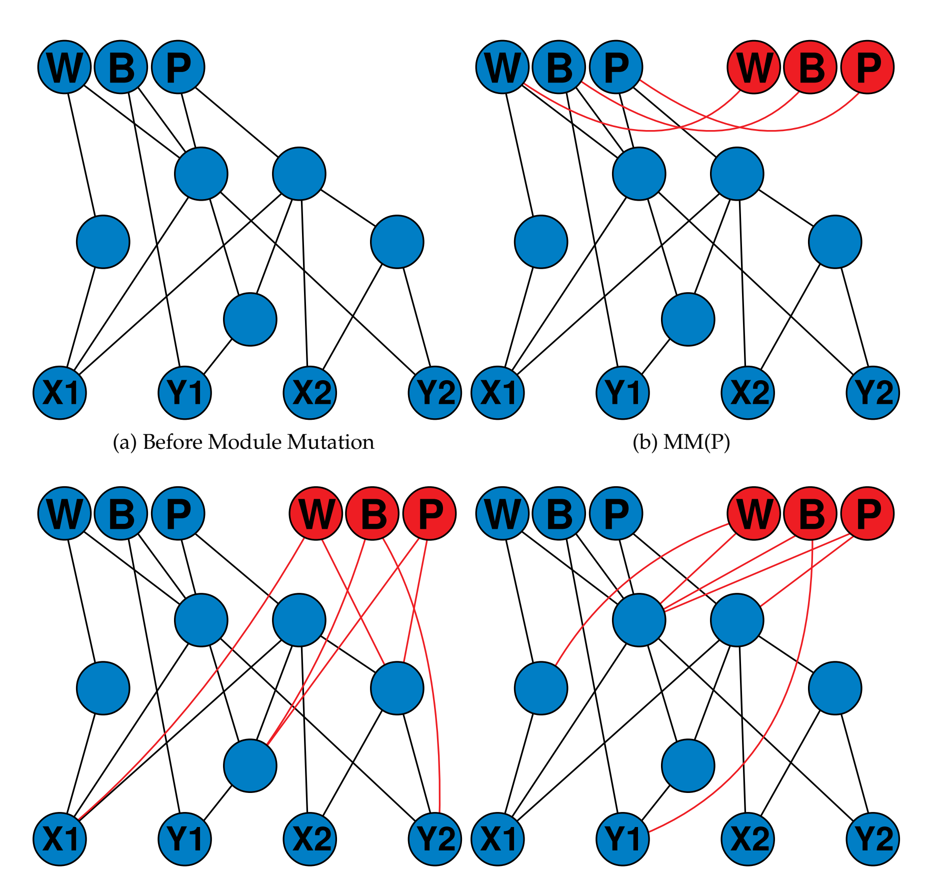


Figure 2: Three Types of Module Mutation.

- (a) Before module mutation: each CPPN has a single module.
- (b) MM(P), for previous: new module has lateral connections from a randomly chosen previous module.
- (c) MM(R), for random: new module has connections from random source neurons with random synaptic weights.
- (d) MM(D), for duplicate: new module duplicates the behavior of a randomly chosen previous module by copying its incoming links.

Generating Substrate Brains

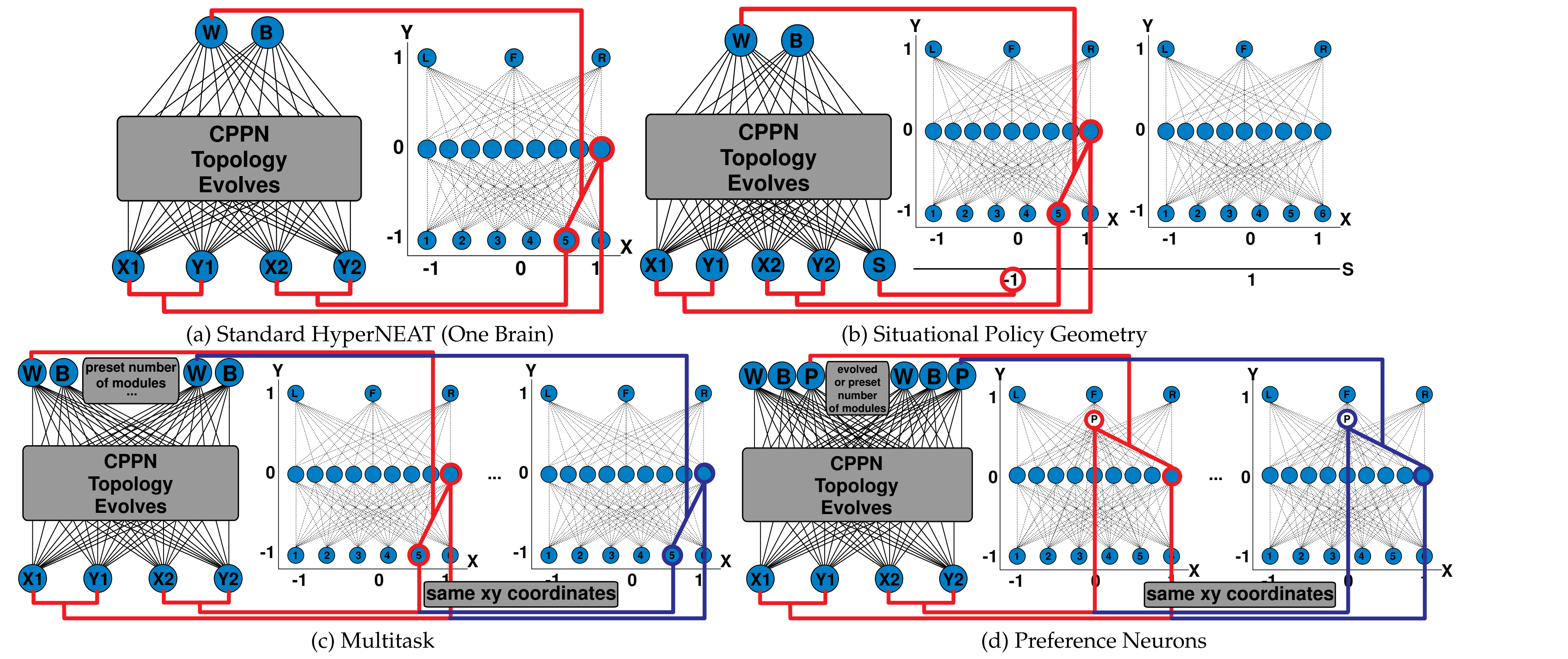
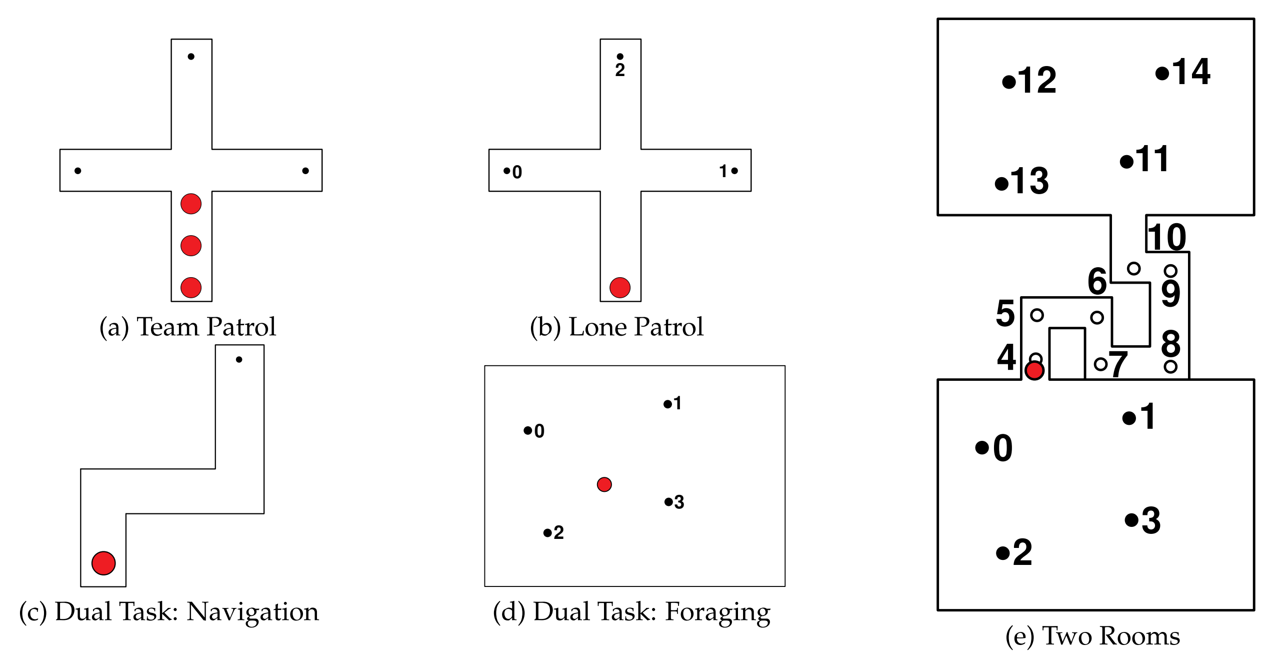


Figure 1: Methods for Generating Substrate Brains. (a) A **standard HyperNEAT CPPN** creates a single-brain substrate [4]. For each possible connection, the xy-coordinates of neurons are input into the CPPN. The W output determines its weight (B determines a fixed bias). (b) **Situational policy geometry [1]** gives CPPNs an extra S input. There is a separate brain for each value of S the CPPN is queried with. The next two approaches are new to this work: (c) A **multitask CPPN** has a group of outputs for each brain (no S dimension). When the CPPN is queried, each output module supplies the corresponding connection weight for a different brain. (d) **Preference neuron CPPNs** add a P neuron to each output module that is used only when defining preference neuron link weights. The preference neuron (white) of each brain has potential connections to all neurons of the hidden and input layers. All brains are activated on each time step, but only the one with the highest preference output matters.

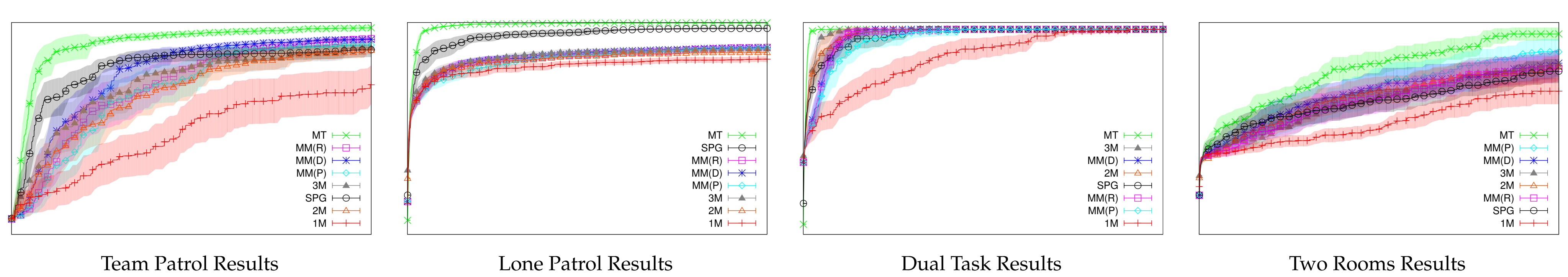
4 Experimental Domains

Agents are evolved in four domains.

- **(a) Team patrol [1]):**
  - Three robots (red circles) advance to visit different waypoints (black dots).
  - Robots return to start when recalled.
- **(b) Lone patrol:**
  - New domain to this work.
  - Single robot visits each waypoint in order.
- **(c)-(d) Dual task [2]:**
  - Domain consists of two environments.
  - Navigation requires robot to traverse hallway.
  - Foraging requires robot to visit all waypoints.
- **(d) Two rooms:**
  - New domain to this work.
  - Combines hallway navigation and foraging.
  - Robot must visit each waypoint in order.
  - Hollow pellets are invisible breadcrumbs.



5 Results



Experimental Results Across Domains. Average champion fitness by generation (2,000 generations) across 30 runs of evolution for each approach in each domain. Transparent regions show 95% confidence intervals. Having multiple brains leads to better levels of performance faster.

6 Conclusions

- Multitask CPPNs are better than all other approaches in each domain.
- Multitask CPPNs are significantly better than situational policy geometry, thus multiple brains without geometric constraints are superior.
- Preference neurons beat standard single-brain HyperNEAT, and sometimes tie with situational policy geometry.
- Preference neurons do not depend on a human-specified task division, thus require less knowledge than situational policy geometry.
- Multi-Brain HyperNEAT is a promising set of tools with potential applications in challenging domains requiring multimodal behavior.

7 More Information

Movies of evolved behavior, source code, and a full-length paper presenting this research are all available online at:
`http://southwestern.edu/~schrum2/re/mb-hyperneat.html`

References

[1] D. B. D'Ambrosio, J. Lehman, S. Risi, and K. O. Stanley. Task Switching in Multirobot Learning Through Indirect Encoding. In *International Conference on Intelligent Robots and Systems*, 2011.

[2] S. Risi and K. O. Stanley. An Enhanced Hypercube-Based Encoding for Evolving the Placement, Density and Connectivity of Neurons. *Artificial Life*, 18(4):331–363, 2012.

[3] J. Schrum and R. Miikkulainen. Discovering Multimodal Behavior in Ms. Pac-Man through Evolution of Modular Neural Networks. *IEEE Transactions on Computational Intelligence and AI in Games*, 8(1):67–81, 2016.

[4] K. O. Stanley, D. B. D'Ambrosio, and J. Gauci. A Hypercube-based Encoding for Evolving Large-scale Neural Networks. *Artificial Life*, 15(2):185–212, 2009.