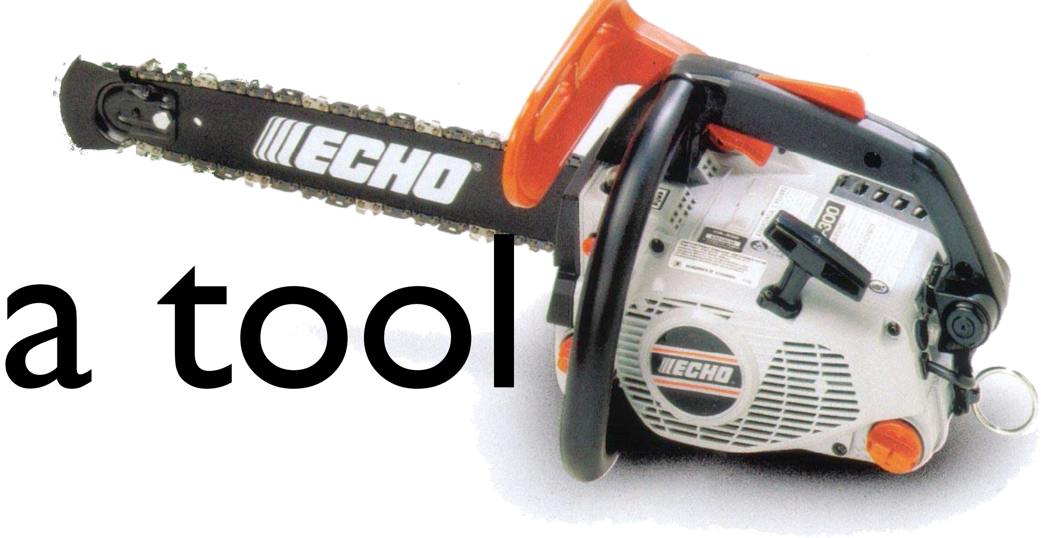
The role of belief updating in human sequential decision making with uncertainty

Brian J. Stankiewicz
Department of Psychology
Center for Perceptual Systems
University of Texas, Austin

Confessions

Bayes as a tool

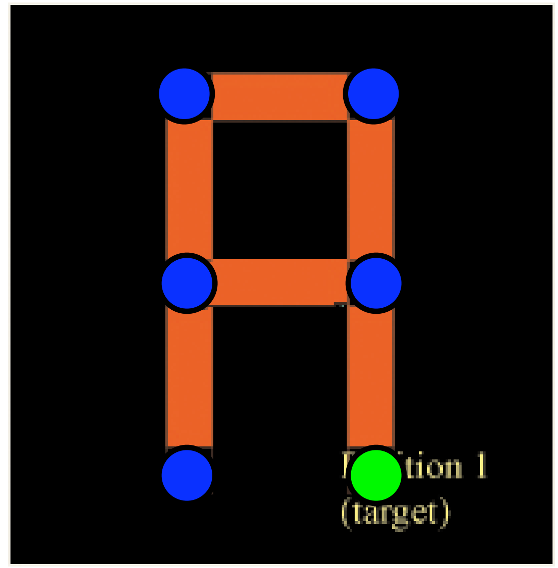


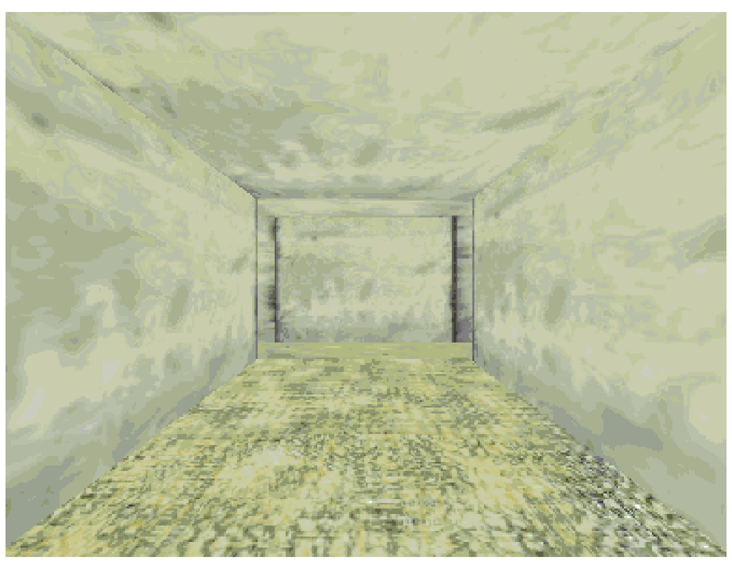
- Provides a vocabulary for understanding/describing a task.
  - What should a rational system do?
- Provides a benchmark assuming no processing inefficiencies or resource limitations.
  - If humans are optimal, assume an understanding of processes and resources for that task
  - If not optimal, localize how/why people are sub-optimal

Sequential Decision Making with Uncertainty

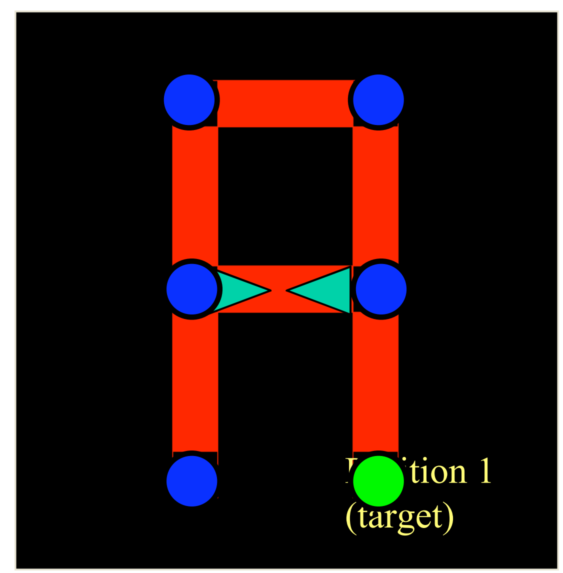
- Examples
  - Medical diagnosis
  - Localization in space
  - Scientific exploration
  - etc.
- Maximize reward given hidden state

Sequential Decision Making with Uncertainty





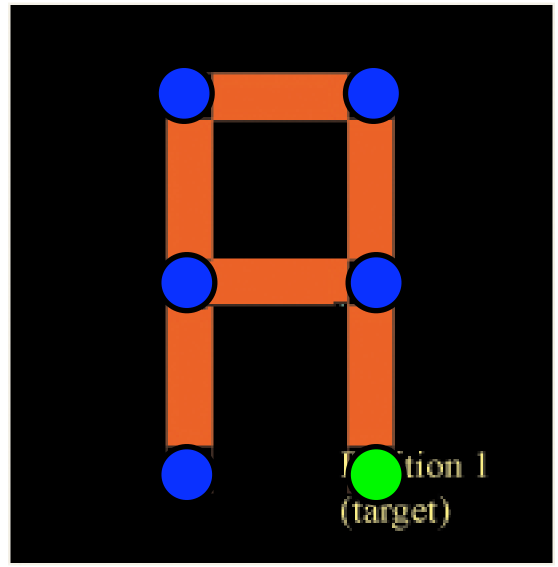
Observation



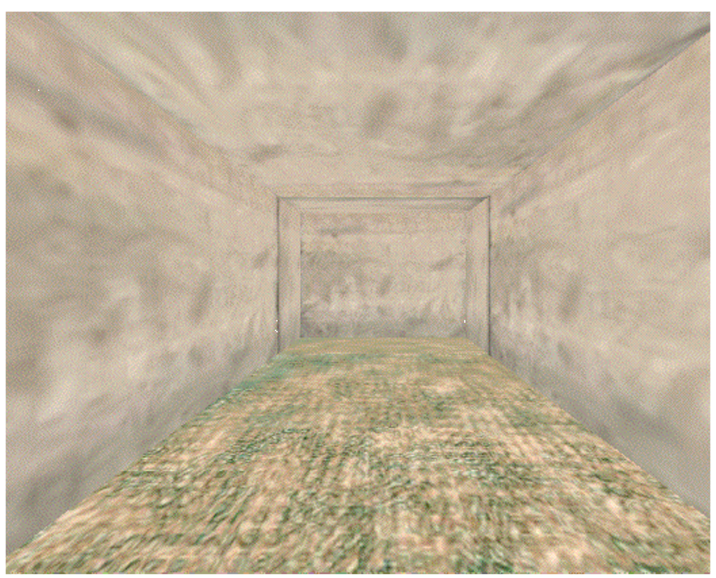
Belief

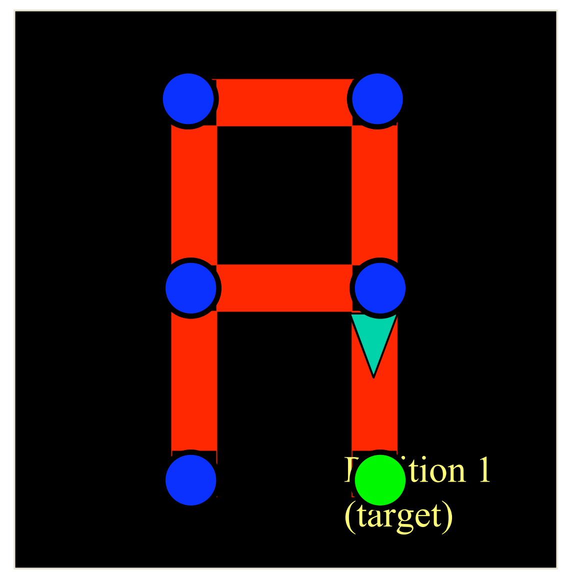
"Rotate Left"

Sequential Decision Making with Uncertainty



Next View





Observation

Belief

Structure of SDMU Task

- States: S
- Actions: A=<“Left”, “Right”, “Forward”>
- Observations: O=<T-Junction, L-Junction, etc.>
- Cost/Rewards: C=<$5,$200,...,$2,000>

Partially Observable Markov Decision Process

Transition function: $p(s'|s,a)$

Observation function: $p(o|s,a)$

Belief Updating: $b_t(s') = p(s'|b_{t-1}, a, o)$

$$b_t(s') = \frac{p(o|s',a)\sum_s p(s'|s,a)b_{t-1}(s)}{p(o|a,b_{t-1})}$$

Computing Reward $Reward(a,s)$

Maximize the expected accumulated reward over time.

Partially Observable Markov Decision Process

References

[1] A. R. Cassandra. *Exact and Approximate Algorithms for Partially Observable Markov Decision Processes.* PhD thesis, Department of Computer Science, Brown University, May 1998.

[2] A. R. Cassandra, M. L. Littman, and N. L. Zhang. Incremental pruning: A simple, fast, exact method for partially observable Markov decision processes. In D. Geiger and P. P. Shenoy, editors, *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–97)*, pages 54–61, San Francisco, CA, 1997. Morgan Kaufmann Publishers.

[3] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[4] E. Sondik. *The Optimal Control of Partially Observable Markov Decision Processes.* Ph.d. thesis, Stanford University, 1971.

Spatial Navigation

- Questions:
  - 1. How efficiently do people navigate with uncertainty in a known environment?
  - 2. If inefficient, determine why inefficient
    - a. Observations
    - b. Transition function
    - c. Belief updating
    - d. Decision strategy

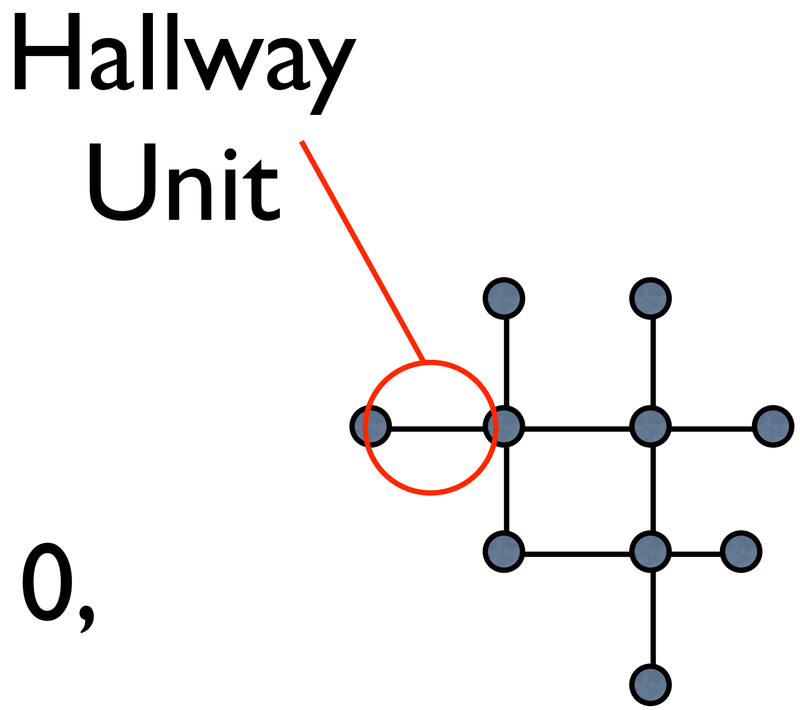
Spatial Navigation

- Develop a task that would be sensitive to all factors involved.
- Indoor virtual reality navigation task in which subjects navigate from unspecified state to a goal state.
- Vary the complexity/size of the environment

Experiment 1

What is the effect of layout complexity/size on human navigation performance?

Environments composed of 10, 20, 40 and 80 hallway units.



Quantifying the task

- Simplify Environments
  - Indoor Environments (based on Cartesian grid)
- Simplify Visual Information (observations)
  - No "Landmark Information"
  - Structural Cues (e.g., T-junctions, L-junctions, etc.)
- Simplify State Space
  - Quantize space into discrete set of states
    - Finite set of Locations and Orientations
- Simplify Actions
  - Move Forward
  - Rotate Right 90°
  - Rotate Left 90°

Simplify Visual Information & Actions

- Desktop VR
- Visually Sparse
  - Perceptual Aliasing
- Actions made by keypress
  - Forward 1 Hallway Unit
  - Rotate Left 90°
  - Rotate Right 90°

General Procedure

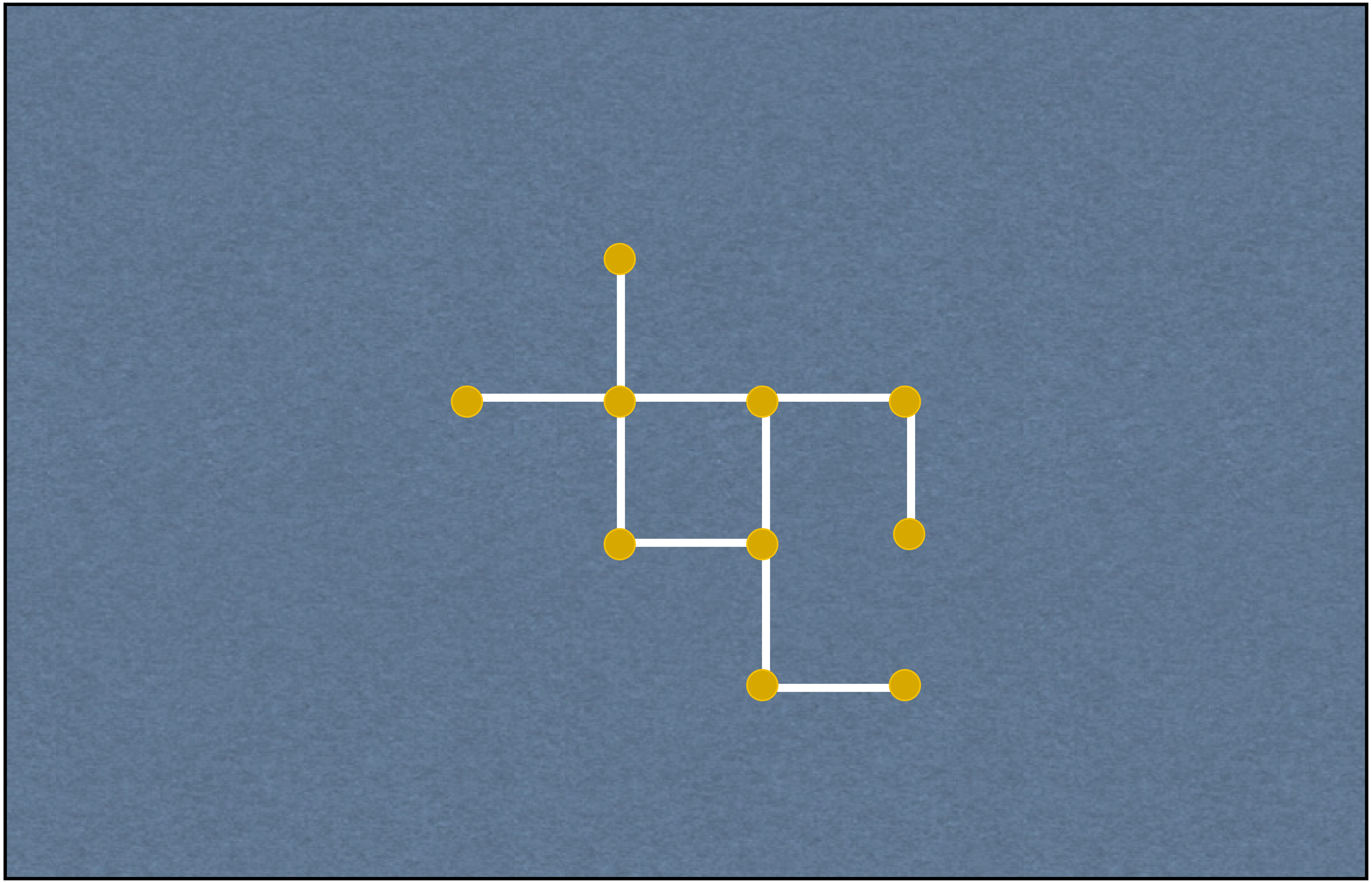
Training (Human):

- Freely explore environment for limited time (3 minutes)
- 1 Target Location in environment
  - Specified by auditory signal
- Draw environment on blank grid.
- Repeat exploration and drawing until map drawn correctly twice in a row.

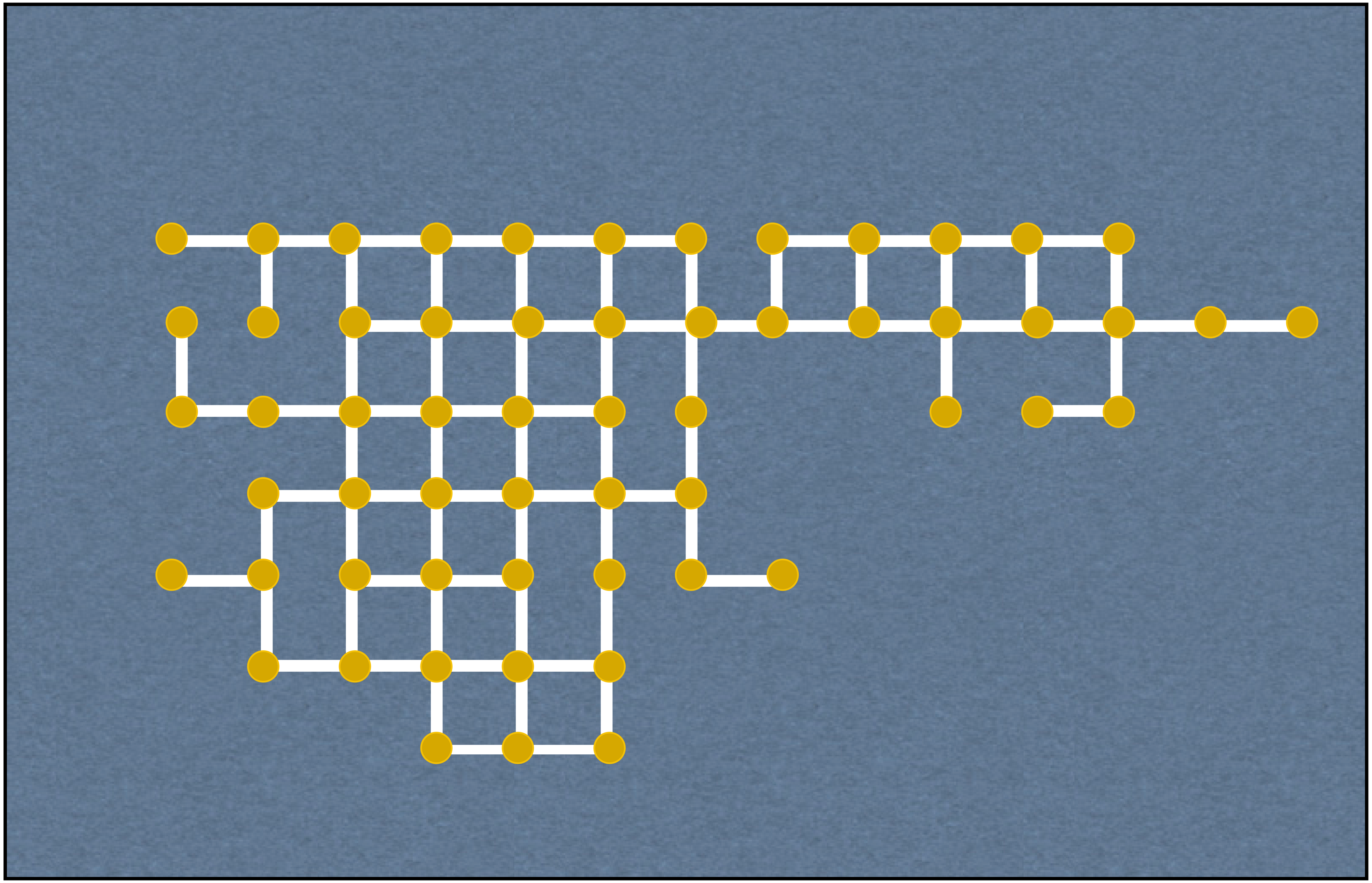
Test (Human and Ideal):

- Start from random location in environment
- Move to target location using as few actions as possible
- Indicate when they reached target location
- Measure number of actions to reach target

Number Hallway Units = 10



Number Hallway Units = 80



Experiment 1: Design

3 Subjects

2 Environments / Condition

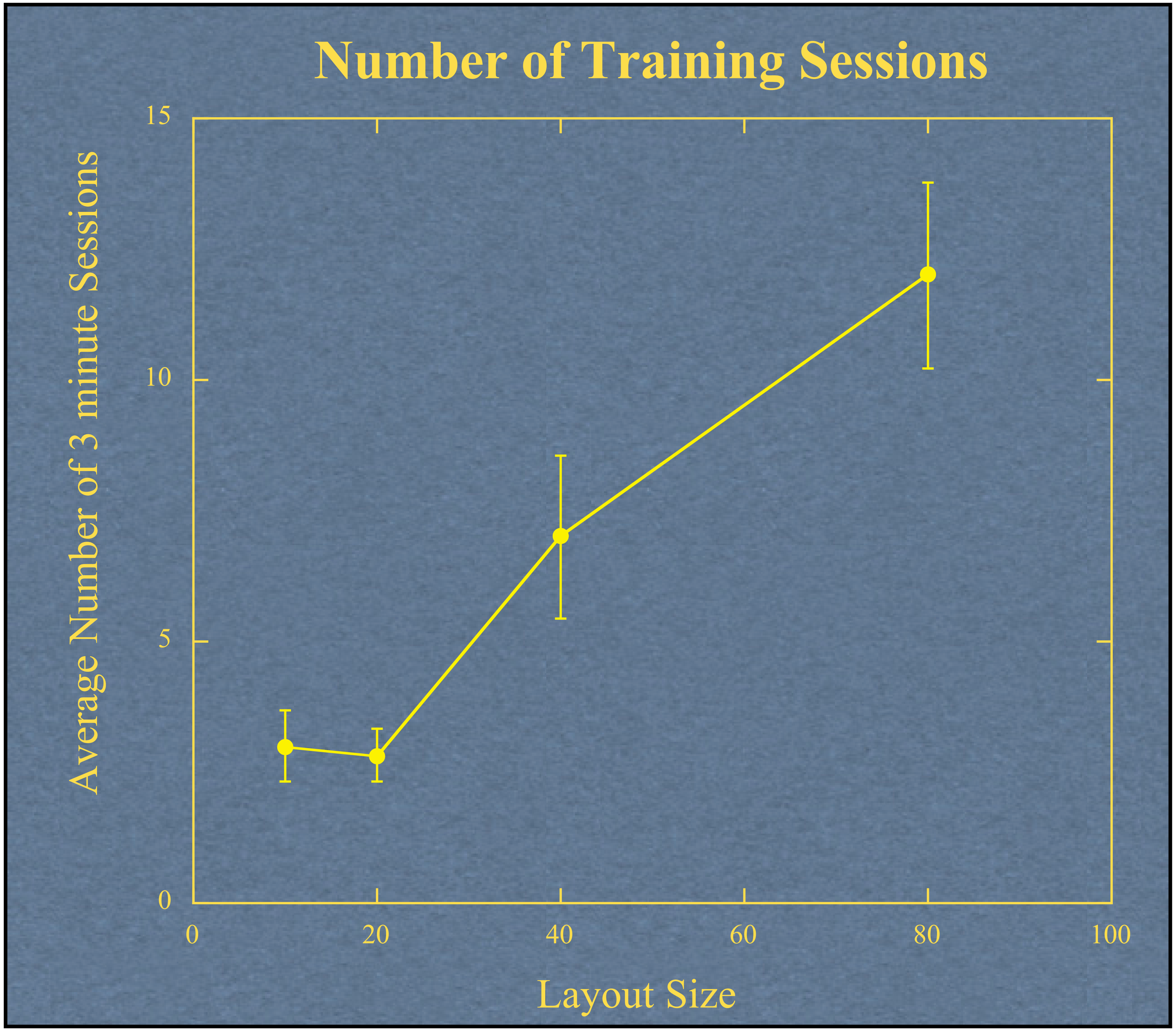
Each subject ran in each environment.

- Random order

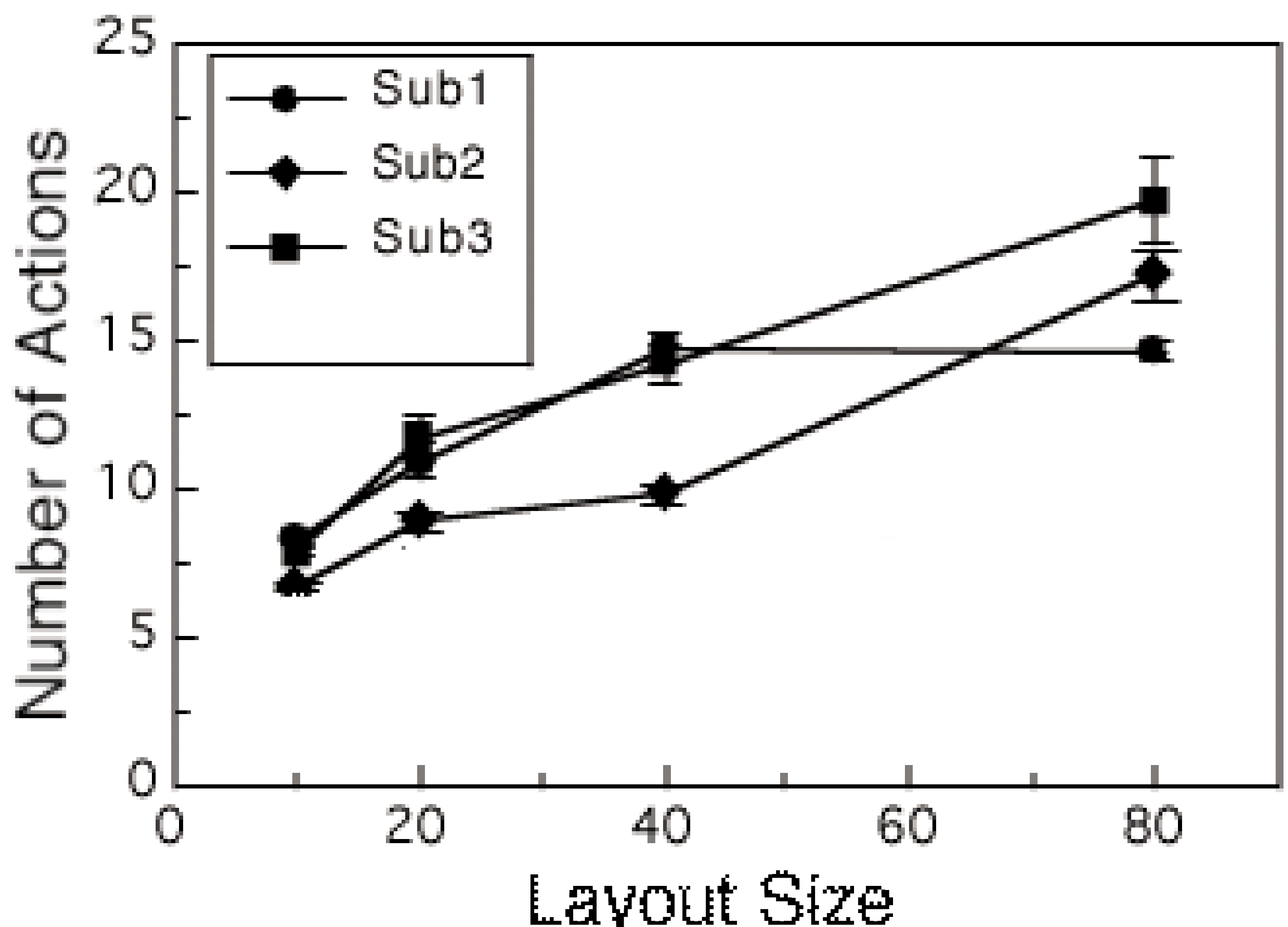
Started from each position equal number of times.

- Ran multiple trials from each position
- Equal number of trials / condition

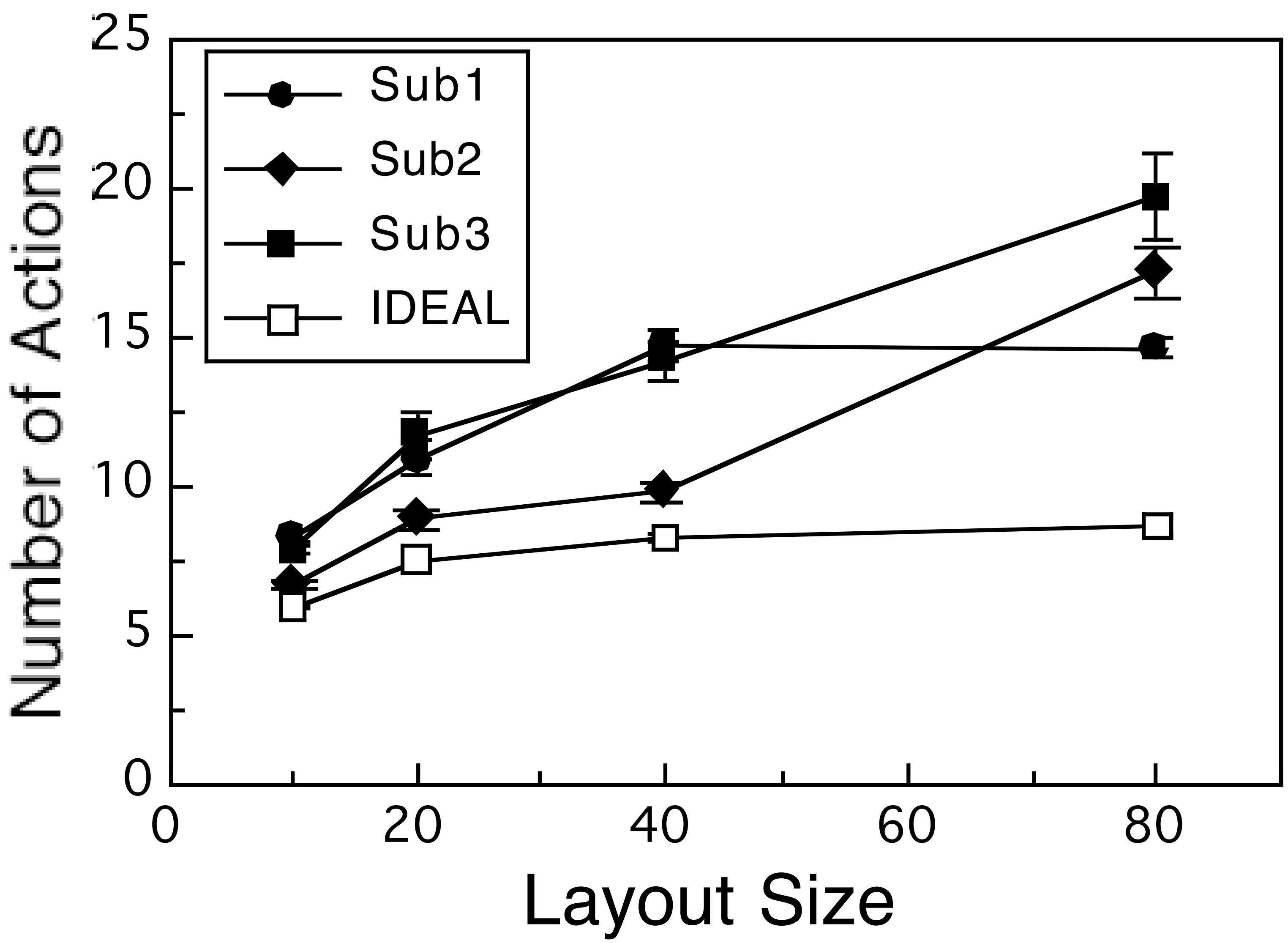
Training Sessions



Number of Actions as a Function Of Layout Size



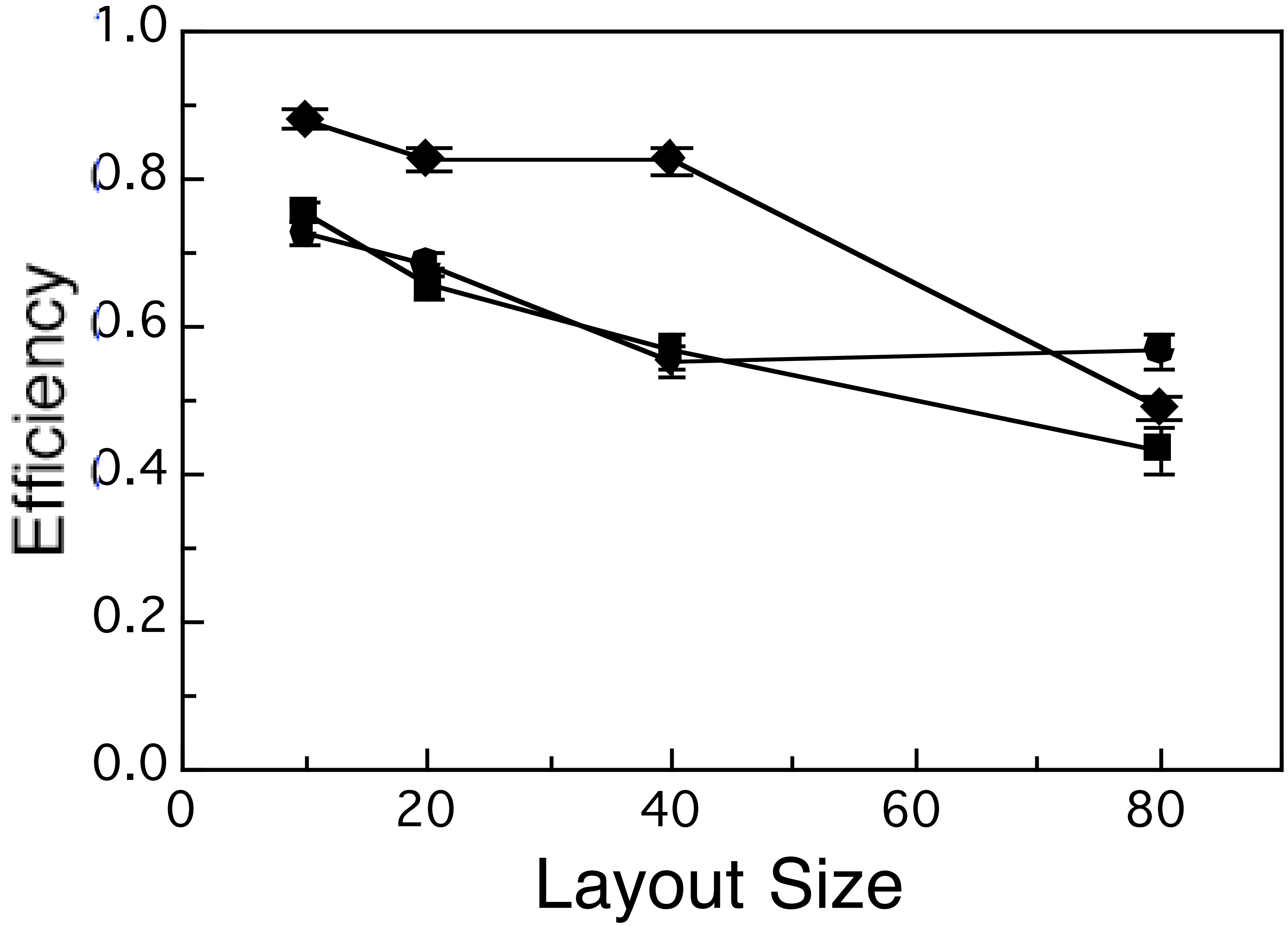
Number of Actions as a Function Of Layout Size



Experiment 1: Analysis

$$\text{"Efficiency"} = \frac{\text{Number of moves Ideal}}{\text{Number of moves Human}}$$

Action Efficiency



Experiment 1: Summary

Clear Effect of Layout Size on Efficiency.

Where is the inefficiency?

- Experiment 2
  - Inefficient visual processing (observations)?
- Experiment 3
  - Accessing the Cog. Map (transition function)
  - Belief Vector Generation (belief updating)
  - Decision Strate

Experiment 2

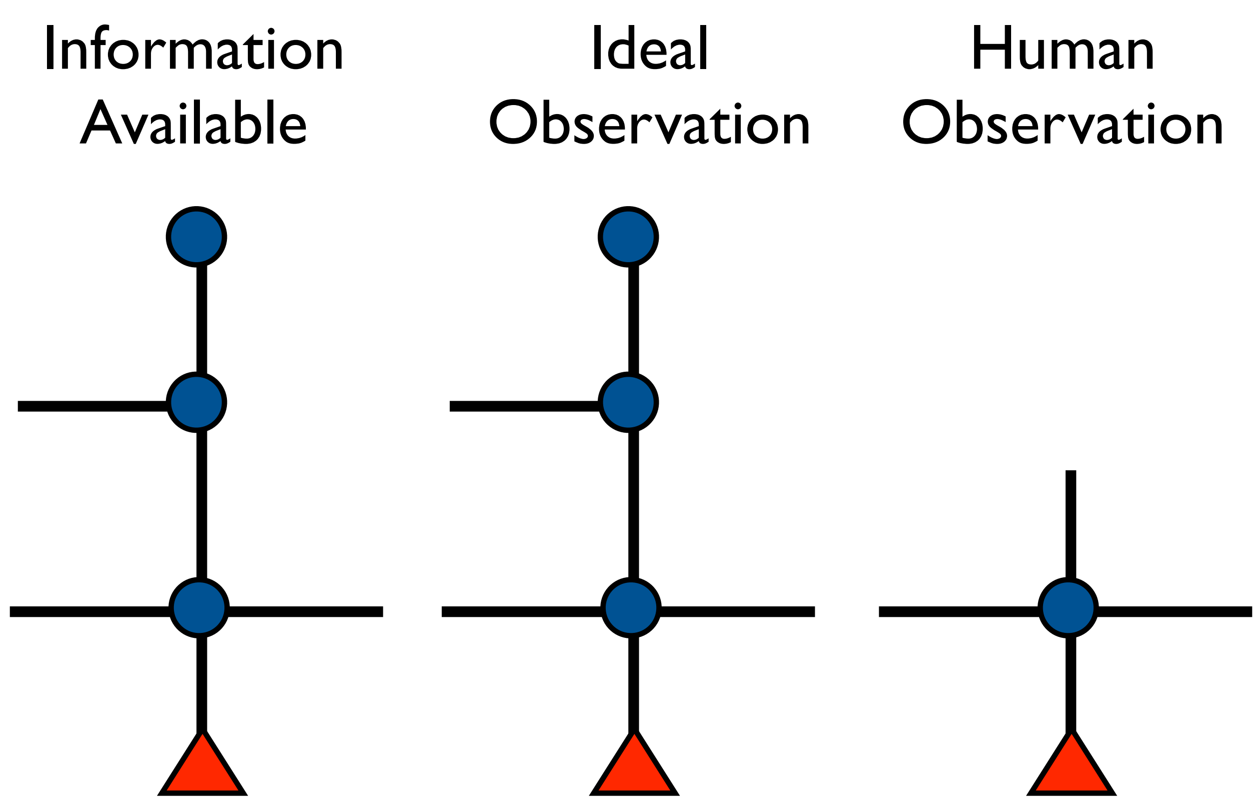
- Can inefficient visual processing explain inefficiencies?
  - Manipulate the amount of visual information available to the human observer.

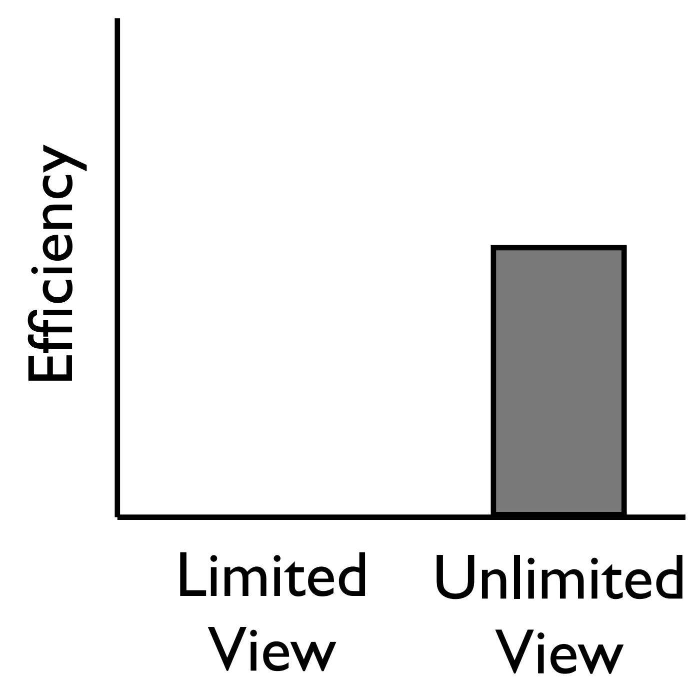
Experiment 2: Limited View

Added virtual fog

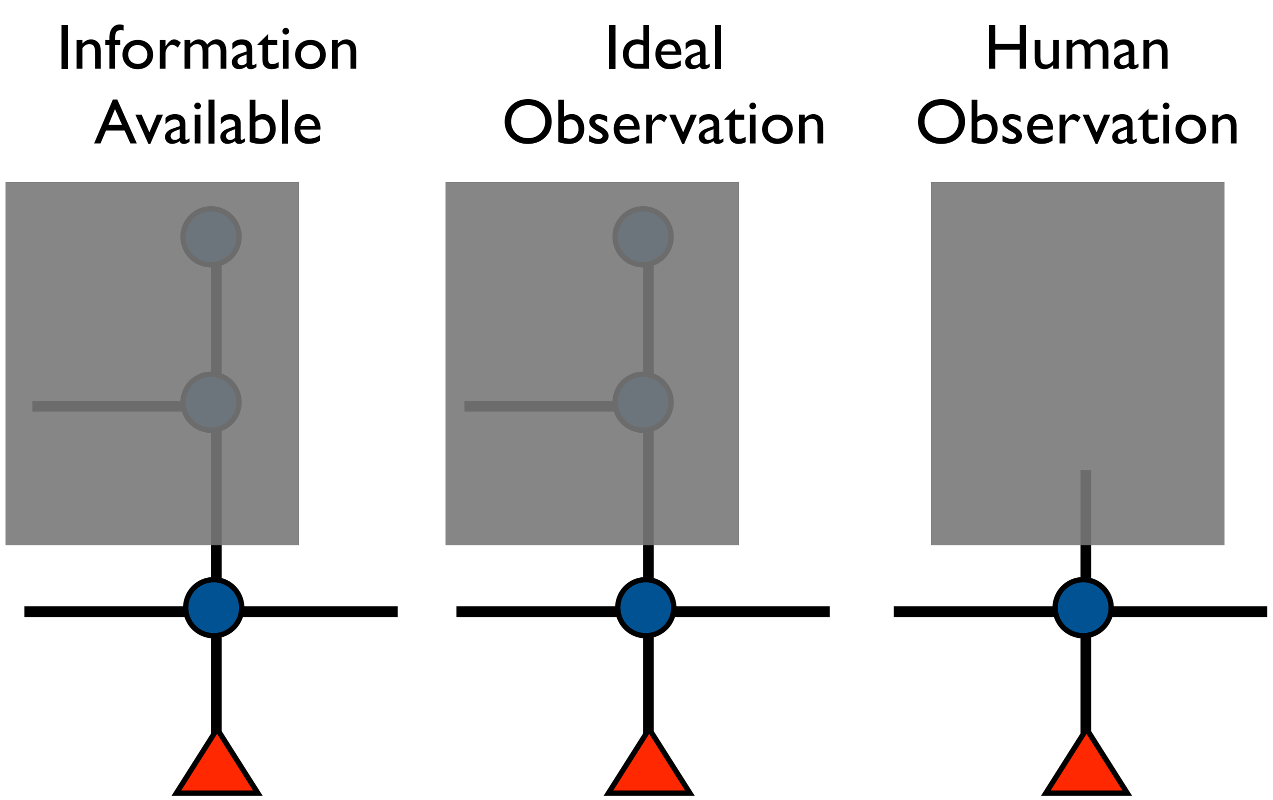
Only able to see down 1 hallway unit

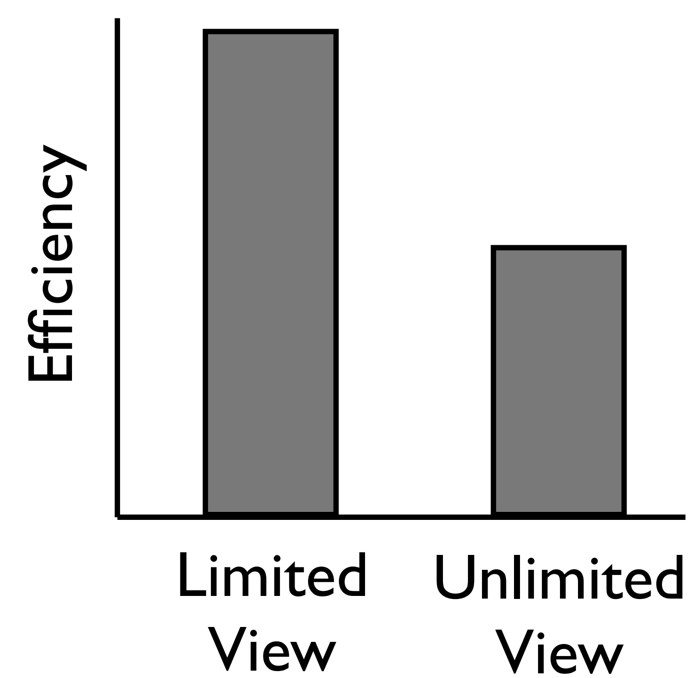
Predictions: Inefficient VP



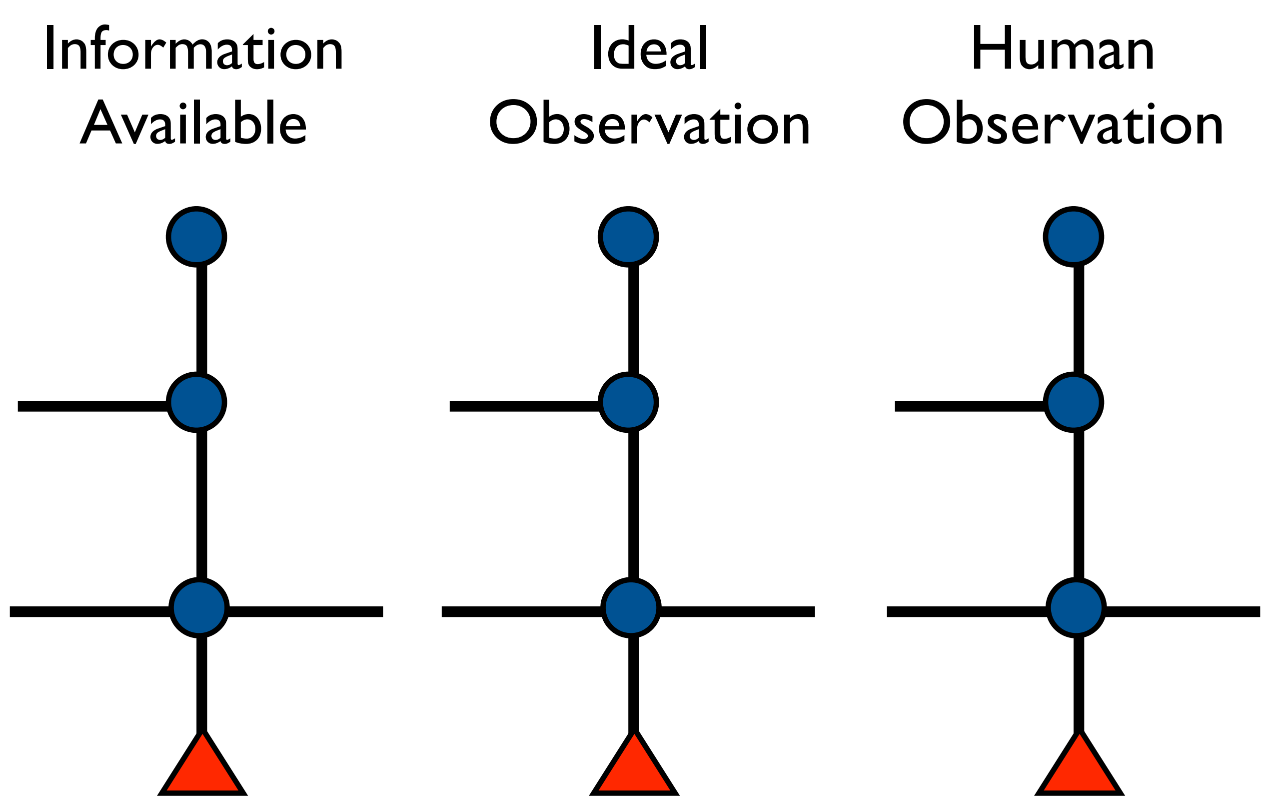


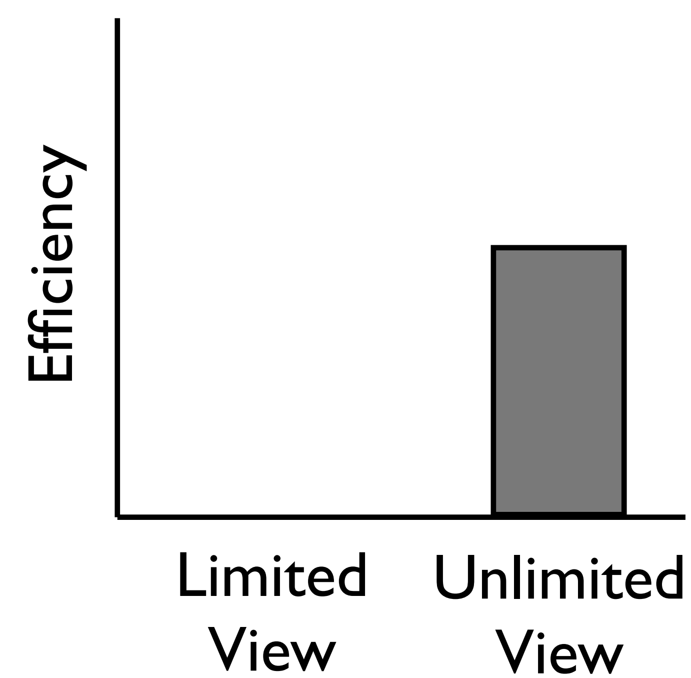
Predictions: Inefficient VP



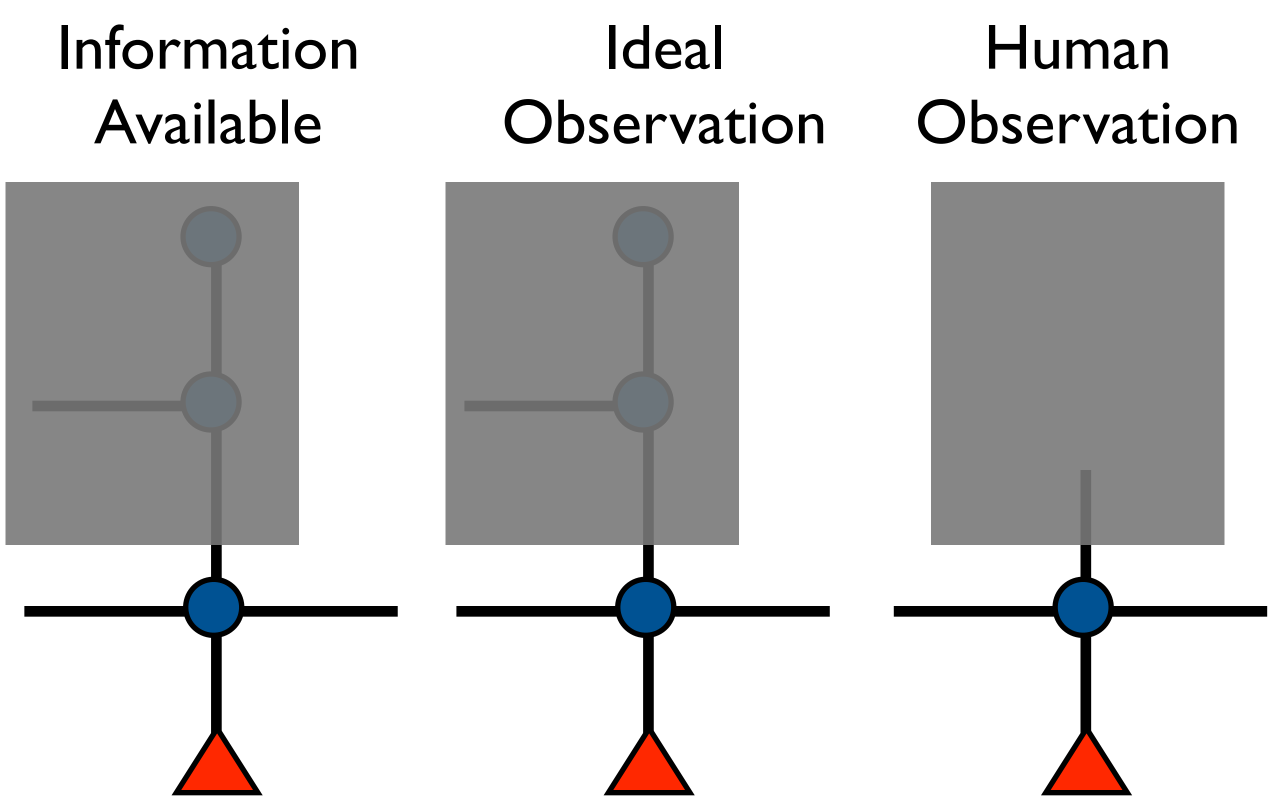


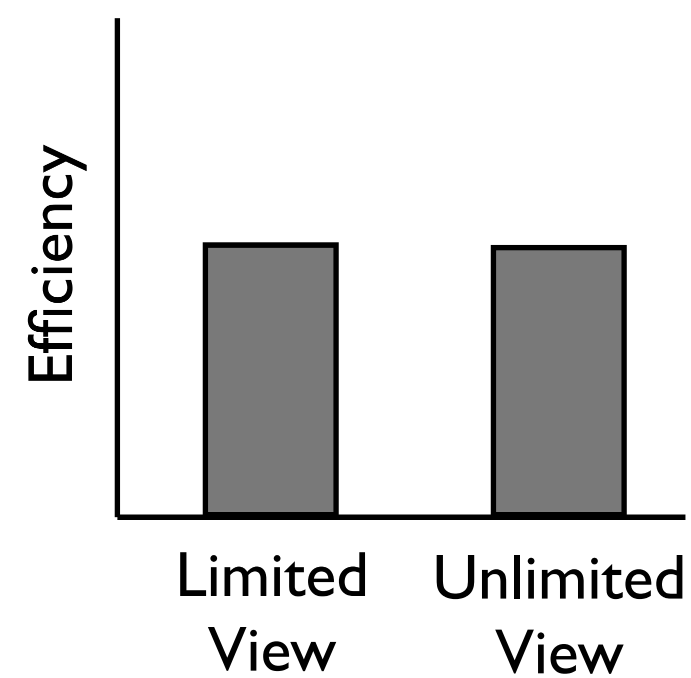
Predictions: Efficient VP





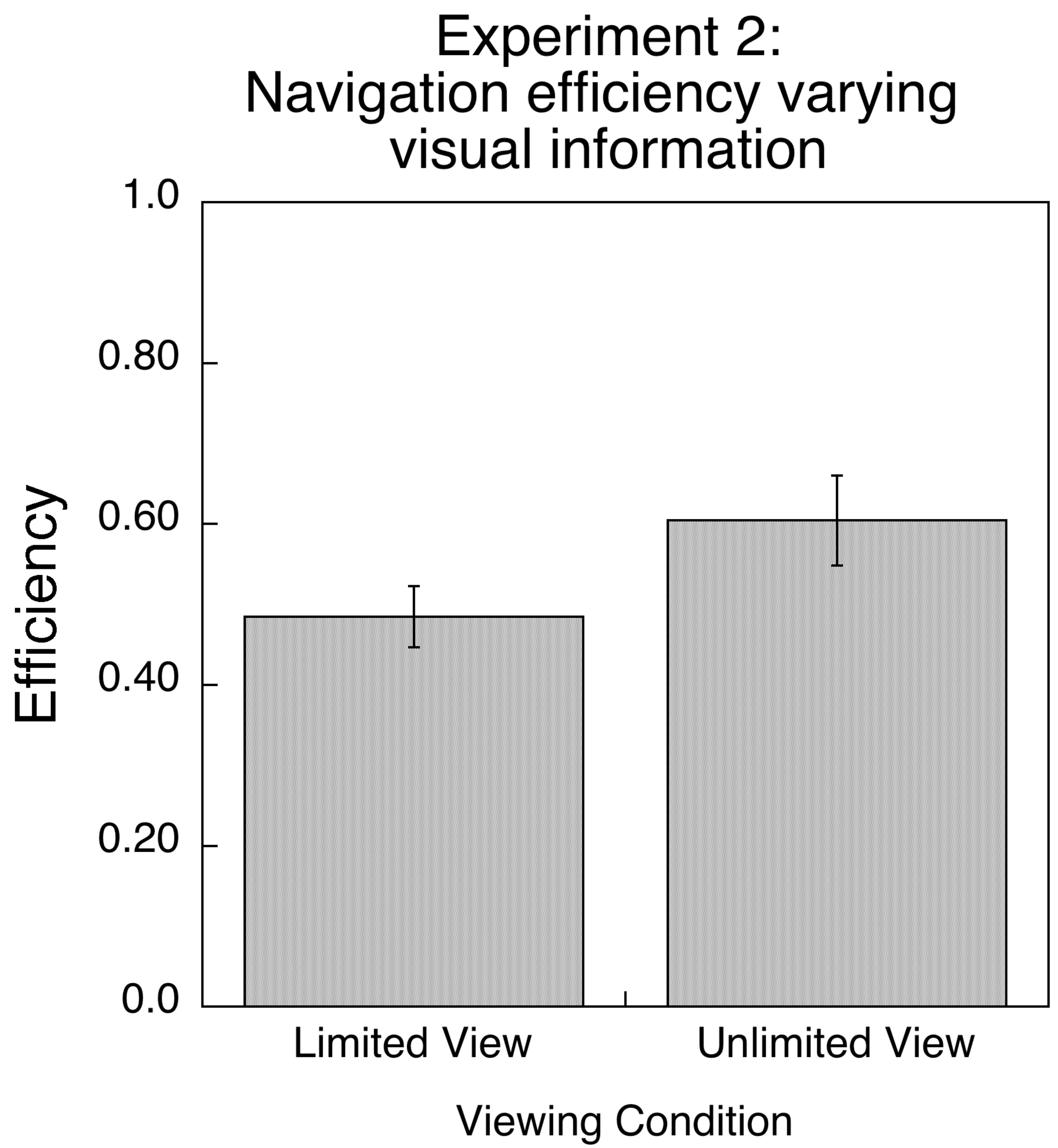
Predictions: Efficient VP





Experiment 2: Design

- 6 Subjects
  - Normal or corrected vision
- Ran in two different environments
  - Limited and Unlimited views
- Started from every state in the environment 6 times
  - 960 trials / environment


Experiment 2:
Navigation efficiency varying
visual information
1.0
0.80
0.60
0.40
0.20
0.0
Efficiency
Limited View
Unlimited View
Viewing Condition

Experiment 2: Summary

Does inefficient visual processing explain the inefficiencies found in Experiment 1?

No

Experiment 2:Discussion

~~Perception~~

Accessing cognitive map

- Transition Function

Spatial Updating

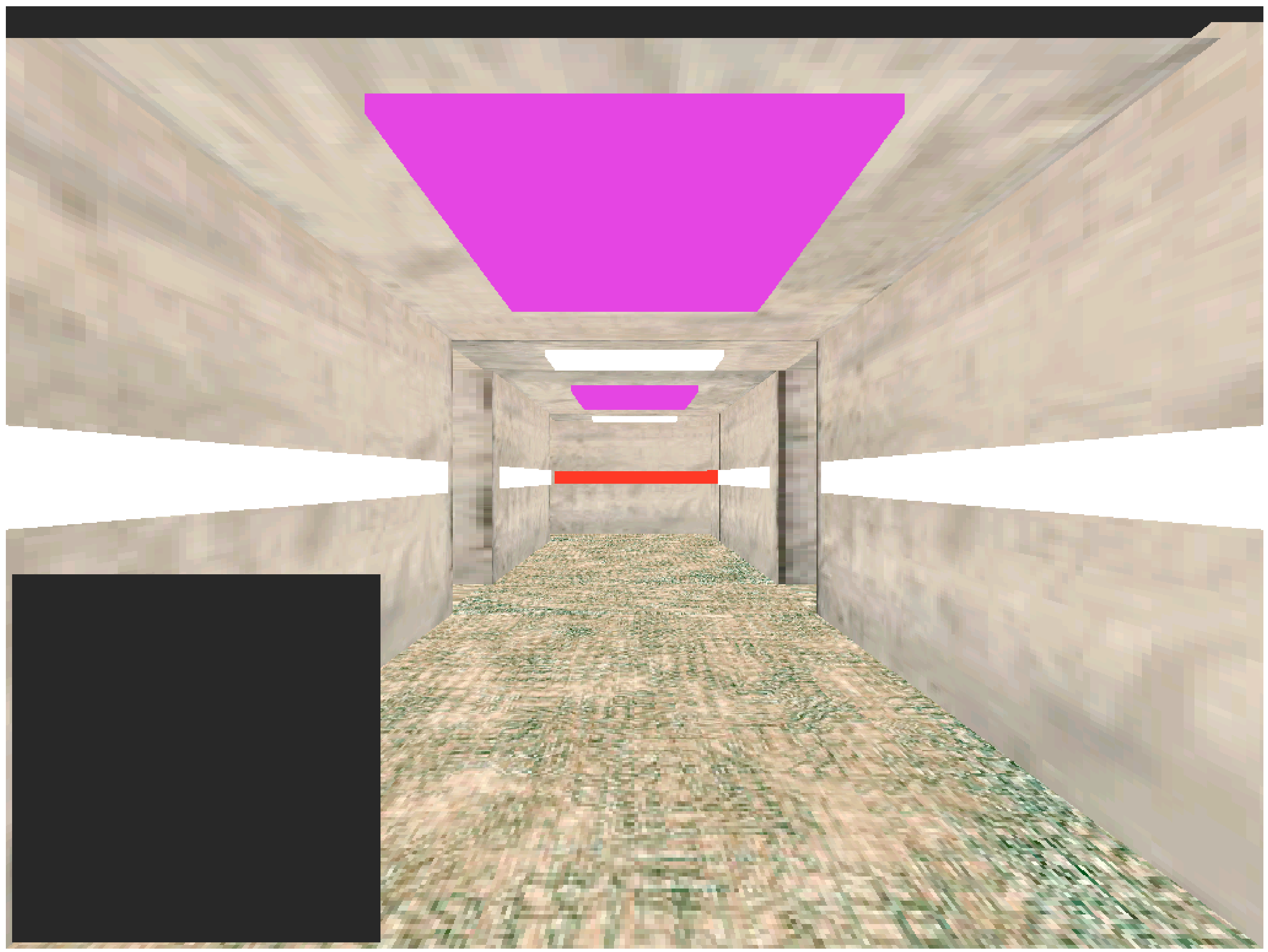
- Belief updating

Decision Strategy

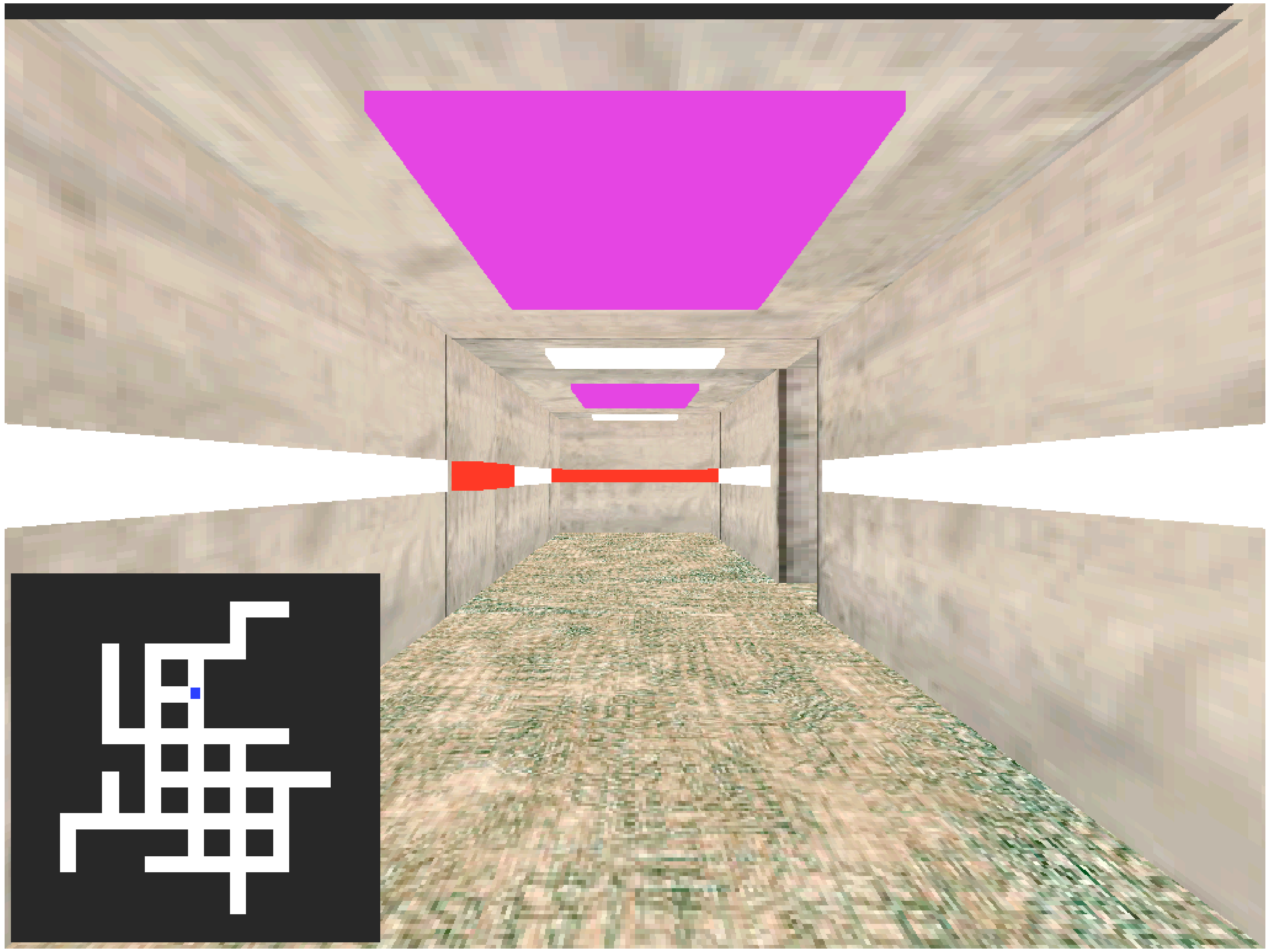
Experiment 3: Procedure

- Training
  - Same as Experiment 1
- Test
  - Same as Experiment 1 with an additional map display
  - Three map conditions

Sample Display (No Map)

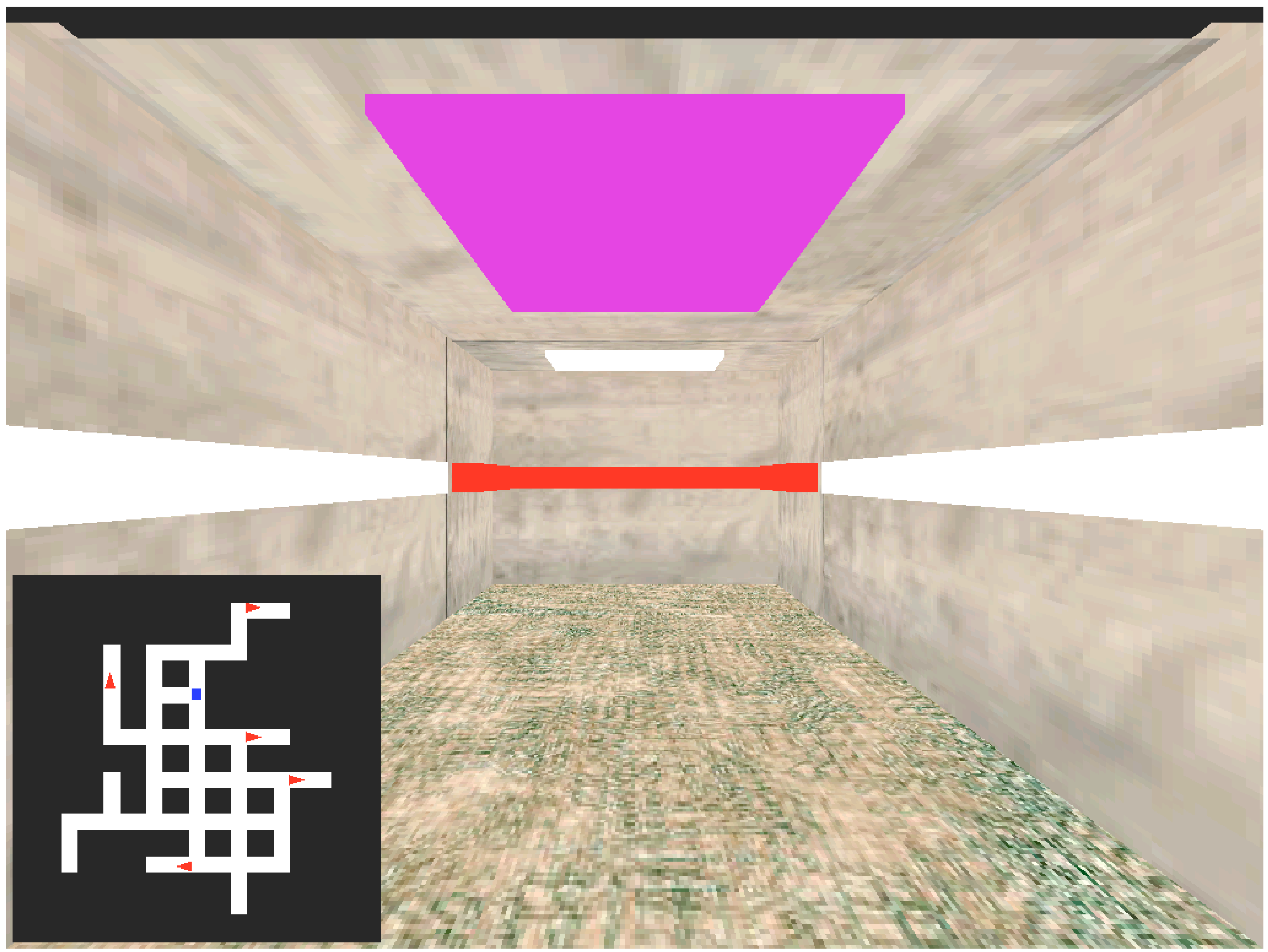


Sample Display (Map)

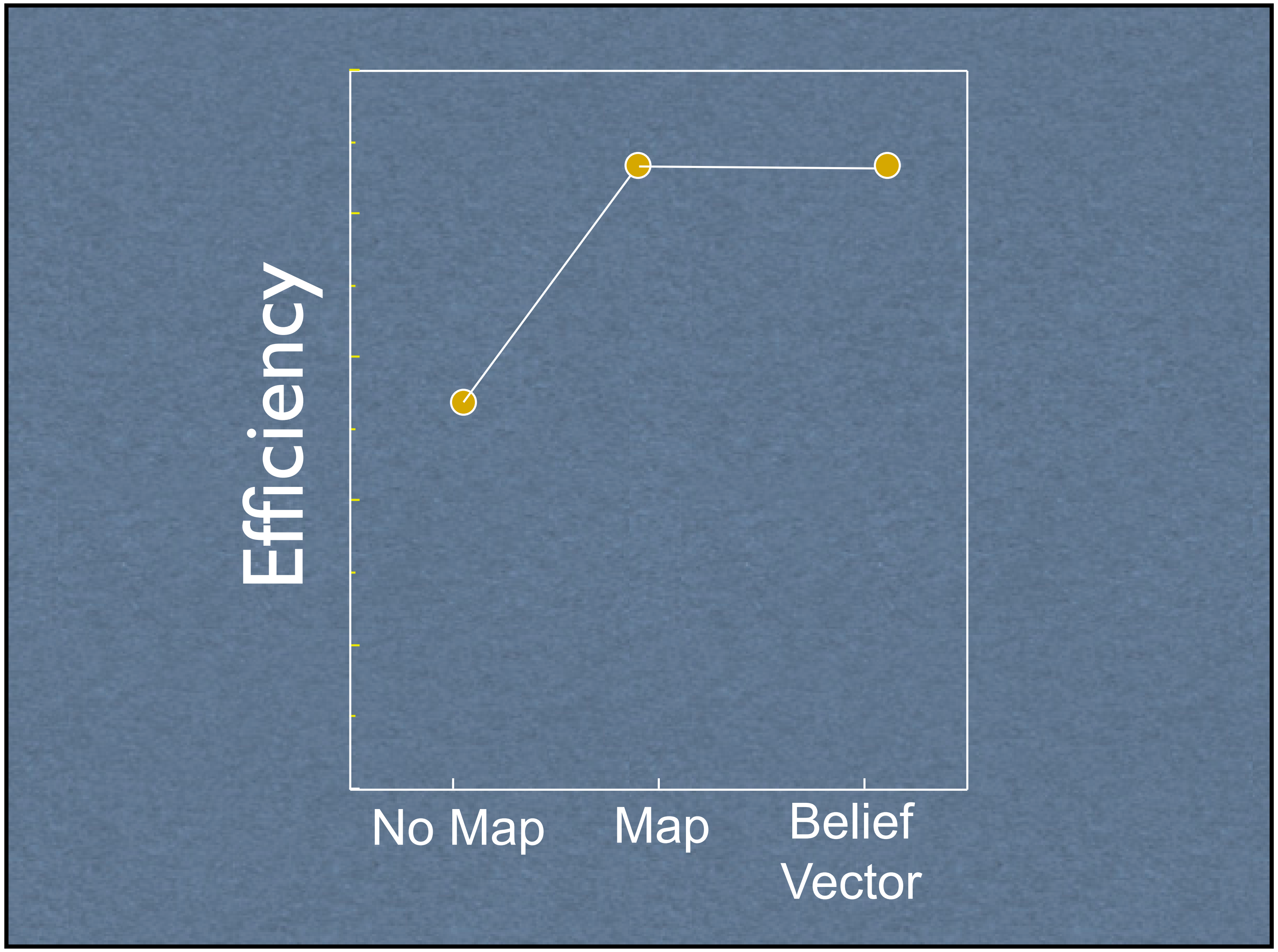


Sample Display

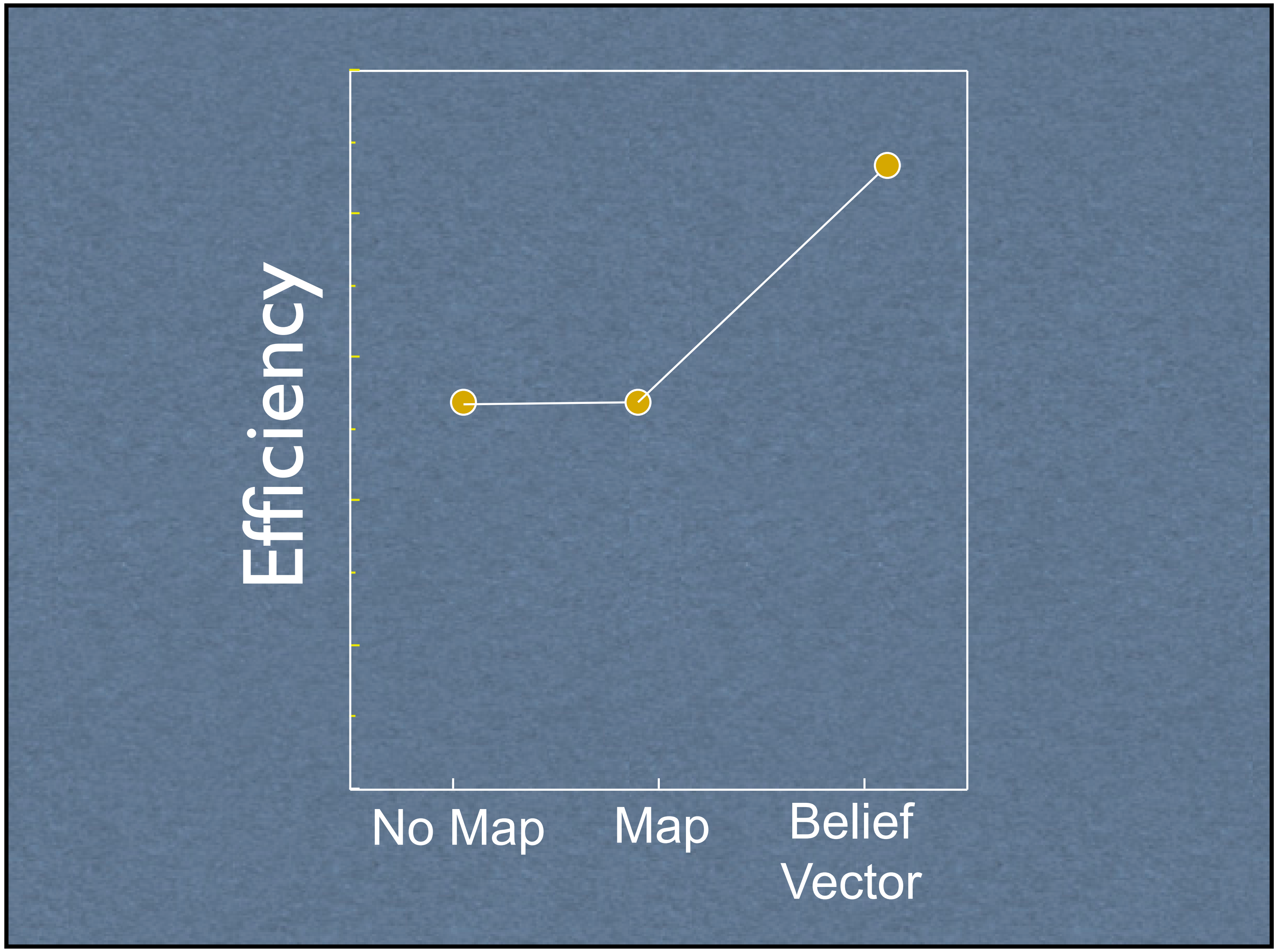
Map + Belief Vector



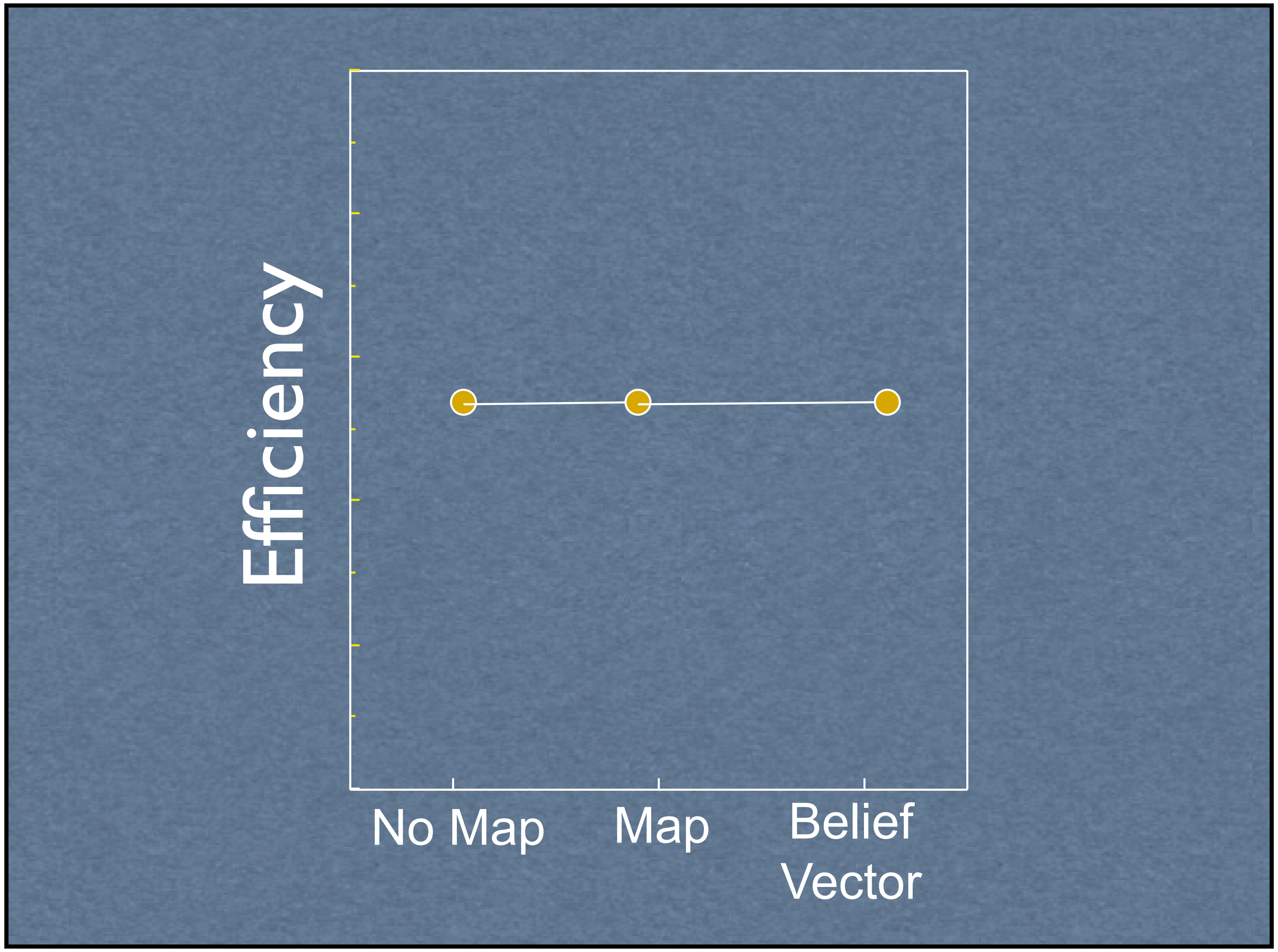
Predictions: Accessing Cog. Map



Predictions: Belief Updating



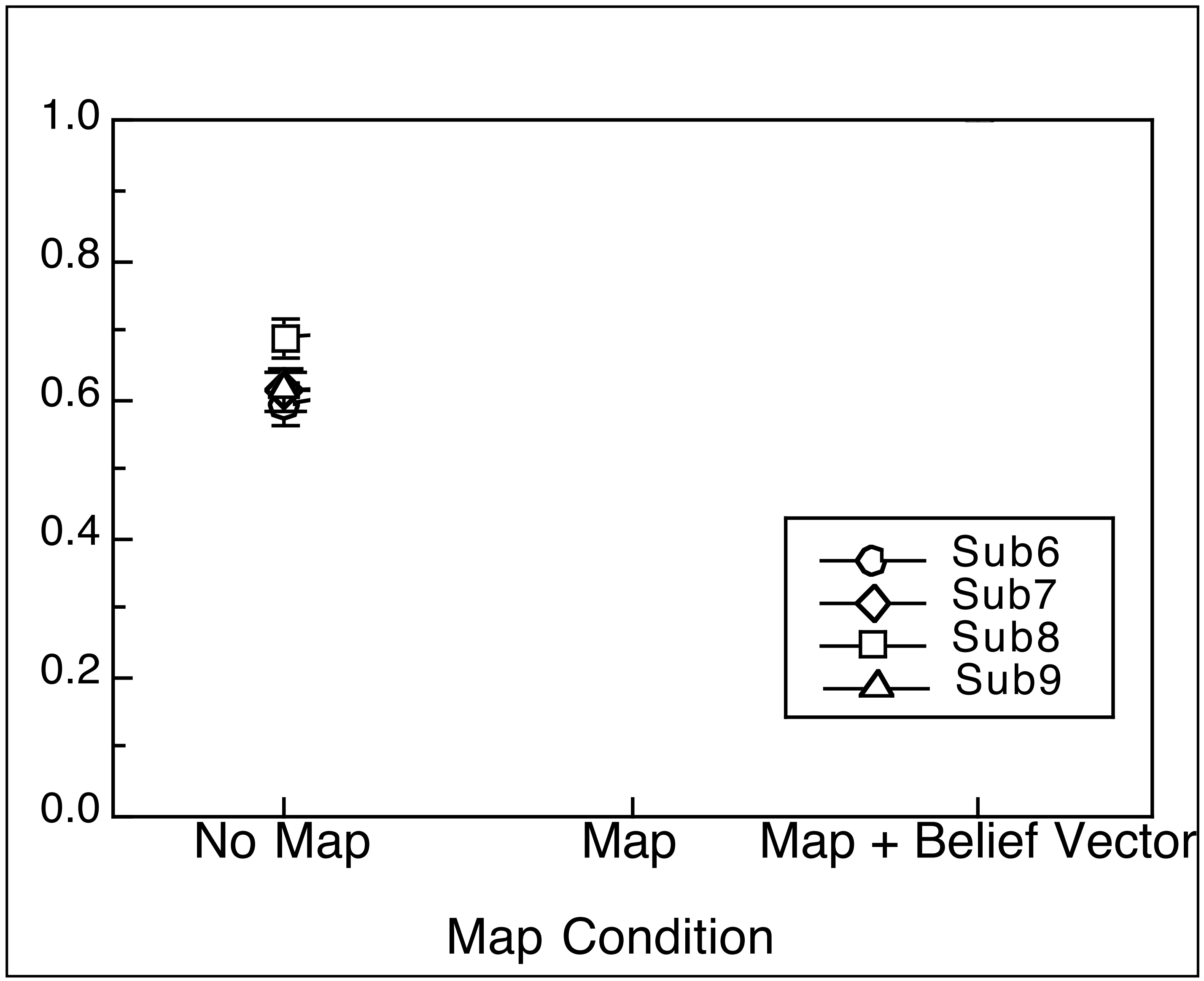
Predictions: Decision Strategy



Experiment 3: Design

- 4 Subjects
- 2 Different environments
  - Each subject ran in 1 environment
- Conditions were ran in random order
- Subjects ran in all three conditions
- Started from every state in each condition 3 times
  - 1440 Trials / subject

Experiment 3: Results



Experiment 3

- Providing explicit belief updating improves efficiency
  - What about non-spatial navigation tasks?
  - What if observations and actions are probabilistic

Seek & Destroy

Seek & Destroy

- In a space, there is an enemy.
- Enemy's position remains static
- Find the enemy and destroy the enemy
  - Actions and observations are probabilistic

Seek & Destroy

- "Seek & Destroy Task"
- 5x5 Area
- 1 Enemy
- Actions
    - Reconnaissance
    - Artillery
    - Declare Finished

Seek & Destroy

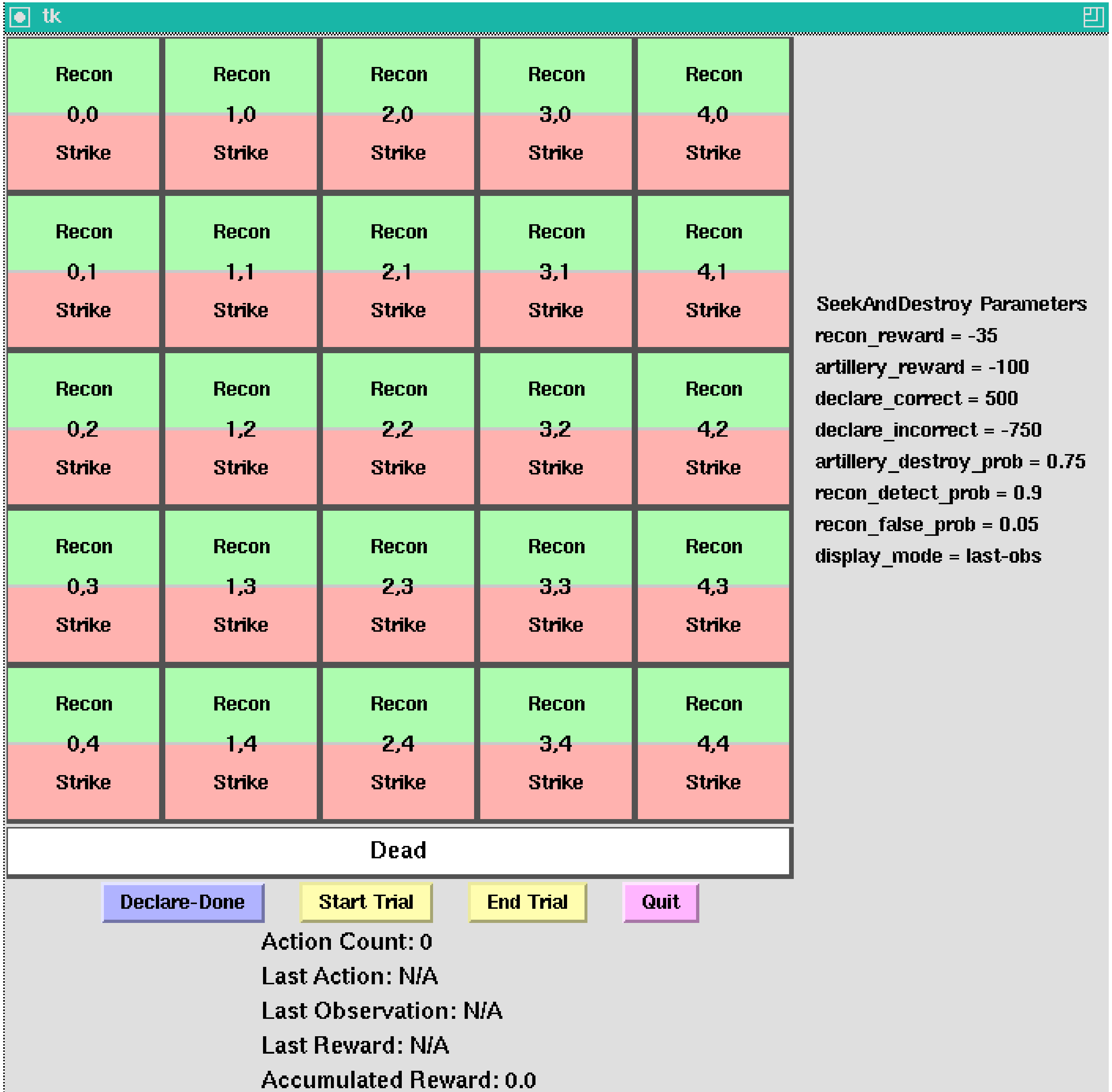
- Reward(Recon)=-35
- Reward(Artillery)=-100
- Reward(Declare|Dead)=500
- Reward(Declare|!Dead)=-750

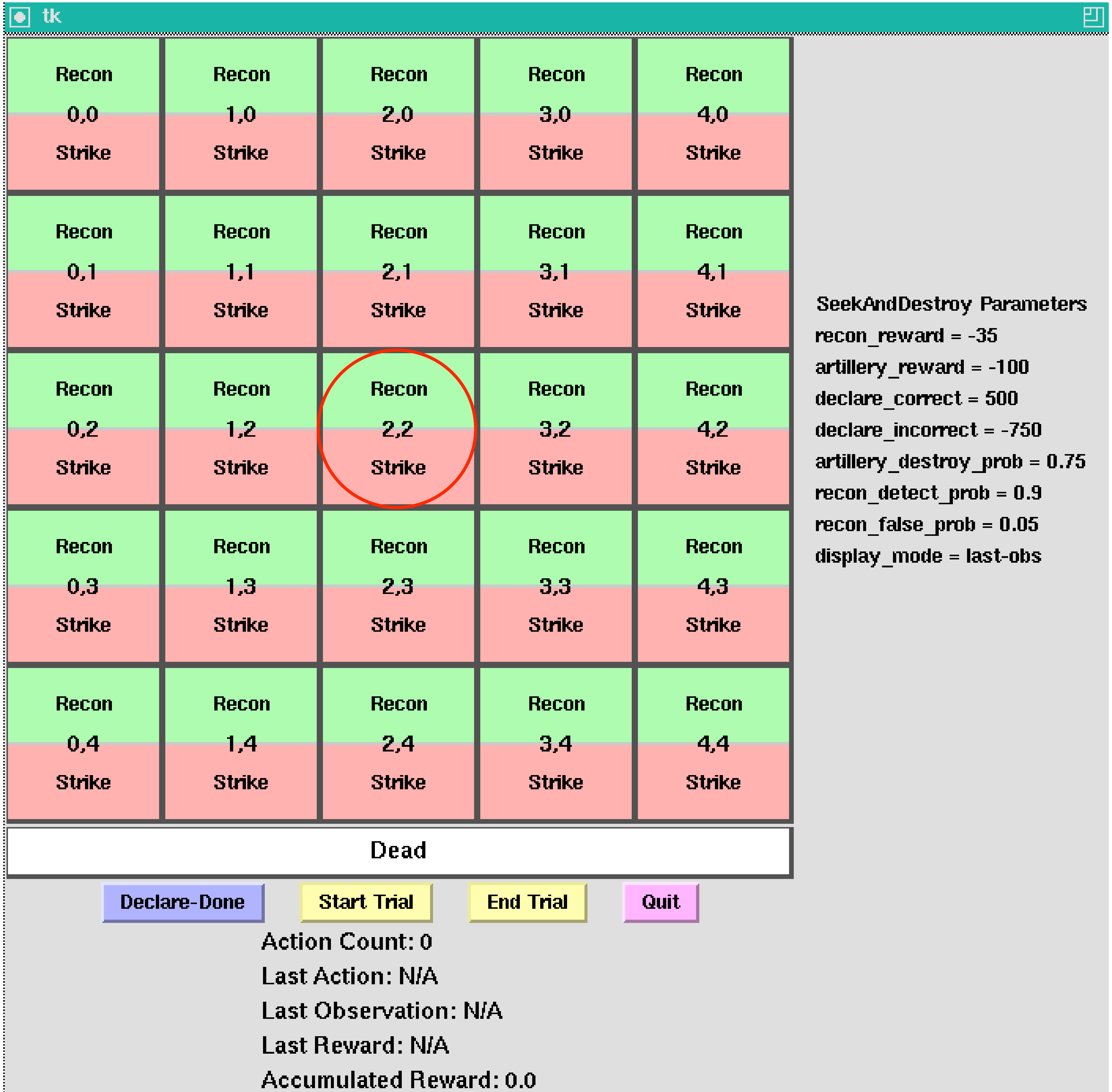
Seek & Destroy

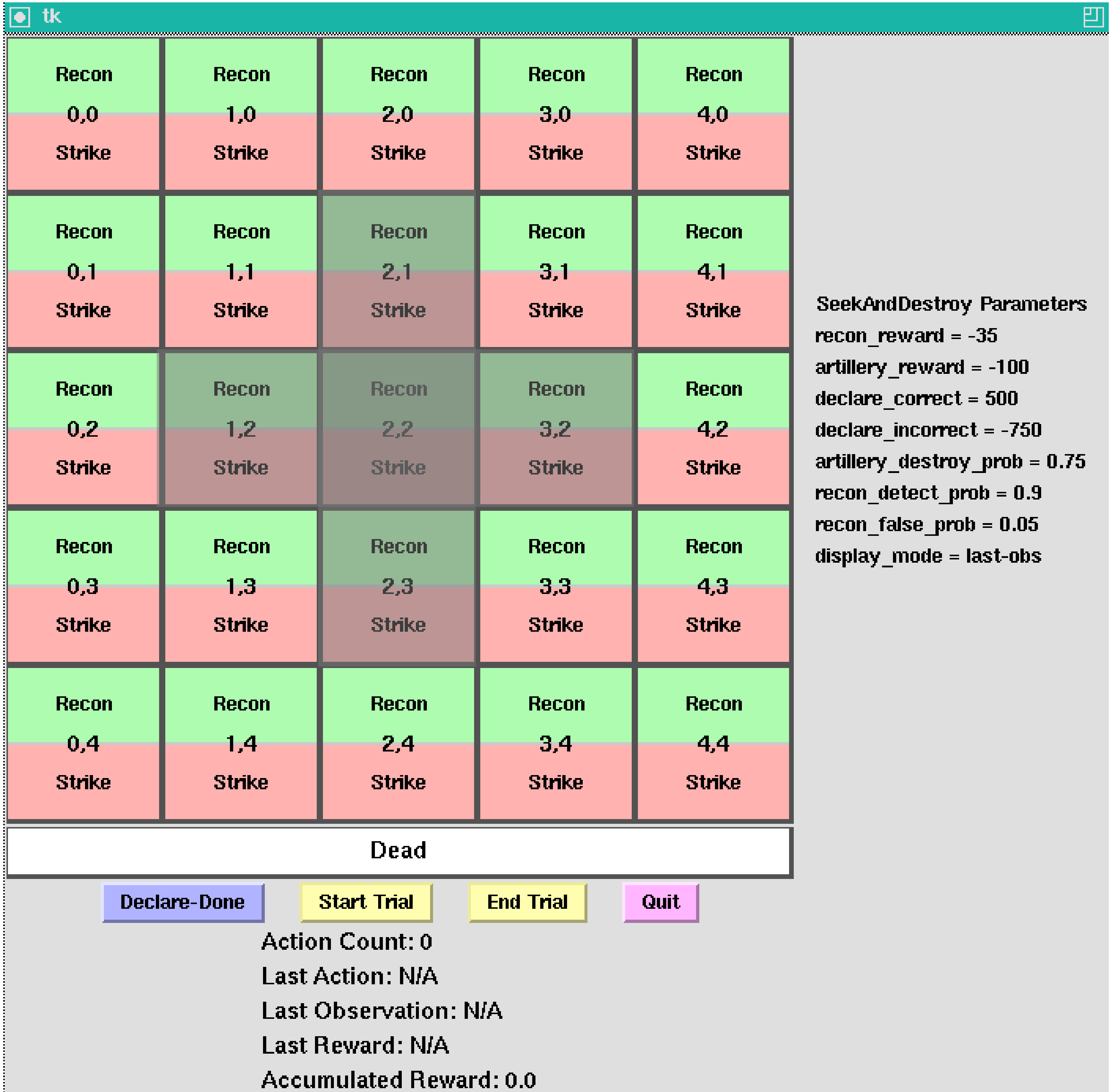
- p(Positive|Enemy)=0.9
- p(Positive|!Enemy)=0.05
- p(s'=Dead|Artillery(s),Enemy(s))=0.75

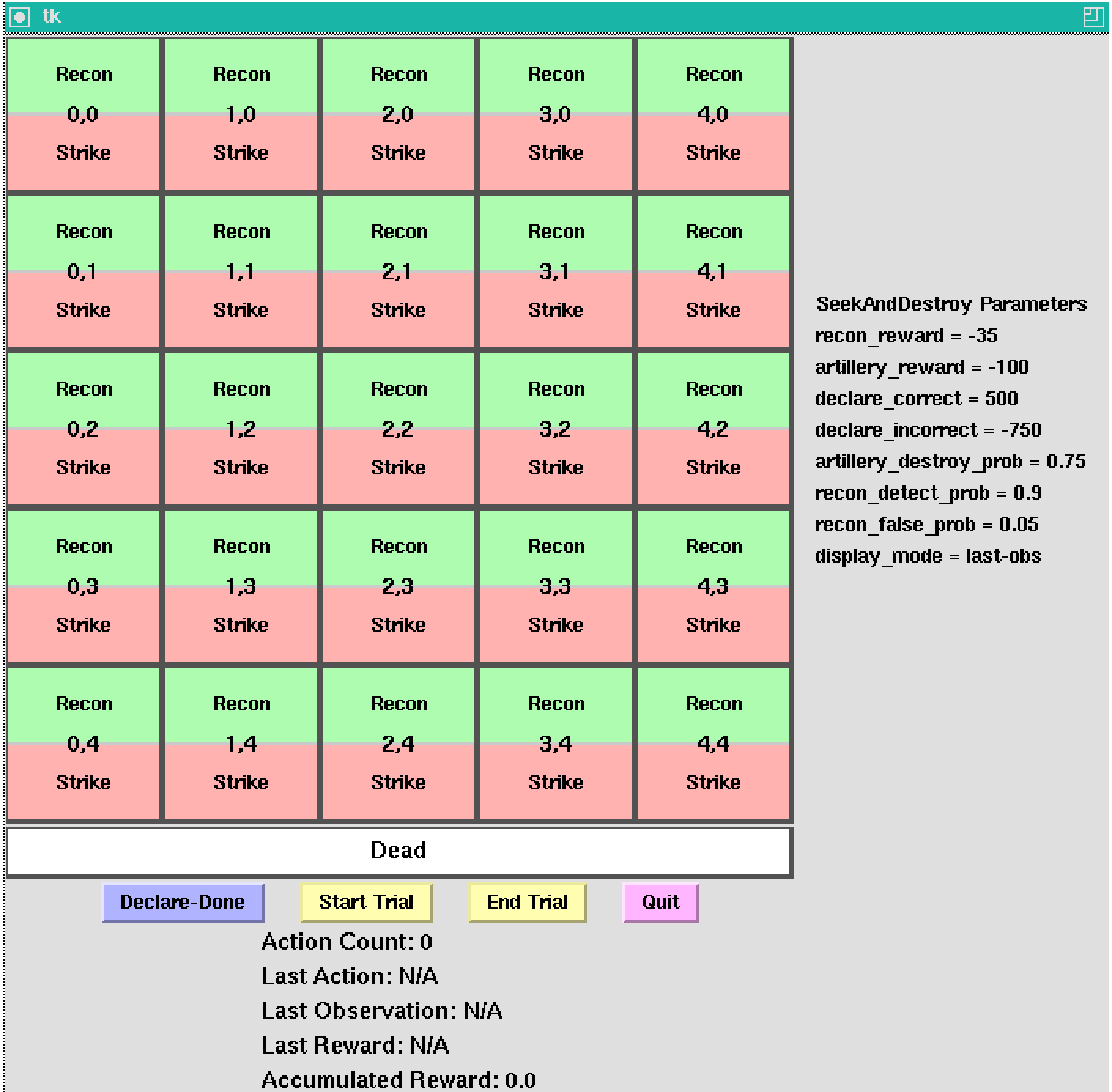
Seek & Destroy

- Three conditions
  - Last action and observations
  - All latest actions and observations
  - Current belief given actions and observations


tk
Recon
0,0
Strike
Recon
1,0
Strike
Recon
2,0
Strike
Recon
3,0
Strike
Recon
4,0
Strike
Recon
0,1
Strike
Recon
1,1
Strike
Recon
2,1
Strike
Recon
3,1
Strike
Recon
4,1
Strike
Recon
0,2
Strike
Recon
1,2
Strike
Recon
2,2
Strike
Recon
3,2
Strike
Recon
4,2
Strike
Recon
0,3
Strike
Recon
1,3
Strike
Recon
2,3
Strike
Recon
3,3
Strike
Recon
4,3
Strike
Recon
0,4
Strike
Recon
1,4
Strike
Recon
2,4
Strike
Recon
3,4
Strike
Recon
4,4
Strike
Dead
Declare-Done
Start Trial
End Trial
Quit
Action Count: 0
Last Action: N/A
Last Observation: N/A
Last Reward: N/A
Accumulated Reward: 0.0
SeekAndDestroy Parameters
recon_reward = -35
artillery_reward = -100
declare_correct = 500
declare_incorrect = -750
artillery_destroy_prob = 0.75
recon_detect_prob = 0.9
recon_false_prob = 0.05
display_mode = last-obs


tk
Recon 0,0 Strike
Recon 1,0 Strike
Recon 2,0 Strike
Recon 3,0 Strike
Recon 4,0 Strike
Recon 0,1 Strike
Recon 1,1 Strike
Recon 2,1 Strike
Recon 3,1 Strike
Recon 4,1 Strike
Recon 0,2 Strike
Recon 1,2 Strike
Recon 2,2 Strike
Recon 3,2 Strike
Recon 4,2 Strike
Recon 0,3 Strike
Recon 1,3 Strike
Recon 2,3 Strike
Recon 3,3 Strike
Recon 4,3 Strike
Recon 0,4 Strike
Recon 1,4 Strike
Recon 2,4 Strike
Recon 3,4 Strike
Recon 4,4 Strike
SeekAndDestroy Parameters
recon_reward = -35
artillery_reward = -100
declare_correct = 500
declare_incorrect = -750
artillery_destroy_prob = 0.75
recon_detect_prob = 0.9
recon_false_prob = 0.05
display_mode = last-obs
Dead
Declare-Done
Start Trial
End Trial
Quit
Action Count: 0
Last Action: N/A
Last Observation: N/A
Last Reward: N/A
Accumulated Reward: 0.0

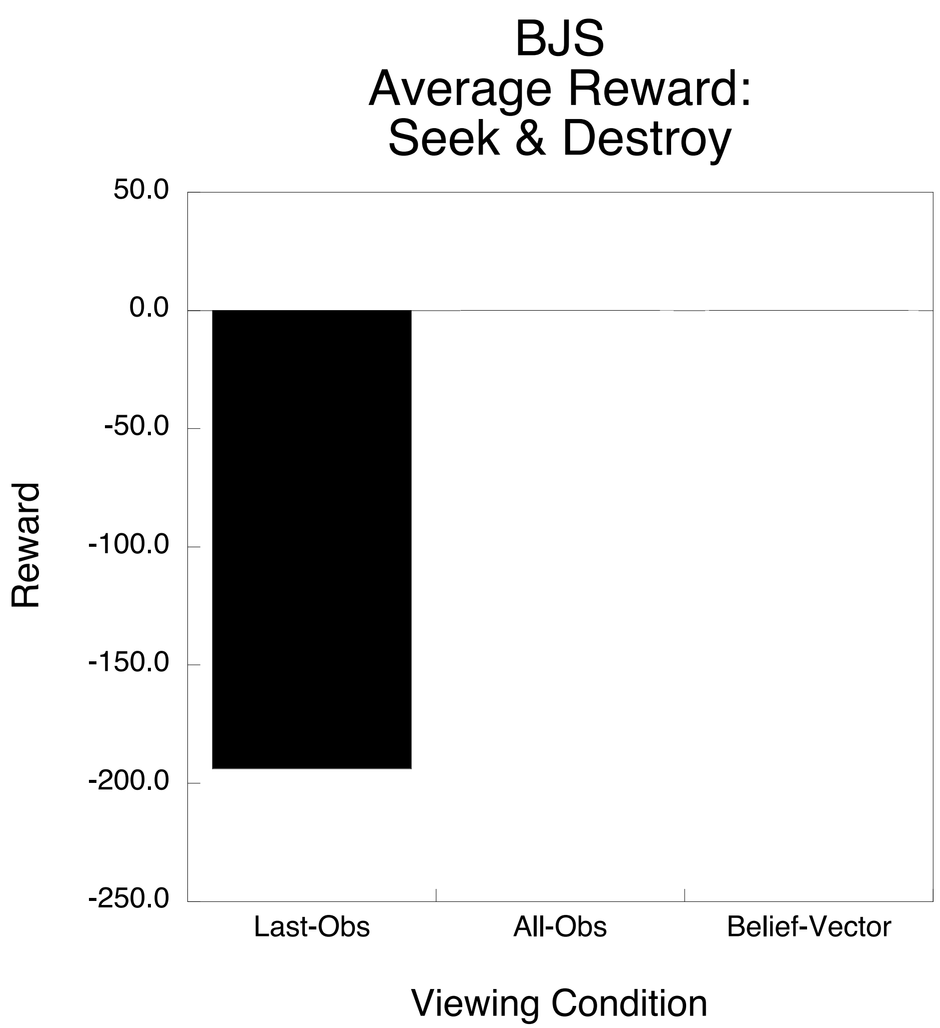

tk
Recon
0,0
Strike
Recon
1,0
Strike
Recon
2,0
Strike
Recon
3,0
Strike
Recon
4,0
Strike
Recon
0,1
Strike
Recon
1,1
Strike
Recon
2,1
Strike
Recon
3,1
Strike
Recon
4,1
Strike
Recon
0,2
Strike
Recon
1,2
Strike
Recon
2,2
Strike
Recon
3,2
Strike
Recon
4,2
Strike
Recon
0,3
Strike
Recon
1,3
Strike
Recon
2,3
Strike
Recon
3,3
Strike
Recon
4,3
Strike
Recon
0,4
Strike
Recon
1,4
Strike
Recon
2,4
Strike
Recon
3,4
Strike
Recon
4,4
Strike
Dead
Declare-Done
Start Trial
End Trial
Quit
Action Count: 0
Last Action: N/A
Last Observation: N/A
Last Reward: N/A
Accumulated Reward: 0.0
SeekAndDestroy Parameters
recon_reward = -35
artillery_reward = -100
declare_correct = 500
declare_incorrect = -750
artillery_destroy_prob = 0.75
recon_detect_prob = 0.9
recon_false_prob = 0.05
display_mode = last-obs

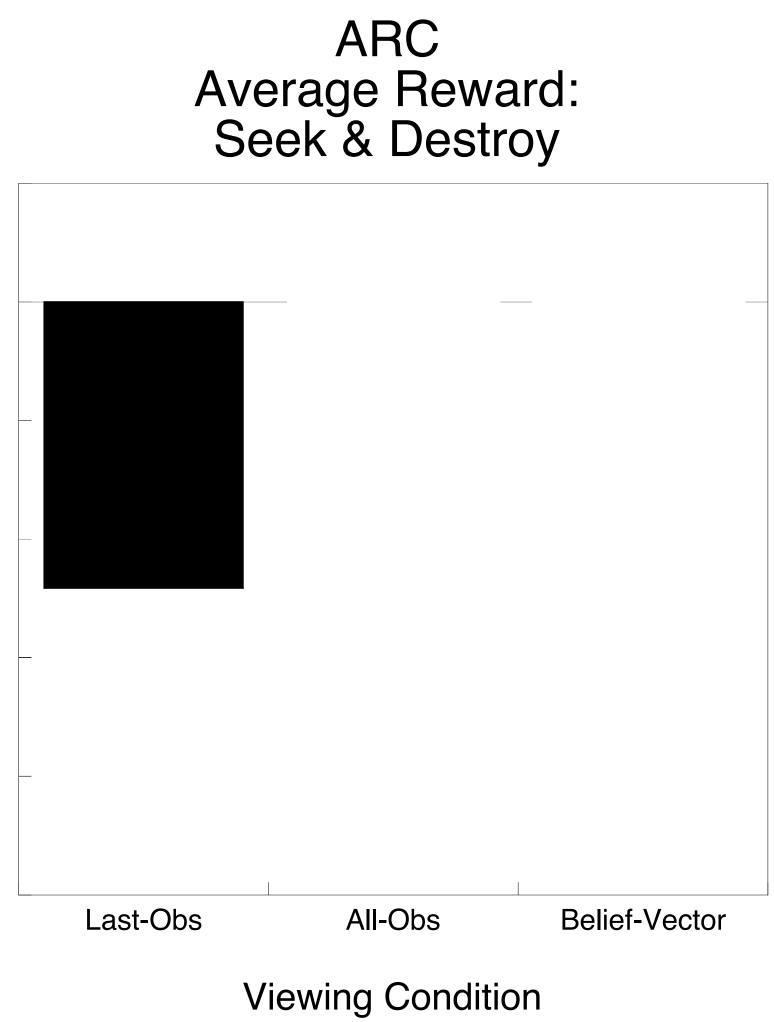

tk
Recon
0,0
Strike
Recon
1,0
Strike
Recon
2,0
Strike
Recon
3,0
Strike
Recon
4,0
Strike
Recon
0,1
Strike
Recon
1,1
Strike
Recon
2,1
Strike
Recon
3,1
Strike
Recon
4,1
Strike
Recon
0,2
Strike
Recon
1,2
Strike
Recon
2,2
Strike
Recon
3,2
Strike
Recon
4,2
Strike
Recon
0,3
Strike
Recon
1,3
Strike
Recon
2,3
Strike
Recon
3,3
Strike
Recon
4,3
Strike
Recon
0,4
Strike
Recon
1,4
Strike
Recon
2,4
Strike
Recon
3,4
Strike
Recon
4,4
Strike
SeekAndDestroy Parameters
recon_reward = -35
artillery_reward = -100
declare_correct = 500
declare_incorrect = -750
artillery_destroy_prob = 0.75
recon_detect_prob = 0.9
recon_false_prob = 0.05
display_mode = last-obs
Dead
Declare-Done
Start Trial
End Trial
Quit
Action Count: 0
Last Action: N/A
Last Observation: N/A
Last Reward: N/A
Accumulated Reward: 0.0


tk
Recon (?) 0,0 Strike
Recon (?) 1,0 Strike
Recon (?) 2,0 Strike
Recon (?) 3,0 Strike
Recon (?) 4,0 Strike
Recon (?) 0,1 Strike
Recon (-) 1,1 Strike
Recon (?) 2,1 Strike
Recon (?) 3,1 Strike
Recon (?) 4,1 Strike
Recon (?) 0,2 Strike
Recon (?) 1,2 Strike
Recon (?) 2,2 Strike
Recon (-) 3,2 Strike
Recon (?) 4,2 Strike
Recon (?) 0,3 Strike
Recon (+) 1,3 Strike
Recon (+) 2,3 Strike
Recon (?) 3,3 Strike
Recon (?) 4,3 Strike
Recon (?) 0,4 Strike
Recon (?) 1,4 Strike
Recon (?) 2,4 Strike
Recon (?) 3,4 Strike
Recon (?) 4,4 Strike
Dead
Declare-Done
Start Trial
End Trial
Quit
Action Count: 4
Last Action: Recon-2-3
Last Observation: EnemySighted
Last Reward: -35.0
Accumulated Reward: -140.0
SeekAndDestroy Parameters
recon_reward = -35
artillery_reward = -100
declare_correct = 500
declare_incorrect = -750
artillery_destroy_prob = 0.75
recon_detect_prob = 0.9
recon_false_prob = 0.05
display_mode = all-latest-obs


tk
Recon
0,0 [0.01]
Strike
Recon
1,0 [0.00]
Strike
Recon
2,0 [0.01]
Strike
Recon
3,0 [0.00]
Strike
Recon
4,0 [0.01]
Strike
Recon
0,1 [0.00]
Strike
Recon
1,1 [0.00]
Strike
Recon
2,1 [0.00]
Strike
Recon
3,1 [0.00]
Strike
Recon
4,1 [0.01]
Strike
Recon
0,2 [0.01]
Strike
Recon
1,2 [0.00]
Strike
Recon
2,2 [0.03]
Strike
Recon
3,2 [0.00]
Strike
Recon
4,2 [0.00]
Strike
Recon
0,3 [0.01]
Strike
Recon
1,3 [0.27]
Strike
Recon
2,3 [0.27]
Strike
Recon
3,3 [0.00]
Strike
Recon
4,3 [0.01]
Strike
Recon
0,4 [0.01]
Strike
Recon
1,4 [0.01]
Strike
Recon
2,4 [0.27]
Strike
Recon
3,4 [0.01]
Strike
Recon
4,4 [0.01]
Strike
Dead [0.00]
Declare-Done
Start Trial
End Trial
Quit
Action Count: 11
Last Action: Recon-2-3
Last Observation: EnemySighted
Last Reward: -35.0
Accumulated Reward: -385.0
SeekAndDestroy Parameters
recon_reward = -35
artillery_reward = -100
declare_correct = 500
declare_incorrect = -750
artillery_destroy_prob = 0.75
recon_detect_prob = 0.9
recon_false_prob = 0.05
display_mode = belief-state

Seek & Destroy: Data





Summary

- Sequential Decision Making with Uncertainty task that is a non-spatial navigation task
  - How does belief updating affect performance
  - Providing explicit belief updating improves performance
    - Consistent with Navigation task.

General Summary

- Sequential Decision Making with Uncertainty.
  - Where are the "cognitive bottlenecks" in solving these tasks efficiently.
  - Currently: Primary limitation is belief updating.

Other thoughts...

- Computationally, belief updating is simple, computing optimal action difficult
  - Belief updating:
    - Human bad; computer good
  - Optimal action
    - Human good (given BV); Computer Laborious

Other thoughts...

- Learning about an environment
  - Belief updating over possible environments.
  - Constrain set of environments
    - Number of hallways
    - “footprint” of environment
  - Drawing is estimate of current hypothesis

Collaborators

- Anthony Cassandra
- Gordon E. Legge
- J. Stephen Mansfield
- Erik Schlicht
- Kyler Eastman
- Chris Goodson

Thank you