Understanding Human Sequential Decision Making Under Uncertainty Using Ideal Observer Analysis

Brian J. Stankiewicz, Ph.D.

University of Texas, Austin
Department Of Psychology & Center for Perceptual Systems

June 17, 2005

Collaborators

- University of Texas, Austin
  - Matthew deBrecht
  - Chris Goodson
  - Kyler Eastman
  - Matthew McCabe
  - Anthony Cassandra
- University of Minnesota
  - Gordon E. Legge
  - Erik Schlicht
- SUNY Plattsburgh
  - J. Stephan Mansfield
- Army Research Lab
  - Sam Middlebrooks

University XXI / Army Research Labs

National Institute of Health

Air Force Office of Scientific Research

Examples Sequential Decision Making with Uncertainty

- Medical Diagnosis
- Scientific Exploration
- Re-orientation after becoming lost
- Seek & Destroy
- Tiger Problem
- Etc.

Examples Sequential Decision Making with Uncertainty

- Medical Diagnosis
- Scientific Exploration
- Re-orientation after becoming lost
- Seek & Destroy
- Tiger Problem
- Etc.

Sequential Decision Making: Structure

- Structure of Sequential Decision Making with Uncertainty Tasks.
  1. States (S)
     - True state is hidden
  2. Actions (A)
  3. Observations (O)
  4. Transition Matrix (T)
  5. Rewards (R)
  6. Beliefs (B)

Medical Diagnosis: States

- Patient can be in one of a set of states. (S)
    - Influenza
    - Pancreatic cancer
    - Appendicitis
    - etc.
- Unable to directly observe the current state.
    - True state is **Hidden**.

Medical Diagnosis: States

- Patient can be in one of a set of states. (S)
  - Influenza
  - Pancreatic cancer
  - Appendicitis
  - etc.
- Unable to directly observe the current state.
  - True state is **Hidden**.

Medical Diagnosis: States

- Patient can be in one of a set of states. (S)
  - Influenza
  - Pancreatic cancer
  - Appendicitis
  - etc.
- Unable to directly observe the current state.
  - True state is **Hidden**.

Medical Diagnosis: States

- Patient can be in one of a set of states. (S)
  - Influenza
  - Pancreatic cancer
  - Appendicitis
  - etc.
- Unable to directly observe the current state.
  - True state is **Hidden**.

Medical Diagnosis: Actions

- A set of **Actions** available to decision maker. (A)
  - Take temperature
  - Blood workup
  - MRI scan
  - Surgery
  - etc.

Medical Diagnosis: Actions

- A set of **Actions** available to decision maker. (A)
  - Take temperature
  - Blood workup
  - MRI scan
  - Surgery
  - etc.

Medical Diagnosis: Rewards

- Actions have costs and rewards.
  - $5,000 for surgery
  - $0.32 for taking temperature
- Rewards/Costs may be dependent upon the state of the patient
  - E.g. Surgery given patient has Appendicitis
    - $Reward(Surgery|Appendicitis)$ =-$5,000+$4,000=-$1,000
    - $Reward(Surgery|NotAppendicitis)$ =-$5,000

Medical Diagnosis: Rewards

- Actions have costs and rewards.
  - \$5,000 for surgery
  - \$0.32 for taking temperature
- Rewards/Costs may be dependent upon the state of the patient
  - E.g. Surgery given patient has Appendicitis
    - $Reward(Surgery|Appendicitis)$ =-\$5,000+\$4,000=-\$1,000
    - $Reward(Surgery|NotAppendicitis)$ =-\$5,000

Medical Diagnosis: Rewards

- Actions have costs and rewards.
  - \$5,000 for surgery
  - \$0.32 for taking temperature
- Rewards/Costs may be dependent upon the state of the patient
  - E.g. Surgery given patient has Appendicitis
    - $Reward(Surgery|Appendicitis)$ =-\$5,000+\$4,000=-\$1,000
    - $Reward(Surgery|NotAppendicitis)$ =-\$5,000

Medical Diagnosis: Rewards

- Actions have costs and rewards.
  - \$5,000 for surgery
  - \$0.32 for taking temperature
- Rewards/Costs may be dependent upon the state of the patient
  - E.g. Surgery given patient has Appendicitis
    - $Reward(Surgery|Appendicitis)$ =-\$5,000+\$4,000=-\$1,000
    - $Reward(Surgery|NotAppendicitis)$ =-\$5,000

Medical Diagnosis: Rewards

- Actions have costs and rewards.
  - \$5,000 for surgery
  - \$0.32 for taking temperature
- Rewards/Costs may be dependent upon the state of the patient
  - E.g. Surgery given patient has Appendicitis
    - $Reward(Surgery|Appendicitis)$ =-\$5,000+\$4,000=-\$1,000
    - $Reward(Surgery|NotAppendicitis)$ =-\$5,000

Medical Diagnosis: Observations

- Actions result in **Observations**. (O)
  - E.g., Temperature = Normal
  - No elevated white blood cells
- Observations help to determine the true state of the system (Hidden state).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
- Observations can be noisy (i.e., non-deterministic)
  - E.g., Nurse may write down wrong temperature on chart
  - Thermometer might be broken or used incorrectly
  - $1.0 > p(Observation|State, Action) > 0.0$

Medical Diagnosis: Observations

- Actions result in **Observations**. (O)
  - E.g., Temperature = Normal
  - No elevated white blood cells
- Observations help to determine the true state of the system (Hidden state).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
- Observations can be noisy (i.e., non-deterministic)
  - E.g., Nurse may write down wrong temperature on chart
  - Thermometer might be broken or used incorrectly
  - 1.0> $p(Observation|State, Action)$ >0.0

Medical Diagnosis: Observations

- Actions result in **Observations**. (O)
  - E.g., Temperature = Normal
  - No elevated white blood cells
- Observations help to determine the true state of the system (Hidden state).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
- Observations can be noisy (i.e., non-deterministic)
  - E.g., Nurse may write down wrong temperature on chart
  - Thermometer might be broken or used incorrectly
  - $1.0 > p(Observation|State, Action) > 0.0$

Medical Diagnosis: Observations

- Actions result in **Observations**. (O)
  - E.g., Temperature = Normal
  - No elevated white blood cells
- Observations help to determine the true state of the system (Hidden state).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
- Observations can be noisy (i.e., non-deterministic)
  - E.g., Nurse may write down wrong temperature on chart
  - Thermometer might be broken or used incorrectly
  - $1.0 > p(Observation|State, Action) > 0.0$

Medical Diagnosis: Observations

- Actions result in **Observations**. (O)
  - E.g., Temperature = Normal
  - No elevated white blood cells
- Observations help to determine the true state of the system (Hidden state).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
- Observations can be noisy (i.e., non-deterministic)
  - E.g., Nurse may write down wrong temperature on chart
  - Thermometer might be broken or used incorrectly
  - 1.0> $p(Observation|State, Action)$ >0.0

Medical Diagnosis: Observations

- Actions result in **Observations**. (O)
  - E.g., Temperature = Normal
  - No elevated white blood cells
- Observations help to determine the true state of the system (Hidden state).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
- Observations can be noisy (i.e., non-deterministic)
  - E.g., Nurse may write down wrong temperature on chart
  - Thermometer might be broken or used incorrectly
  - $1.0 > p(Observation|State, Action) > 0.0$

Medical Diagnosis: Beliefs

- We can represent the decision maker's current belief about the true state of the system in terms of likelihoods.
  - E.g., $< p(Influenza), p(Pancreaticcancer), \ldots, p(Healthy) >$
- Observations and actions modify this belief (i.e., information gathering).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
  - p(Pancreatic cancer) will go up slightly given that the temperature is normal.

Medical Diagnosis: Beliefs

- We can represent the decision maker's current belief about the true state of the system in terms of likelihoods.
  - E.g., $< p(Influenza), p(Pancreaticcancer), \ldots, p(Healthy) >$
- Observations and actions modify this belief (i.e., information gathering).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
  - p(Pancreatic cancer) will go up slightly given that the temperature is normal.

Medical Diagnosis: Beliefs

- We can represent the decision maker's current belief about the true state of the system in terms of likelihoods.
  - E.g., $< p(Influenza), p(Pancreaticcancer), \ldots, p(Healthy) >$
- Observations and actions modify this belief (i.e., information gathering).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
  - p(Pancreatic cancer) will go up slightly given that the temperature is normal.

Medical Diagnosis: Beliefs

- We can represent the decision maker's current belief about the true state of the system in terms of likelihoods.
  - E.g., $< p(Influenza), p(Pancreaticcancer), \ldots, p(Healthy) >$
- Observations and actions modify this belief (i.e., information gathering).
  - E.g. $p(Influenza)$ goes down if temperature is normal.
  - p(Pancreatic cancer) will go up slightly given that the temperature is normal.

Medical Diagnosis: Transitions

- Executing an action may move the system from one state to another. (T)
  - E.g., Surgery may move the patient from Appendicitis to Healthy.
- Results of actions may be non-deterministic (1.0$>p(s'|s,a)>$0.0)
  - $p(Healthy|Appendicitis, Surgery)$=0.97
  - $p(Dead|Appendicitis, Surgery)$=0.03

Medical Diagnosis: Transitions

- Executing an action may move the system from one state to another. (T)
  - E.g., Surgery may move the patient from Appendicitis to Healthy.
- Results of actions may be non-deterministic ($1.0 > p(s'|s,a) > 0.0$)
  - $p(Healthy|Appendicitis, Surgery) = 0.97$
  - $p(Dead|Appendicitis, Surgery) = 0.03$

Medical Diagnosis: Transitions

- Executing an action may move the system from one state to another. (T)
  - E.g., Surgery may move the patient from Appendicitis to Healthy.
- Results of actions may be non-deterministic $(1.0 > p(s'|s,a) > 0.0)$
  - $p(Healthy|Appendicitis, Surgery) = 0.97$
  - $p(Dead|Appendicitis, Surgery) = 0.03$

Medical Diagnosis: Transitions

- Executing an action may move the system from one state to another. (T)
  - E.g., Surgery may move the patient from Appendicitis to Healthy.
- Results of actions may be non-deterministic $(1.0 > p(s'|s,a) > 0.0)$
  - $p(Healthy|Appendicitis, Surgery)$=0.97
  - $p(Dead|Appendicitis, Surgery)$=0.03

Sequential Decision Making: Structure

- Medical diagnosis can be described with the following structures.
  1. States (S)
     - True state is hidden
  2. Actions (A)
  3. Observations (O)
  4. Transition Matrix (T)
  5. Rewards (R)
  6. Beliefs (B)

Medical Diagnosis: Summary

- Description of Medical Diagnosis problem.

Table: General description of decision making under uncertainty task in medical diagnosis.

States	Set of disorders (Cancer, Influenza, etc.)
Hidden State	Patient's current disorder(s)
Actions	MRI scan, blood work, surgery, etc.
Observations	Elevated white blood cells, temperature, etc.
Transitions	Pancreatic Cancer $\xrightarrow{Surgery}$ No cancer
Rewards	Money, time, personnel, etc.
Beliefs	Hypothesis about the patient's current disorder

Re-orientation task

- Other sequential decision making processes have the same basic structure.

Table: Re-orientation after becoming lost.

States	Position & Orientation in environment
Hidden State	Current position & orientation
Actions	Move forward, rotate-left, rotate-right, etc.
Observations	Visual Landmarks, Proprioception, Compass, etc.
Transitions	$< X, Y, \Theta > \overset{Translation}{\longrightarrow} < X', Y', \Theta >$
Rewards	Energy expended, time, etc.
Beliefs	Hypothesis about your current position

Seek & Destroy task

- Other sequential decision making processes have the same basic structure.

Table: Structure of seek & destroy problem.

States	Possible locations of enemy
Hidden State	Current position of enemy
Actions	Reconnaissance, Artillery, Declare "Finished", etc.
Observations	Enemy sighted, Enemy not sighted, etc.
Transitions	Enemy:X,Y $\stackrel{Artillery}{\longrightarrow}$ Enemy: Destroyed
Rewards	Cost of Unmanned Air Vehicle assignment, time, etc.
Beliefs	Hypothesis about the enemy's current position

Beyond the structure of the problem

- We have provided a description of the structure of a sequential decision making with uncertainty task.
- Does not provide with a description of what one **should** do if they are optimal/rational.
- One can generate the optimal behavior using **Partially Observable Markov Decision Processes** (POMDP)
  - Bayesian statistics method for computing the optimal decision given the types of specifications above.
- To formulate a problem we need to fully specify problem.
- Describe a simple example to illustrate the mathematics.

Beyond the structure of the problem

- We have provided a description of the structure of a sequential decision making with uncertainty task.
- Does not provide with a description of what one **should** do if they are optimal/rational.
- One can generate the optimal behavior using **Partially Observable Markov Decision Processes** (POMDP)
  - Bayesian statistics method for computing the optimal decision given the types of specifications above.
- To formulate a problem we need to fully specify problem.
- Describe a simple example to illustrate the mathematics.

Beyond the structure of the problem

- We have provided a description of the structure of a sequential decision making with uncertainty task.
- Does not provide with a description of what one **should** do if they are optimal/rational.
- One can generate the optimal behavior using **Partially Observable Markov Decision Processes** (POMDP)
  - Bayesian statistics method for computing the optimal decision given the types of specifications above.
- To formulate a problem we need to fully specify problem.
- Describe a simple example to illustrate the mathematics.

Beyond the structure of the problem

- We have provided a description of the structure of a sequential decision making with uncertainty task.
- Does not provide with a description of what one **should** do if they are optimal/rational.
- One can generate the optimal behavior using **Partially Observable Markov Decision Processes** (POMDP)
  - Bayesian statistics method for computing the optimal decision given the types of specifications above.
- To formulate a problem we need to fully specify problem.
- Describe a simple example to illustrate the mathematics.

Beyond the structure of the problem

- We have provided a description of the structure of a sequential decision making with uncertainty task.
- Does not provide with a description of what one **should** do if they are optimal/rational.
- One can generate the optimal behavior using **Partially Observable Markov Decision Processes** (POMDP)
    - Bayesian statistics method for computing the optimal decision given the types of specifications above.
- To formulate a problem we need to fully specify problem.
- Describe a simple example to illustrate the mathematics.

Tiger Problem

1. Tiger Problem
   - Simple example of Sequential Decision Making under Uncertainty task.
   - Illustration to provide intuitive understanding of **POMDP** architecture.

Tiger Problem: States

- Two doors:
  - Behind one door is Tiger
  - Behind other door is “pot of gold”

Tiger Problem: Actions

- Three Actions:
  1. Listen
  2. Open Left-Door
  3. Open Right-Door

Tiger Problem: Observations

- Two Observations:
  1. Hear Tiger Left ($Hear_{Left}$)
  2. Hear Tiger Right ($Hear_{Right}$)

Observation Structure

$$p(Hear_{Left} | Tiger_{Left}, Listen) = 0.85$$

$$p(Hear_{Right} | Tiger_{Right}, Listen) = 0.85$$

$$p(Hear_{Right} | Tiger_{Left}, Listen) = 0.15$$

$$p(Hear_{Left} | Tiger_{Right}, Listen) = 0.15$$

Tiger Problem: Observations

- Two Observations:
  1. Hear Tiger Left ($Hear_{Left}$)
  2. Hear Tiger Right ($Hear_{Right}$)

Observation Structure

$$p(Hear_{Left}|Tiger_{Left}, Listen) = 0.85$$

$$p(Hear_{Right}|Tiger_{Right}, Listen) = 0.85$$

$$p(Hear_{Right}|Tiger_{Left}, Listen) = 0.15$$

$$p(Hear_{Left}|Tiger_{Right}, Listen) = 0.15$$

Tiger Problem: Rewards

Table: Reward Structure for Tiger Problem

	Tiger=Left	Tiger=Right
Listen	-1	-1
Open-Left	-100	10
Open-Right	10	-100

Formalizing a SDMUU Task

1. Belief Vector ($p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$)
    - The decision maker's current belief about the true state of the system.
    - Represented as the probability of being in each state.
2. Hidden state (s_{true})
    - The position of the Tiger
    - Not **directly** observable
3. Goal
    - Maximize the expected reward.

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1			
2			
3			
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen		
2			
3			
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	
2			
3			
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2			
3			
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	
3			
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3			
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Listen	$Hear_{Right}$	
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Listen	$Hear_{Right}$	0.85
4			
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Listen	$Hear_{Right}$	0.85
4	Listen	$Hear_{Left}$	
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b, o) = \frac{p(o|s', b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Listen	$Hear_{Right}$	0.85
4	Listen	$Hear_{Left}$	0.9698
5			

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Listen	$Hear_{Right}$	0.85
4	Listen	$Hear_{Left}$	0.9698
5	Listen	$Hear_{Left}$	

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Listen	$Hear_{Right}$	0.85
4	Listen	$Hear_{Left}$	0.9698
5	Listen	$Hear_{Left}$	0.9945

Tiger Problem: Belief Updating

Belief Updating

$$p(s'|b,o) = \frac{p(o|s',b)p(s'|b))}{p(o|b)}$$

Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Listen	$Hear_{Right}$	0.85
4	Listen	$Hear_{Left}$	0.9698
5	Listen	$Hear_{Left}$	0.9945

But what about action selection?

Tiger Problem: Action Selection

Table: Reward Structure for Tiger Problem

	Tiger=Left	Tiger=Right
Listen	-1	-1
Open-Left	-100	10
Open-Right	10	-100

Computing Expected Value

- $\rho(b,a) = \sum_{s \in S} R(s,a)b(s)$
  - Immediate reward
- $R = r(0) + \sum_{t=1}^{\infty} r(t)$
  - Consider immediate and future rewards
- $V(b) = \max_{a \in A} \left[\rho(b,a) + \sum_{b' \in B} \tau(b,a,b')V(b')\right]$
  - Expected Value Function

Computing Expected Value

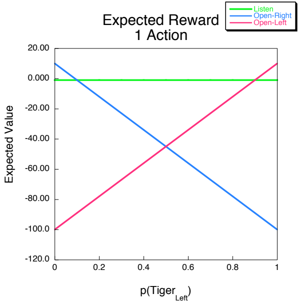
- $\rho(b,a) = \sum_{s \in S} R(s,a)b(s)$
  - Immediate reward
- $R = r(0) + \sum_{t=1}^{\infty} r(t)$
  - Consider immediate and future rewards
- $V(b) = \max_{a \in A} \left[\rho(b,a) + \sum_{b' \in B} \tau(b,a,b')V(b')\right]$
  - Expected Value Function

Computing Expected Value

- $\rho(b, a) = \sum_{s \in S} R(s, a) b(s)$
  - Immediate reward
- $R = r(0) + \sum_{t=1}^{\infty} r(t)$
  - Consider immediate and future rewards
- $V(b) = \max_{a \in A} \left[\rho(b, a) + \sum_{b' \in B} \tau(b, a, b') V(b') \right]$
  - Expected Value Function

Tiger Problem: Expected Reward

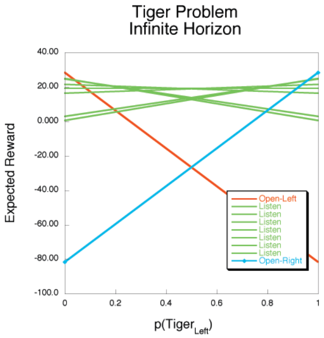
- $\rho(b,a) = \sum_{s \in S} R(s,a)b(s)$
  - Immediate reward



- Need to calculate the expected reward.
- Reward(Action,State)

Tiger Problem: Action Selection

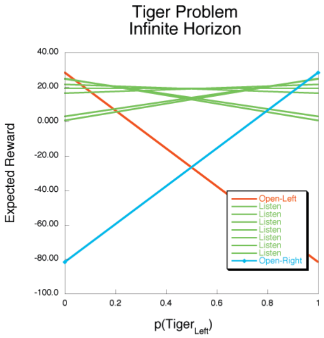
- $R = r(0) + \sum_{t=1}^{\infty} r(t)$
  - Immediate and future rewards
- $V(b) = \max_{a \in A} \left[\rho(b,a) + \sum_{b' \in B} \tau(b,a,b') V(b') \right]$
  - Expected Value Function



- Expected reward functions for multiple future actions with an infinite horizon.

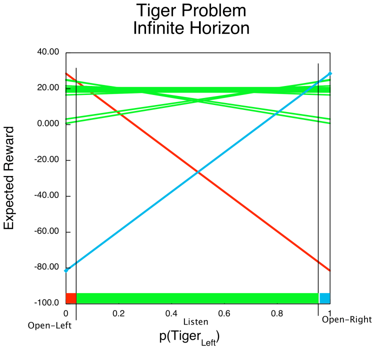
Tiger Problem: Action Selection

- $R = r(0) + \sum_{t=1}^{\infty} r(t)$
  - Immediate and future rewards
- $V(b) = \max_{a \in A} \left[\rho(b, a) + \sum_{b' \in B} \tau(b, a, b') V(b') \right]$
  - Expected Value Function



- Expected reward functions for multiple future actions with an infinite horizon.

Tiger Problem: Action Selection



- From value function (Expected Rewards) we can generate a policy based upon our current belief (belief vector).

Tiger Problem: Action Selection

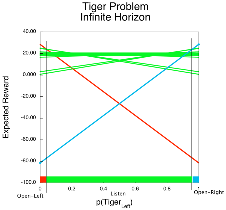


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1			
2			
3			

Tiger Problem: Action Selection

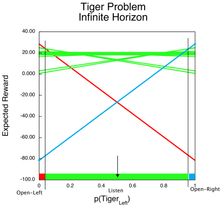


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen		
2			
3			

Tiger Problem: Action Selection

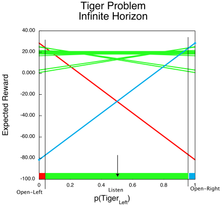


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	
2			
3			

Tiger Problem: Action Selection

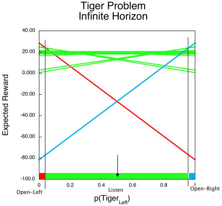


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2			
3			

Tiger Problem: Action Selection

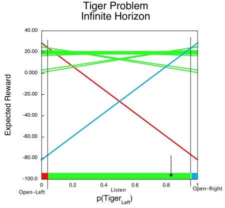


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2			
3			

Tiger Problem: Action Selection

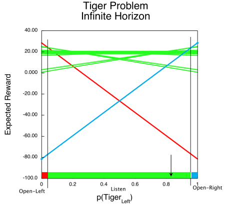


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	
3			

Tiger Problem: Action Selection

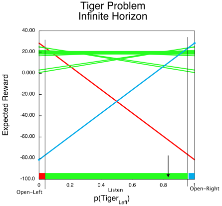


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3			

Tiger Problem: Action Selection

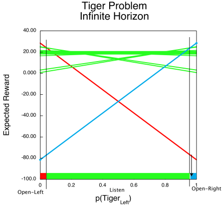


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3			

Tiger Problem: Action Selection

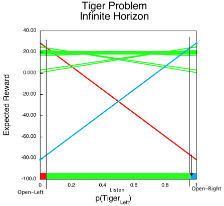


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Open-Right	*Reward*	

Tiger Problem: Action Selection

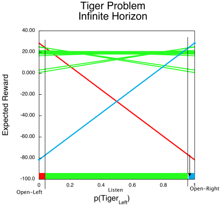


Table: Belief Updating for Tiger Problem

Act. Num	Action	Observation	$p(Tiger_{Left})$
0	—-	—-	0.5
1	Listen	$Hear_{Left}$	0.85
2	Listen	$Hear_{Left}$	0.9698
3	Open-Right	*Reward*	0.5

POMDP: Computing Expected Value

1. Using a POMDP we can generate a policy graph for a **Sequential Decision Making Under Uncertainty Task**.
   - Policy graph provides us with the optimal action given a belief about the true state of the system.
2. Using a POMDP we can compute the **Expected Reward** given the initial belief state and optimal action selection.
   - Using the optimal expected reward structure we can compare human performance to the optimal performance.
   - By comparing human behavior to the optimal Expected Reward we can get a measure of **efficiency**.

POMDP: Computing Expected Value

1. Using a POMDP we can generate a policy graph for a **Sequential Decision Making Under Uncertainty Task**.
    - Policy graph provides us with the optimal action given a belief about the true state of the system.
2. Using a POMDP we can compute the **Expected Reward** given the initial belief state and optimal action selection.
    - Using the optimal expected reward structure we can compare human performance to the optimal performance.
    - By comparing human behavior to the optimal Expected Reward we can get a measure of **efficiency**.

Utility of Optimal Observer

- What is the **utility** of computing the optimal solution?
- Provides a way to normalize for **task difficulty**.
- For example, Stankiewicz, Legge, Mansfield & Schlicht (Under Review)

Utility of Optimal Observer

- What is the **utility** of computing the optimal solution?
- Provides a way to normalize for **task difficulty**.
- For example, Stankiewicz, Legge, Mansfield & Schlicht (Under Review)

Utility of Optimal Observer

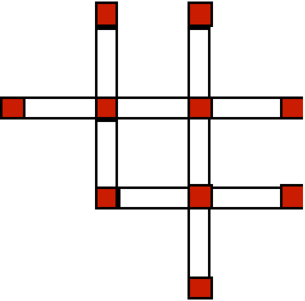
- What is the **utility** of computing the optimal solution?
- Provides a way to normalize for **task difficulty**.
- For example, Stankiewicz, Legge, Mansfield & Schlicht (Under Review)

Navigating with uncertainty: General

- Needed to carefully control the environment (system) so that we can provide the optimal navigator (POMDP) with the same information as the human.
  - Used virtual reality indoor environments.
  - Environments were randomly generated on a Cartesian grid.
  - Environments were visually sparse.
    - No object landmarks
    - Able to quantify **Observations** ($p(o|s)$)
    - Observations are not unique. More than one state can generate the same observation ($p(o|s) \leq 1.0$)
  - Specific set of simple actions.
    1. Translate forward 1 hallway unit (to be described)
    2. Rotate right 90°
    3. Rotate left 90°

Navigating with uncertainty: Environment

Sample map of environment



- Randomly generated environments.
- Generated on Cartesian grid.
- Translations move from one red square to another

Navigating with uncertainty: Environment

Sample of navigating through virtual environment

Experiment 1: Procedure

- Training:
  1. Allow free exploration for 3 minutes.
     - One goal position.
     - Indicated by an auditory signal when subject "walks" over the position.
  2. Have subjects draw environment on cartesian grid.
  3. Check if map is accurately drawn.
  4. Return to exploration.
  5. Repeat until environment is drawn correctly twice in a row.

Experiment 1: Procedure

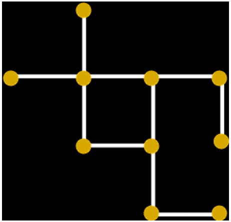
- Testing:
  - Subject starts from a random state in the environment
  - Subject's task is to reach the goal state in the fewest number of actions possible (minimize cost).
    - Every action (translation and rotation) has an equal cost.
    - Subjects have to be certain that they are there and indicate that they are at the goal with a button press.
  - Actions are deterministic ($p(s'|a, s) = 1.0$ or $p(s'|a, s) = 0.0$)
  - Observations are deterministic ($p(o|s) = 1.0$ or $p(o|s) = 0.0$
  - Measure the number of actions to reach the goal state and be certain that they are there.

Experiment 1: Manipulation

- Manipulated the number of hallway units that composed the environment.
- 10, 20, 40, and 80 hallway units.
- Increasing the layout size increased the memory/processing in the four areas that we are interested in understanding.
    - Inefficient processing of observations (O)
    - Inefficient access to their cognitive map (T)
    - Inefficient belief updating (B)
    - Inefficient decision process (POMDP)

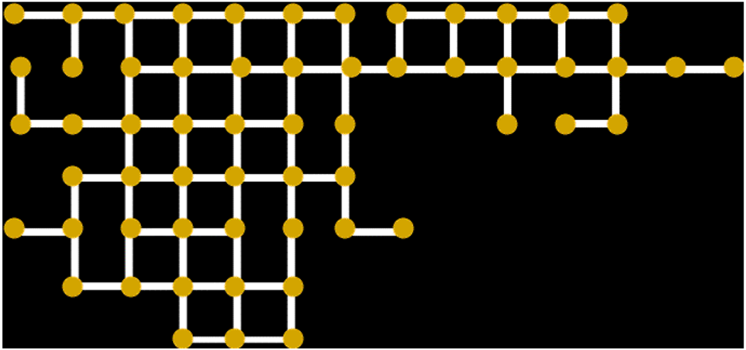
Experiment 1: Environments

10 Hallway Environment



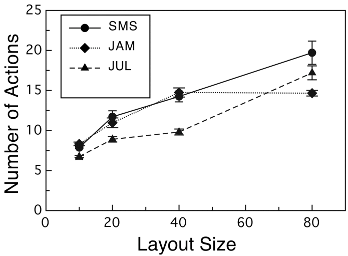
Experiment 1: Environments

80 Hallway Environment



Experiment 1: Results

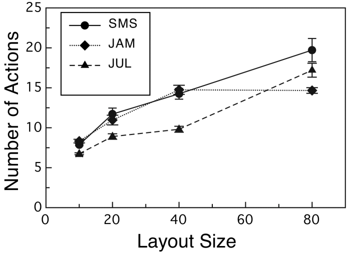
Effect of layout complexity on navigation performance



- Increasing layout size increased number of actions to reach goal state.
- Not too surprising . . . need to travel farther.
- Need to control for task difficulty in different environments.
- Ideal navigator will be sensitive to task difficulty.

Experiment 1: Results

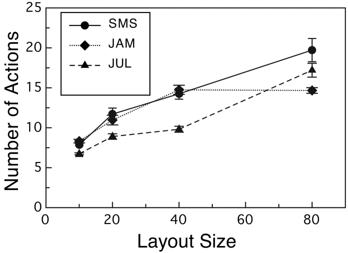
Effect of layout complexity on navigation performance



- Increasing layout size increased number of actions to reach goal state.
- Not too surprising . . . need to travel farther.
- Need to control for task difficulty in different environments.
- Ideal navigator will be sensitive to task difficulty.

Experiment 1: Results

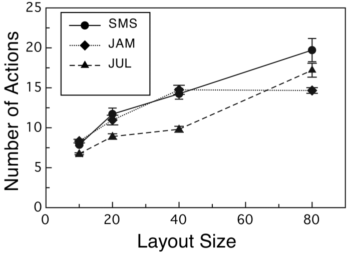
Effect of layout complexity on navigation performance



- Increasing layout size increased number of actions to reach goal state.
- Not too surprising . . . need to travel farther.
- Need to control for task difficulty in different environments.
- Ideal navigator will be sensitive to task difficulty.

Experiment 1: Results

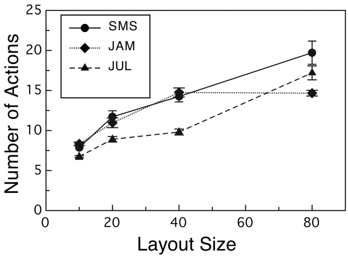
Effect of layout complexity on navigation performance



- Increasing layout size increased number of actions to reach goal state.
- Not too surprising . . . need to travel farther.
- Need to control for task difficulty in different environments.
- Ideal navigator will be sensitive to task difficulty.

Experiment 1: Results

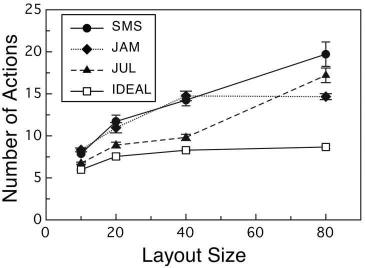
Effect of layout complexity on navigation performance



- Increasing layout size increased number of actions to reach goal state.
- Not too surprising . . . need to travel farther.
- Need to control for task difficulty in different environments.
- Ideal navigator will be sensitive to task difficulty.

Experiment 1: Results

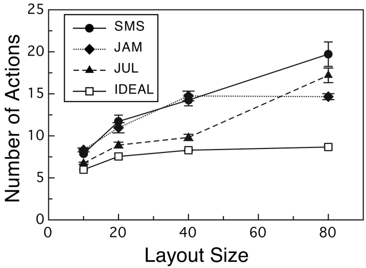
Effect of layout complexity on navigation performance



- Using ideal navigator we can compute the optimal performance.
- POMDP provides optimal performance with no **cognitive limitations**.
- Use performance to get a measure of **efficiency**
  - $Efficiency = \frac{IdealPerformance}{HumanPerformance}$

Experiment 1: Results

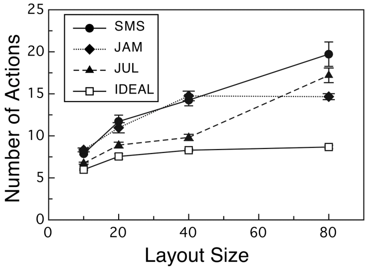
Effect of layout complexity on navigation performance



- Using ideal navigator we can compute the optimal performance.
- POMDP provides optimal performance with no **cognitive limitations**.
- Use performance to get a measure of **efficiency**
  - $Efficiency = \frac{IdealPerformance}{HumanPerformance}$

Experiment 1: Results

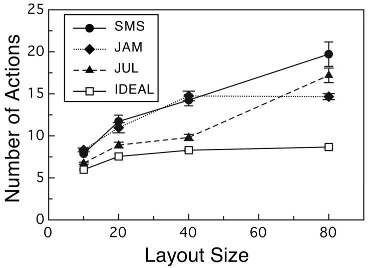
Effect of layout complexity on navigation performance



- Using ideal navigator we can compute the optimal performance.
- POMDP provides optimal performance with no **cognitive limitations**.
- Use performance to get a measure of **efficiency**
  - $Efficiency = \frac{IdealPerformance}{HumanPerformance}$

Experiment 1: Results

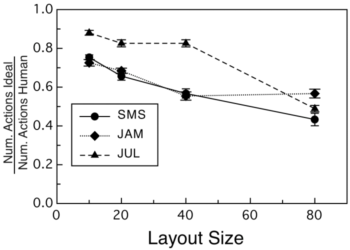
Effect of layout complexity on navigation performance



- Using ideal navigator we can compute the optimal performance.
- POMDP provides optimal performance with no **cognitive limitations**.
- Use performance to get a measure of **efficiency**
  - $Efficiency = \frac{IdealPerformance}{HumanPerformance}$

Experiment 1: Results

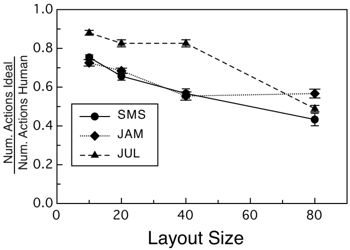
Effect of layout complexity on navigation performance



- Increasing layout complexity decreased navigation efficiency.
- Controlled for task difficulty using ideal navigator.

Experiment 1: Results

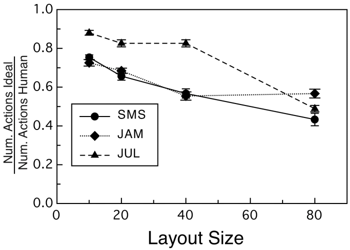
Effect of layout complexity on navigation performance



- Increasing layout complexity decreased navigation efficiency.
- Controlled for task difficulty using ideal navigator.

Experiment 1: Results

Effect of layout complexity on navigation performance



- Increasing layout complexity decreased navigation efficiency.
- Controlled for task difficulty using ideal navigator.

Experiment 1: Summary

- **Why** does increasing layout complexity reduce human efficiency when navigating with uncertainty?
- **Inefficient visual processing**? (Experiment 2)
  - Longer hallways in larger environments.
  - Difficult to process all the way to the end.
- **Limited access to cognitive map**? (Experiment 3)
  - More to remember in larger environments.
  - Perhaps difficult to access entire map when localizing.
- **Belief updating**? (Experiment 3)
  - More uncertainty in larger environment.
  - Have to consider more places in larger environments.
- **Suboptimal Decision Process**? (Experiment 3)
  - More decision needed to be made in larger environment.

Experiment 1: Summary

- **Why** does increasing layout complexity reduce human efficiency when navigating with uncertainty?
- **Inefficient visual processing**? (Experiment 2)
  - Longer hallways in larger environments.
  - Difficult to process all the way to the end.
- **Limited access to cognitive map**? (Experiment 3)
  - More to remember in larger environments.
  - Perhaps difficult to access entire map when localizing.
- **Belief updating**? (Experiment 3)
  - More uncertainty in larger environment.
  - Have to consider more places in larger environments.
- **Suboptimal Decision Process**? (Experiment 3)
  - More decision needed to be made in larger environment.

Experiment 1: Summary

- **Why** does increasing layout complexity reduce human efficiency when navigating with uncertainty?
- **Inefficient visual processing**? (Experiment 2)
  - Longer hallways in larger environments.
  - Difficult to process all the way to the end.
- **Limited access to cognitive map**? (Experiment 3)
  - More to remember in larger environments.
  - Perhaps difficult to access entire map when localizing.
- **Belief updating**? (Experiment 3)
  - More uncertainty in larger environment.
  - Have to consider more places in larger environments.
- **Suboptimal Decision Process**? (Experiment 3)
  - More decision needed to be made in larger environment.

Experiment 1: Summary

- **Why** does increasing layout complexity reduce human efficiency when navigating with uncertainty?
- **Inefficient visual processing**? (Experiment 2)
  - Longer hallways in larger environments.
  - Difficult to process all the way to the end.
- **Limited access to cognitive map**? (Experiment 3)
  - More to remember in larger environments.
  - Perhaps difficult to access entire map when localizing.
- **Belief updating**? (Experiment 3)
  - More uncertainty in larger environment.
  - Have to consider more places in larger environments.
- **Suboptimal Decision Process**? (Experiment 3)
  - More decision needed to be made in larger environment.

Experiment 1: Summary

- **Why** does increasing layout complexity reduce human efficiency when navigating with uncertainty?
- **Inefficient visual processing**? (Experiment 2)
  - Longer hallways in larger environments.
  - Difficult to process all the way to the end.
- **Limited access to cognitive map**? (Experiment 3)
  - More to remember in larger environments.
  - Perhaps difficult to access entire map when localizing.
- **Belief updating**? (Experiment 3)
  - More uncertainty in larger environment.
  - Have to consider more places in larger environments.
- **Suboptimal Decision Process**? (Experiment 3)
  - More decision needed to be made in larger environment.

Experiment 1: Summary

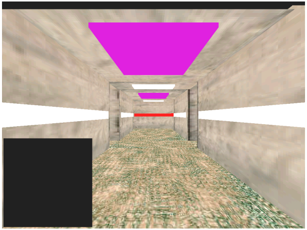
- **Why** does increasing layout complexity reduce human efficiency when navigating with uncertainty?
- **Inefficient visual processing**? (Experiment 2)
  - Longer hallways in larger environments.
  - Difficult to process all the way to the end.
- **Limited access to cognitive map**? (Experiment 3)
  - More to remember in larger environments.
  - Perhaps difficult to access entire map when localizing.
- **Belief updating**? (Experiment 3)
  - More uncertainty in larger environment.
  - Have to consider more places in larger environments.
- **Suboptimal Decision Process**? (Experiment 3)
  - More decision needed to be made in larger environment.

Experiment 3: Introduction

- Provide subject with explicit knowledge that should be available internally.
- Information provided is already available in the task.
- Present information on computer screen using a "heads-up" display.
- Three conditions:
  1. No Map
  2. Map
  3. Map + Belief Vector

Experiment 3: Introduction

No Map

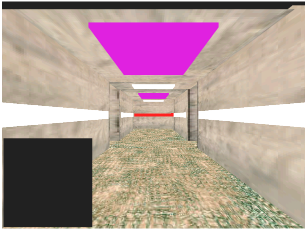


Map

Map + Belief Vector

Experiment 3: Introduction

No Map

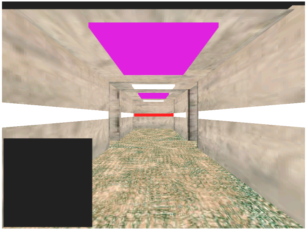


Map

Map + Belief Vector

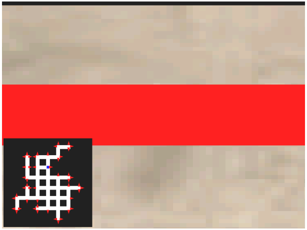
Experiment 3: Introduction

No Map



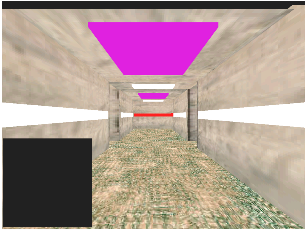
Map

Map + Belief Vector



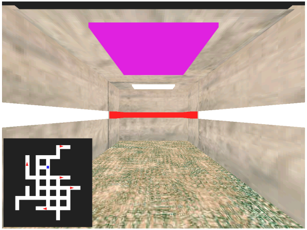
Experiment 3: Introduction

No Map



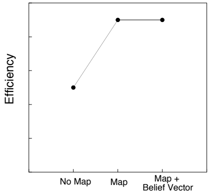
Map

Map + Belief Vector



Experiment 3: Predictions

Accessing Cognitive Map

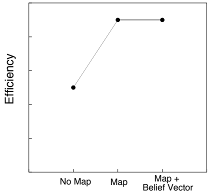


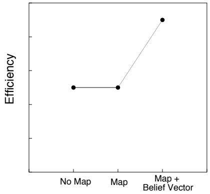
Belief Updating

Map + Belief Vector

Experiment 3: Predictions

Accessing Cognitive Map Belief Updating

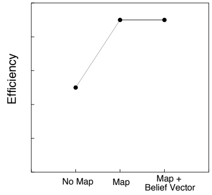




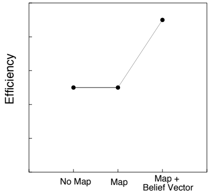
Map + Belief Vector

Experiment 3: Predictions

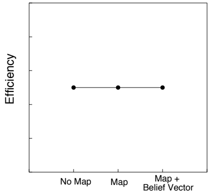
Accessing Cognitive Map



Belief Updating



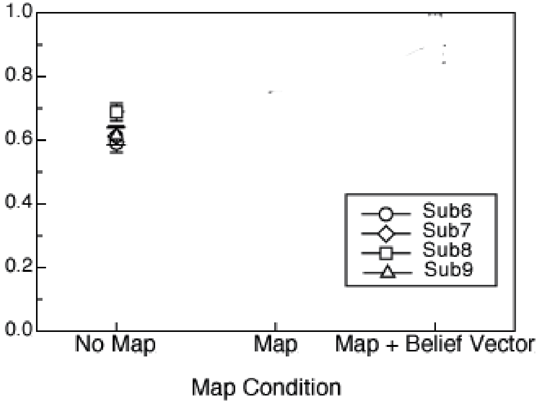
Map + Belief Vector



Experiment 3: Design

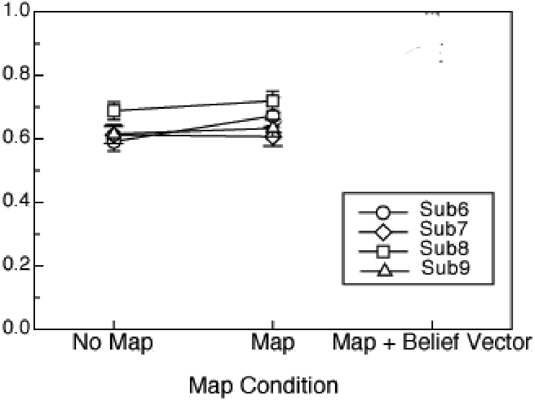
- Training: Same as Experiment 1
- Test: Same as Experiment 1 with an additional "heads-up" map display
- Three map conditions:
  1. No Map
  2. Map
  3. Map + Belief Vector

Experiment 3: Results



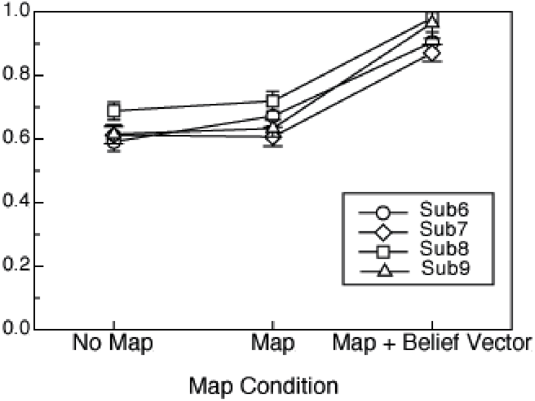
- Improved performance when belief vector is made explicit.
- Suggests that the limiting cognitive factor in navigating with uncertainty is an inefficient belief updating process.

Experiment 3: Results



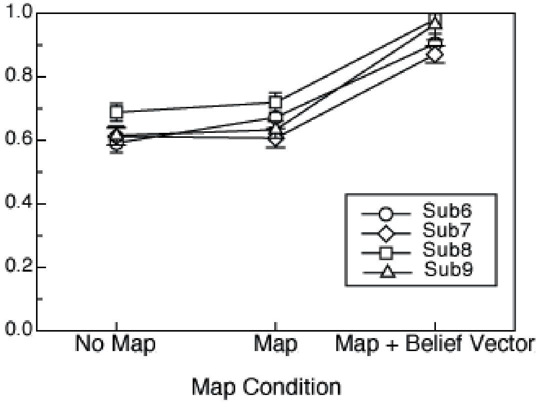
- Improved performance when belief vector is made explicit.
- Suggests that the limiting cognitive factor in navigating with uncertainty is an inefficient belief updating process.

Experiment 3: Results



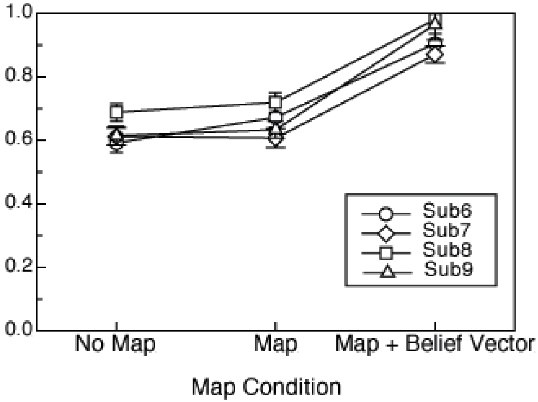
- Improved performance when belief vector is made explicit.
- Suggests that the limiting cognitive factor in navigating with uncertainty is an inefficient belief updating process.

Experiment 3: Results



- Improved performance when belief vector is made explicit.
- Suggests that the limiting cognitive factor in navigating with uncertainty is an inefficient belief updating process.

Experiment 3: Results



- Improved performance when belief vector is made explicit.
- Suggests that the limiting cognitive factor in navigating with uncertainty is an inefficient belief updating process.

Summary

- Investigated human sequential decision making under uncertainty.
- Used Ideal Observer analysis (Bayesian; POMDP) to study cognitive limitations in two different tasks.
- Localized cognitive bottleneck in generating, updating and maintaining an accurate belief vector.
- Currently studying if effect generalizes to non-navigational task.
  - Seek & Destroy problem
  - Results suggest belief updating is also the bottleneck.

Summary

- Investigated human sequential decision making under uncertainty.
- Used Ideal Observer analysis (Bayesian; POMDP) to study cognitive limitations in two different tasks.
- Localized cognitive bottleneck in generating, updating and maintaining an accurate belief vector.
- Currently studying if effect generalizes to non-navigational task.
    - Seek & Destroy problem
    - Results suggest belief updating is also the bottleneck.

Summary

- Investigated human sequential decision making under uncertainty.
- Used Ideal Observer analysis (Bayesian; POMDP) to study cognitive limitations in two different tasks.
- Localized cognitive bottleneck in generating, updating and maintaining an accurate belief vector.
- Currently studying if effect generalizes to non-navigational task.
  - Seek & Destroy problem
  - Results suggest belief updating is also the bottleneck.

Summary

- Investigated human sequential decision making under uncertainty.
- Used Ideal Observer analysis (Bayesian; POMDP) to study cognitive limitations in two different tasks.
- Localized cognitive bottleneck in generating, updating and maintaining an accurate belief vector.
- Currently studying if effect generalizes to non-navigational task.
  - Seek & Destroy problem
  - Results suggest belief updating is also the bottleneck.

Summary

- Investigated human sequential decision making under uncertainty.
- Used Ideal Observer analysis (Bayesian; POMDP) to study cognitive limitations in two different tasks.
- Localized cognitive bottleneck in generating, updating and maintaining an accurate belief vector.
- Currently studying if effect generalizes to non-navigational task.
    - Seek & Destroy problem
    - Results suggest belief updating is also the bottleneck.

Summary

- Interesting note:
  - Computationally, updating and generating an accurate belief vector is fast and easy for the computer.
  - However for human, it is difficult
  - Selecting the optimal action given the belief vector is difficult computationally.
  - However, data suggests that it is relatively easy for the human.
  - Suggests a symbiotic relationship between computer and human.

Summary

- Interesting note:
  - Computationally, updating and generating an accurate belief vector is fast and easy for the computer.
  - However for human, it is difficult
  - Selecting the optimal action given the belief vector is difficult computationally.
  - However, data suggests that it is relatively easy for the human.
  - Suggests a symbiotic relationship between computer and human.

Summary

- Interesting note:
  - Computationally, updating and generating an accurate belief vector is fast and easy for the computer.
  - However for human, it is difficult
  - Selecting the optimal action given the belief vector is difficult computationally.
  - However, data suggests that it is relatively easy for the human.
  - Suggests a symbiotic relationship between computer and human.

Summary

- Interesting note:
  - Computationally, updating and generating an accurate belief vector is fast and easy for the computer.
  - However for human, it is difficult
  - Selecting the optimal action given the belief vector is difficult computationally.
  - However, data suggests that it is relatively easy for the human.
  - Suggests a symbiotic relationship between computer and human.

Summary

- Interesting note:
  - Computationally, updating and generating an accurate belief vector is fast and easy for the computer.
  - However for human, it is difficult
  - Selecting the optimal action given the belief vector is difficult computationally.
  - However, data suggests that it is relatively easy for the human.
  - Suggests a symbiotic relationship between computer and human.

Summary

- Interesting note:
  - Computationally, updating and generating an accurate belief vector is fast and easy for the computer.
  - However for human, it is difficult
  - Selecting the optimal action given the belief vector is difficult computationally.
  - However, data suggests that it is relatively easy for the human.
  - Suggests a symbiotic relationship between computer and human.

Thank you

Thank You