Eye Movements

Daniela Pamplona

U2IS - ENSTA - IPParis

ecampus moodle: MI210 - Modèles neuro-computationnels de la vision (P4 - 2020-21)

daniela.pamplona@ensta.fr

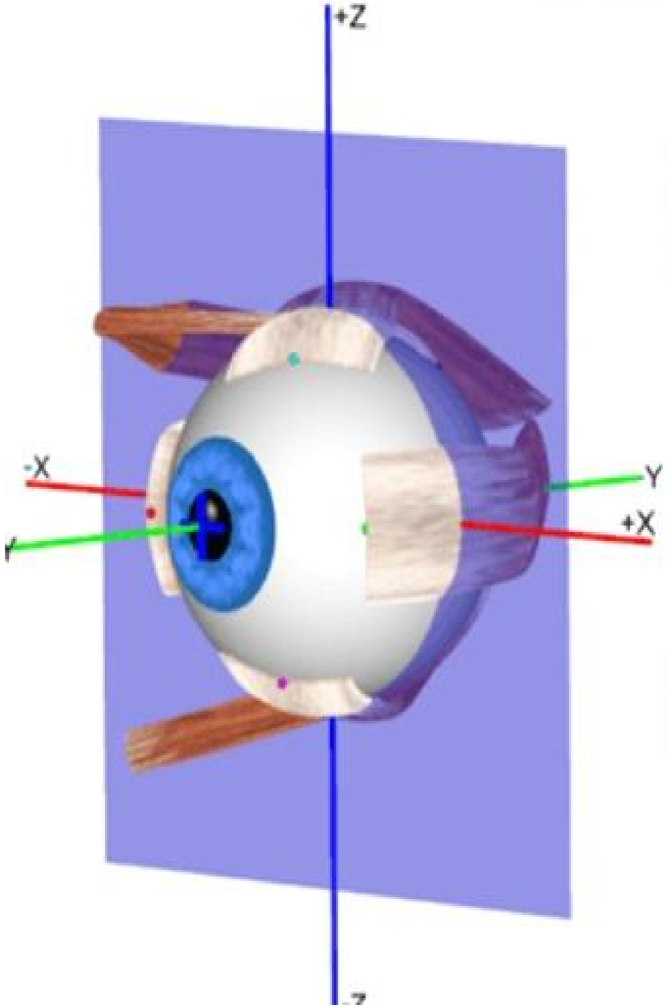
Contents

1. Major types of eye movements
2. What triggers saccades?
3. Reinforcement learning
    1. Markov decision processes
    2. Q learning with Monte Carlo and SARSA
4. Eye movements to learn to solve visually guided tasks

Contents

1. Major types of eye movements
2. What triggers saccades?
3. Reinforcement learning
    1. Markov decision processes
    2. Q learning with Monte Carlo and SARSA
4. Eye movements to learn to solve visually guided tasks

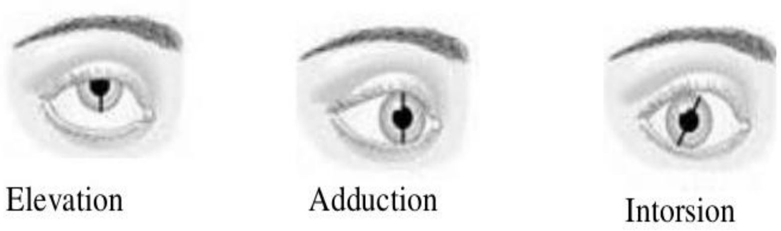
3 groups of eye movements

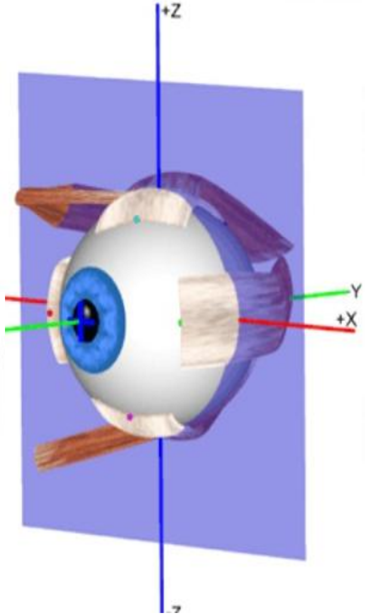


https://www.tapatalk.com/groups/geoffmangumsflatstickforum/eyeballs-and-golfballs-gaze-and-neural-imaging-t910.html

3 groups of eye movements

- Ductions: isolated movements of a single eye





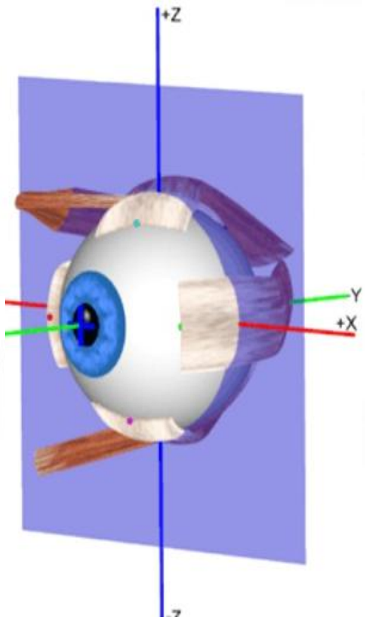
3 groups of eye movements

- Ductions: isolated movements of a single eye

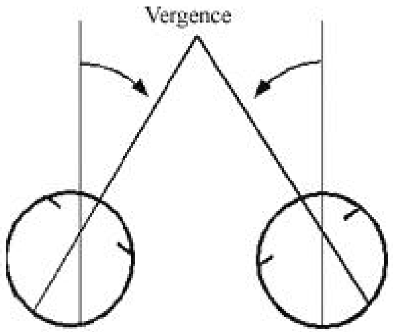
Elevation

Adduction

Intorsion



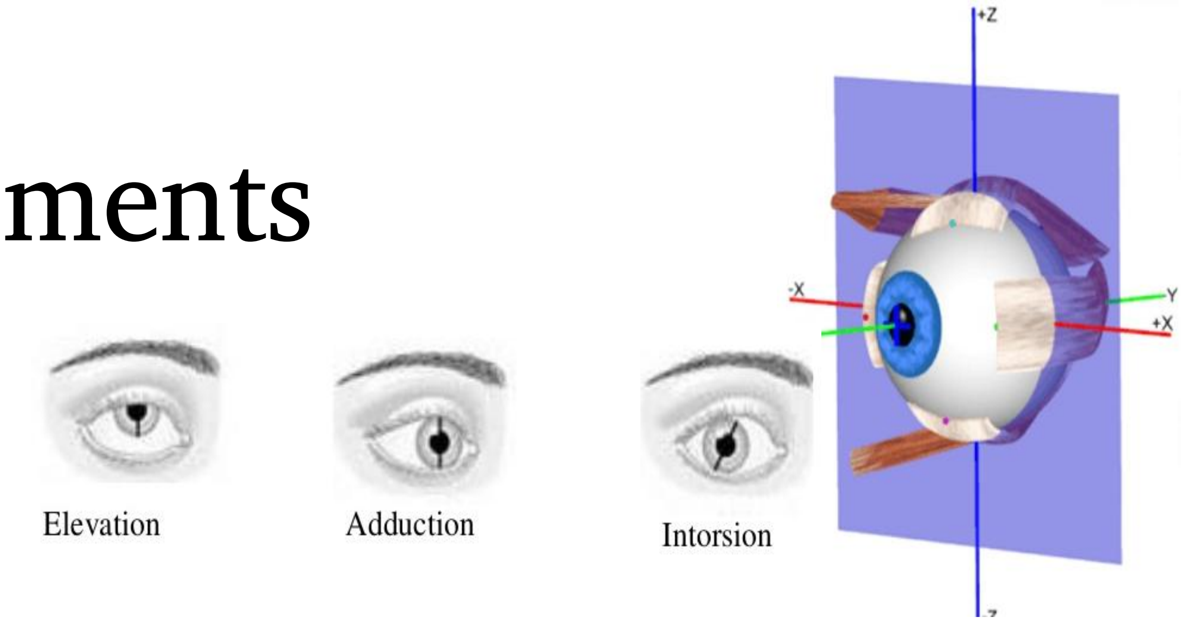
- Vergences: vergence movements are mirror image movements, being equal and opposite



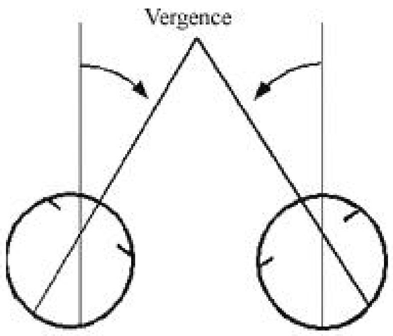
http://schorlab.berkeley.edu/passpro/oculomotor/html/chapter_19.html

3 groups of eye movements

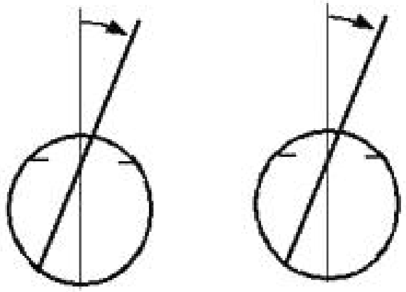
- Ductions: isolated movements of a single eye



- Vergences: vergence movements are mirror image movements, being equal and opposite



- Versions: conjugate gaze movements of both eyes



http://schorlab.berkeley.edu/passpro/oculomotor/html/chapter_19.html

Major types of versions movements

Image stabilization during body movements (involuntary)

Major types of versions movements

Image stabilization during body movements (involuntary)

a) Vestibulo–ocular reflex: compensate head movement to keep object in the fovea, max velo: 350 deg/s

b) Optokinetic reflex: compensate large motions on the visual field, max velo:??

Major types of versions movements

Image tracking/search of objects,
acquiring information (voluntaty)

Major types of versions movements

Image tracking/search of objects, acquiring information (voluntaty)

a) Smooth pursuit: follow continuously a moving object, max velo: 100deg/sec

b) Saccades: bring object to fovea, jerky and abrupt, max velo: 900deg/sec

Major types of versions movements

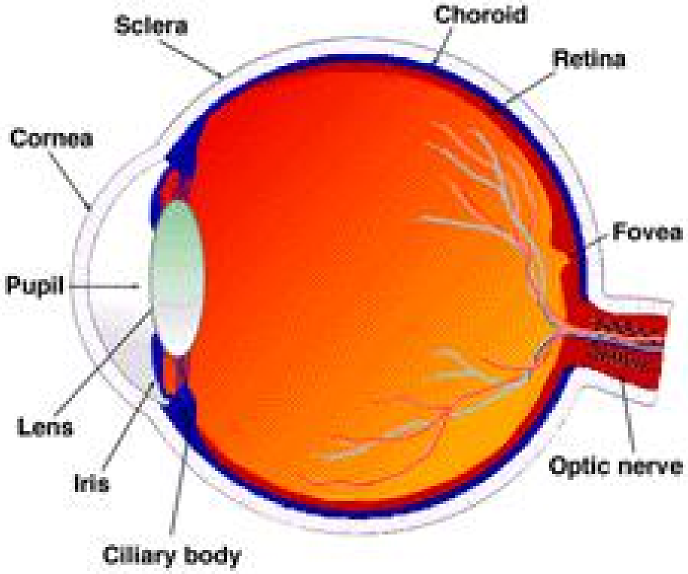
Image tracking/search of objects, acquiring information (voluntaty)

a) Smooth pursuit: follow continuously a moving object, max velo: 100deg/sec

b) Saccades: bring object to fovea, jerky and abrupt, max velo: 900deg/sec

Why do we need to bring objects to the fovea?

Why do we need to bring objects to the fovea?



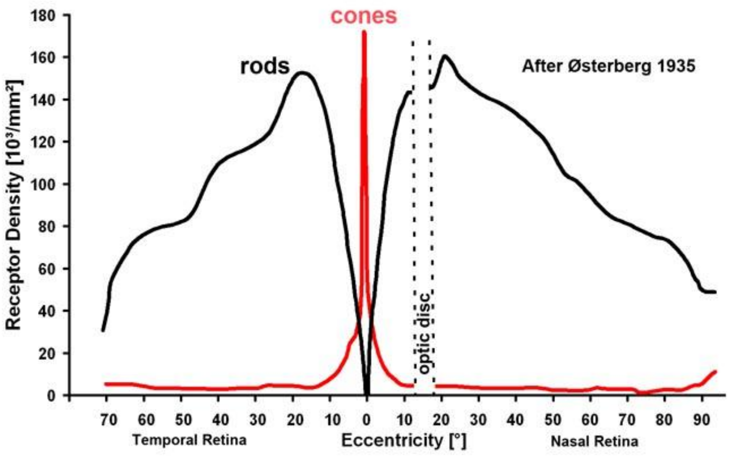
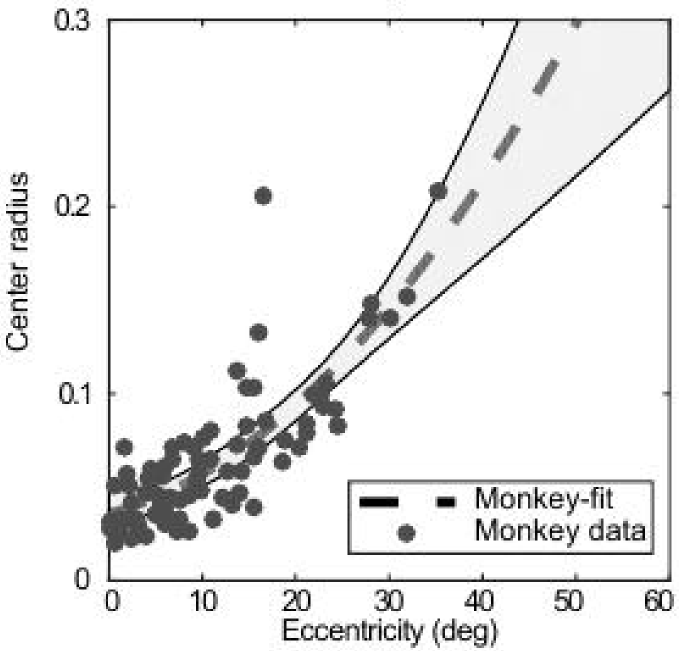
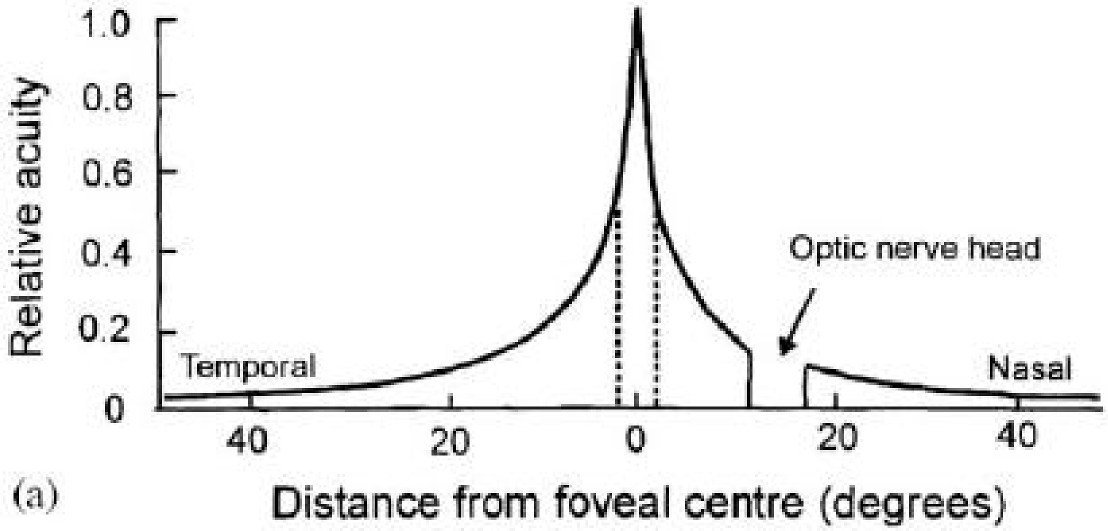


Fig. 20. Graph to show rod and cone densities along the horizontal meridian.



Why do we need to bring objects to the fovea?



Where is the oculomotor control center?

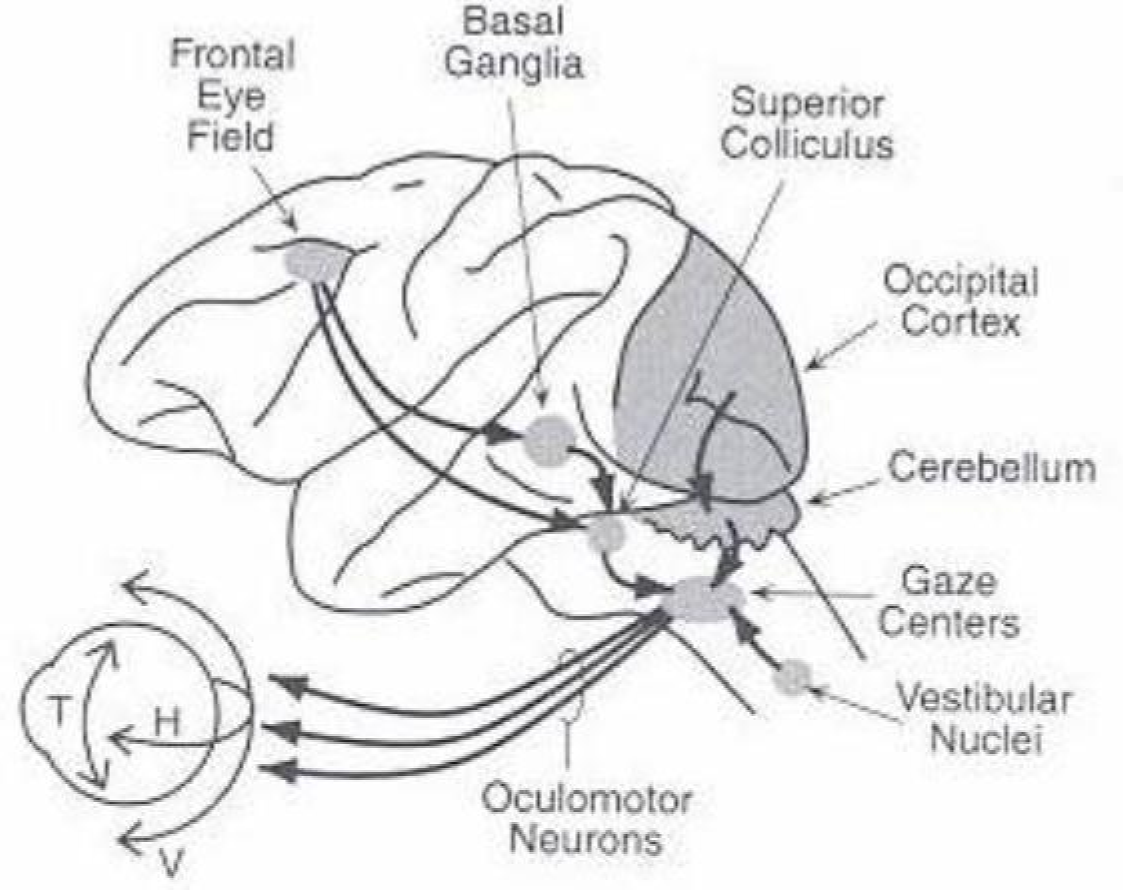


Figure 2.1 Brain areas that control eye movements

Where is the oculomotor control center?

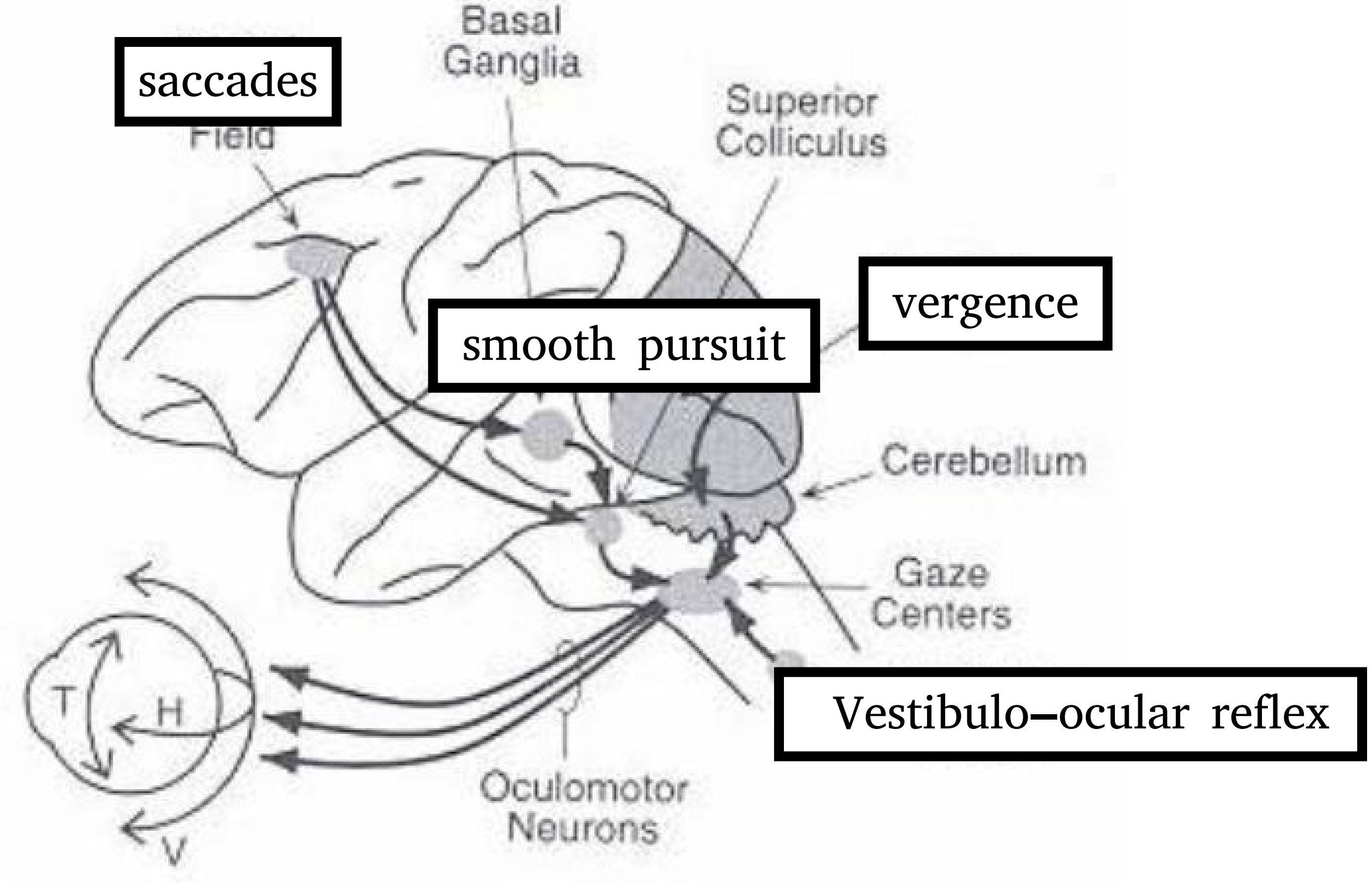


Figure 2.1 Brain areas that control eye movements

Contents

1. Major types of eye movements
2. What triggers saccades?
3. Reinforcement learning
    1. Markov decision processes
    2. Q learning with Monte Carlo and SARSA
4. Eye movements to learn to solve visually guided tasks

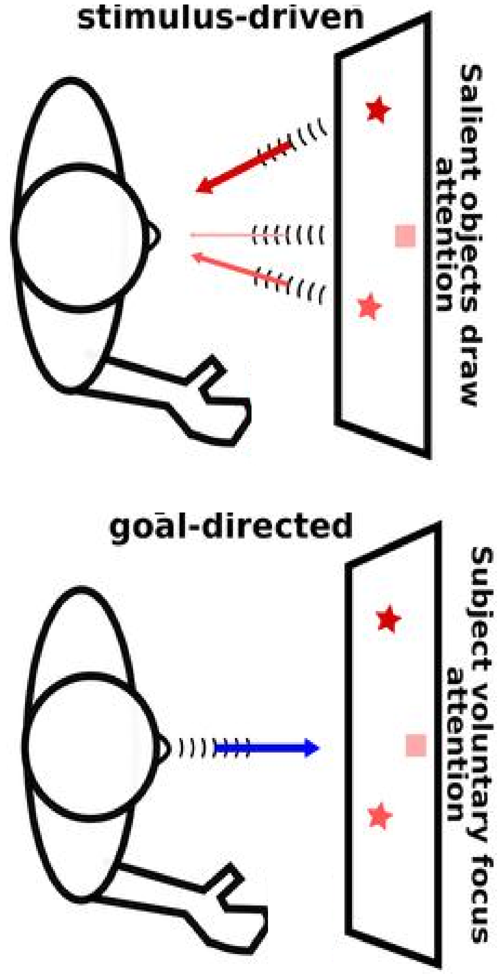
What triggers saccades?

What triggers saccades? Models of attention

1. Bottom up theories

2. Top down theories

3. Mixture


stimulus-driven
Salient objects draw attention
goal-directed
Subject voluntary focus attention

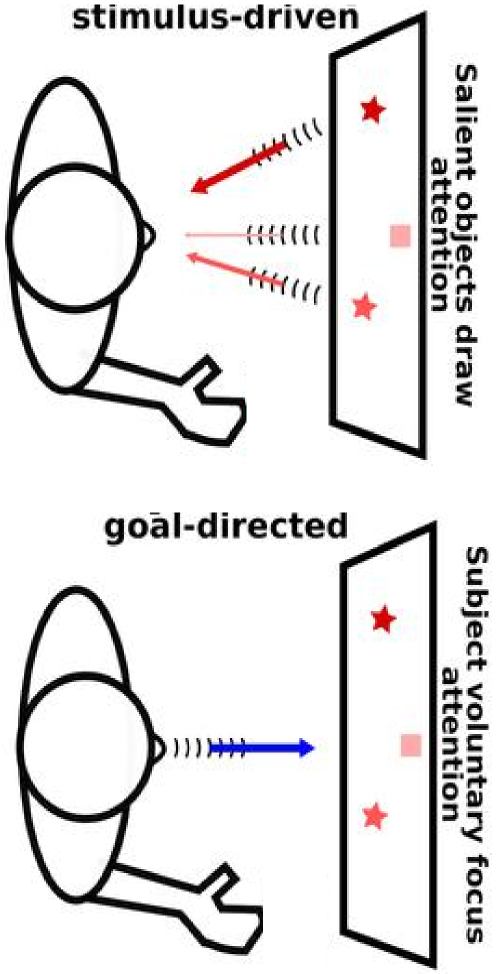
Belkaid M, Cuperlier N, Gaussier P,2017

What triggers saccades? Models of attention

1. Bottom up theories
   1) Saliency
   2) Context

2. Top down theories
   1. Memory
   2. Task

3. Mixture


stimulus-driven
Salient objects draw attention
goal-directed
Subject voluntary focus attention

Belkaid M, Cuperlier N, Gaussier P,2017

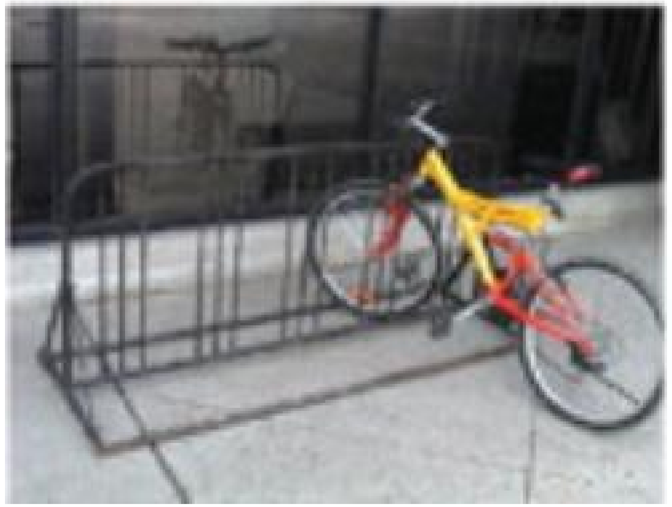
Bottom up: Saliency

Stimulus

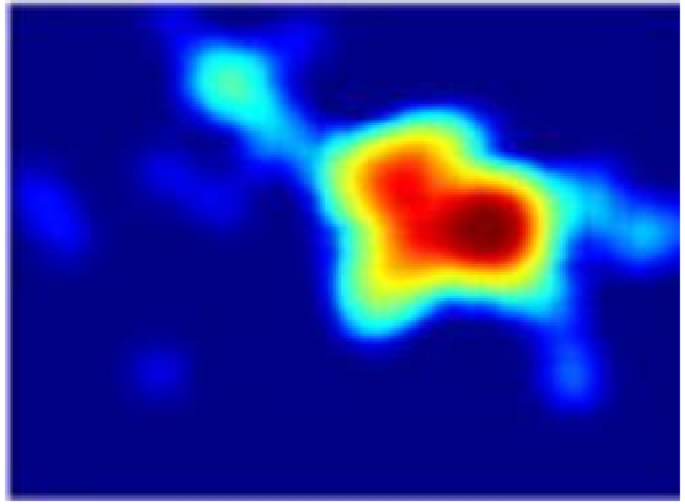
Bruce and Tsotos, 2009

Bottom up: Saliency

Stimulus



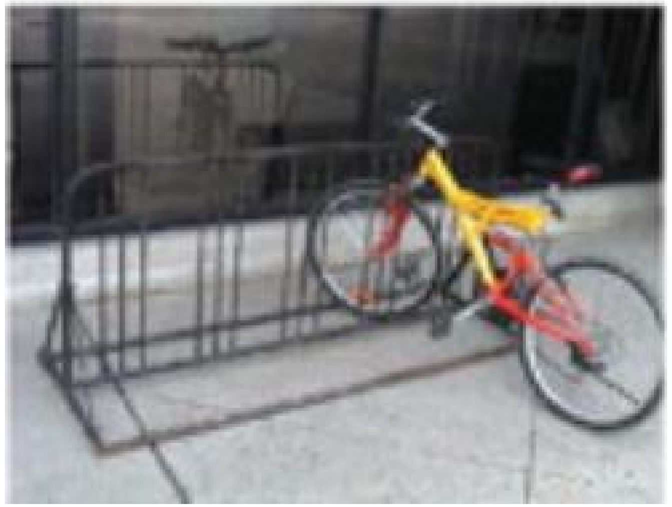
Gazing heat map



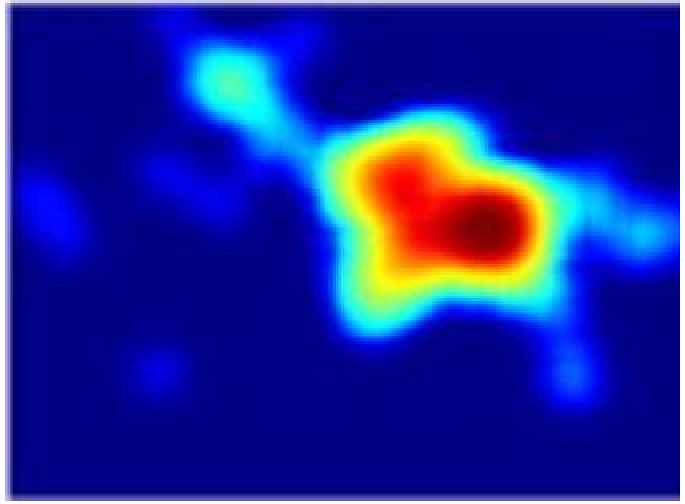
Bruce and Tsotos, 2009

Bottom up: Saliency

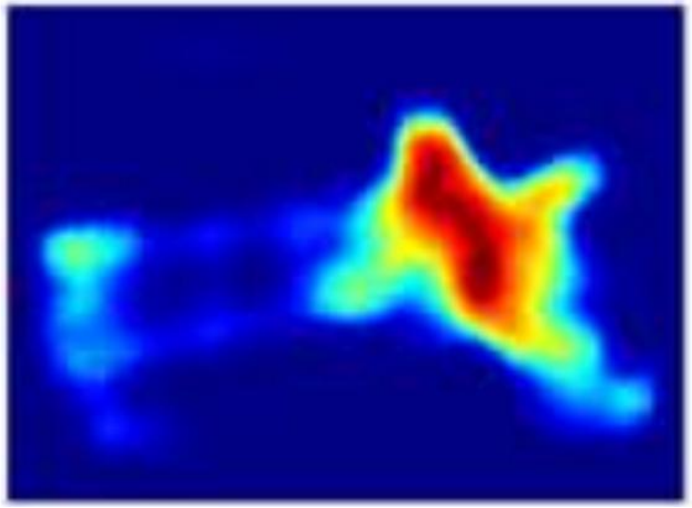
Stimulus



Gazing heat map

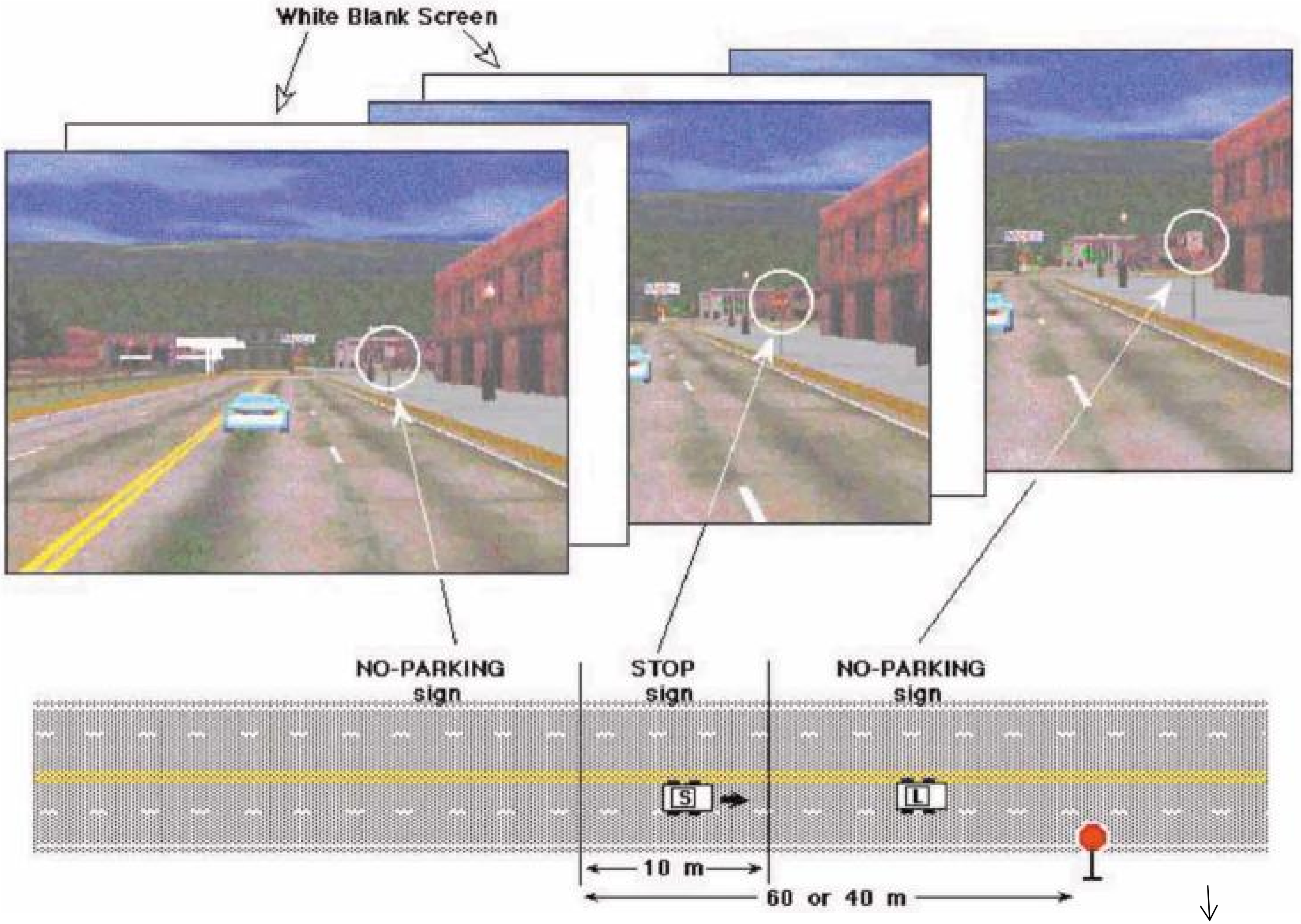


Saliency based on the responses to ICA



Bruce and Tsotos, 2009

Bottom up: Context

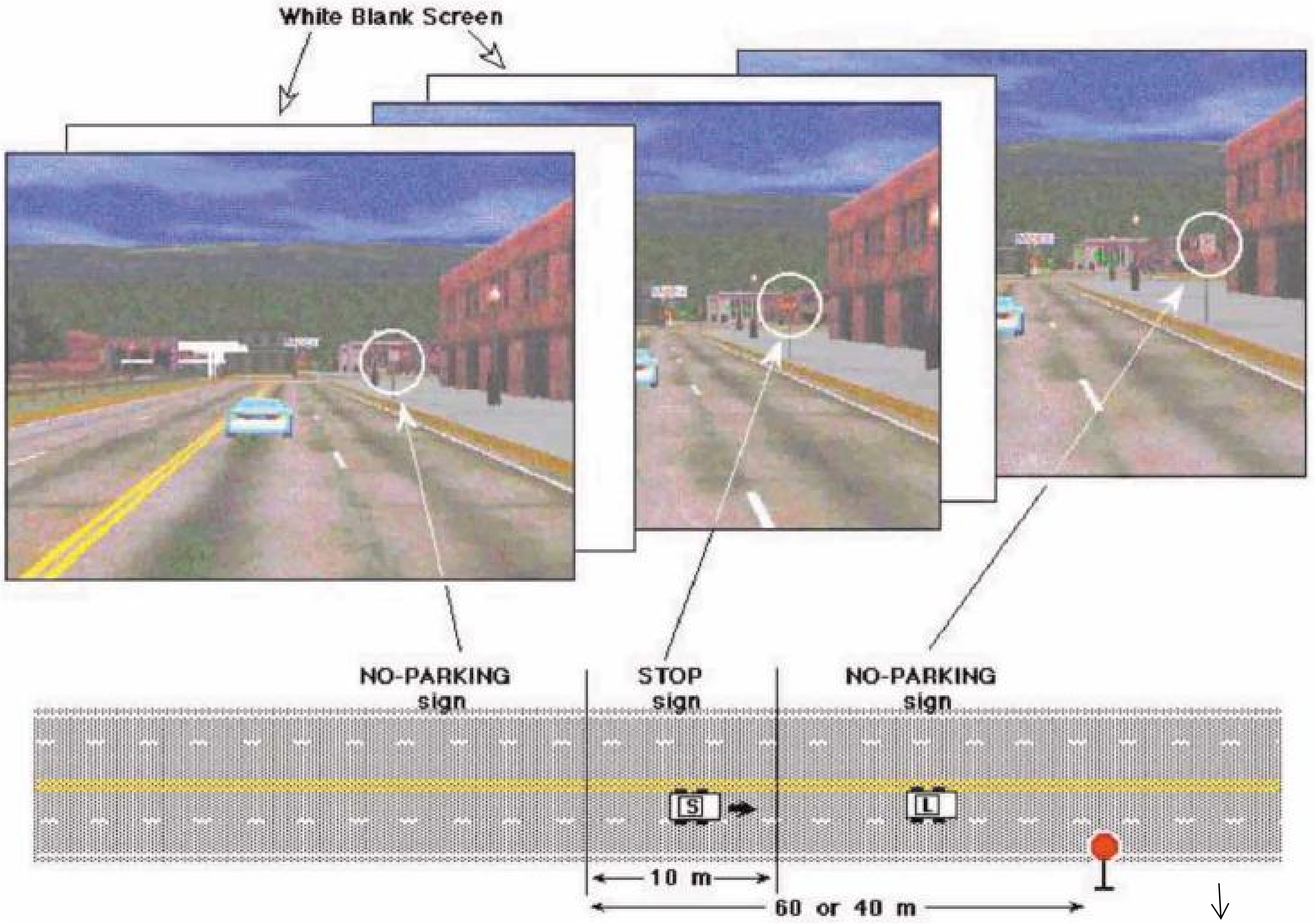


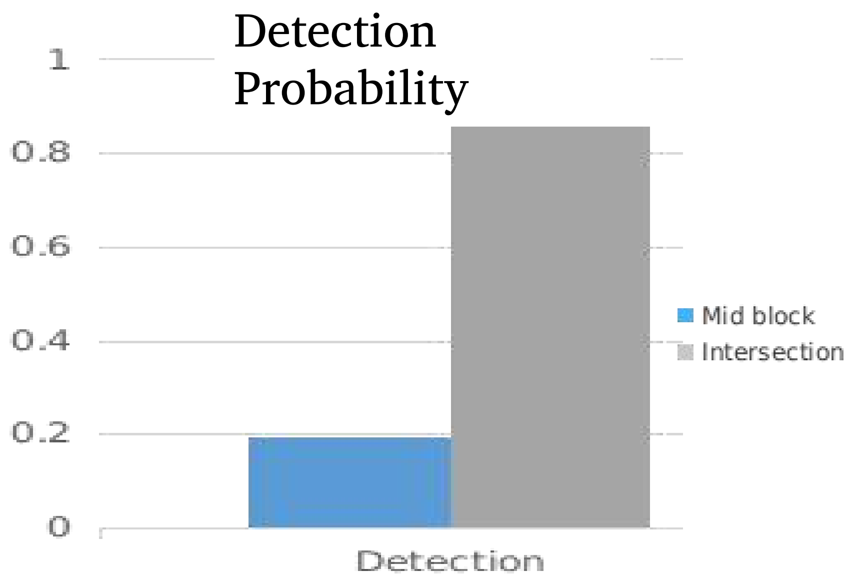
↓

Intersection or mid block

Shinoda et all, 2001

Bottom up: Context





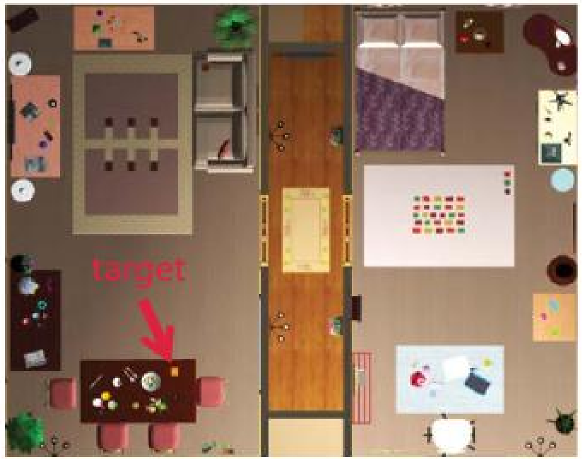
↓

Intersection or mid block

Shinoda et all, 2001

Top down: Memory

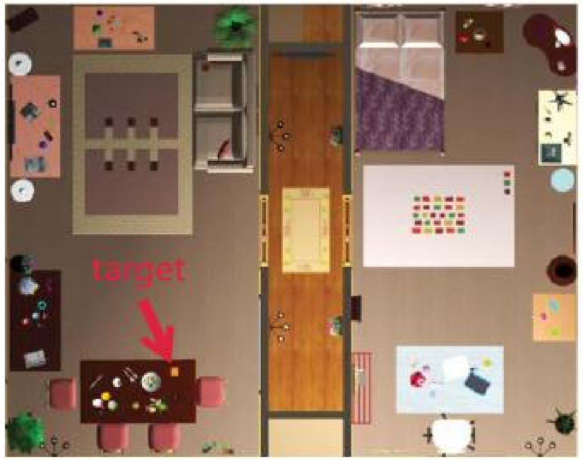
Target

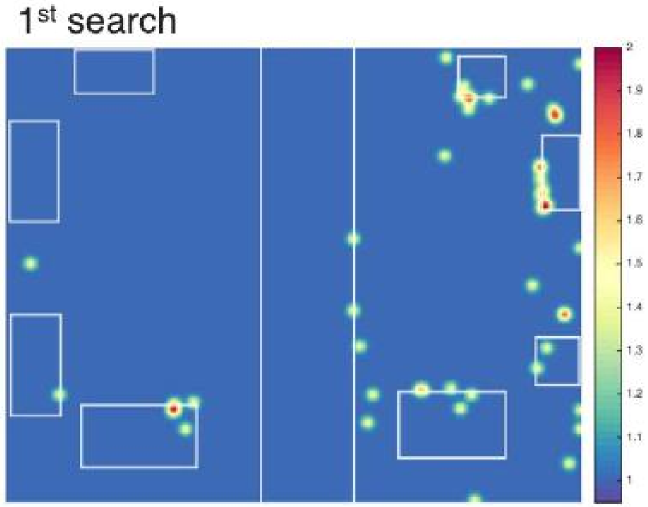


Li et al, 2016

Top down: Memory

Target

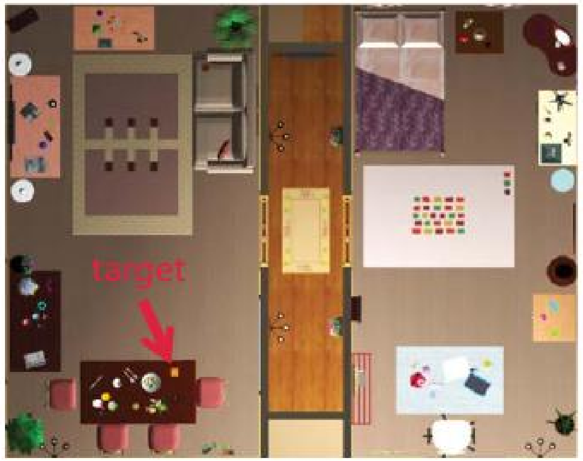




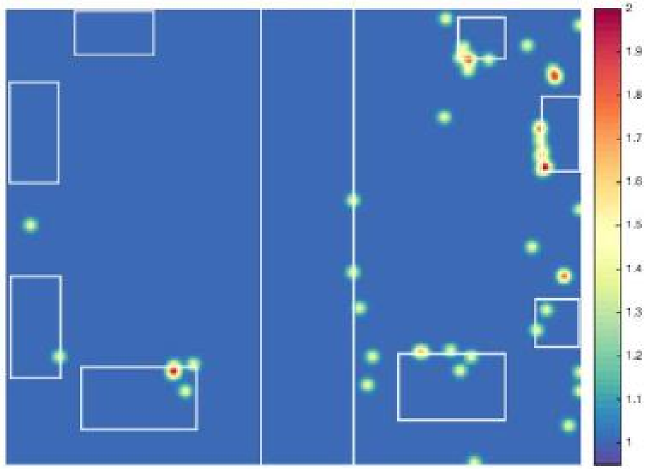
Li et al, 2016

Top down: Memory

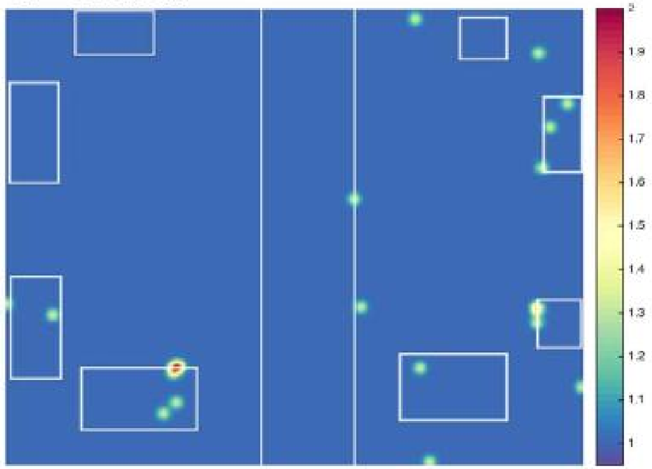
Target



1st search



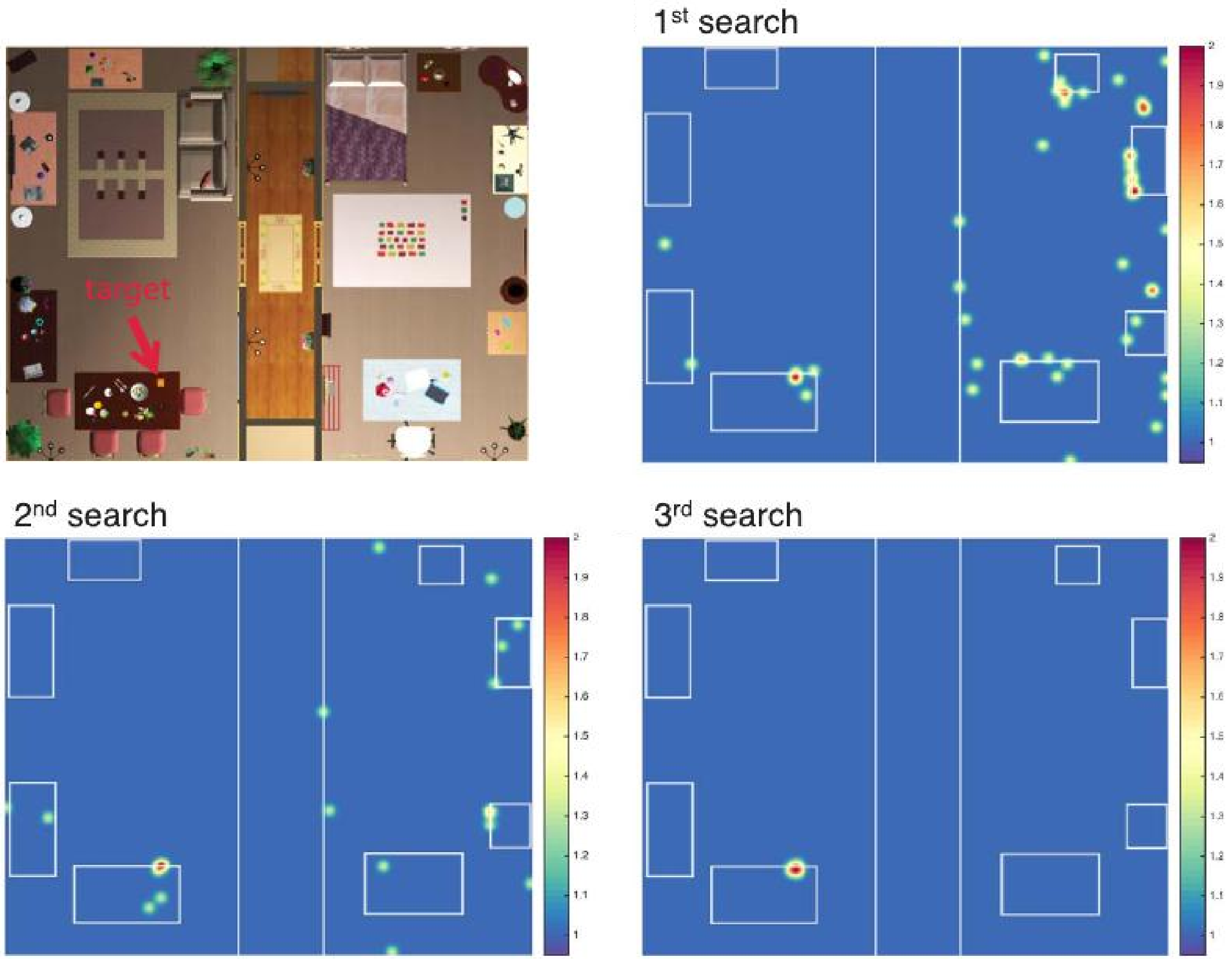
2nd search



Li et al, 2016

Top down: Memory

Target



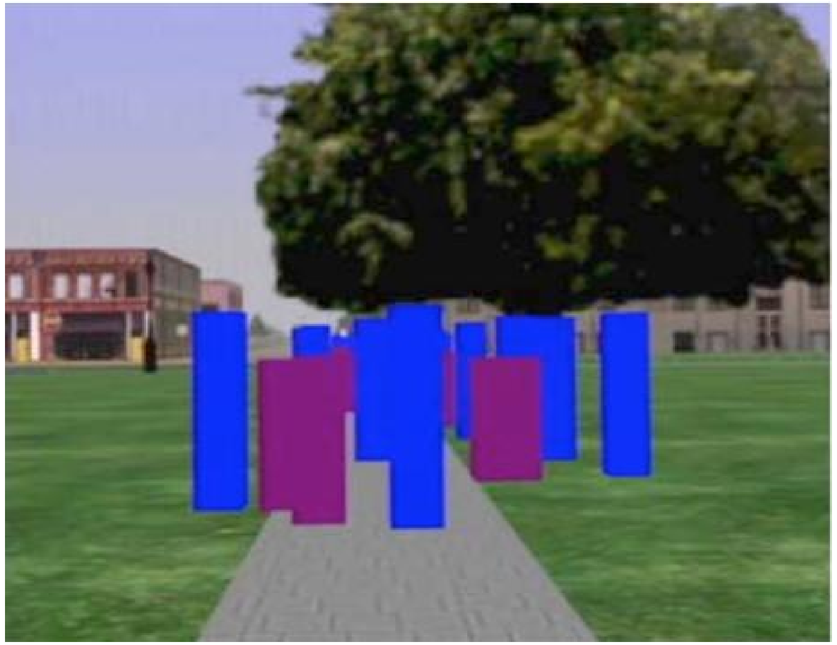
1st search

2nd search

3rd search

Li et al, 2016

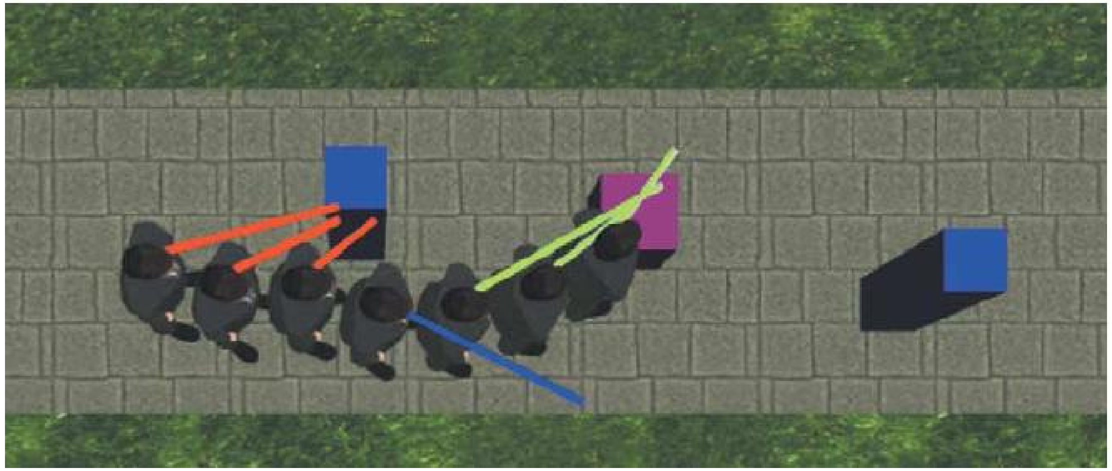
Top down: Task

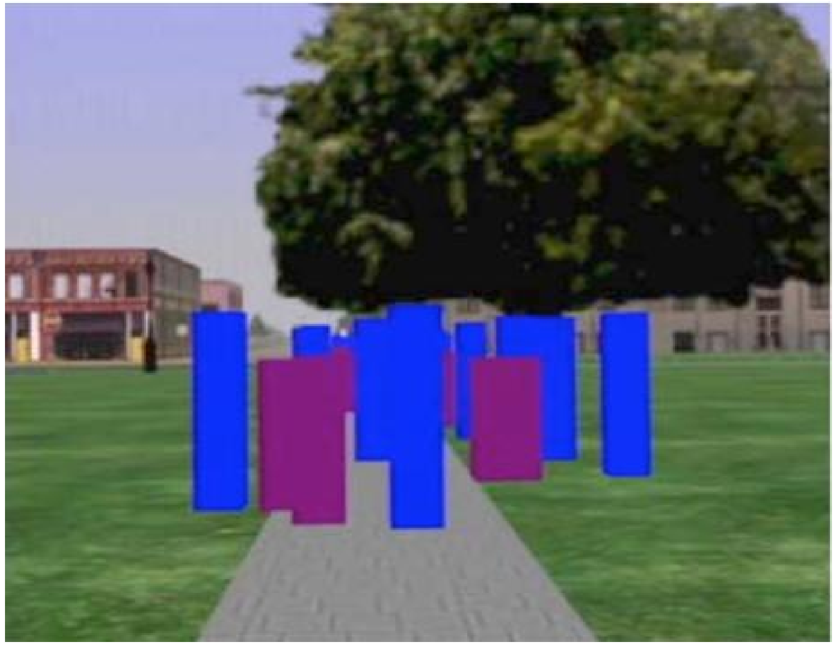


Tasks/behaviors:

1. avoid obstacle
2. sidewalk following
3. pick litter

Top down: Task

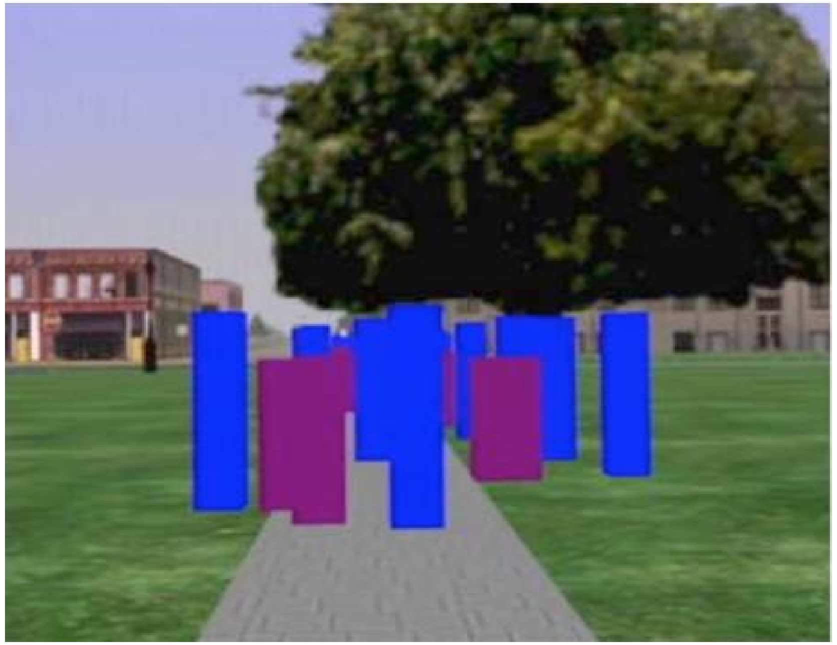


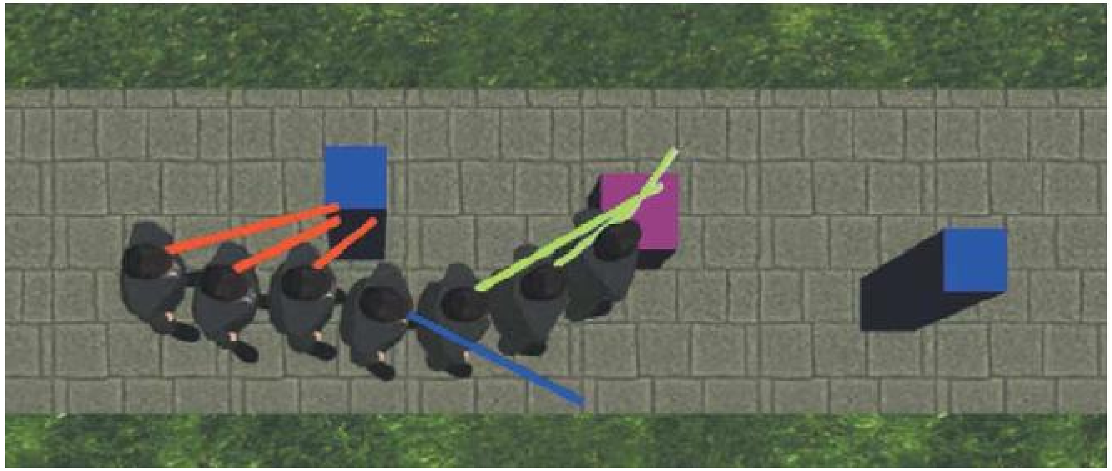


Tasks/behaviors:

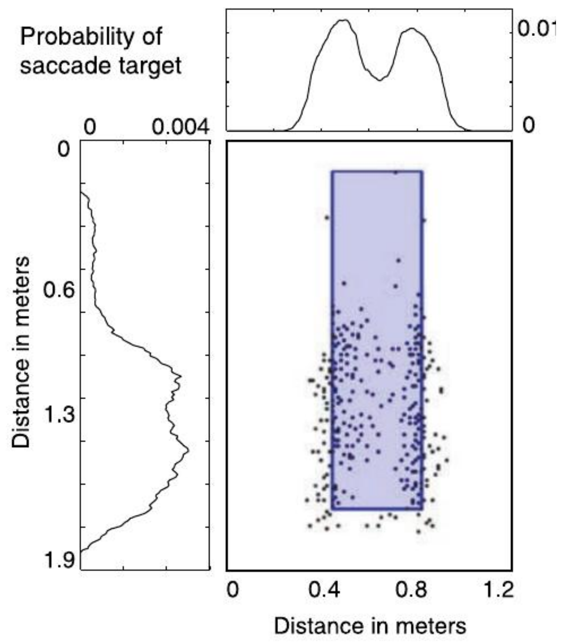
1. avoid obstacle
2. sidewalk following
3. pick litter

Top down: Task

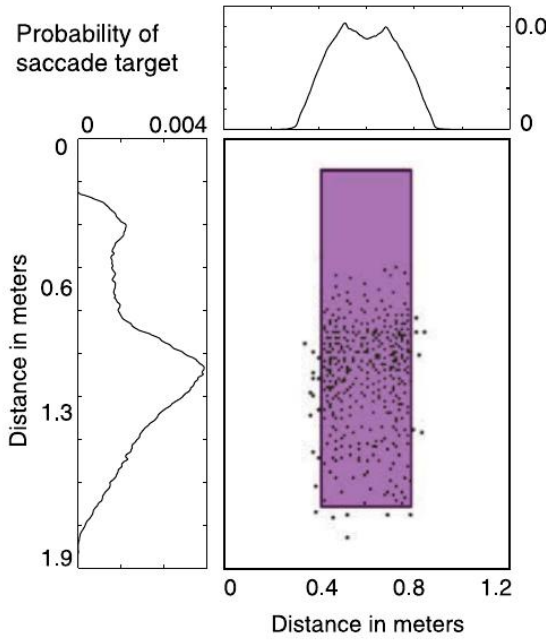




Gaze distribution on obstacles in the “avoid” condition



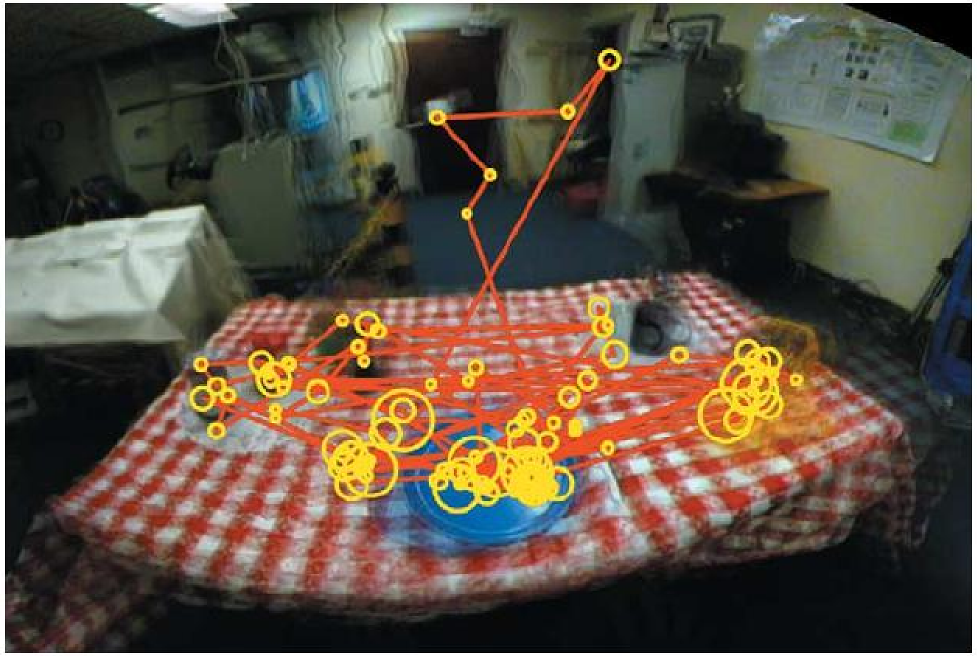
Gaze distribution on litter in the “pickup” condition



Tasks/behaviors:

1. avoid obstacle
2. sidewalk following
3. pick litter

Mixed

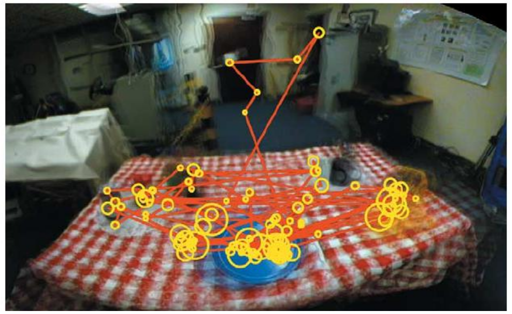


Task: to make a PBJ sandwich

- 50% of objects were irrelevant (scotch tape, forks,etc)
- before task started: 52% of time looking at irrelevant objects
- when task started: 18% of time looking at irrelevant object

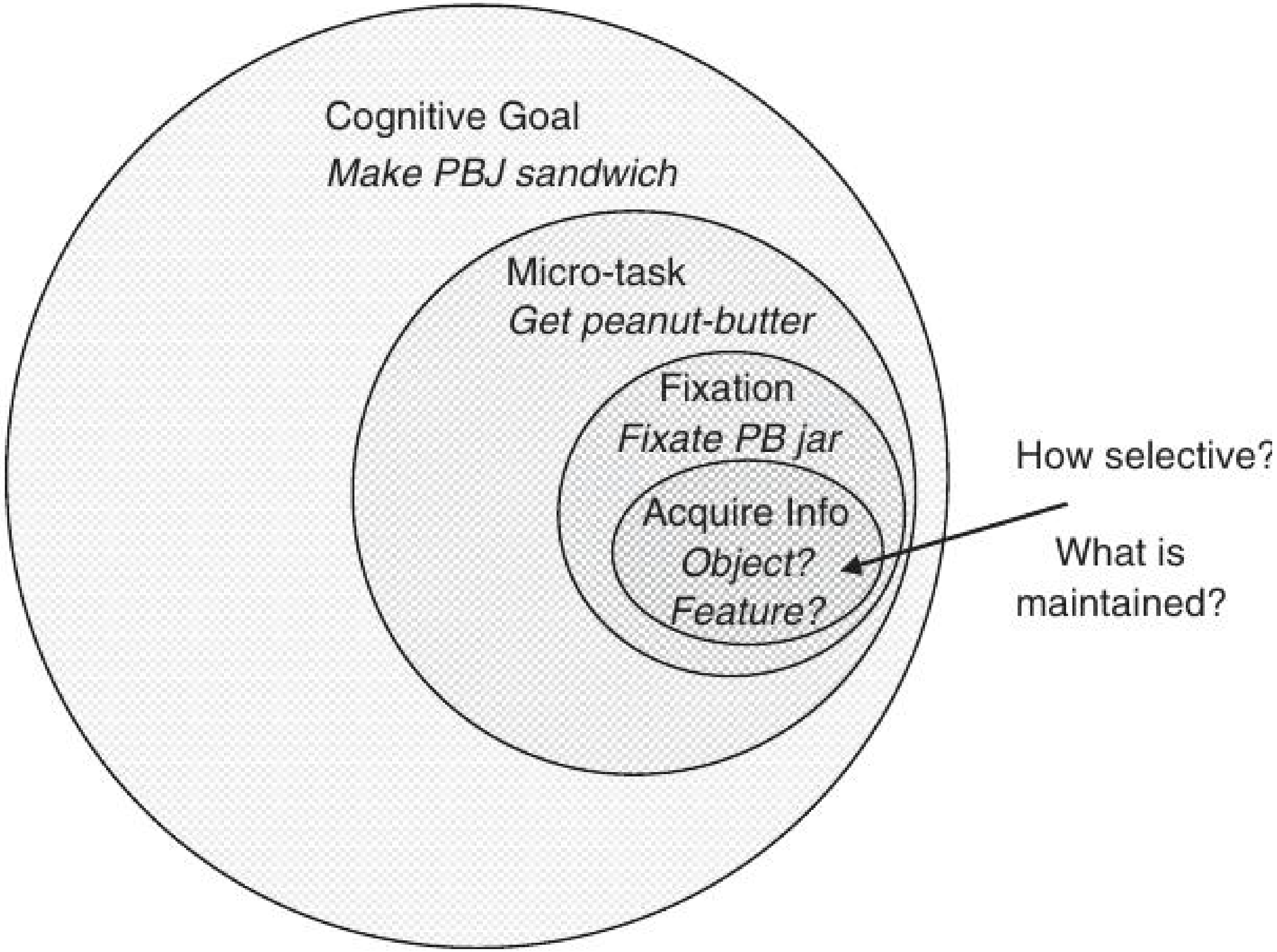
Hayhoe and Rothkopf, 2010
Land, 2006

Mixed



Task: to make a PBJ sandwich

- 50% of objects were irrelevant (scotch tape, forks,etc)
- before task started: 52% of time looking at irrelevant objects
- when task started: 18% of time looking at irrelevant object



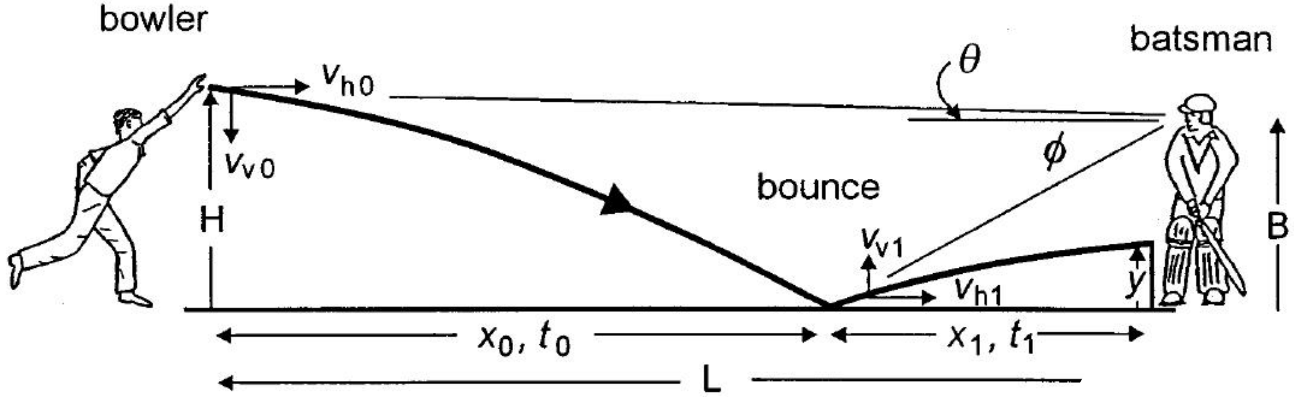
Hayhoe and Rothkopf, 2010
Land, 2006

Break!

Contents

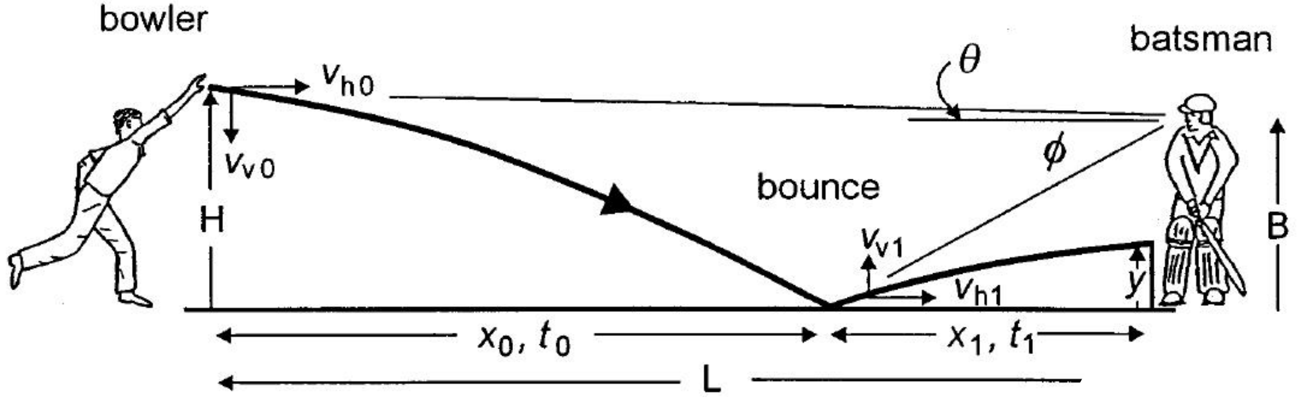
1. Major types of eye movements
2. What triggers saccades?
3. Reinforcement learning
    1. Are eye movements learned?
    2. Markov decision processes
    3. Q learning with Monte Carlo and SARSA
4. Eye movements to learn to solve visually guided tasks

Learning eye movements for playing cricket

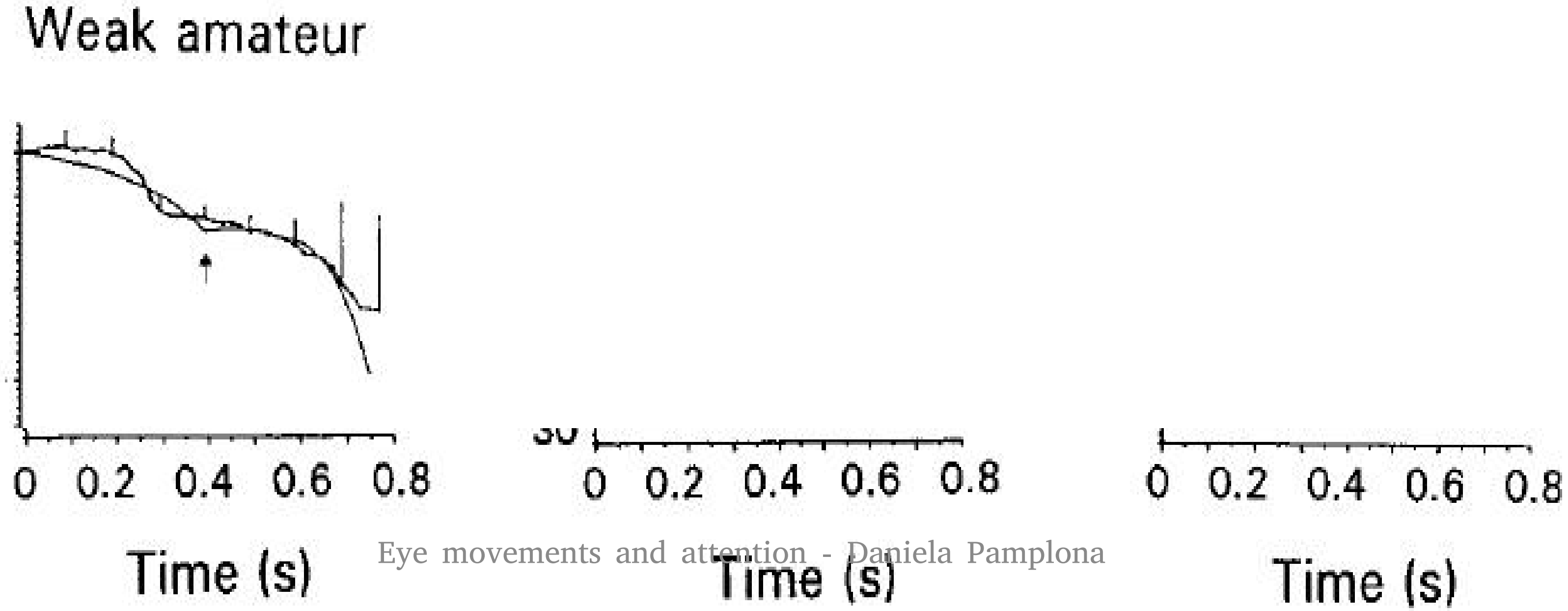


Land and McLeod 2000

Learning eye movements for playing cricket

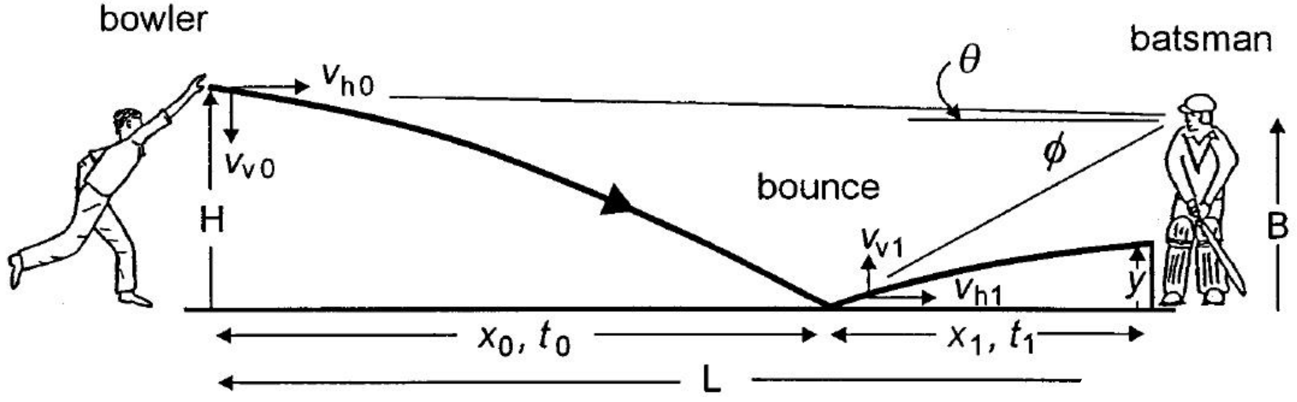


Gaze and ball directions (angles from head)

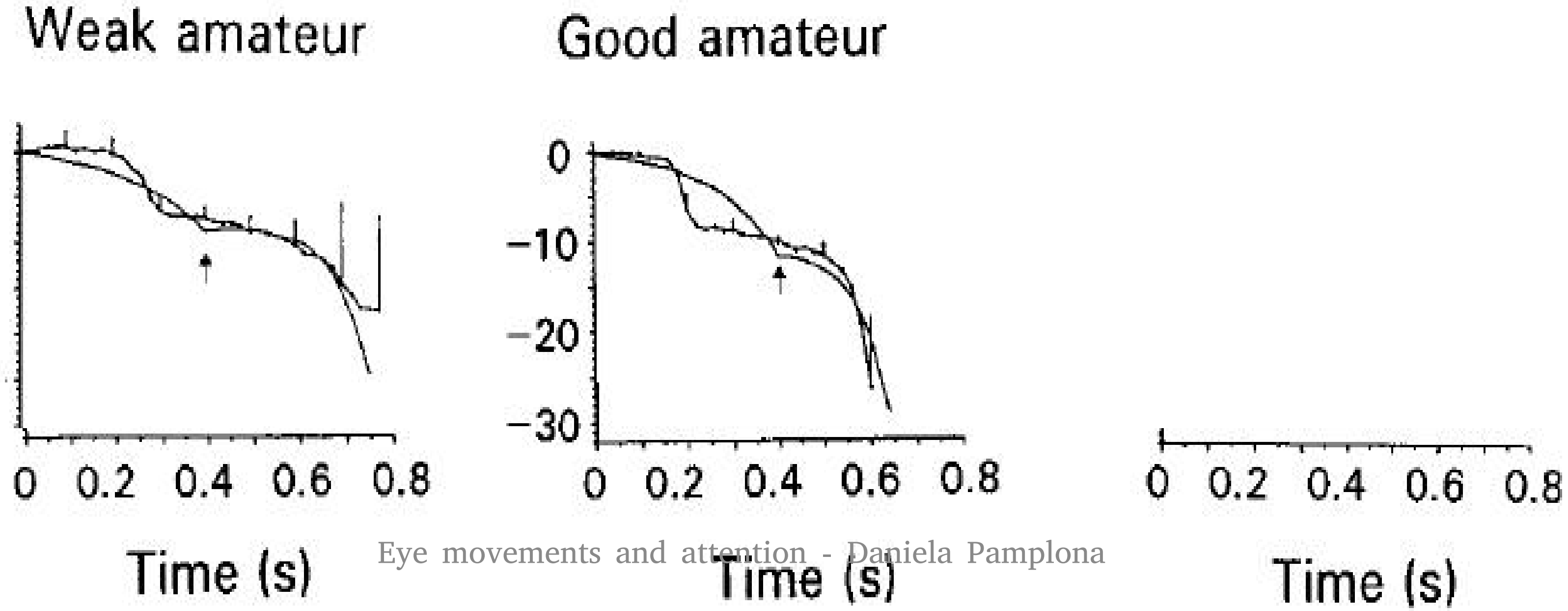

Weak amateur

Land and McLeod 2000

Learning eye movements for playing cricket

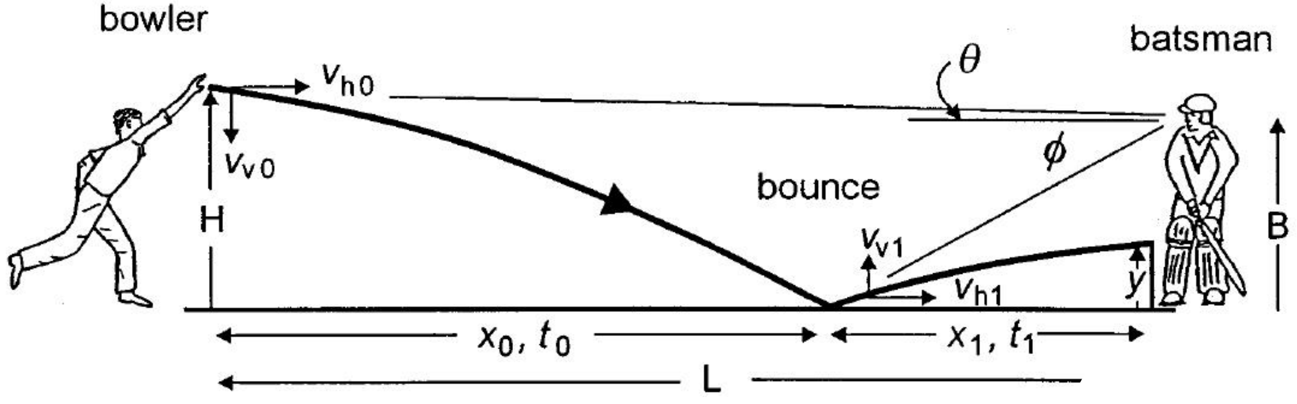


Gaze and ball directions (angles from head)

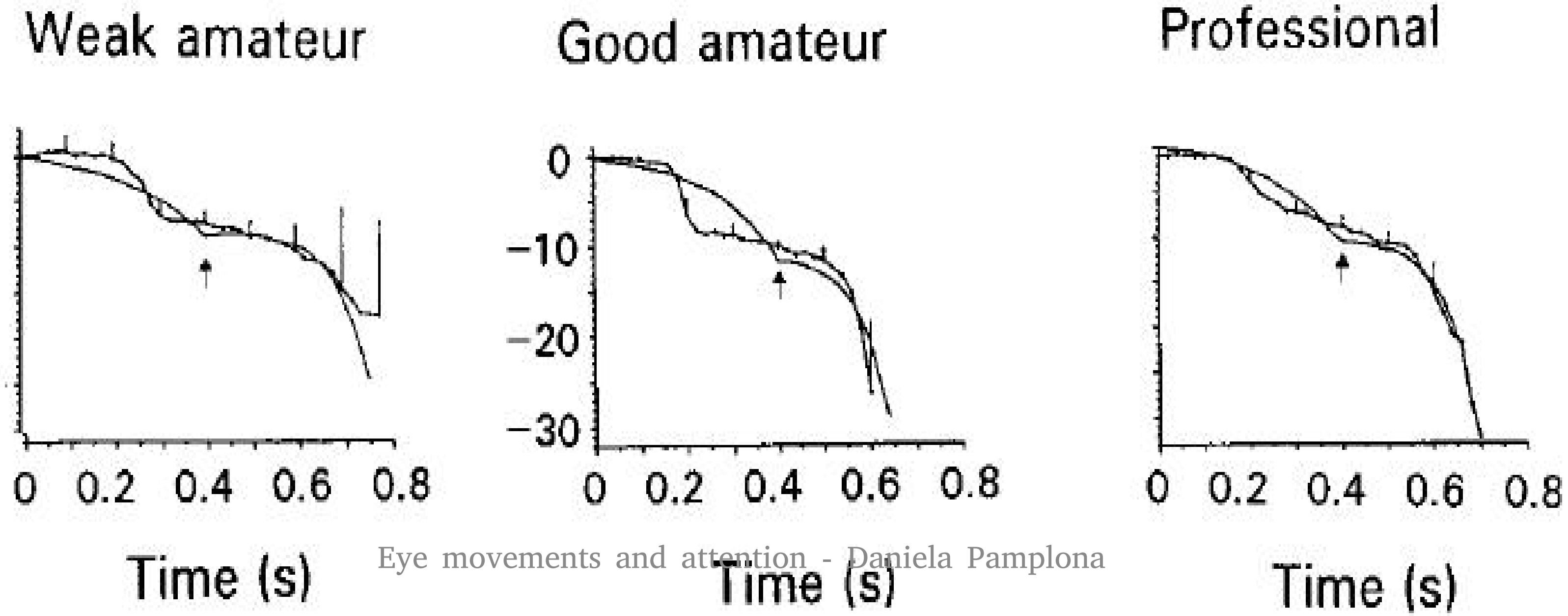


Land and McLeod 2000

Learning eye movements for playing cricket

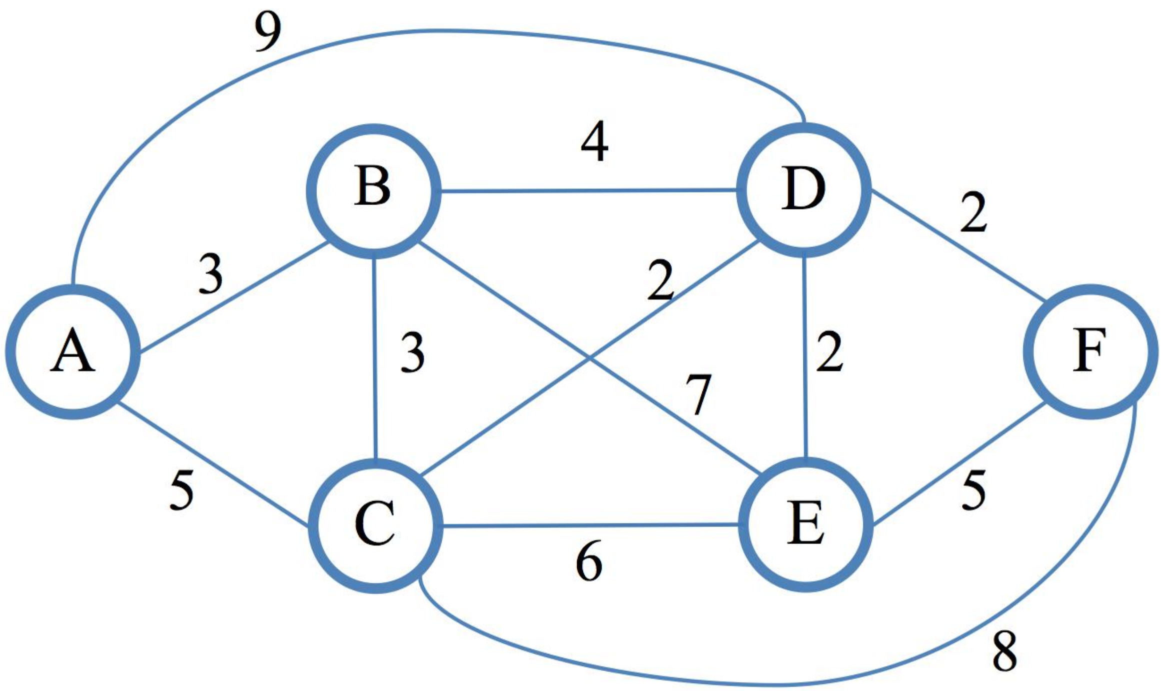


Gaze and ball directions (angles from head)



Land and McLeod 2000

From the shortest path problem to reinforcement learning

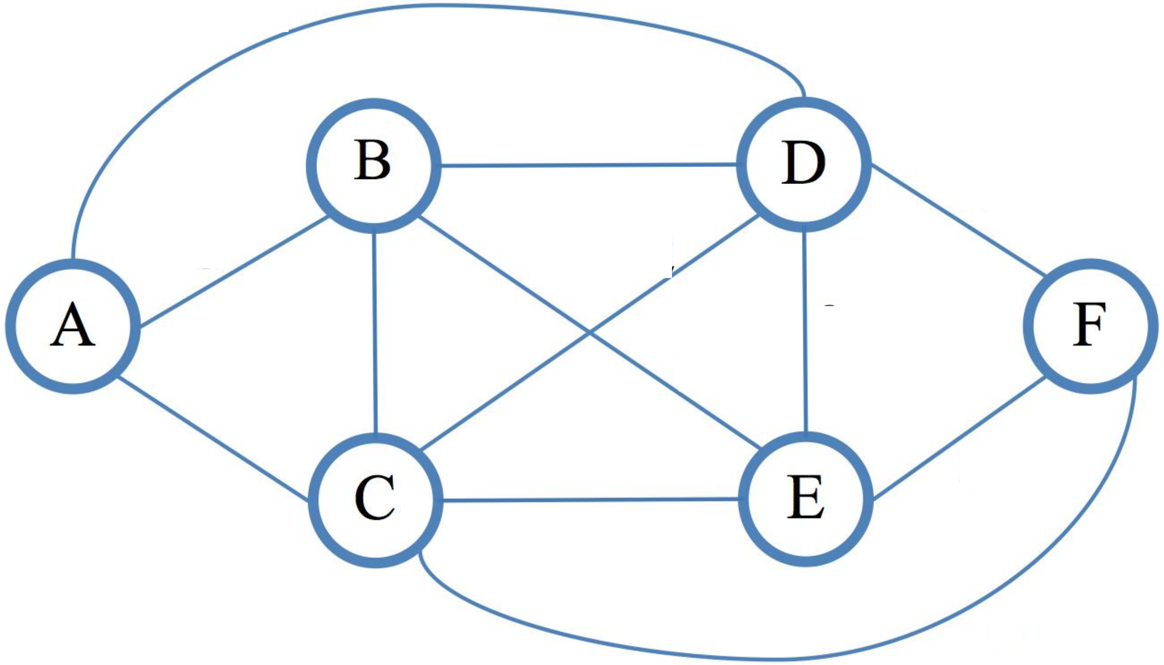


Goal to find the shortest path from A to F without repetitions

- S ≡ set of states: A, B, C…
- A ≡ set of actions: turn 90º, 45º, -30º
- V ≡ distance between states
- R ≡ reward the sum of all neg distances.

  The optimal action sequence can be calculated by Dijksta's algorithm

From the shortest path problem to reinforcement learning

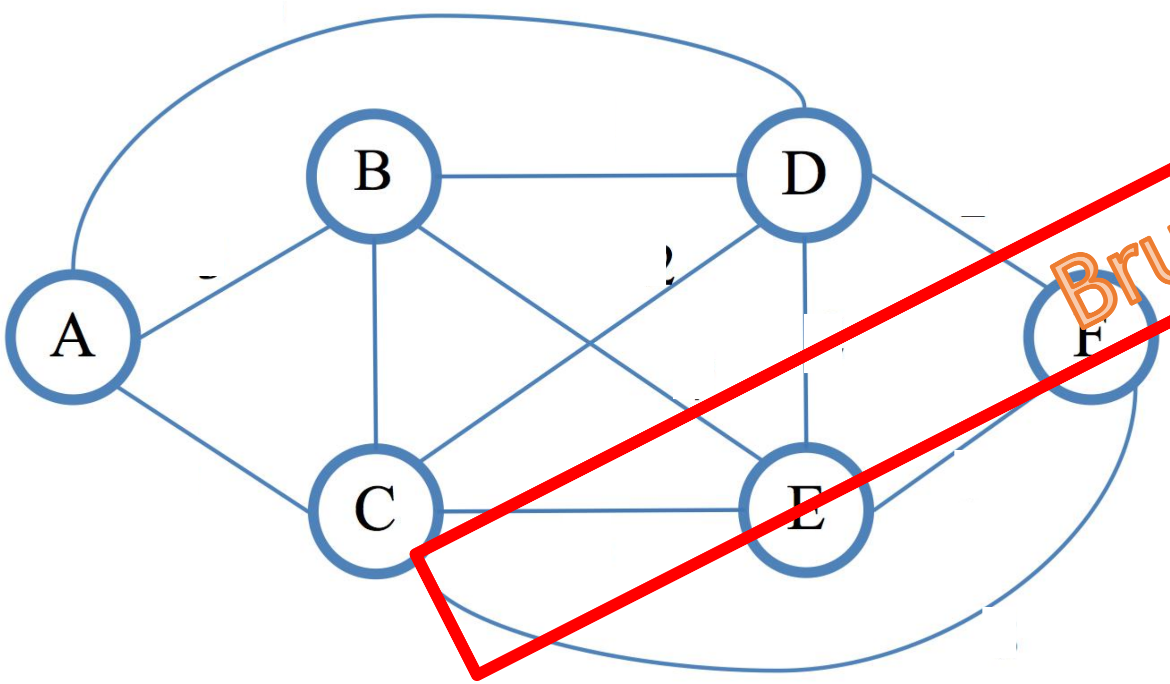


Goal to find the shortest path from A to F without repetitions

- S ≡ set of states: A, B, C…
- A ≡ set of actions: turn 90º, 45º, -30º
- V ≡ ????
- R ≡ reward the sum of all neg distances given

How do we solve this problem?

From the shortest path problem to reinforcement learning



Goal to find the shortest path from A to F without repetitions

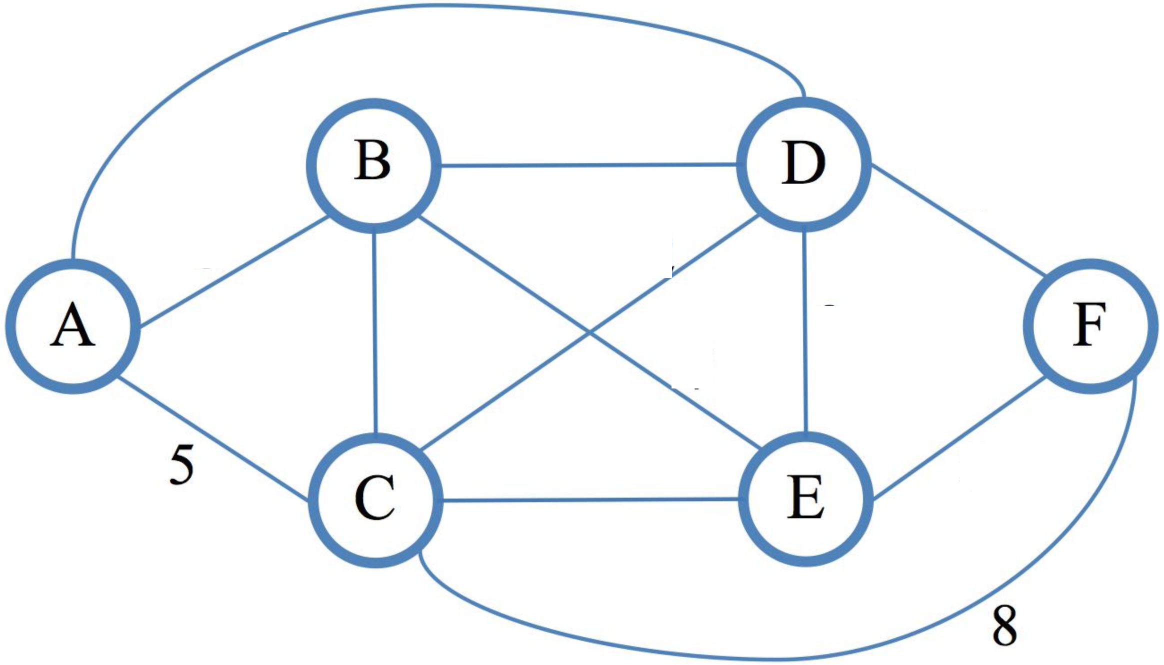
- S ≡ set of states: A, B, C…
- A ≡ set of actions: turn 90º, 45º, -30º
- V ≡ ????
- R ≡ reward the sum of all neg distances

How do we solve this problem?

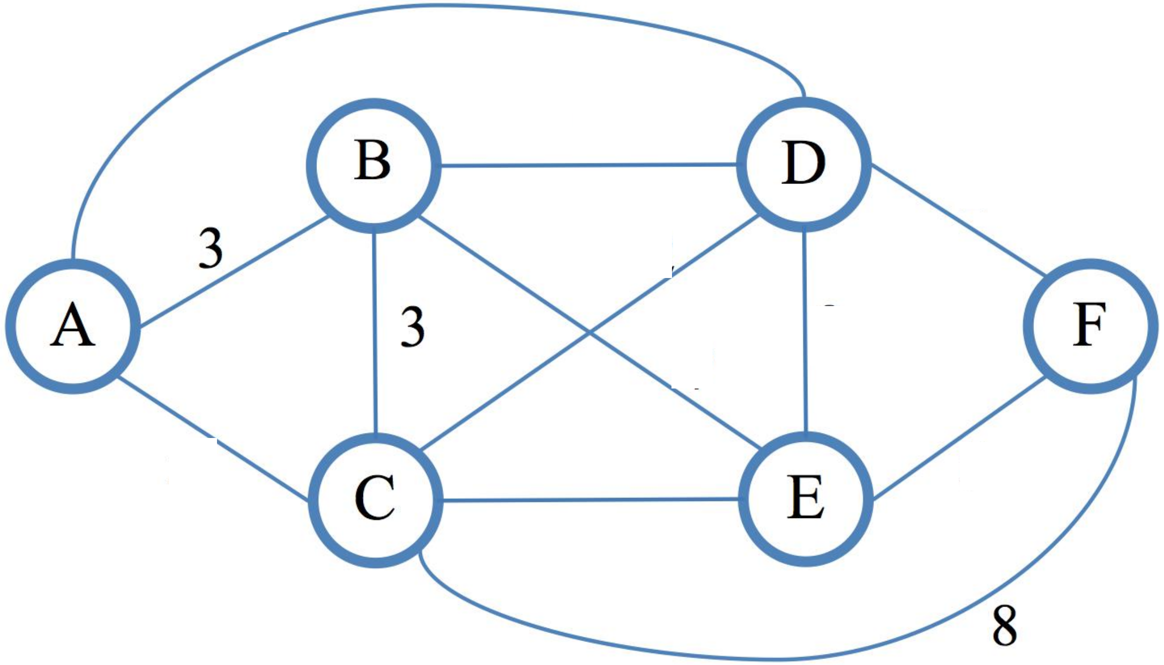
Brute force

From the shortest path problem to reinforcement learning: Brute force

Episode 1: A->C->F: R(Ep.1) = -13



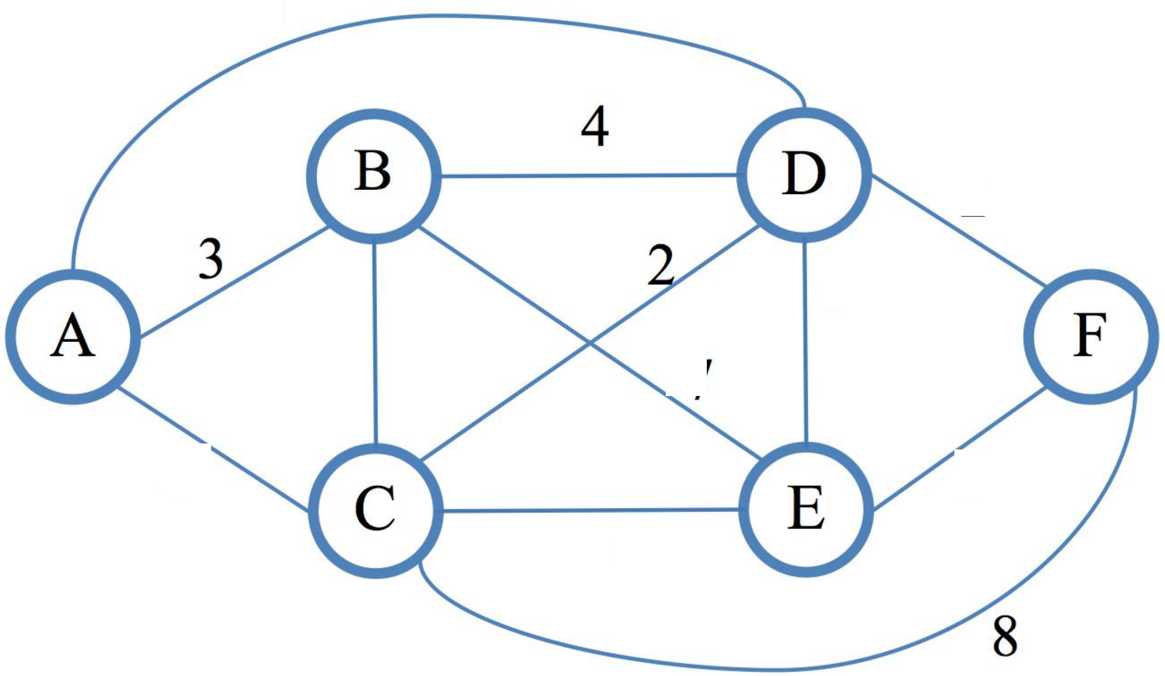
From the shortest path problem to reinforcement learning: Brute force



Episode 1: A->C->F: R(Ep.1) = -13

Episode 2: A->B->C->F: R(Ep.2) = -14

From the shortest path problem to reinforcement learning: Brute force



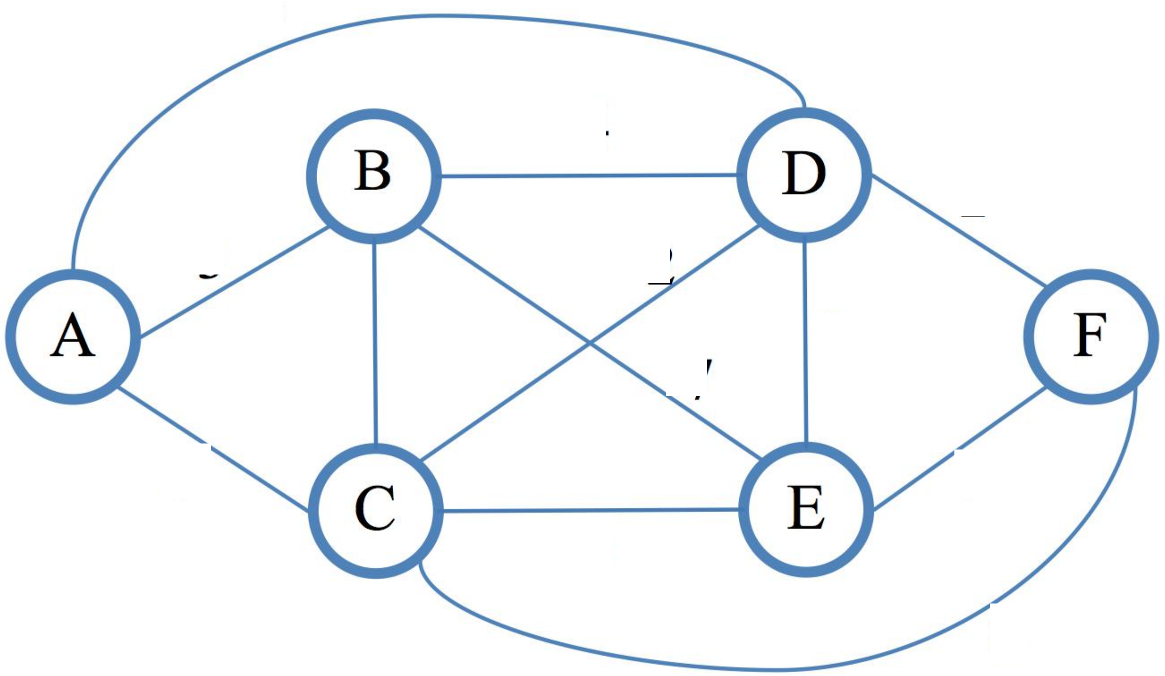
Episode 1: A->C->F: R(Ep.1) = -13

Episode 2: A->B->C->F: R(Ep.2) = -14
V(A,B)+V(B,C)=V(A,C) -1

Episode 3: A->B->D->C->F: R(Ep.3) = -16

...

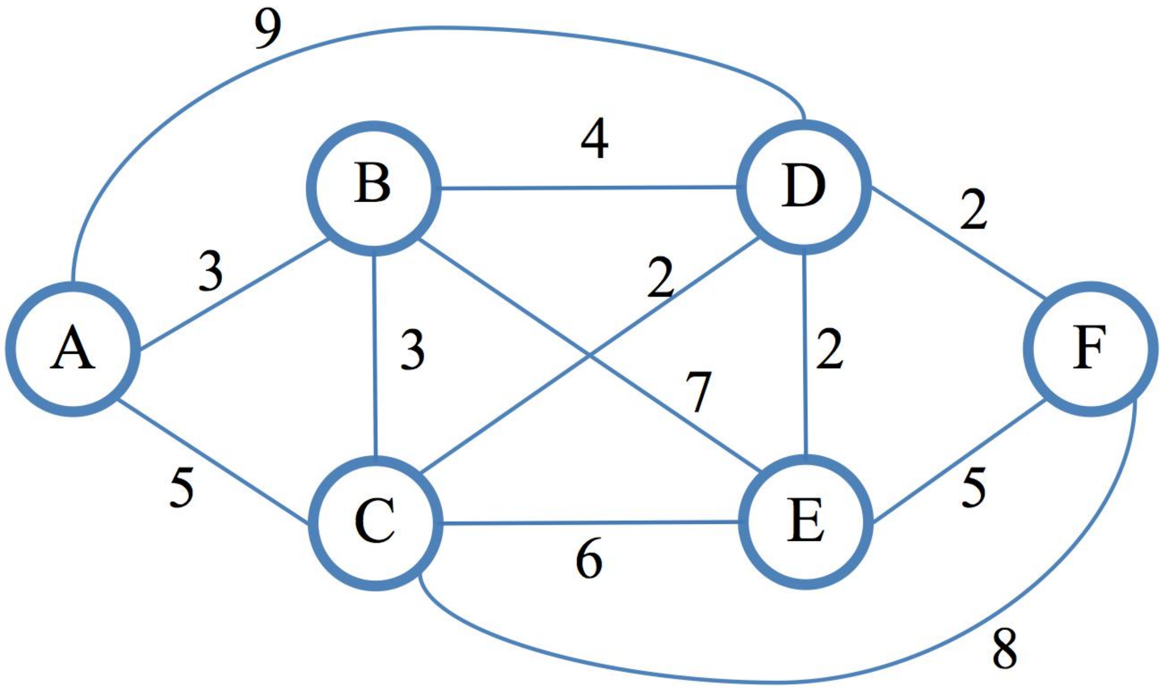
From the shortest path problem to reinforcement learning: Brute force



Problems with brute force solution:

- The number of trajectories grows exponentially
- The number of states might be infinite
- The reward might be stochastic

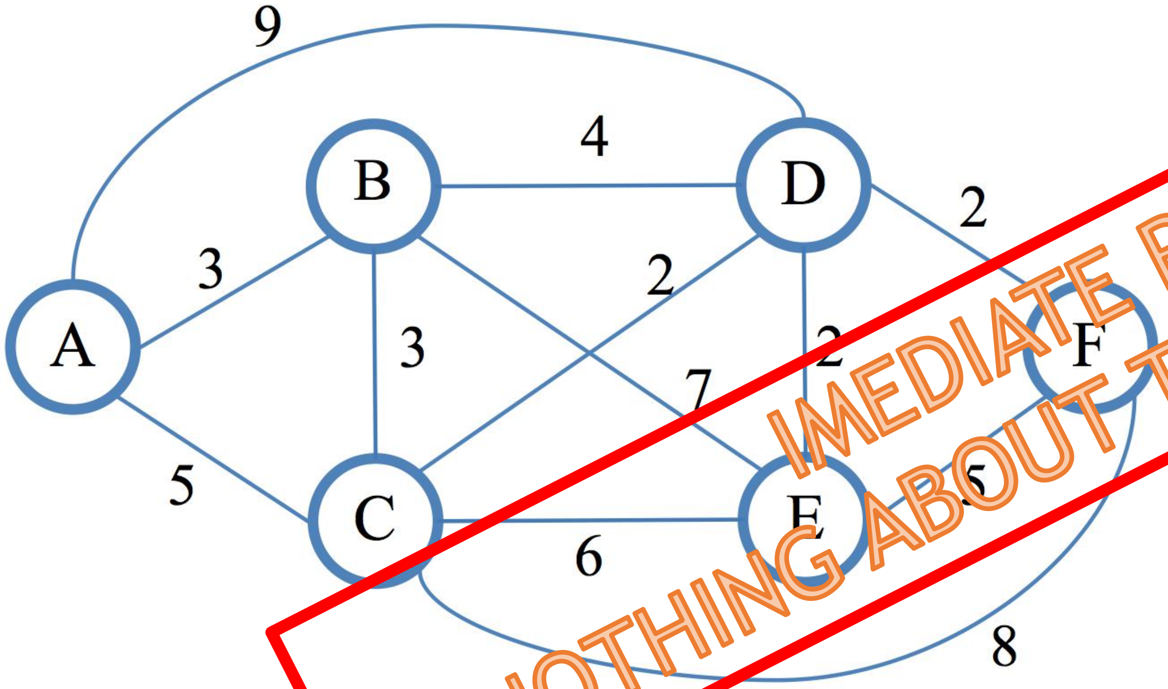
From the shortest path problem to reinforcement learning



One step reward matrix

-	A	B	C	D	E	F
A	-	-3	-5	-9	-	-
B	-3	-3	-4	-	-7	-
C	-5	-4	-	-2	-6	-8
D	-9	-	-2	-	-2	-2
E	-	-7	-6	-2	-	-5
F	-	-	-8	-2	-5	-

From the shortest path problem to reinforcement learning



IMEDIATE REWARD,
NOTHING ABOUT THE TOTAL REWARD

One step reward matrix

-	A	B	C	D	E	F
A	-	-3	-5	-9	-	-
B	-3	-3	-4	-	-7	-
C	-5	-4	-	-2	-6	-8
D	-9	-	-2	-	-2	-2
E	-	-7	-6	-2	-	-5
F	-	-	-8	-2	-5	-

RL with MDP: main functions

- **Expected discounted reward:** ballance between earlier and later rewards.

$$G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \qquad 0 \leq \gamma \leq 1,$$

RL with MDP: main functions

- **Expected discounted cumulative reward:**

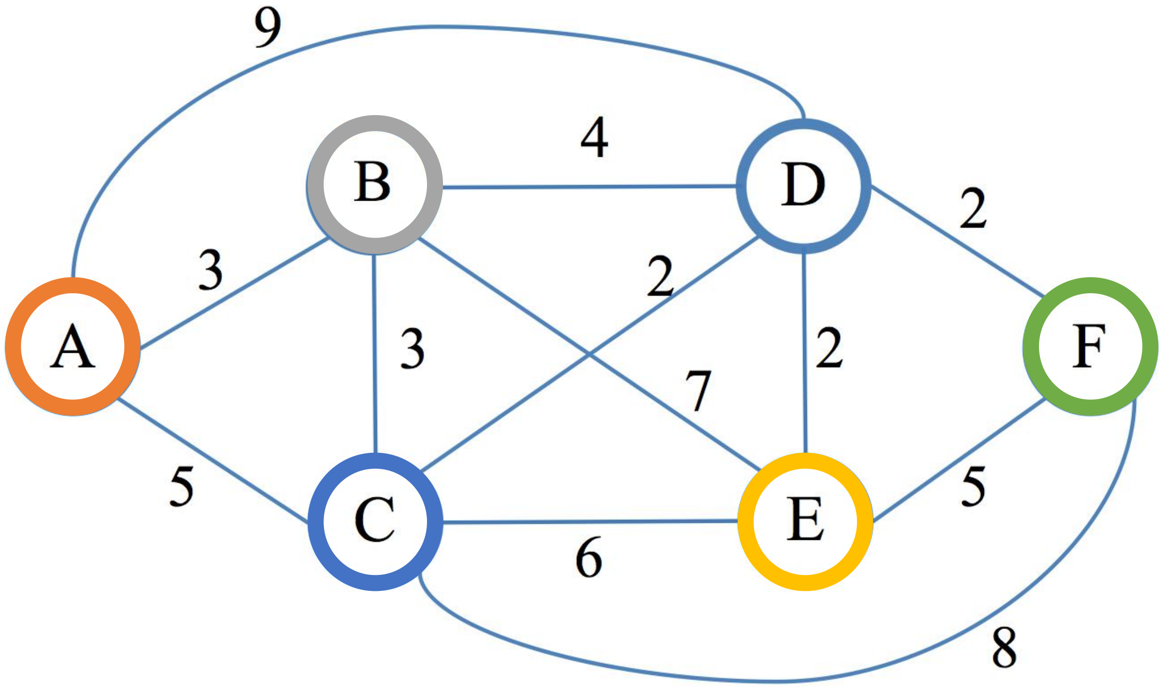
ballance between earlier and later rewards.

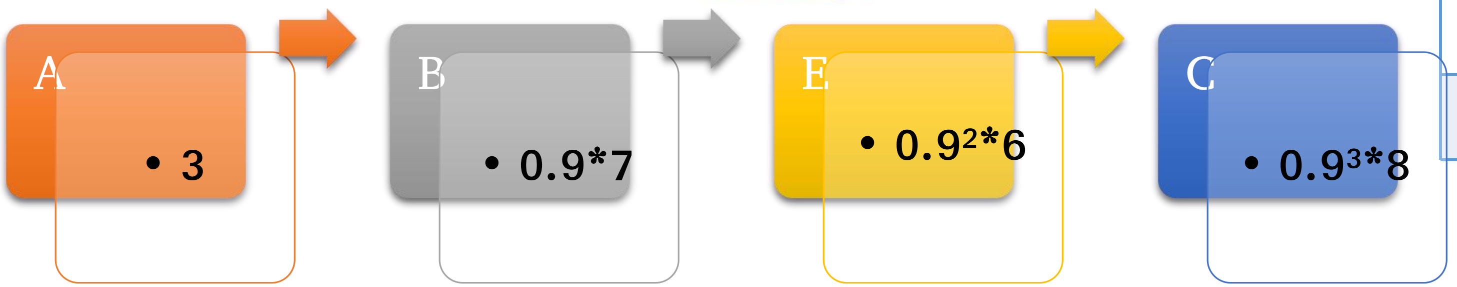
$$G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \qquad 0 \leq \gamma \leq 1,$$

- **value of a state-action under a policy Π:** expected discounted cumulative reward starting from that state and taking that action

$$q_\pi(s,a) \doteq \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s, A_t = a\right]$$

From the shortest path problem to reinforcement learning





A

- 3

B

- 0.9*7

E

- 0.9^2*6

C

- 0.9^3*8

- Q-matrix

-	A	B	C	D	E	F
A	-	20			-	-
B				-		-
C			-			
D		-		-		
E	-				-	
F	-	-				-

RL with MDP: main functions

- **Expected discounted return:** ballance between earlier and later rewards.

$$G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \qquad 0 \leq \gamma \leq 1,$$

- **value of a state-action under a policy Π:** expected discounted return starting from that state and taking that action

$$q_\pi(s,a) \doteq \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s, A_t = a\right]$$

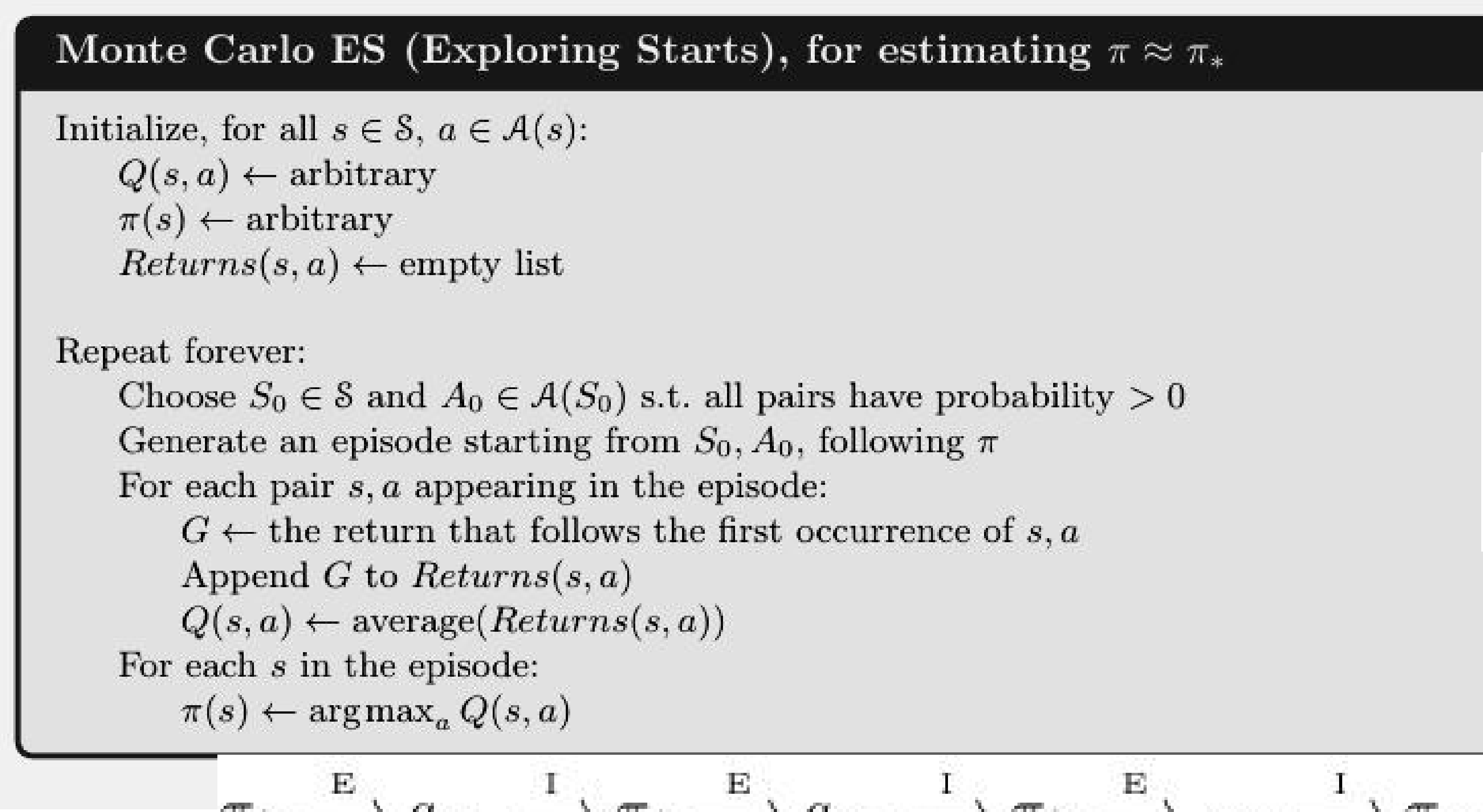
- **Greedy policy:**

$$\Pi^*(s) = \mathrm{argmax}_{\,a}\ Q(s,a)$$

RL with MDP

- https://www.youtube.com/watch?v=bHeeaXgqVig

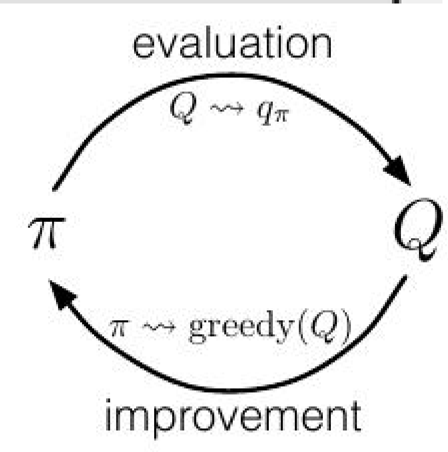
on-policy: Monte Carlo method (extension to brute force)



Monte Carlo ES (Exploring Starts), for estimating $\pi \approx \pi_*$

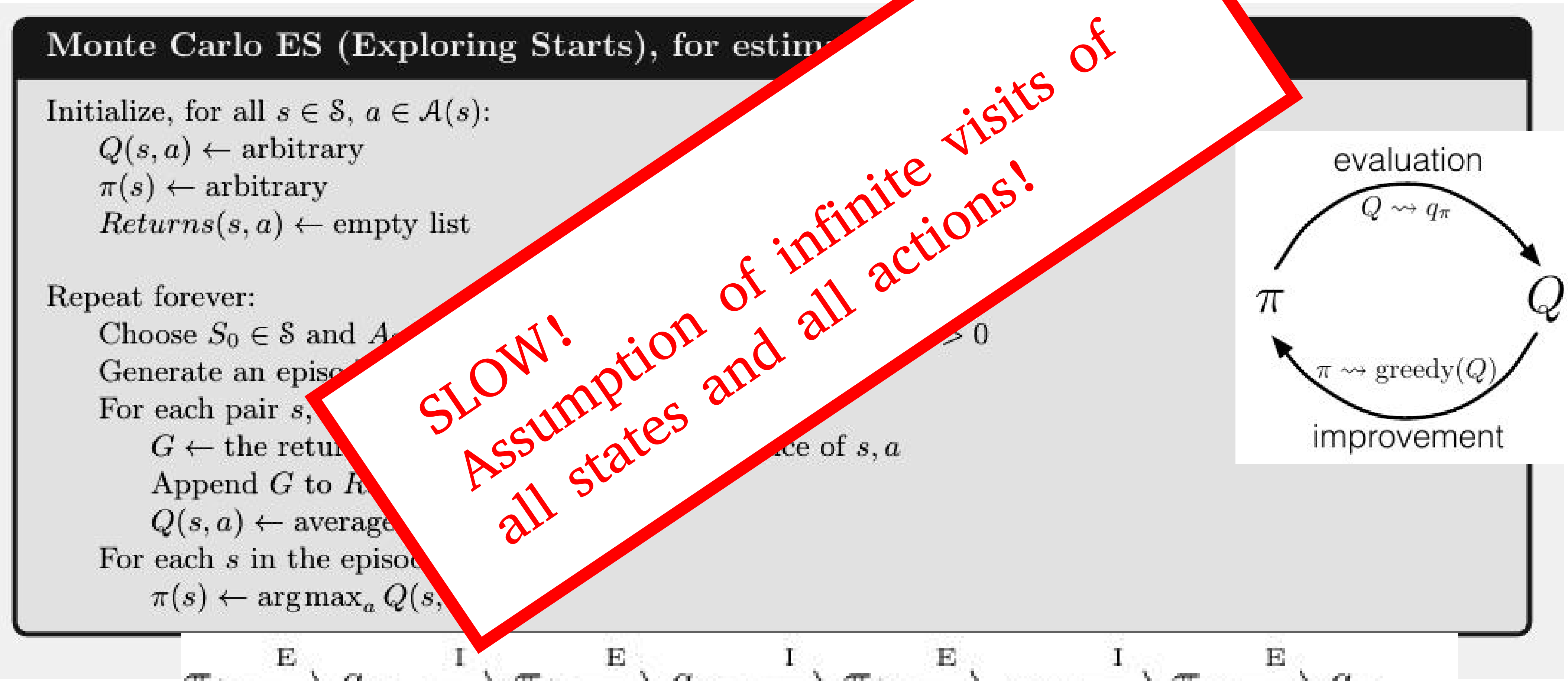
```
Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
    $Q(s,a) \leftarrow$ arbitrary
    $\pi(s) \leftarrow$ arbitrary
    $Returns(s,a) \leftarrow$ empty list

Repeat forever:
    Choose $S_0 \in \mathcal{S}$ and $A_0 \in \mathcal{A}(S_0)$ s.t. all pairs have probability $> 0$
    Generate an episode starting from $S_0, A_0$, following $\pi$
    For each pair $s, a$ appearing in the episode:
        $G \leftarrow$ the return that follows the first occurrence of $s, a$
        Append $G$ to $Returns(s,a)$
        $Q(s,a) \leftarrow$ average$(Returns(s,a))$
    For each $s$ in the episode:
        $\pi(s) \leftarrow \arg\max_a Q(s,a)$
```


evaluation
$Q \rightsquigarrow q_\pi$
π
Q
$\pi \rightsquigarrow \text{greedy}(Q)$
improvement

$$\pi_0 \xrightarrow{\text{E}} q_{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} q_{\pi_1} \xrightarrow{\text{I}} \pi_2 \xrightarrow{\text{E}} \cdots \xrightarrow{\text{I}} \pi_* \xrightarrow{\text{E}} q_*,$$

on-policy: Monte Carlo method (extension to brute force)



Monte Carlo ES (Exploring Starts), for estim

```
Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
    $Q(s,a) \leftarrow$ arbitrary
    $\pi(s) \leftarrow$ arbitrary
    $Returns(s,a) \leftarrow$ empty list

Repeat forever:
    Choose $S_0 \in \mathcal{S}$ and $A$                    $> 0$
    Generate an episo
    For each pair $s,$
        $G \leftarrow$ the retu                  ce of $s, a$
        Append $G$ to $R$
        $Q(s,a) \leftarrow$ average
    For each $s$ in the episo
        $\pi(s) \leftarrow \arg\max_a Q(s,$
```

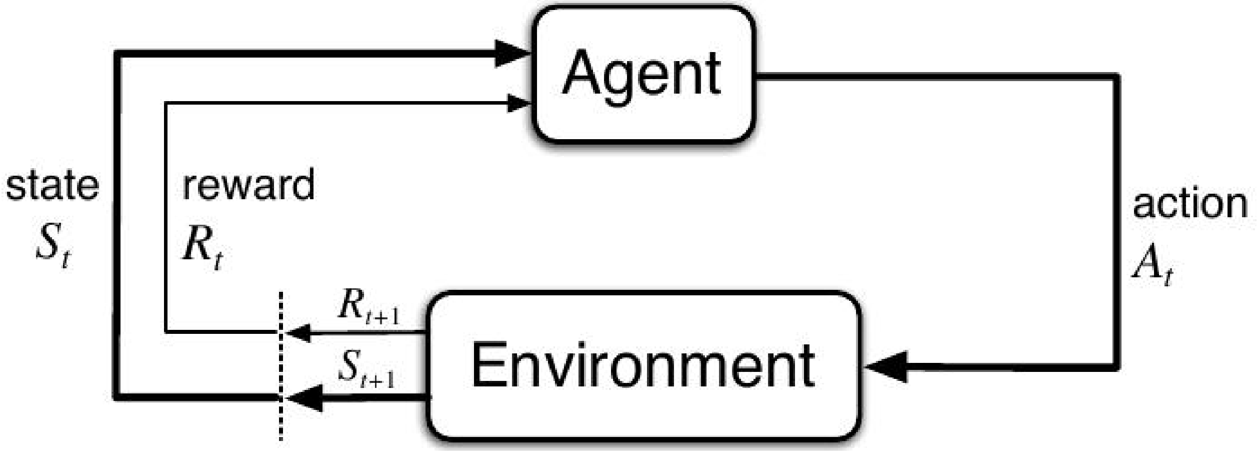
SLOW!

Assumption of infinite visits of all states and all actions!

evaluation
$Q \rightsquigarrow q_\pi$
π
Q
$\pi \rightsquigarrow \text{greedy}(Q)$
improvement

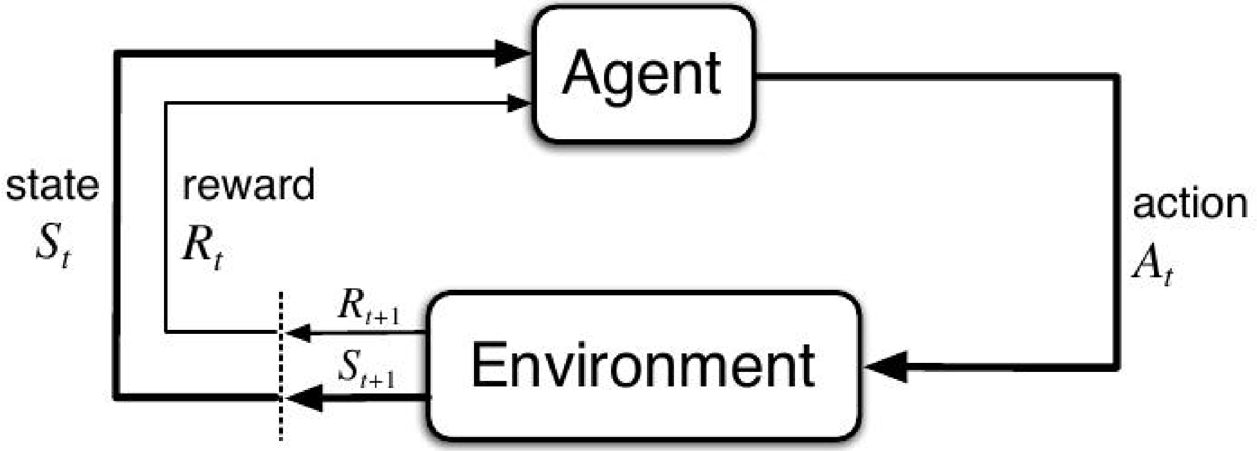
$$\pi_0 \xrightarrow{E} q_{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} q_{\pi_1} \xrightarrow{I} \pi_2 \xrightarrow{E} \cdots \xrightarrow{I} \pi_* \xrightarrow{E} q_*,$$

Reinforcement learning



Sutton and Barto, 2017

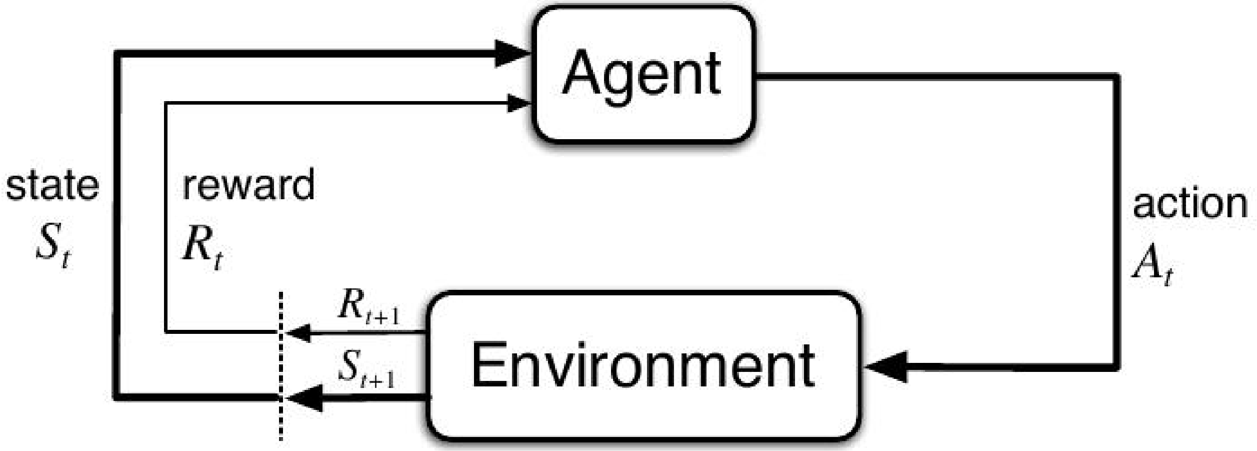
Reinforcement learning with Markov Decision Processes (MDP)



R_{t+1} and S_{t+1} only depend on R_t, S_t and A_t

Sutton and Barto, 2017

Reinforcement learning with Markov Decision Processes (MDP) (S, A, R, T, Π)

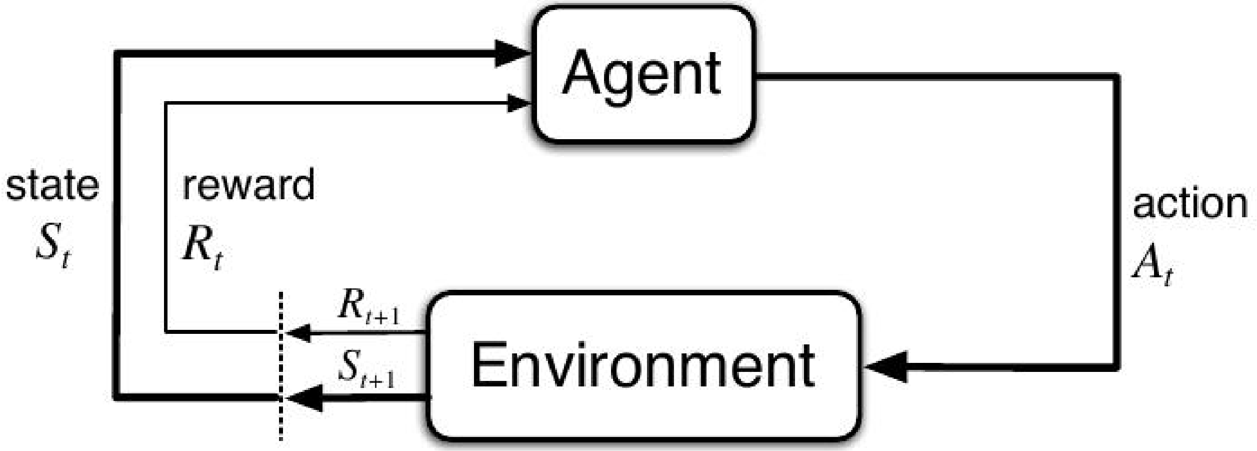


R_{t+1} and S_{t+1} only depend on R_t, S_t and A_t

- S ≡ set of **states:** how the
- environment and the agent are
- A ≡ set of **actions:** what the agent is allowed do
  $A: S \mapsto S$
- R ≡ **reward:** feedback on agent's action (immediate)
  $R: S \mapsto \mathcal{R}$
- Π ≡ **Policy:** agent's stategy: it defines the action to take at each state
  $\Pi: S \mapsto A$

Sutton and Barto, 2017

Reinforcement learning

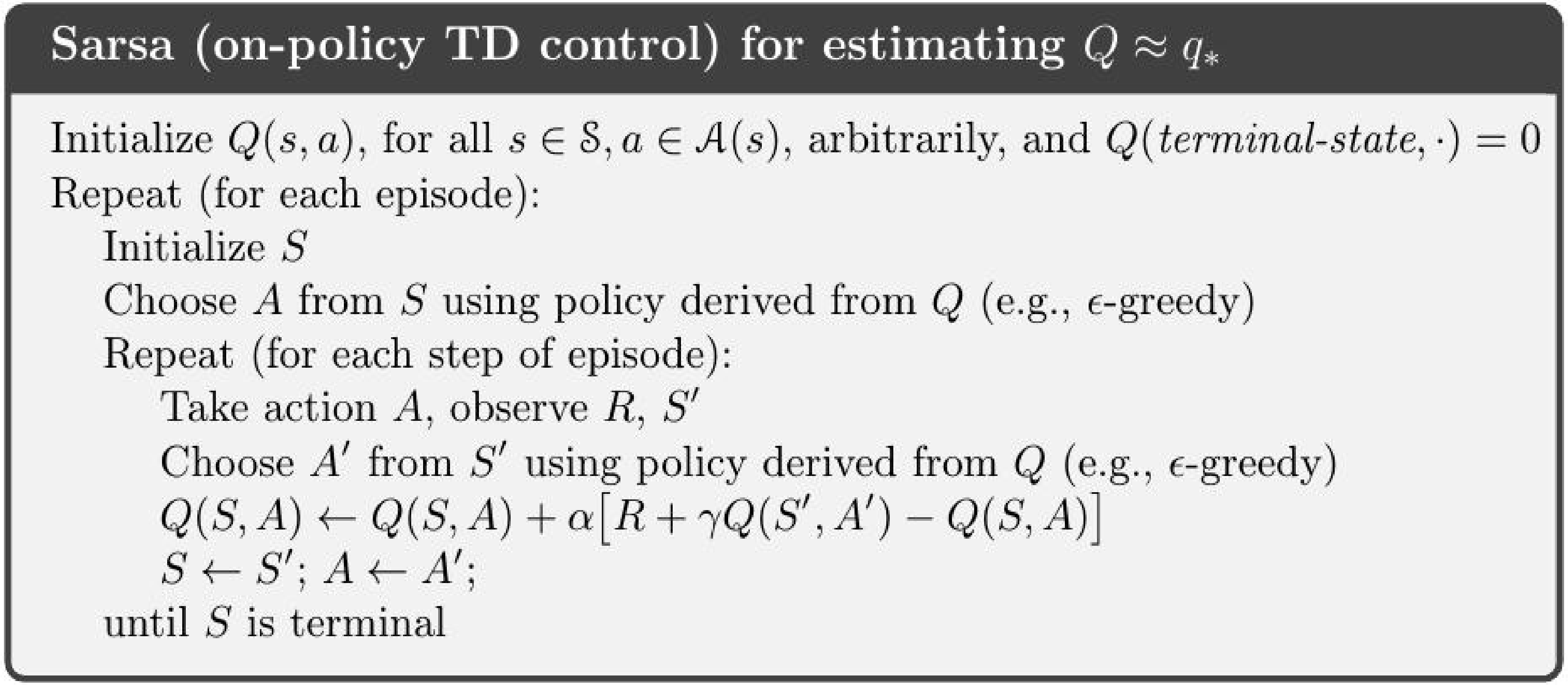


- S ≡ set of **states:** how the
- environment and the agent are
- A ≡ set of **actions:** what the agent is allowed do
- R ≡ **reward:** feedback on agent's action (immediate)
- Π ≡ **Policy:** agent's stategy: it defines the action to take at each state

Sutton and Barto, 2017

RL with MDP: on-policy temporal diference

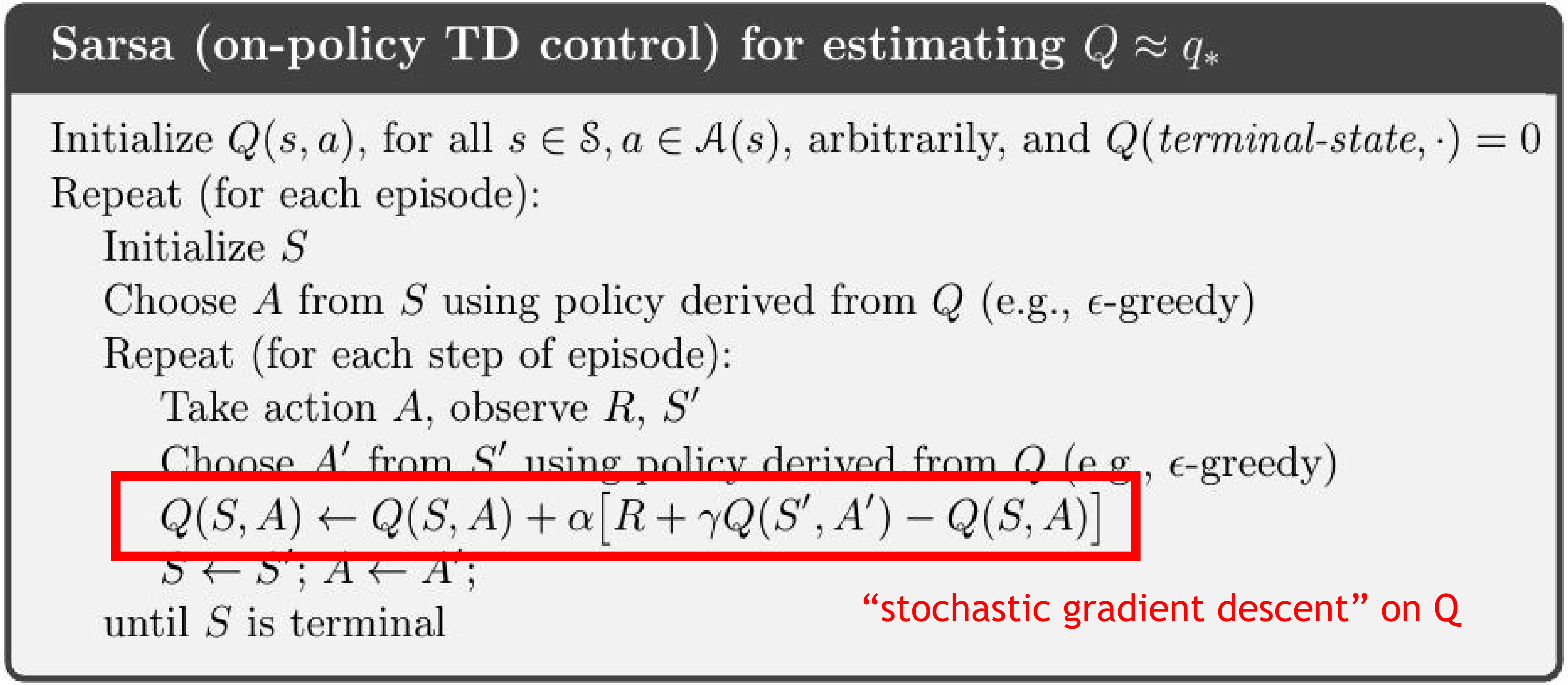
RL with MDP: on-policy temporal diference: SARSA (State$_t$,Action$_t$,Reward$_t$,State$_{t+1}$,Action$_{t+1}$)



Sarsa (on-policy TD control) for estimating $Q \approx q_*$

```
Initialize $Q(s,a)$, for all $s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\textit{terminal-state}, \cdot) = 0$
Repeat (for each episode):
    Initialize $S$
    Choose $A$ from $S$ using policy derived from $Q$ (e.g., $\epsilon$-greedy)
    Repeat (for each step of episode):
        Take action $A$, observe $R$, $S'$
        Choose $A'$ from $S'$ using policy derived from $Q$ (e.g., $\epsilon$-greedy)
        $Q(S,A) \leftarrow Q(S,A) + \alpha\big[R + \gamma Q(S',A') - Q(S,A)\big]$
        $S \leftarrow S'$; $A \leftarrow A'$;
    until $S$ is terminal
```

RL with MDP: on-policy temporal diference method: SARSA (State$_t$,Action$_t$,Reward$_t$,State$_{t+1}$,Action$_{t+1}$)

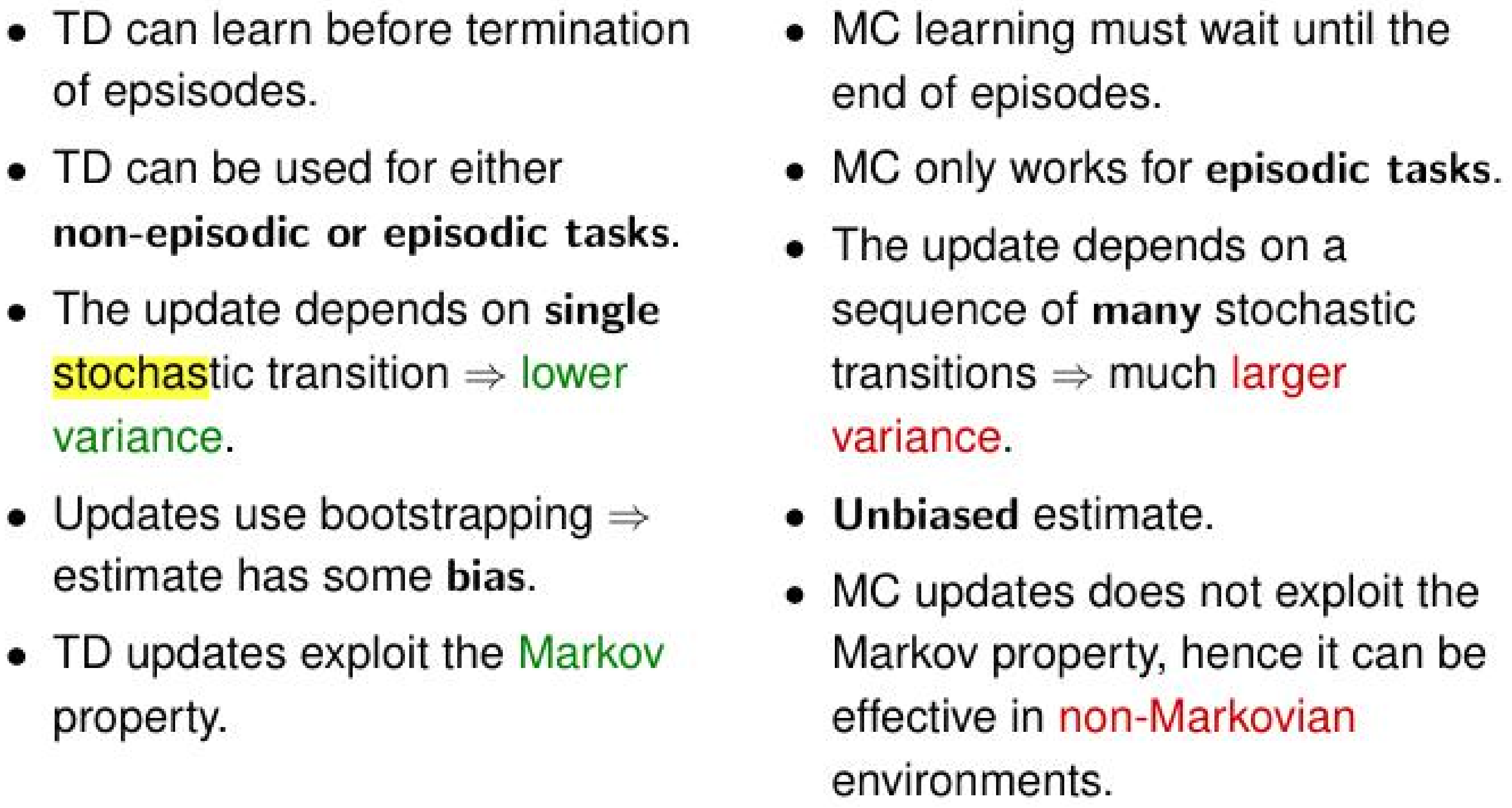


```
Sarsa (on-policy TD control) for estimating $Q \approx q_*$

Initialize $Q(s, a)$, for all $s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\mathit{terminal\text{-}state}, \cdot) = 0$
Repeat (for each episode):
    Initialize $S$
    Choose $A$ from $S$ using policy derived from $Q$ (e.g., $\epsilon$-greedy)
    Repeat (for each step of episode):
        Take action $A$, observe $R$, $S'$
        Choose $A'$ from $S'$ using policy derived from $Q$ (e.g., $\epsilon$-greedy)
        $\boxed{Q(S, A) \leftarrow Q(S, A) + \alpha\big[R + \gamma Q(S', A') - Q(S, A)\big]}$
        $S \leftarrow S'$; $A \leftarrow A'$;
    until $S$ is terminal
```

“stochastic gradient descent” on Q

Temporal Difference vs. Monte Carlo



- TD can learn before termination of epsisodes.
- TD can be used for either **non-episodic or episodic tasks**.
- The update depends on **single** stochastic transition ⇒ lower variance.
- Updates use bootstrapping ⇒ estimate has some **bias**.
- TD updates exploit the Markov property.

- MC learning must wait until the end of episodes.
- MC only works for **episodic tasks**.
- The update depends on a sequence of **many** stochastic transitions ⇒ much larger variance.
- **Unbiased** estimate.
- MC updates does not exploit the Markov property, hence it can be effective in non-Markovian environments.

http://ipvs.informatik.uni-stuttgart.de/mlr/wp-content/uploads/2016/06/03-TemporalDiffHung.pdf

Contents

1. Major types of eye movements
2. What triggers saccades?
3. Reinforcement learning
    1. Markov decision processes
    2. Q learning with Monte Carlo and SARSA
4. Eye movements to learn to solve visually guided tasks

Possible add-ons

- ε-greedy policy

$$\Pi^*(s) = \begin{cases} \text{argmax}_a\ Q(s,a) \text{ with probability } \varepsilon \\ \text{random } a \text{ with probability } 1-\varepsilon \end{cases}$$

- States only partially observable (thus there is uncertainity on the state representation)

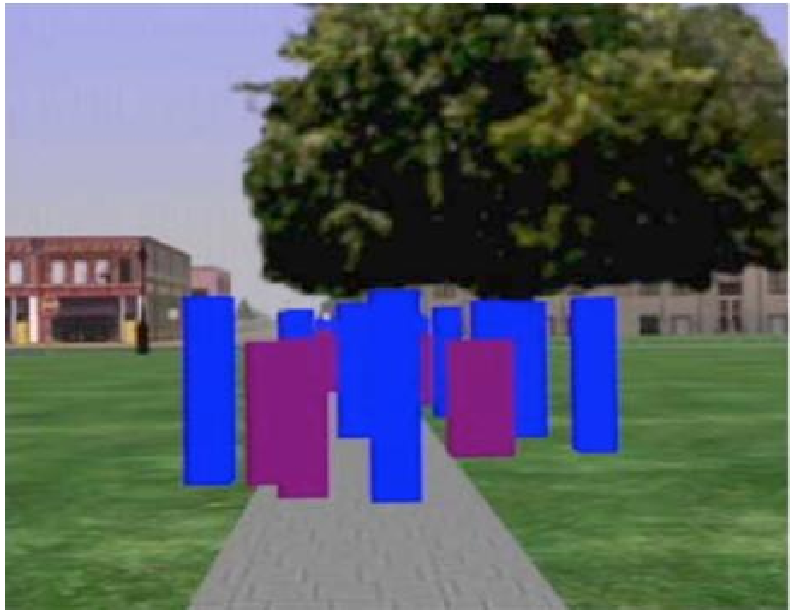
- Highly dimensional states

- Multitask learning

Break

- Questions**?**

Problem formulation (S, A, R, Π)

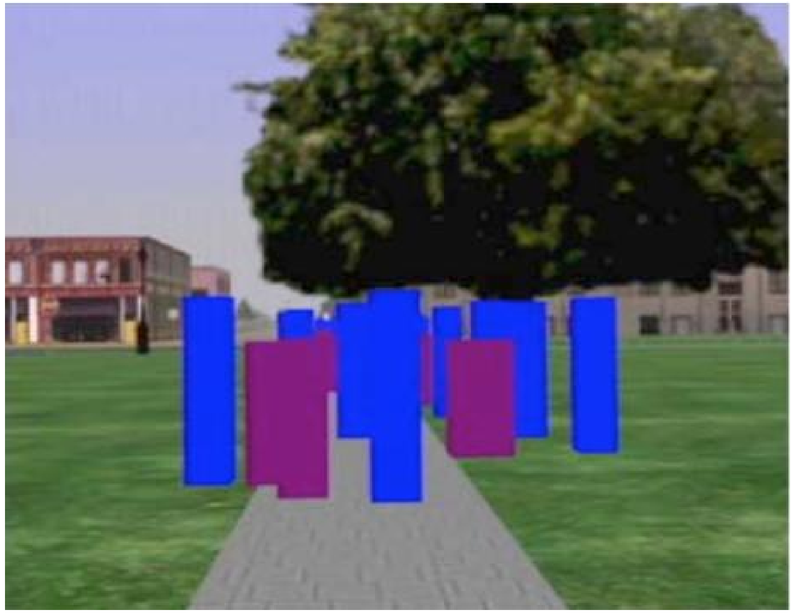


Tasks/behaviors:

1. avoid obstacle
2. sidewalk following
3. pick litter

Sprague and Ballard, 2003
Sutton and Barto, 2012

Problem formulation (S, A, R, Π)



Tasks/behaviors:

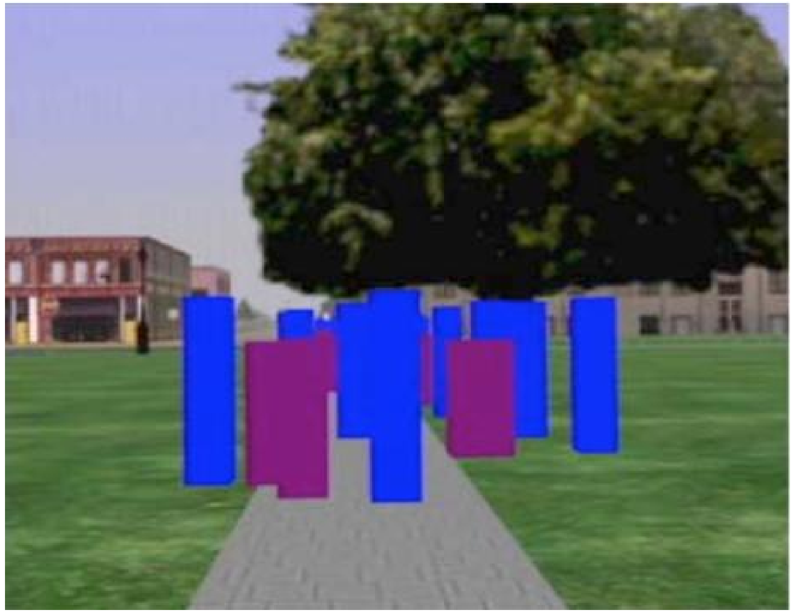
1. avoid obstacle
2. sidewalk following
3. pick litter

- S ≡ set of states (position in the environment angle vs distance). The state is estimated in function of the visual perception (the eyes of the agent)
- A ≡ set of actions (turn -15deg, no turn, turn 15 deg. Walking and action selection at fixed rate)
- R(s) ≡ reward (-4 if hit obstacle, +1 in sidewalk, +2 if pick litter)

- Π(s) ≡ Policy (ε-greedy)

Sprague and Ballard, 2003
Sutton and Barto, 2012

Problem formulation (S, A, R, Π)



Tasks/behaviors:

1. avoid obstacle
2. sidewalk following
3. pick litter

Each task has its 2D state represention:

1. distance and angle, relative to the agent, to the nearest obstacle:
2. angle of the center-line of the sidewalk relative to the agent and signed distance to the center of the sidewalk
3. distance and angle, relative to the agent, to the nearest litter

Sprague and Ballard, 2003
Sutton and Barto, 2012

Modular RL for multi-tasking: behaviors

- **Each behavior (sensory-action control) has the ability to direct the eye,** perform appropriate visual processing to retrieve the information necessary for performance of the behavior's task, and choose an appropriate course of action.
- Each behavior is only allowed to acess perception (thus move the eye) during 300ms. That **behavior updates its state space using a Kalman filter, while the others propagate their estimates and track the uncertainities**

Modular RL for multi-tasking: behaviors

- **Each behavior (sensory-action control) has the ability to direct the eye,** perform appropriate visual processing to retrieve the information necessary for performance of the behavior's task, and choose an appropriate course of action.
- Each behavior is only allowed to acess perception (thus move the eye) during 300ms. That **behavior updates its state space using a Kalman filter, while the others propagate their estimates and track the uncertainities**
- How to mediate between behaviors?

Modular RL for multi-tasking with SARSA

Q learning

Single task

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha(r + \gamma Q(s', a'))$$

Multiple n-tasks

Σ

Sprague and Ballard, 2003
Sutton and Barto, 2012

Modular RL for multi-tasking with SARSA

Single task

Multiple n-tasks

Q learning

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(r + \gamma Q(s',a'))$$

$$Q(s,a) \approx \sum_{i=1}^{n} Q_i(s_i,a)$$

Σ

Sprague and Ballard, 2003
Sutton and Barto, 2012

Modular RL for multi-tasking with SARSA

Single task

Multiple n-tasks

Q learning

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(r + \gamma Q(s',a'))$$

$$Q(s,a) \approx \sum_{i=1}^{n} Q_i(s_i,a)$$

action selection

$$\Pi^*(s) = \begin{cases} \text{argmax}_a\ Q(s,a) \text{ with probability } \varepsilon \\ \text{random a with probability } 1-\varepsilon \end{cases}$$

Σ

Sprague and Ballard, 2003
Sutton and Barto, 2012

Modular RL for multi-tasking with SARSA

Q learning

Single task

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(r + \gamma Q(s',a'))$$

Multiple n-tasks

$$Q(s,a) \approx \sum_{i=1}^{n} Q_i(s_i,a)$$

action selection

$$\Pi^*(s)= \begin{cases} \text{argmax}_{\,a}\ Q(s,a) \text{ with probability } \varepsilon \\ \text{random } a \text{ with probability } 1\text{-}\varepsilon \end{cases}$$

$$\Pi^*(s)= \begin{cases} \text{argmax}_{\,a}\sum Q_i(s,a) \text{ with probability } \varepsilon \\ \text{random } a \text{ with probability } 1\text{-}\varepsilon \end{cases}$$

Sprague and Ballard, 2003
Sutton and Barto, 2012

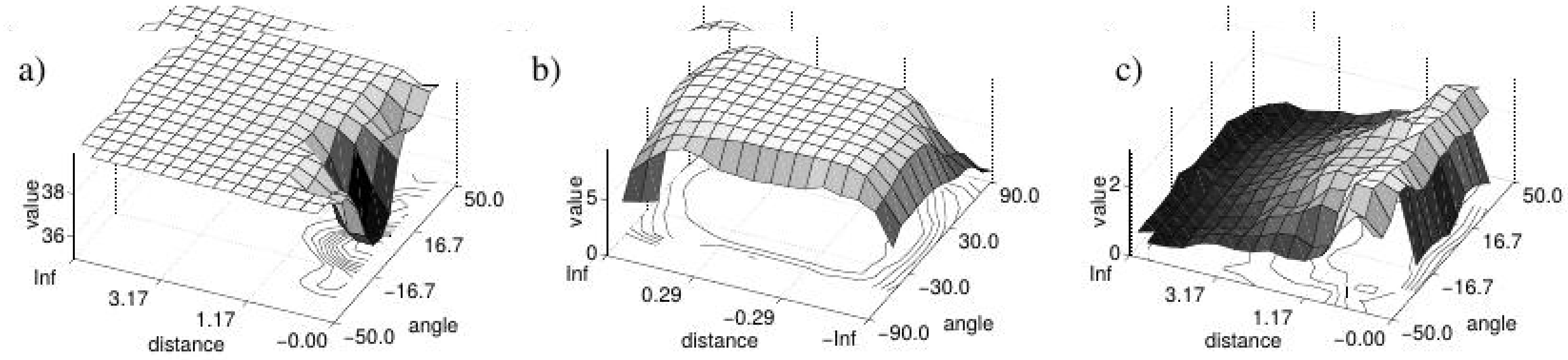
Q-values and Polices

Obstacle Avoidance

Sidewalk following

Litter collection

Q-function



Sprague and Ballard, 2003

Q-values and Polices

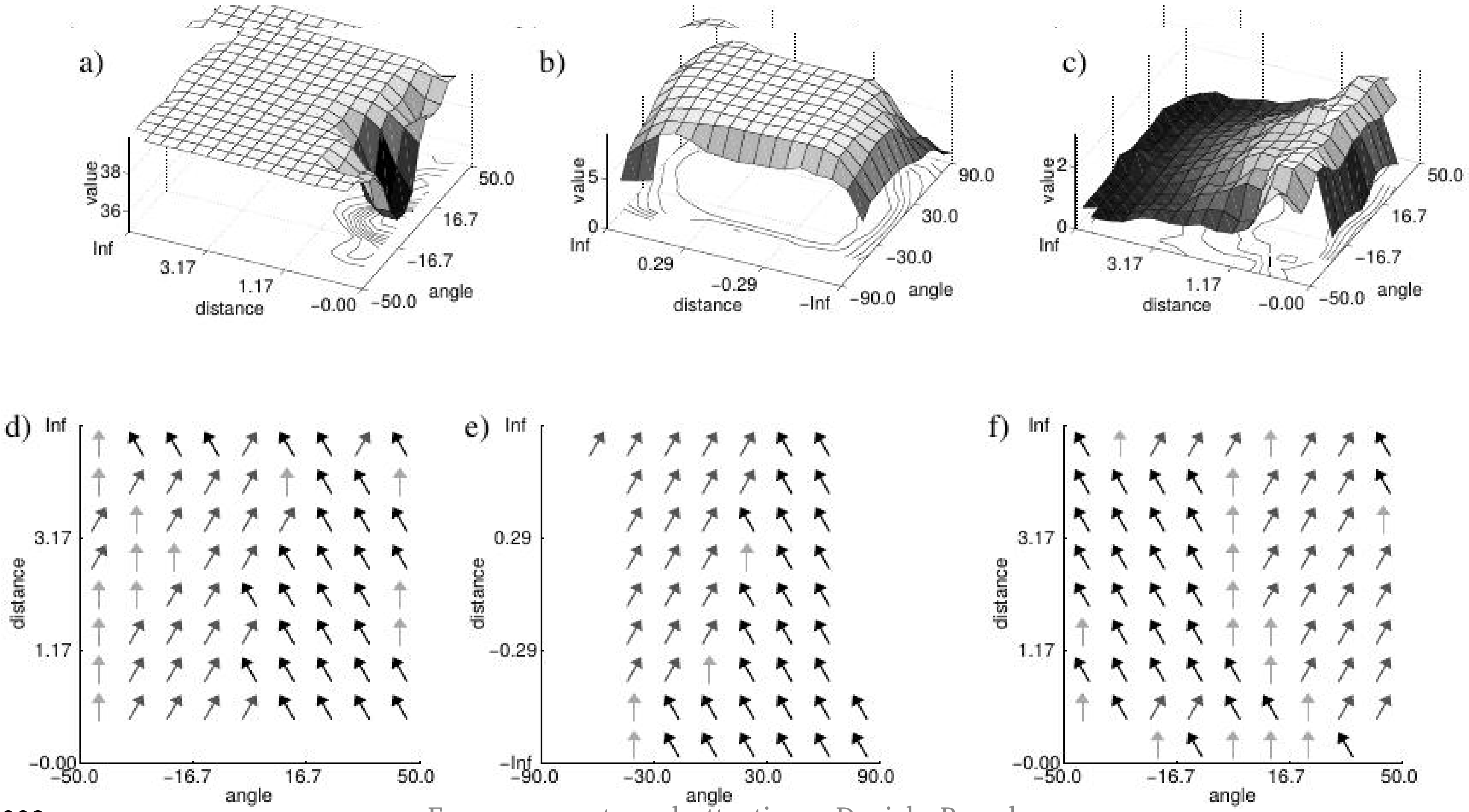
Obstacle Avoidance

Sidewalk following

Litter collection

Q-function

Policy



Sprague and Ballard, 2003

How to select where to look?

Or how to select where to extract features to improve my state representation?

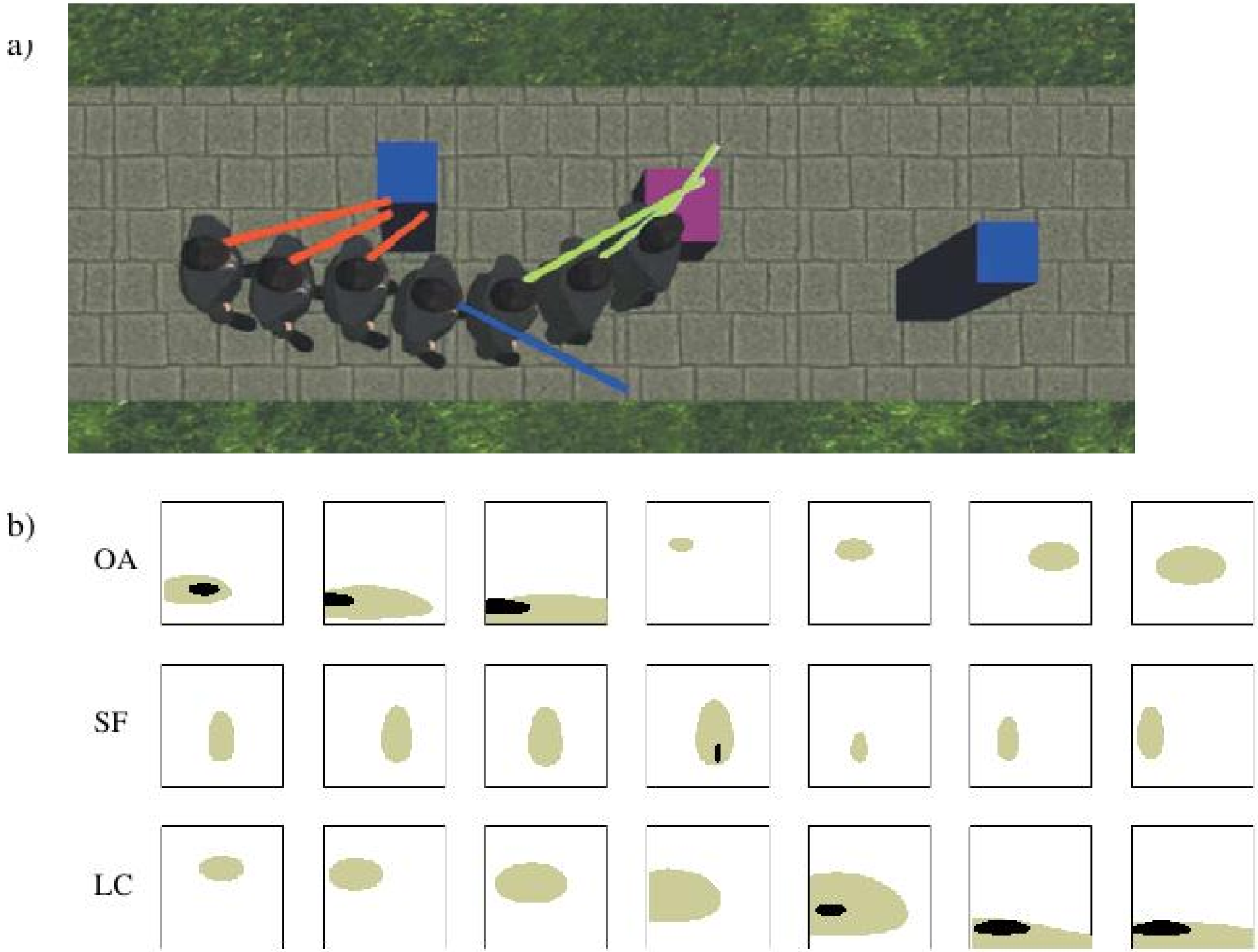
Single behavior

1. For each possible eye movement estimate the **cost of the uncertainty** in the state representation if the movement is not made.
2. Select the **eye movement with highest** potential cost.

Multiple n-behaviors

1. For each possible behavior, for each possible eye movement estimate the **cost of the uncertainty** in the state representation if the movement is not made
2. For each possible behavior, estimate the total cost
3. Select the behavior with larger total cost
4. Select the **eye movement with highest** potential cost.

Eye movements



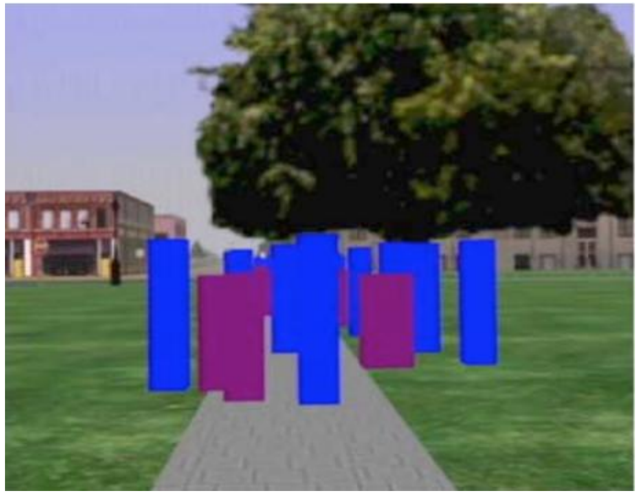
a) An overhead view of the virtual agent during seven time steps of the sidewalk navigation task.
b) State estimates during the same seven time steps. The light gray regions correspond to the 90% confidence bounds before any perception has taken place. When present, the black regions correspond to the 90% confidence bounds after an eye movement has been made.

Sprague and Ballard, 2003

In summary

- 3 tasks
- Shared action space
- Non-shared state representation
- 2 controls: **body** and **eye**
- **body**: actions are selected to maximize reward by RL agent
- **eye:** Each behavior decides where to look/observe its state. Only one behavior is allowed at each time

- Reward is maximized by moving eyes according to the behavior that loses the most if not observed.

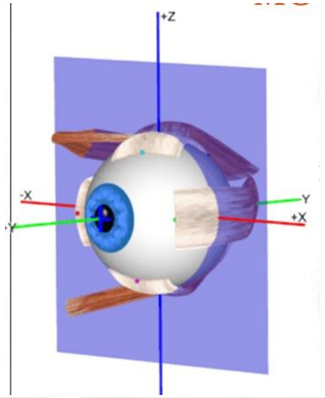


Tasks/behaviors:

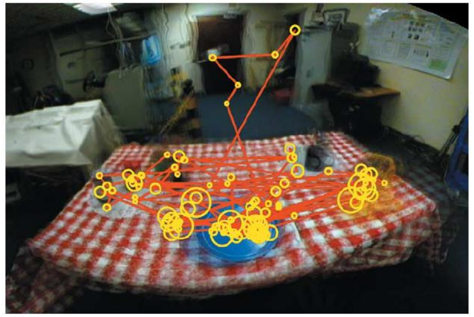
1. avoid obstacle
2. sidewalk following
3. pick litter

Summary

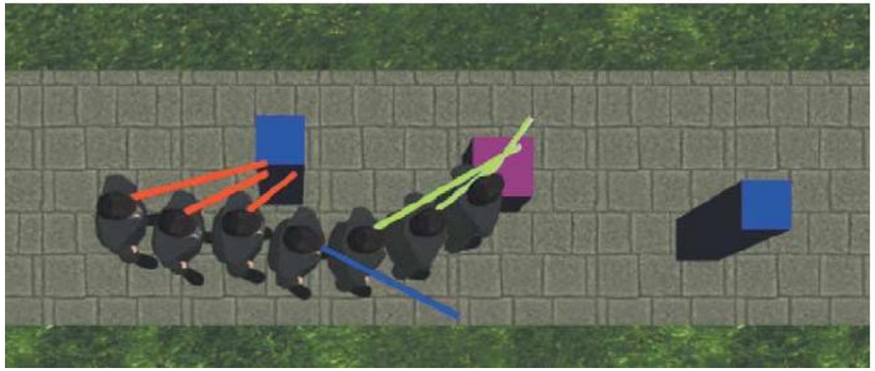
- Major types of eye movements



- What makes eyes move?



- Multi-task RL modeling visually guided behaviors



Bibliography

- Krauzlis, Recasting the Smooth Pursuit Eye Movement System, 2004
- Bruce and Tsotos, Saliency, attention, and visual search: An information theoretic approach, 2009
- Land, Eye movements and the control of actions in everyday life, 2006
- Hayhoe and Rothkop,Vision in the natural world, 2010
- Shinoda et all, What controls attention in natural environments, 2001

Bibliography (cont)

- Hofsten and Rosander, The Development of Gaze Control and Predictive Tracking in Young Infants, 1995
- Li et al, Memory and visual search in naturalistic 2D and 3D environments, 2016
- Sprague and Ballard, Eye Movements for Reward Maximization, 2003
- Sutton and Barto, Reinforcement Learning: An Introduction, 2017, http://incompleteideas.net/book/bookdraft2017nov5.pdf

Extra slides