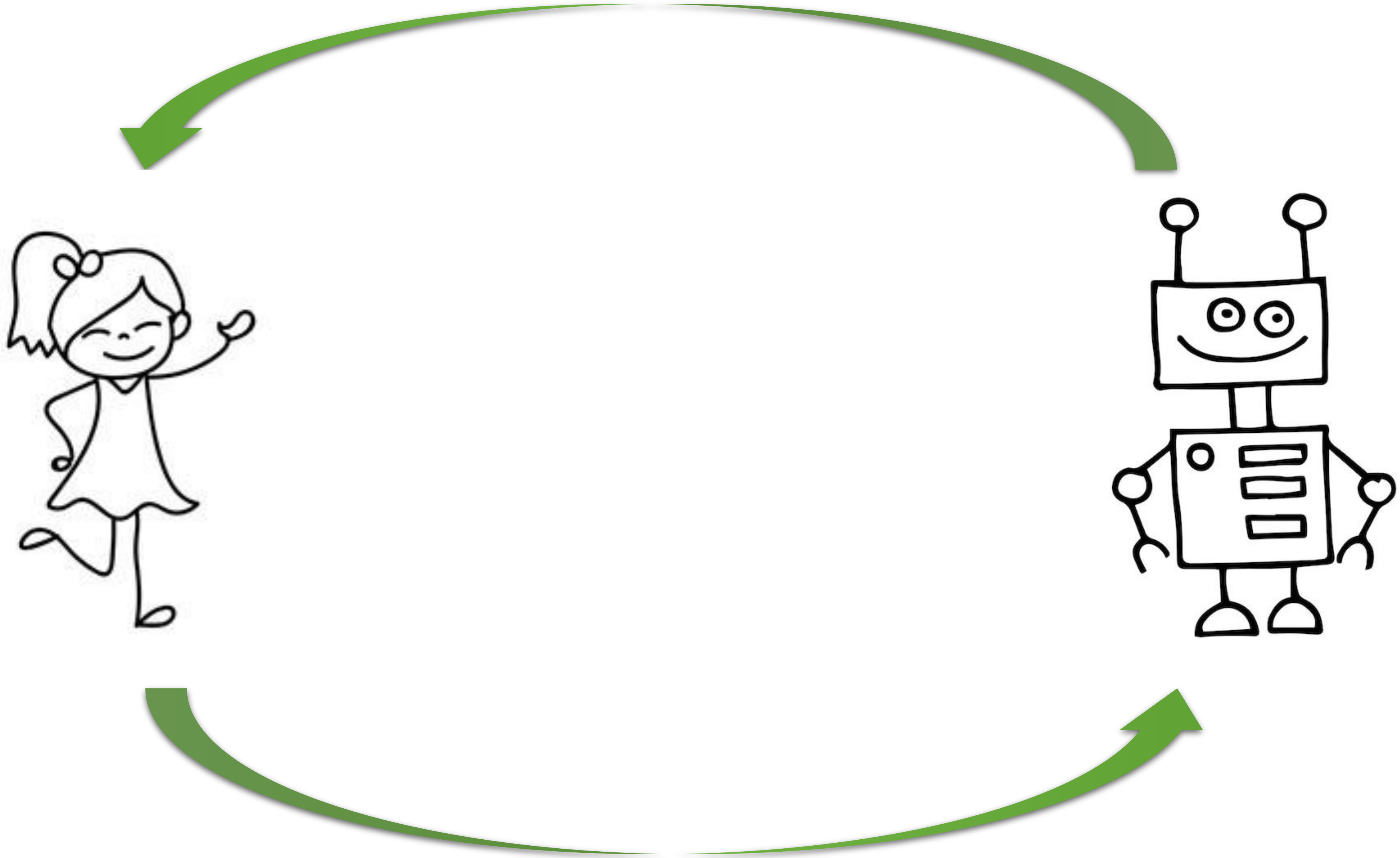
Insights from cognitive sciences into fast and flexible learning

Anne Collins, UC Berkeley

Sharif University of Technology, June 2025

Intelligent behavior necessitates fast and flexible learning

Humans provide unique examples of flexible behavior, complete with bugs!



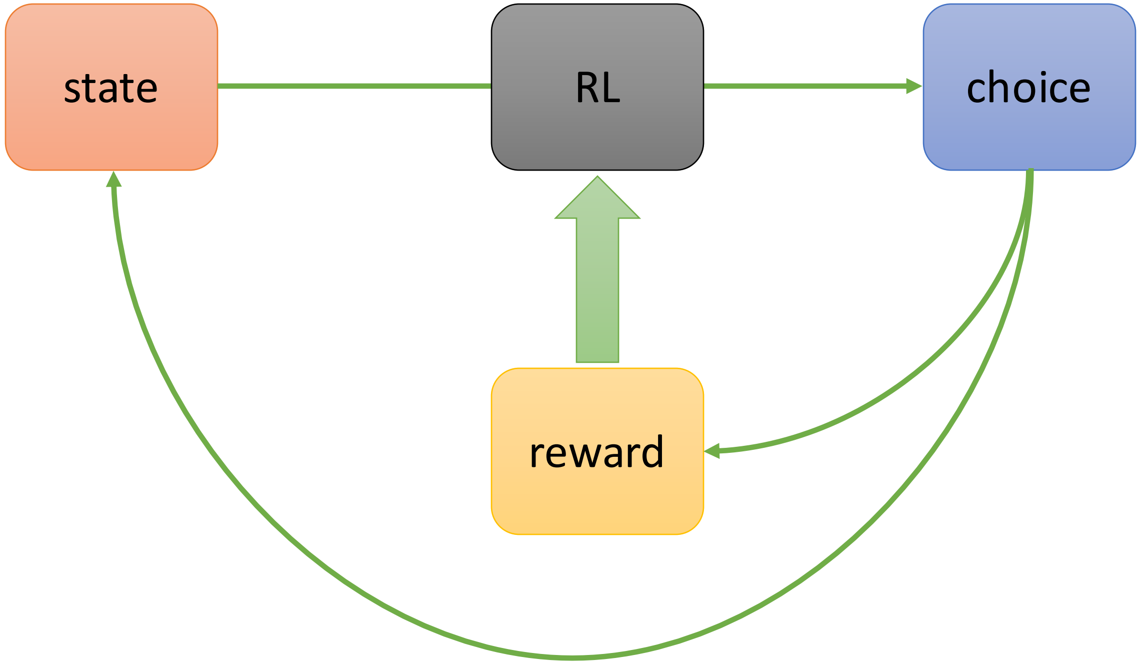
Computations supporting human intelligence:

Bugs or features?

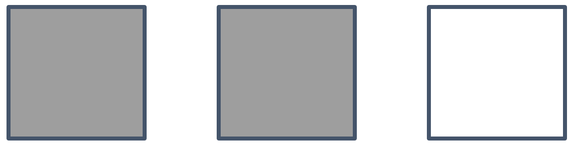
1. *Deconstructing the processes underlying human behavior.*

2. Bugs or features?

Reinforcement learning (RL)



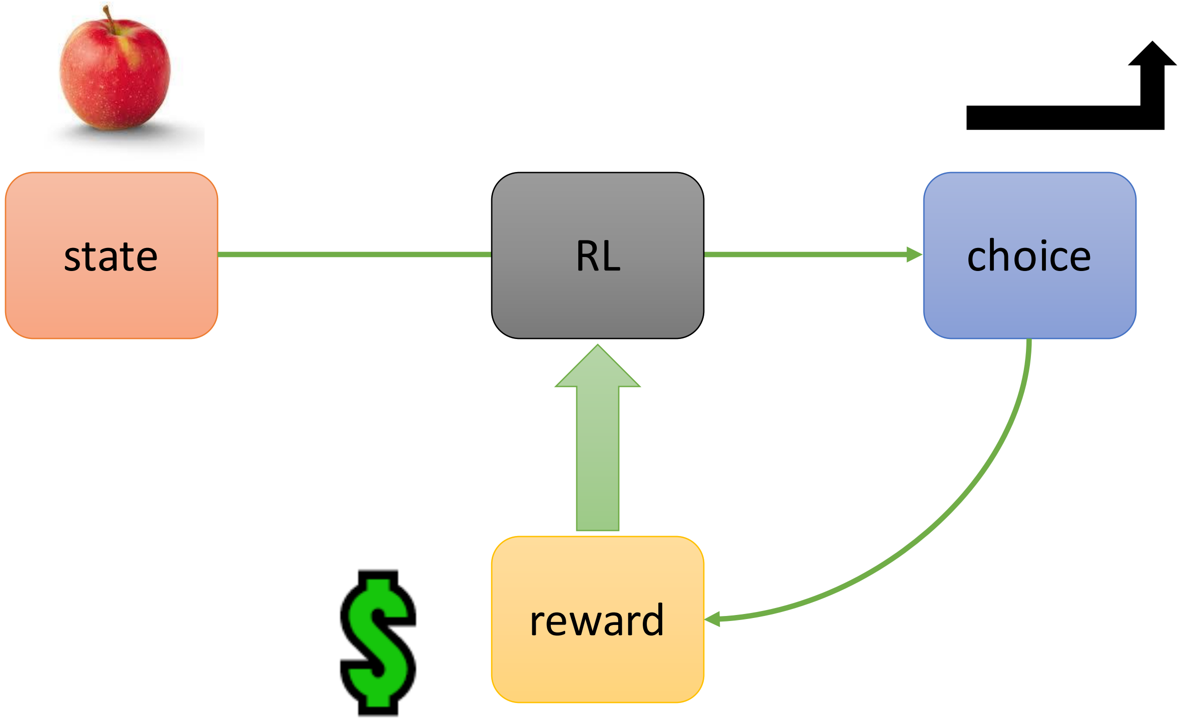
0 points

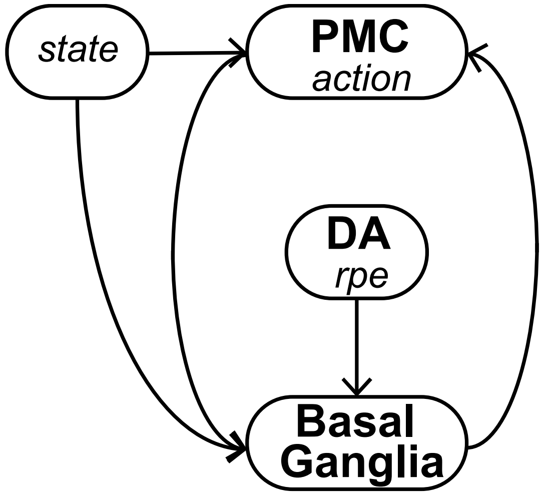


1 points

0 points

Reinforcement learning (RL)



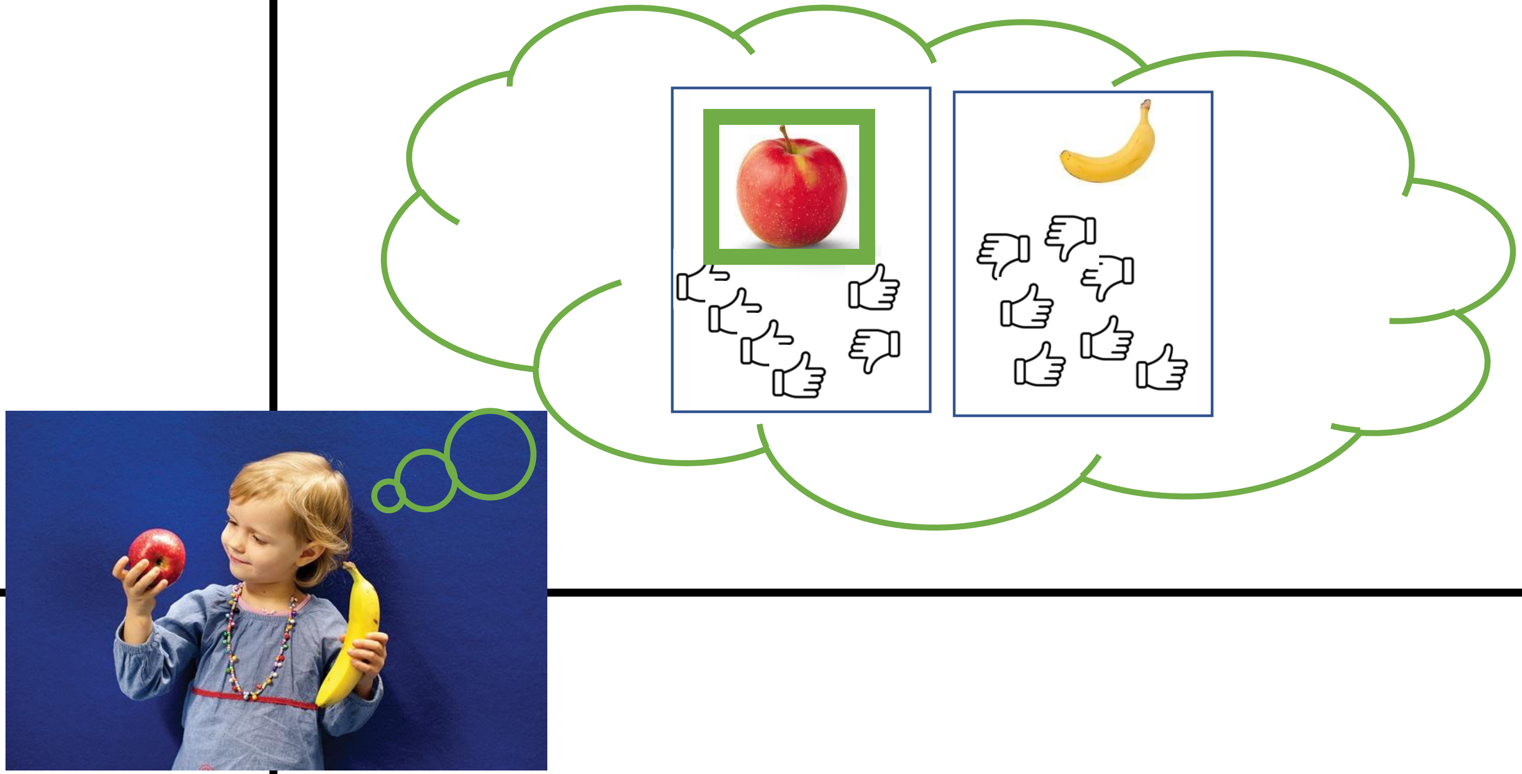


$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha\ (r - Q_t)$$

Functional, algorithmic level

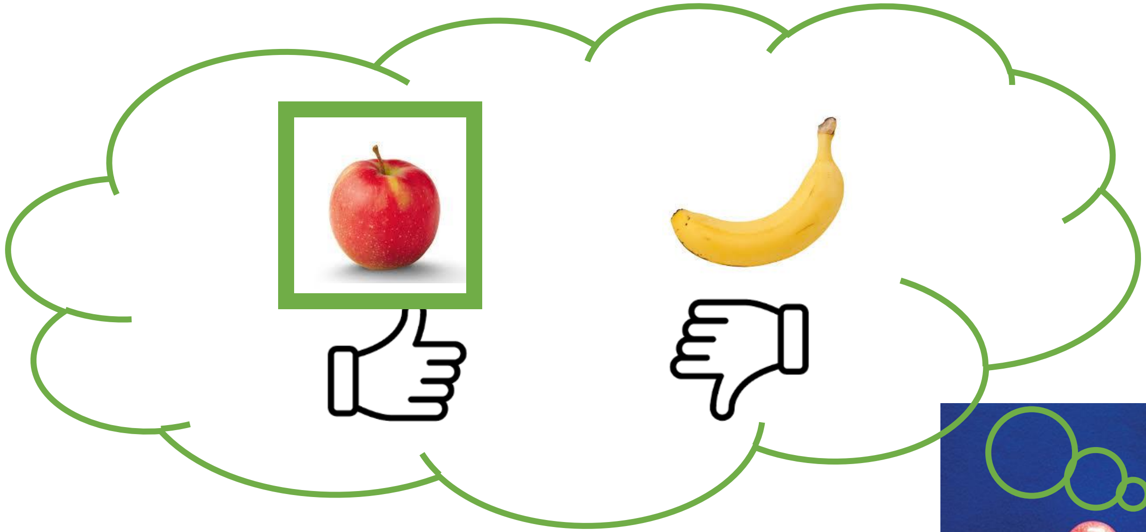
Implementation level

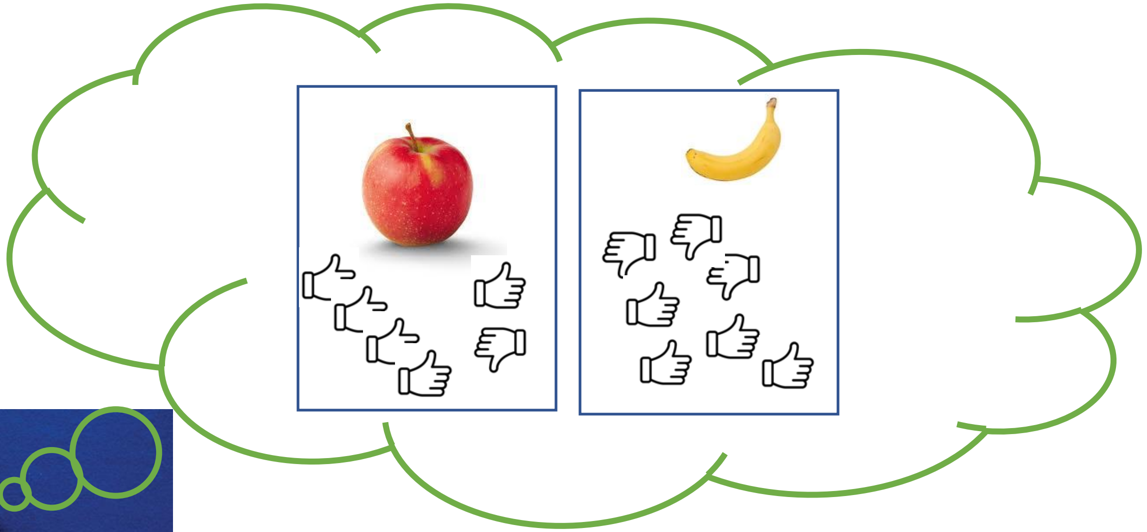
Value learning (RL)



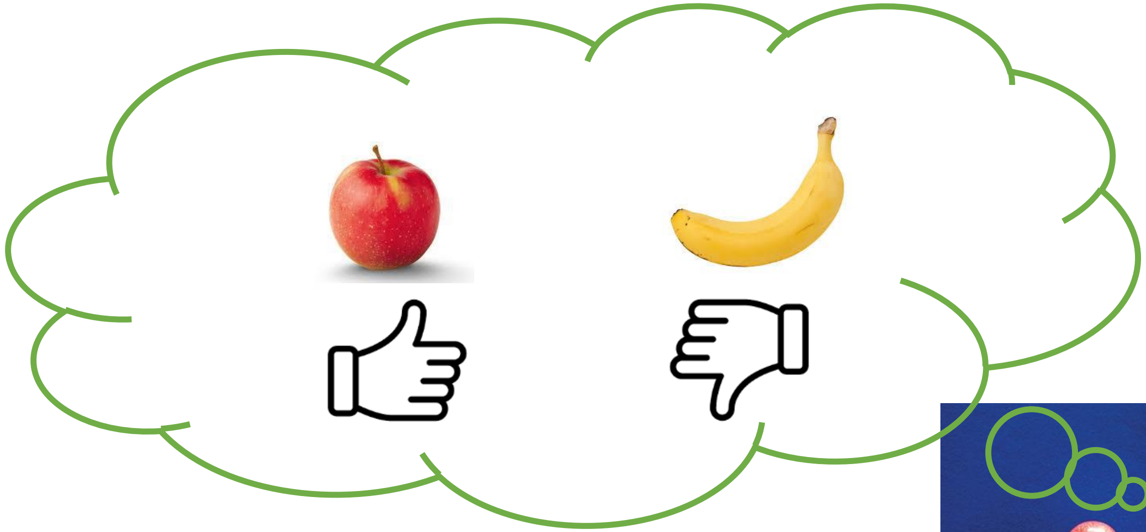
Short term memory

Value learning (RL)

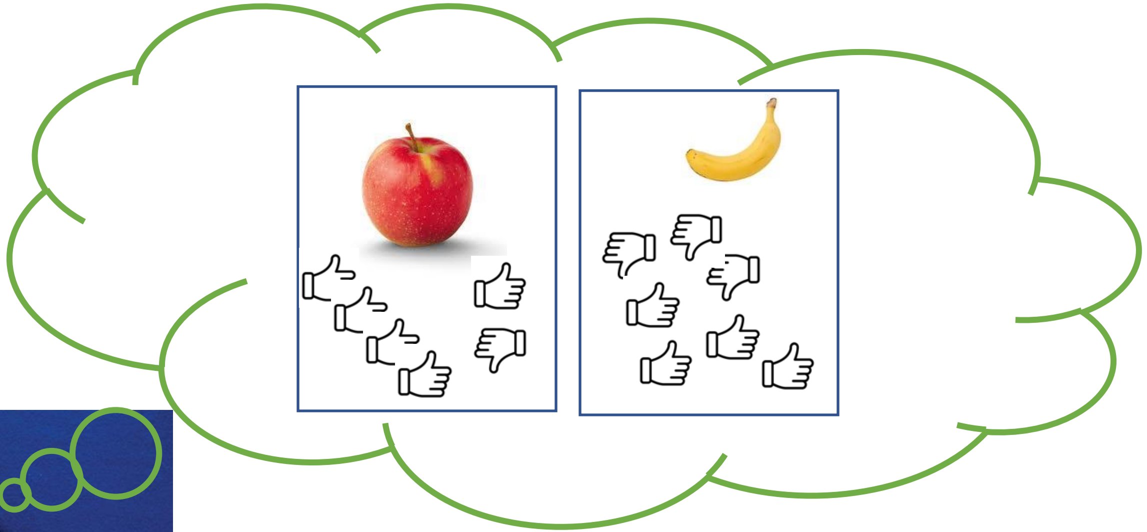




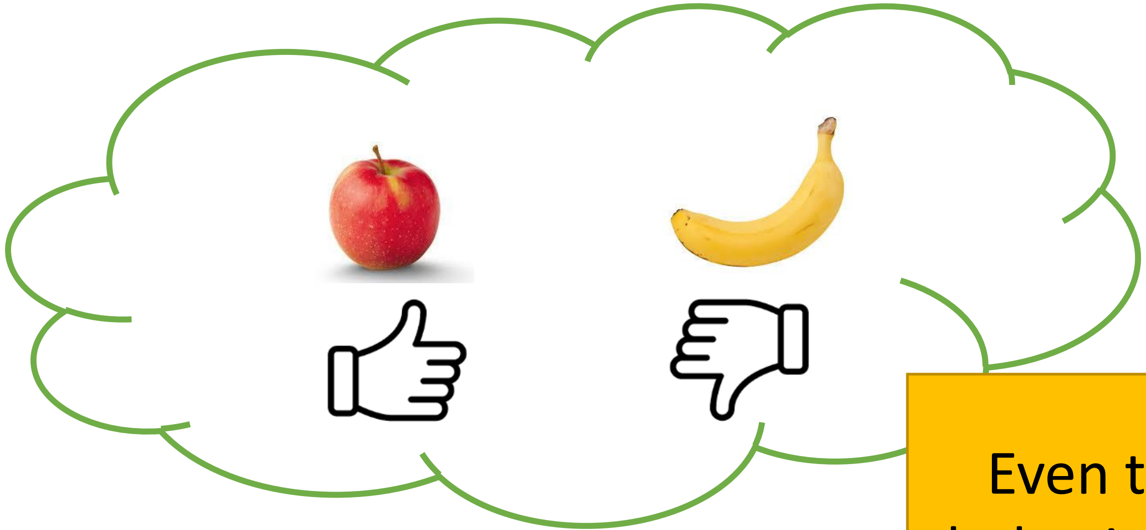
Short term memory



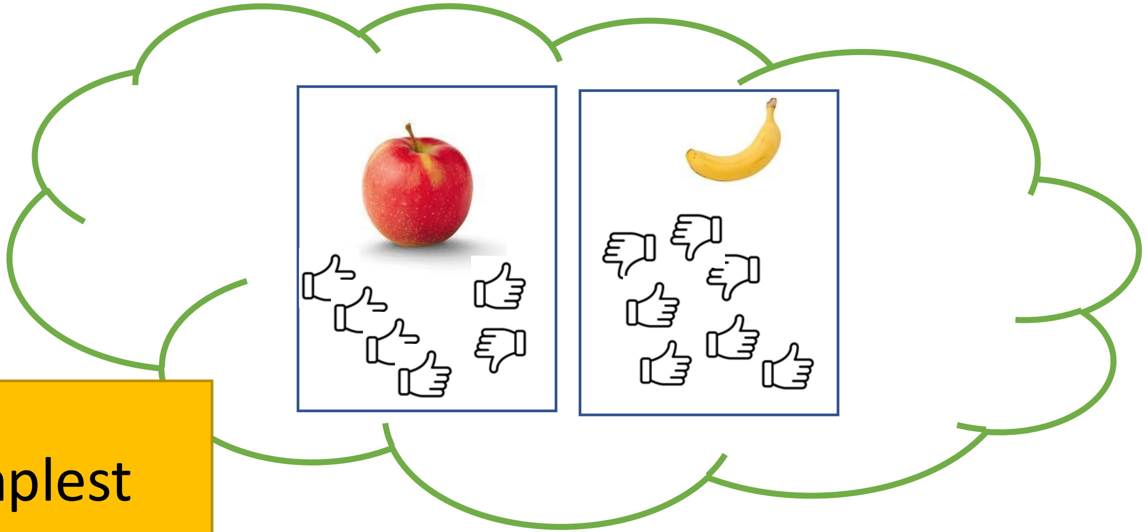
Value learning (RL)



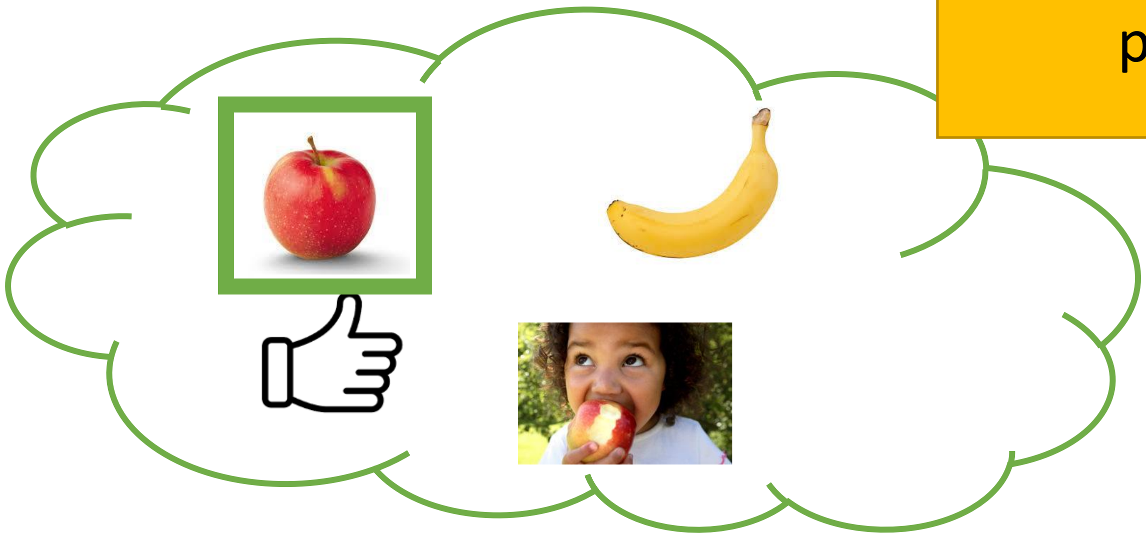
Short term memory



Value learning (RL)

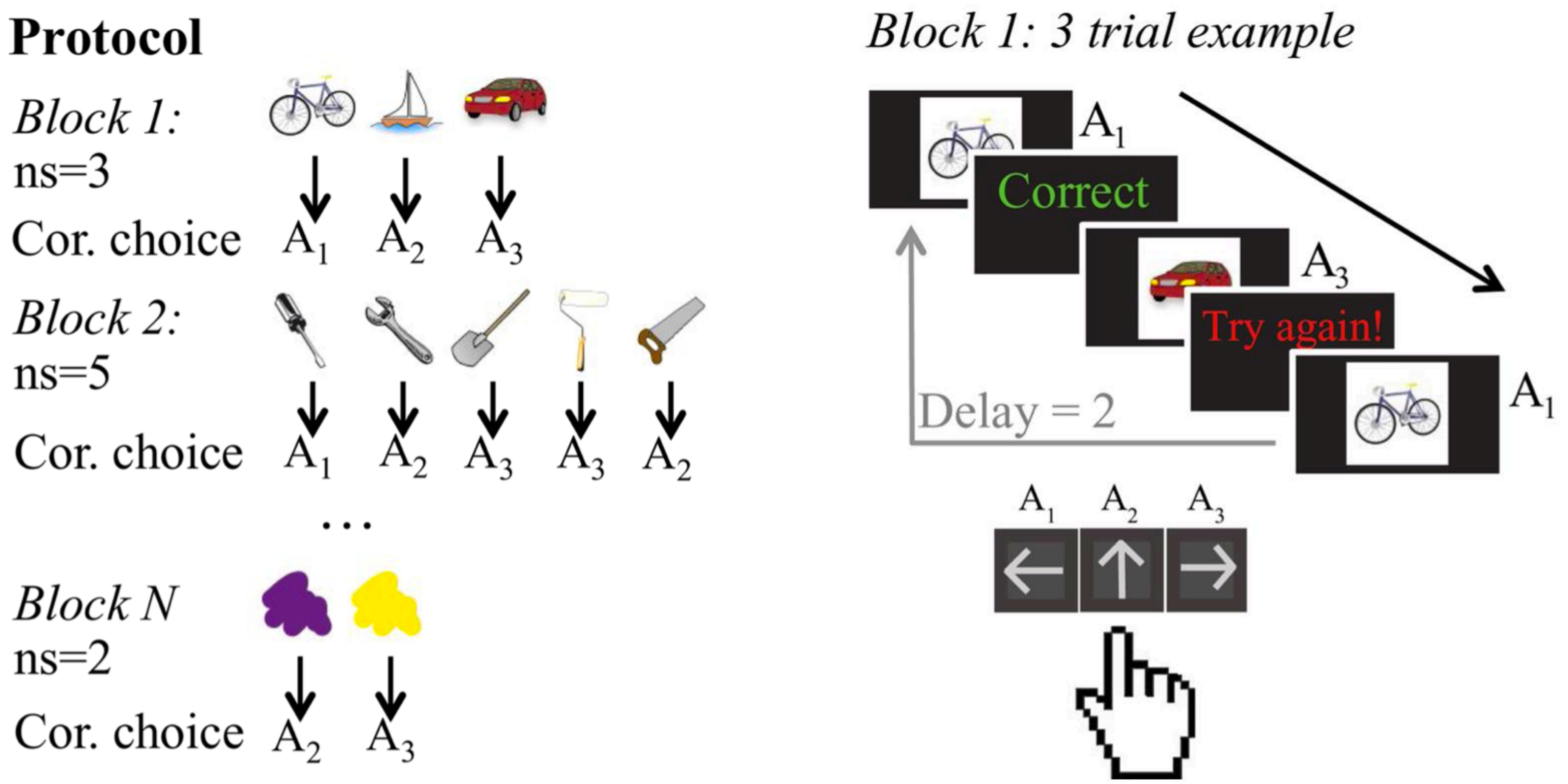


Even the simplest behavior is **complex,** relies on **redundant** processes



Episodic memory

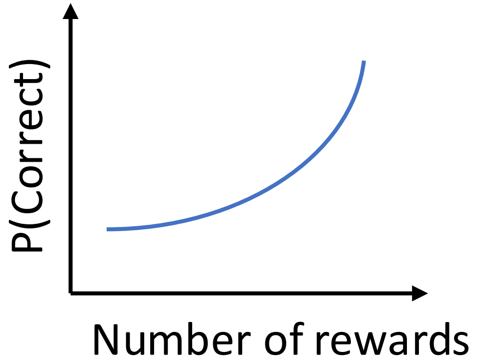
- Exploration
- Heuristics
- Strategies
- …



Protocol
Block 1:
ns=3
Cor. choice A1 A2 A3
Block 2:
ns=5
Cor. choice A1 A2 A3 A3 A2
...
Block N
ns=2
Cor. choice A2 A3
Block 1: 3 trial example
A1
Correct
A3
Try again!
A1
Delay = 2
A1 A2 A3

Value learning (RL)

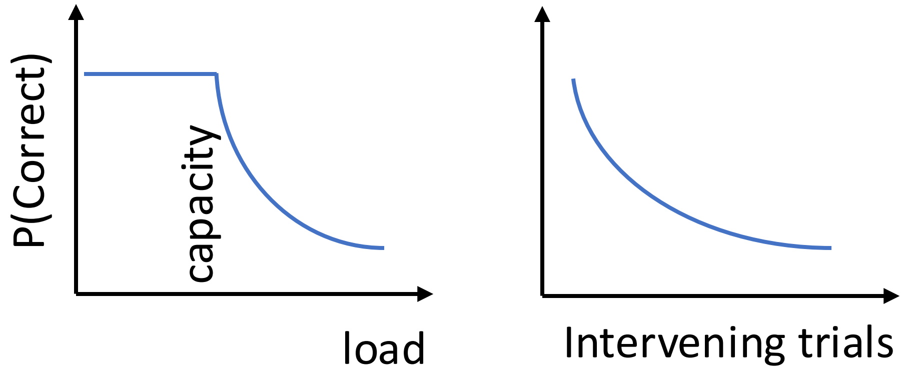
- Effects of cumulative reward
- Effects of positive/negative outcomes



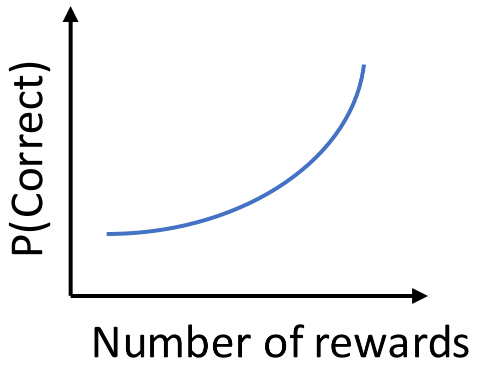
- Value estimation from reward prediction errors:
  $Q(S,A) \leftarrow Q(S,A) + \alpha[r_t - Q(S,A)]$
- Value-dependent choice policy

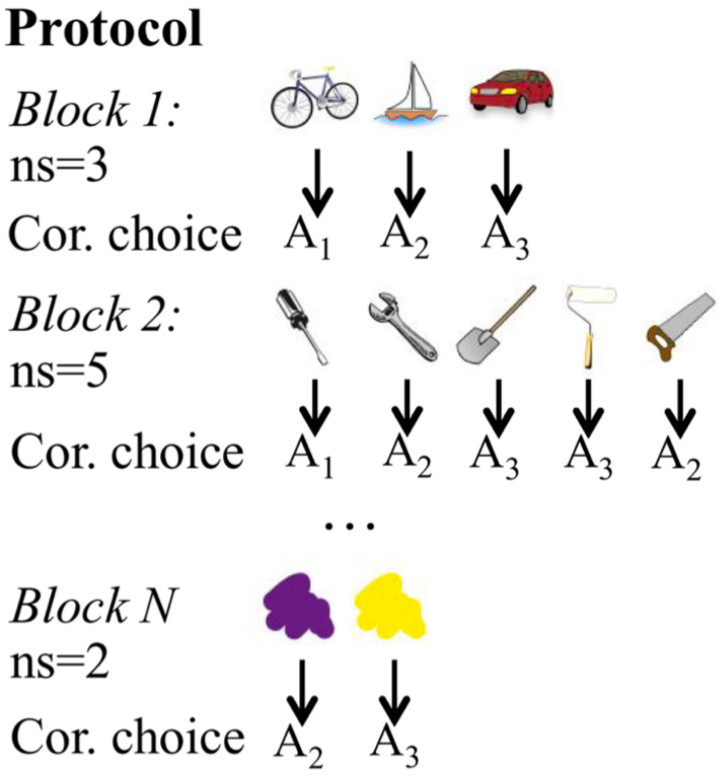
Collins & Frank 2012

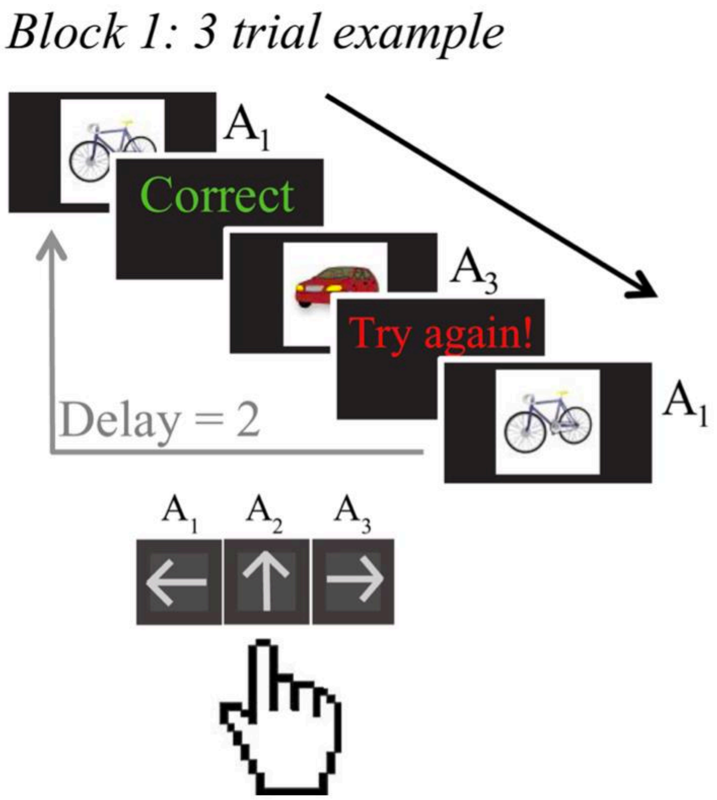
Working memory

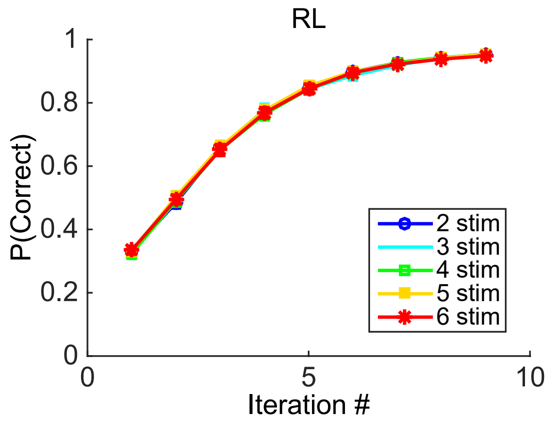


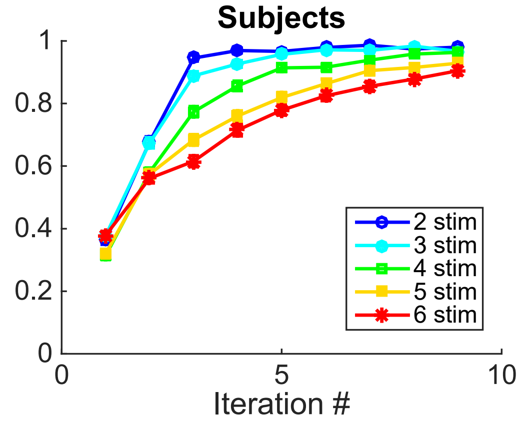
Value learning (RL)







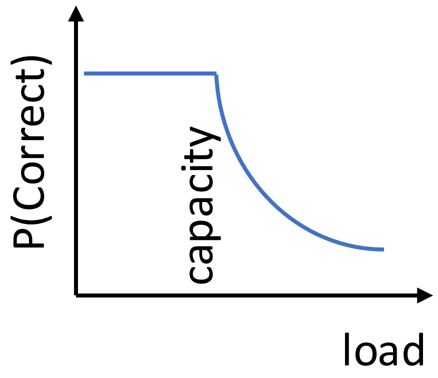


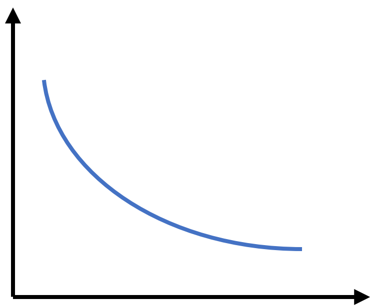


Working memory

Value learning (RL)

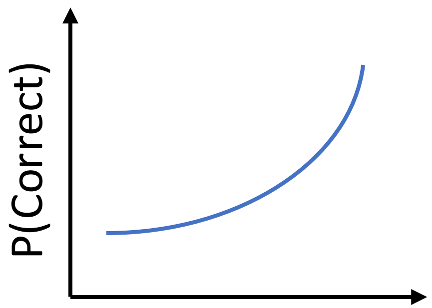
- Load effects
- Short term delay effects





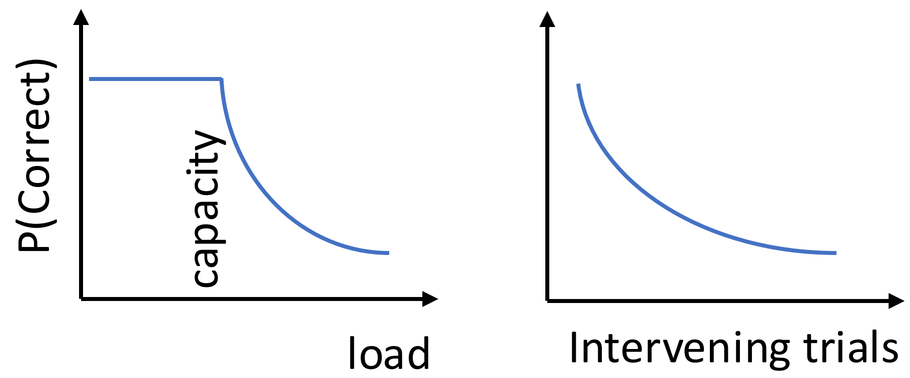
- Perfect memory of last trial:
  $W(S,A) \leftarrow r_t$
- Fast forgetting:
  $W \leftarrow W + \phi[W_0 - W]$
- Capacity dependent policy

- Effects of cumulative reward
- Effects of positive/negative outcomes

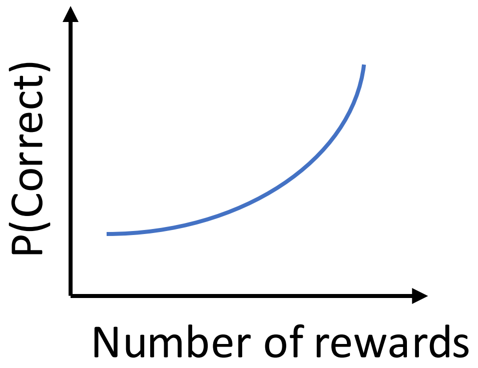


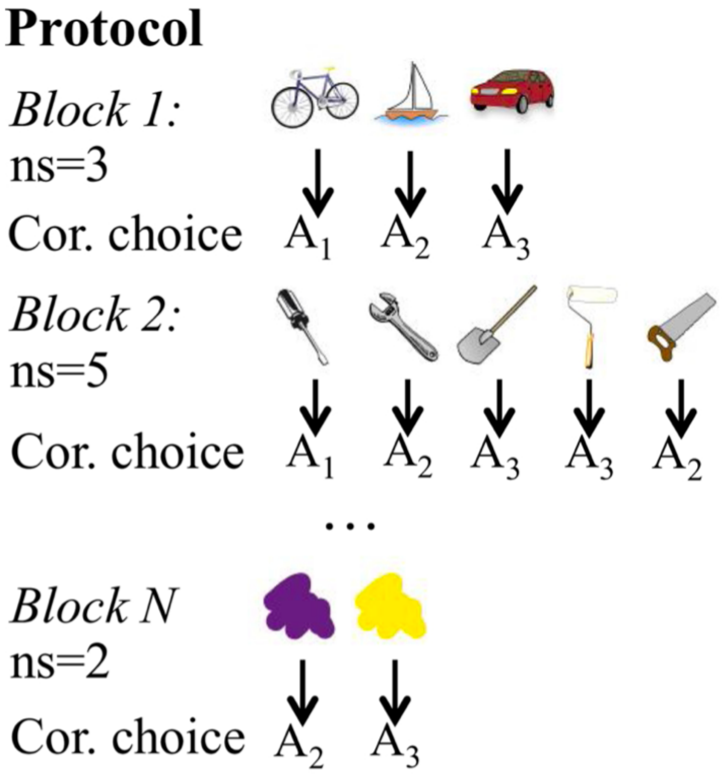
- Value estimation from reward prediction errors:
  $Q(S,A) \leftarrow Q(S,A) + \alpha[r_t - Q(S,A)]$
- Value-dependent choice policy

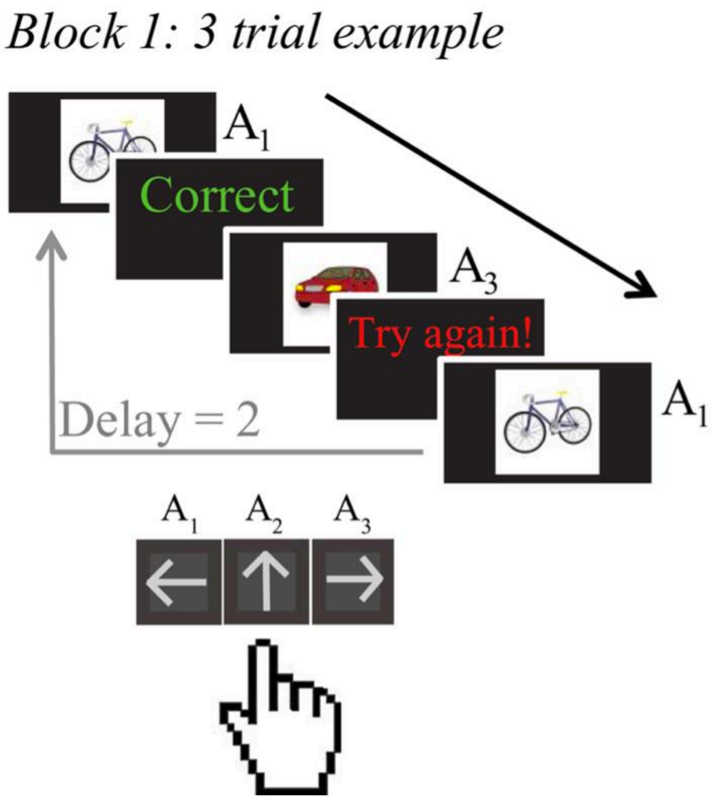
Working memory

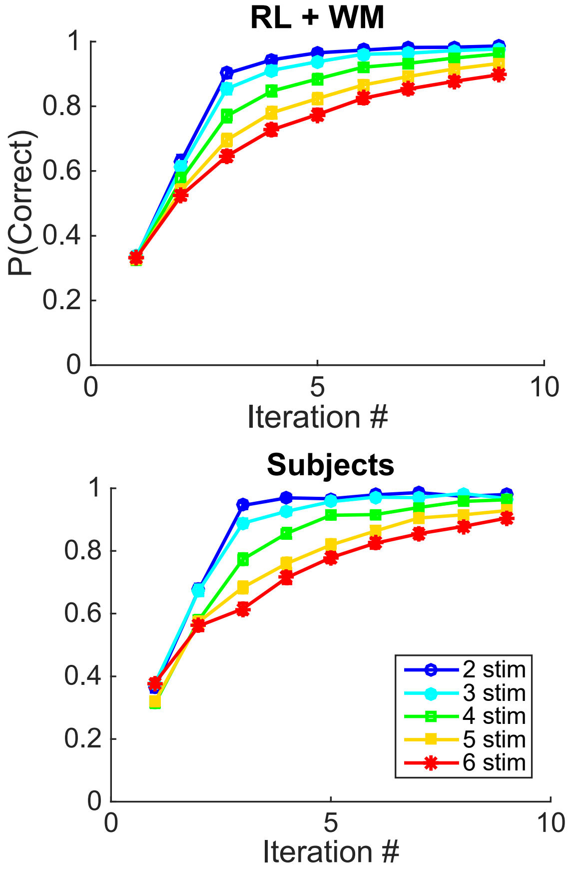


Value learning (RL)







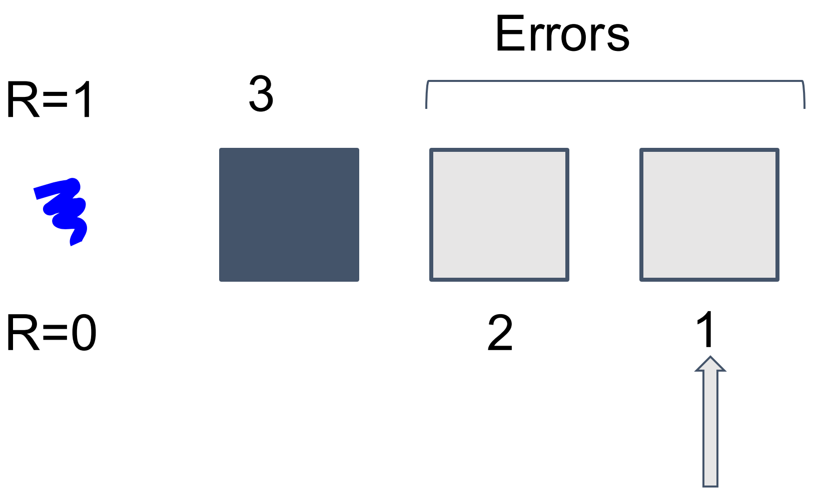


Conclusion so far:
RL is augmented by working memory to support flexible learning?

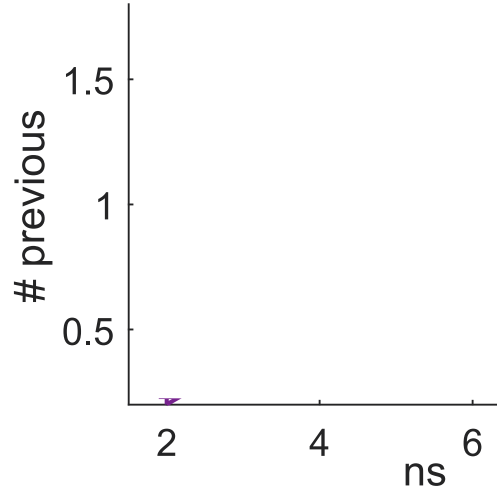
What about RL?

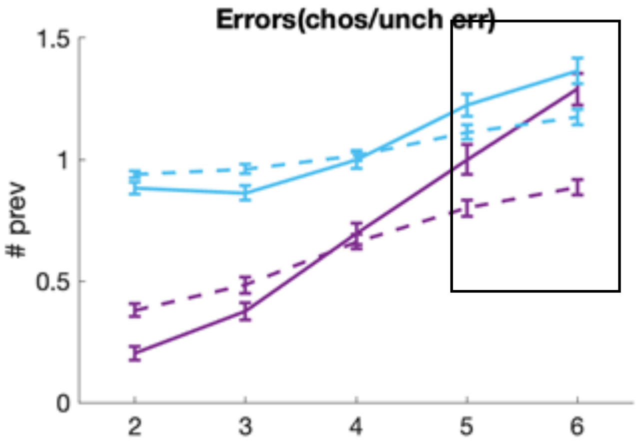
Errors

- Effects of cumulative reward
- Effects of positive/**negative outcomes**



 # previous same errors (|stim)
 # previous different errors (|stim)



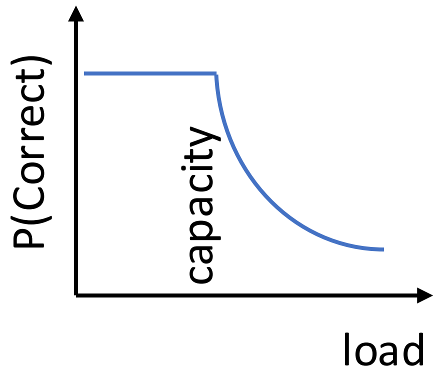


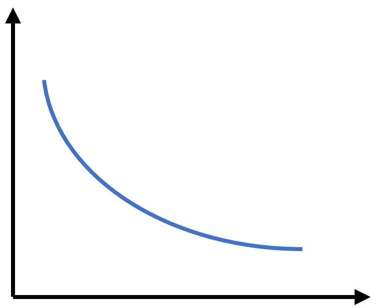
- Good ***avoidance*** of past errors in low set sizes
  Sensitivity to negative feedback ***disappears*** in high set sizes
  RL/WM model cannot capture this well

Working memory

Value learning (RL)

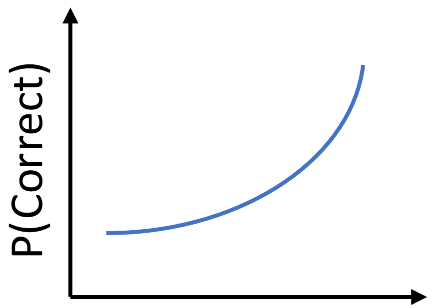
- Load effects
- Short term delay effects





- Perfect memory of last trial:
  $W(S,A) \leftarrow r_t$
- Fast forgetting:
  $W \leftarrow W + \phi[W_0 - W]$
- Capacity dependent policy

- Effects of cumulative reward
- Effects of positive/negative outcomes

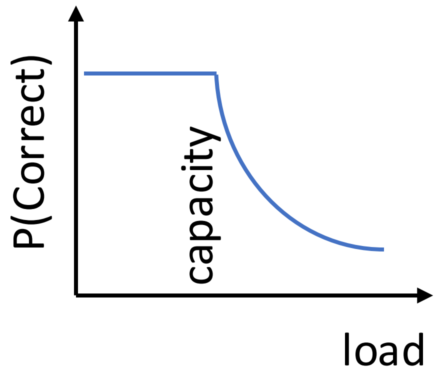


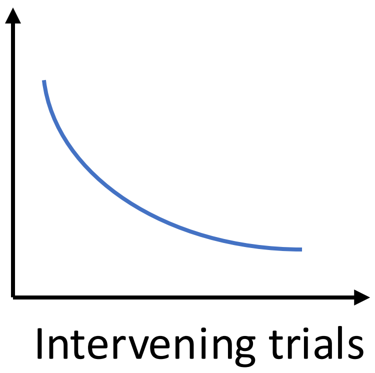
- Value estimation from reward prediction errors:
  $Q(S,A) \leftarrow Q(S,A) + \alpha[r_t - Q(S,A)]$
- Value-dependent choice policy

Collins & Frank 2012

Working memory

- Load effects
- Short term delay effects

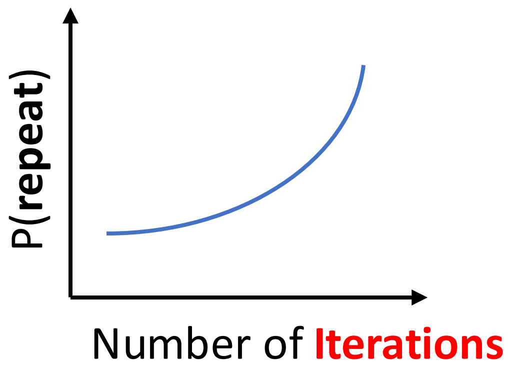




- Perfect memory of last trial:
  $W(S,A) \leftarrow r_t$
- Fast forgetting:
  $W \leftarrow W + \phi[W_0 - W]$
- Capacity dependent policy

Habit
(Value-free learning)

- Effects of cumulative **iterations**
- **No** effects of outcomes

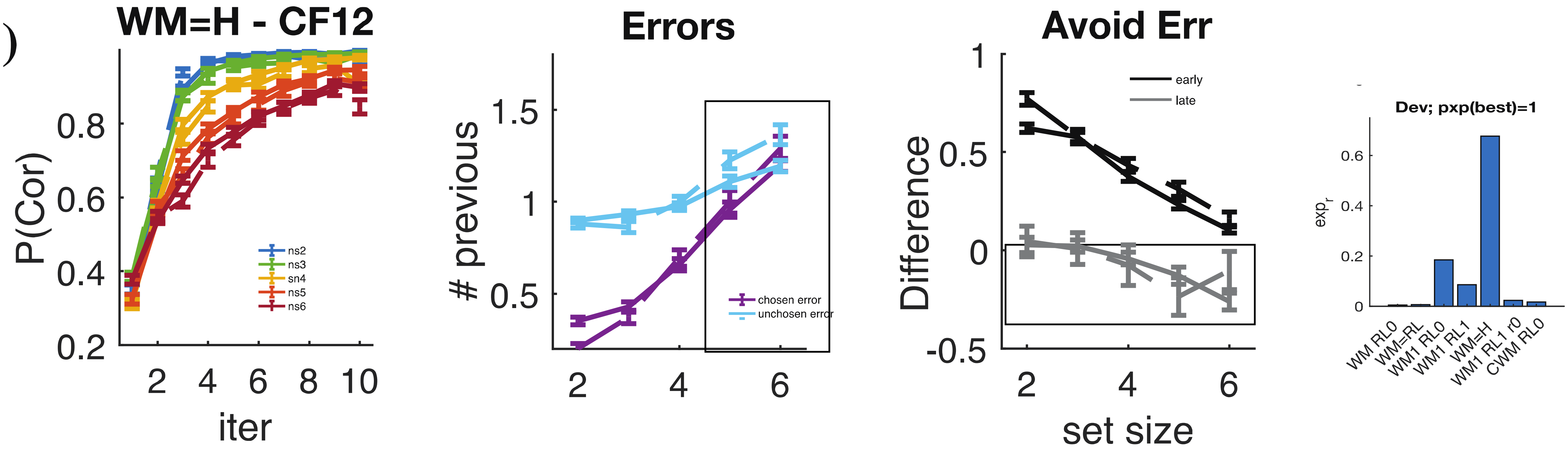


- Learned weights from Hebbian/habit update:
  $H(S,A) \leftarrow H(S,A) + \alpha[1 - Q(S,A)]$
- **No value learning**

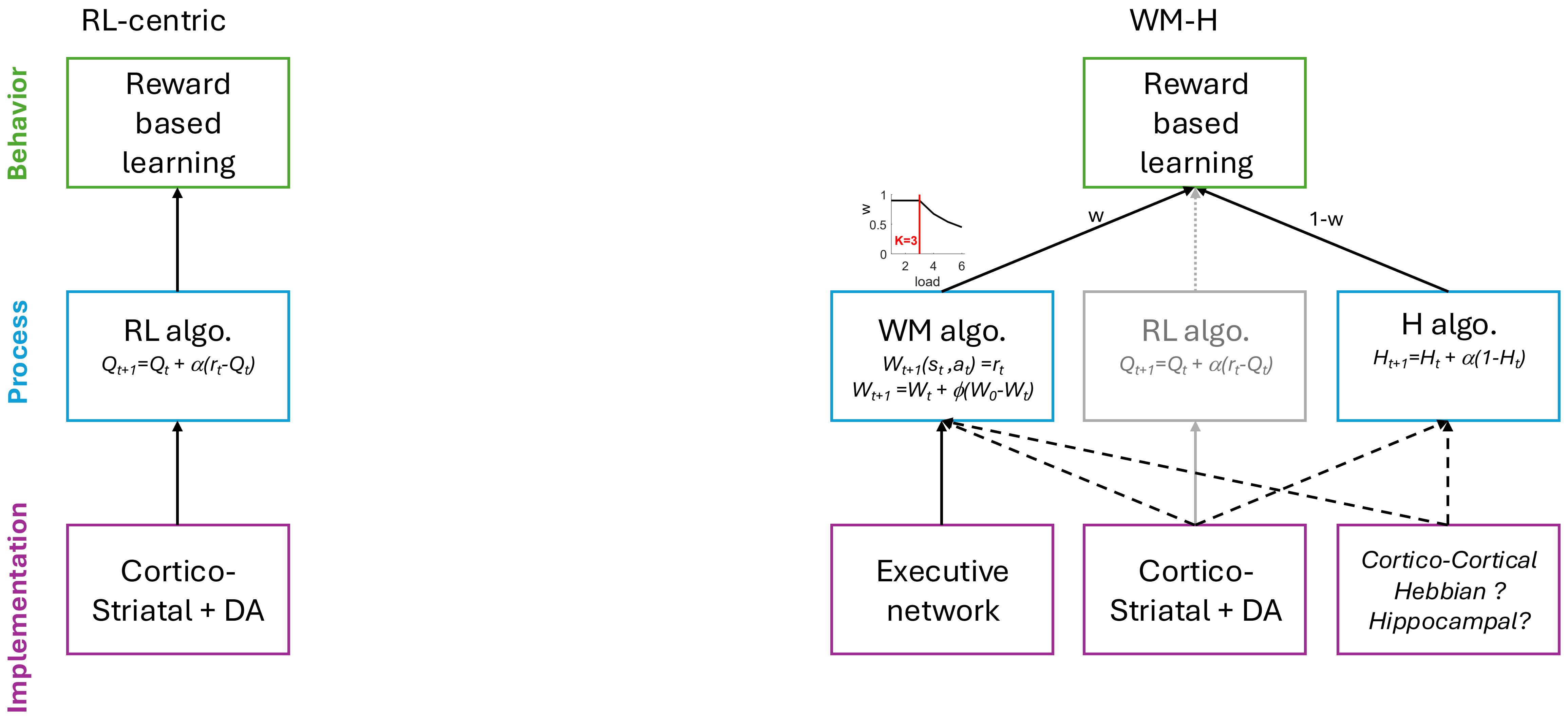
Collins & Frank 2012

WM + habit (not RL) captures error pattern across WM loads

C)



Deconstruction

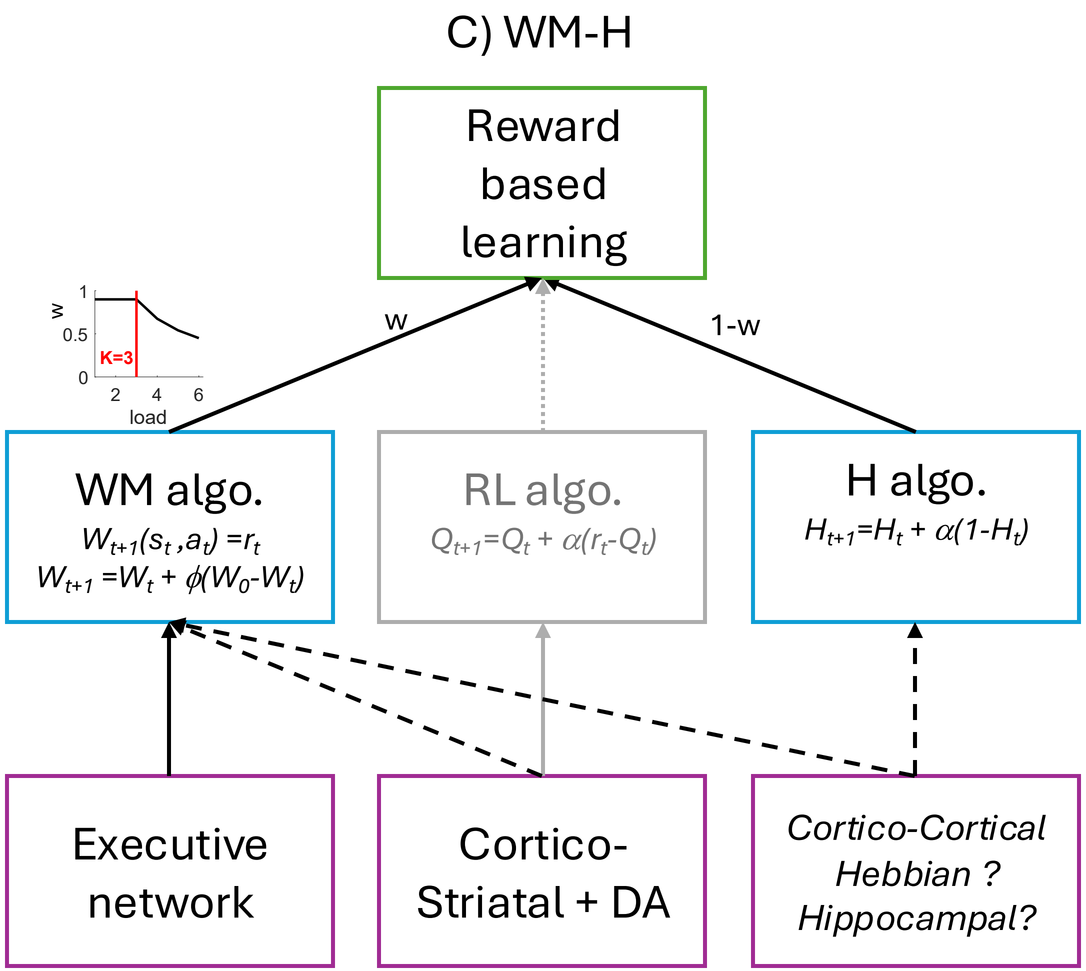


1. Deconstructing the processes underlying human behavior.

2. ***Bugs or features?***

Bugs or features?

1. Simplicity
2. Redundancy
3. Tradeoffs
4. Bottlenecks
5. Complexity

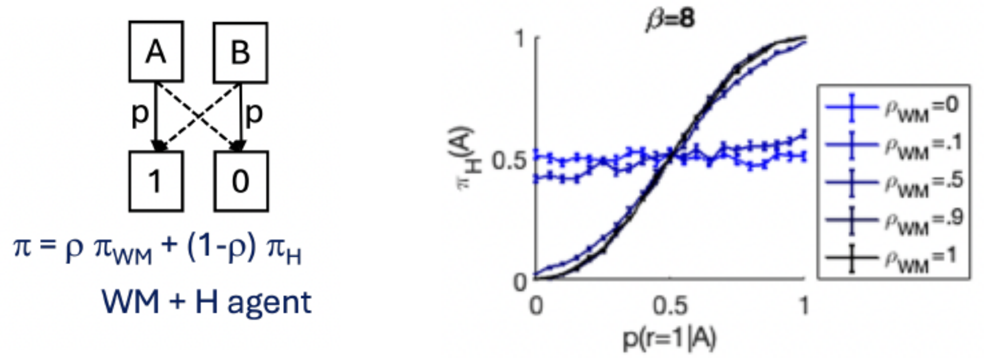


1. Simplicity – bug or feature?

"Habit" $\boxed{H_{t+1}(s,a) = H_t(s,a) + \alpha\,(1-H_t)}$

- Habit process is ***suboptimal for learning***:
  - Does not track value, but frequency
    - Making an error makes it more likely, not less → not RL
  - How is that helpful? Bug?

- Feature
  - In the presence of WM to guide some choices, ***H Learns a good policy***
  - With a ***simpler*** computation, less attention.



Collins, 2024, psyRxiv; submitted
Miller et al, 2019

2. Redundancy – bug or feature?

WM
- Fast & flexible
- Forgetful
- Capacity limited

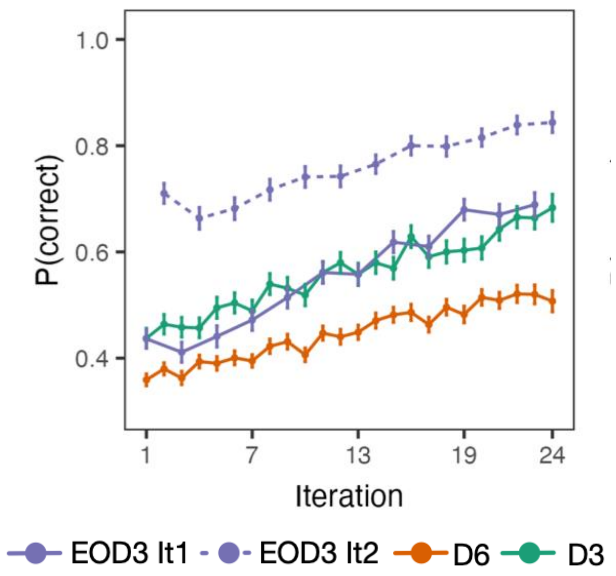
Habit
- Slow & inflexible
- Effortless
- Broad and robust

Partial ***redundancy*** in WM and habit ***mitigates*** each memory system's information tradeoff

Why the ***tradeoffs?***

3. Fast vs. slow tradeoff

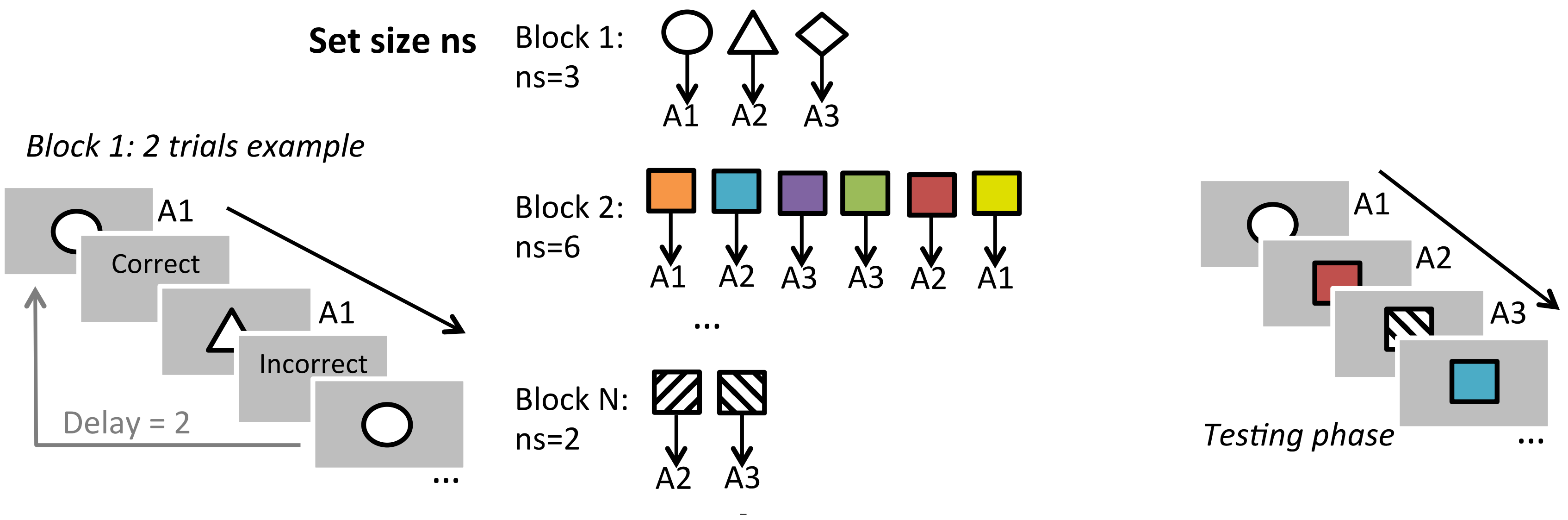
- WM is immediate, but RL is slow
  - E.g. Zhang et al, submitted – need ~25 rewards, up to ~100 iterations to reinforce an assocations
  - Why not faster? Bug or feature?



- Fast (WM) allows fast learning!
  - Essential in ***fast-changing dynamical*** environment
  - But could also imply fast forgetting

- Slow (RL) allows robust, long-term learning
  - Integrates value over noise
  - Essential in ***stable*** environments

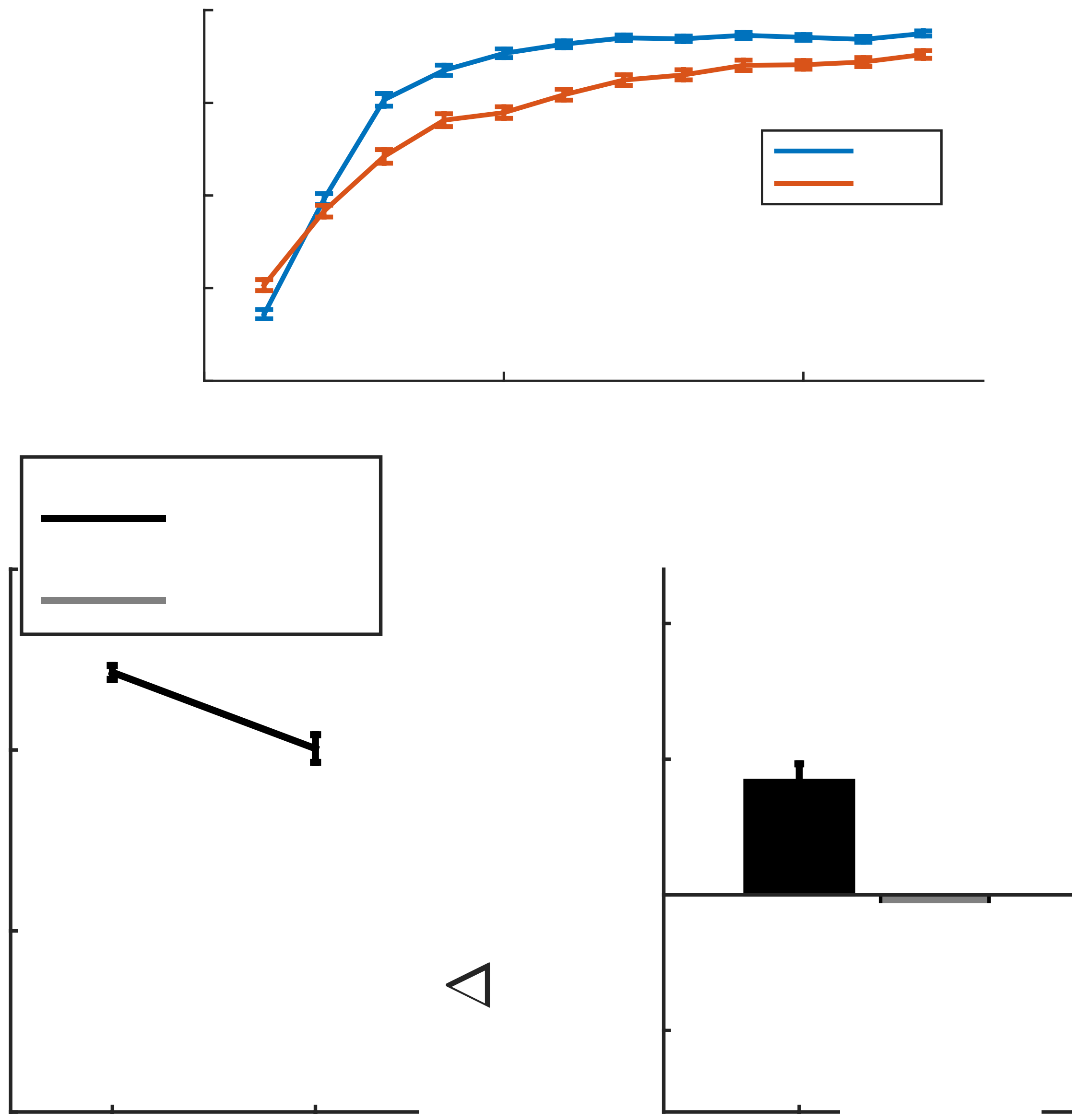
3. Fast vs. slow tradeoff.



RL + WM

RL only

Collins & Frank, 2018
Collins 2018

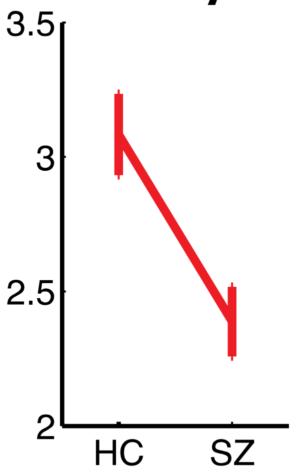


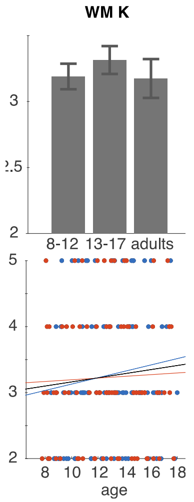
Fast WM learning **blocks** long-term retention.

Slower RL enables more **robust**, long-term learning

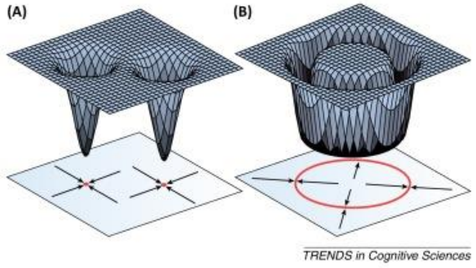
4. Resource tradeoff: low WM capacity

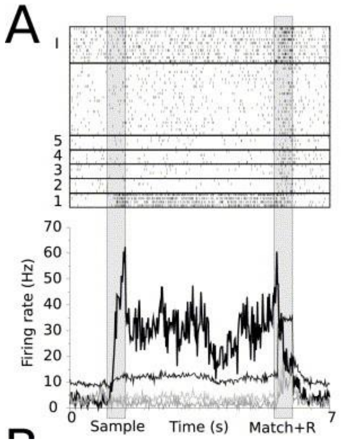
- RL has broad capacity, but WM is very capacity/resource limited
  - 3-4 compounds!!!
  - Why not more? Bug or feature?





- Bug?
  - WM relies on costly active maintenance in spiking neurons (e.g. Compte 2006, Bays 2015)
  - Our brain has not evolved to enable more WM resources?
    - Limitations of attractor networks?
  - → Irrelevant to AI (can provide more resources)?

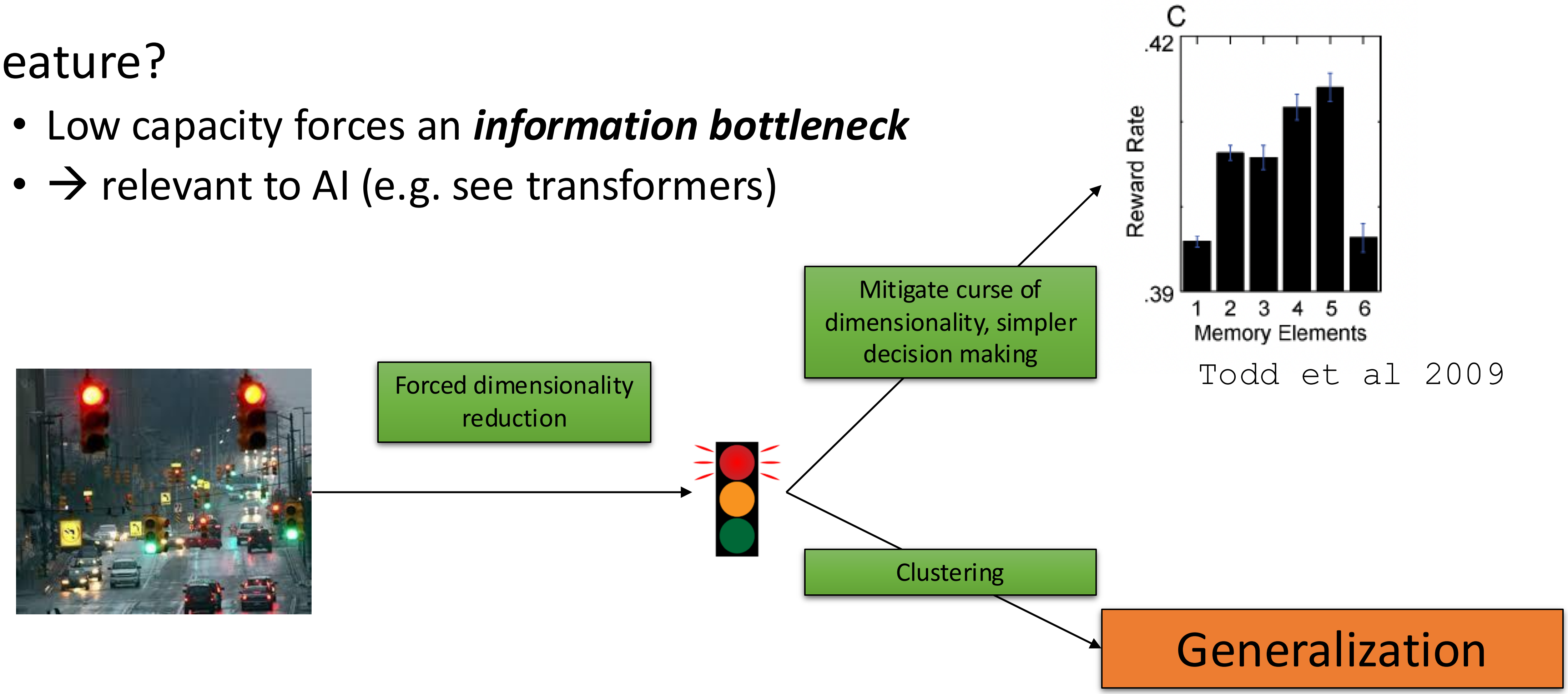




Master et al 2019; Collins et al 2017

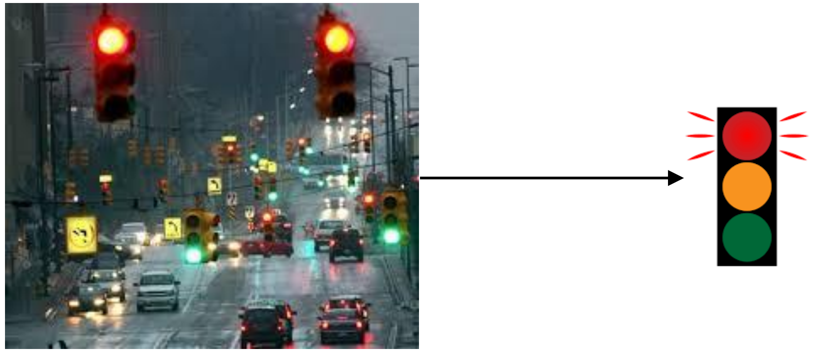
4. Resource tradeoff: low WM capacity

- Feature?
  - Low capacity forces an ***information bottleneck***
  - → relevant to AI (e.g. see transformers)

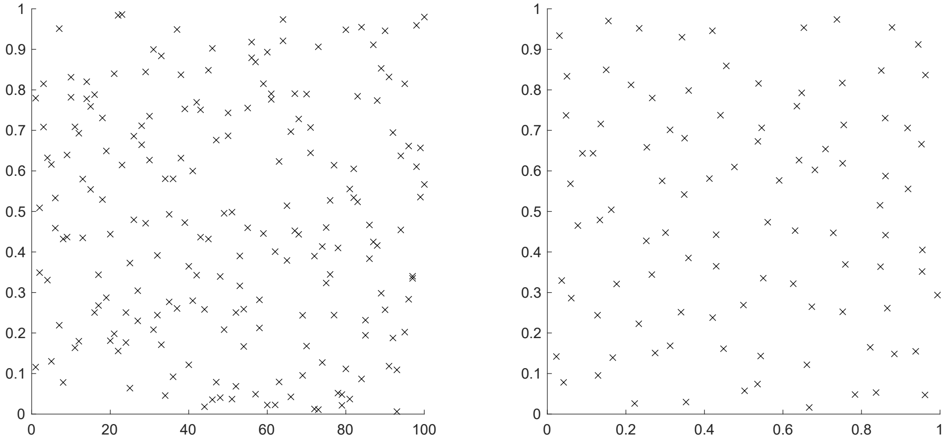


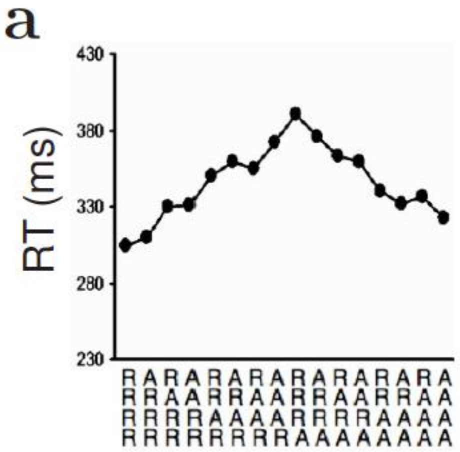
Collins & Frank 2016

Dimensionality reduction

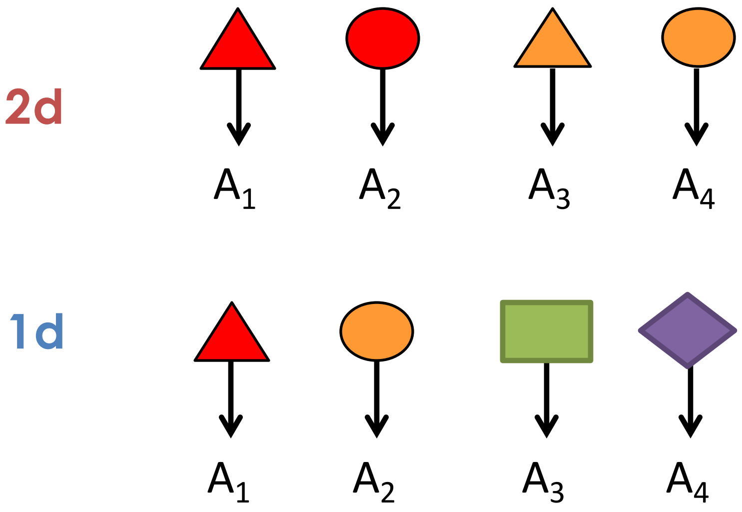


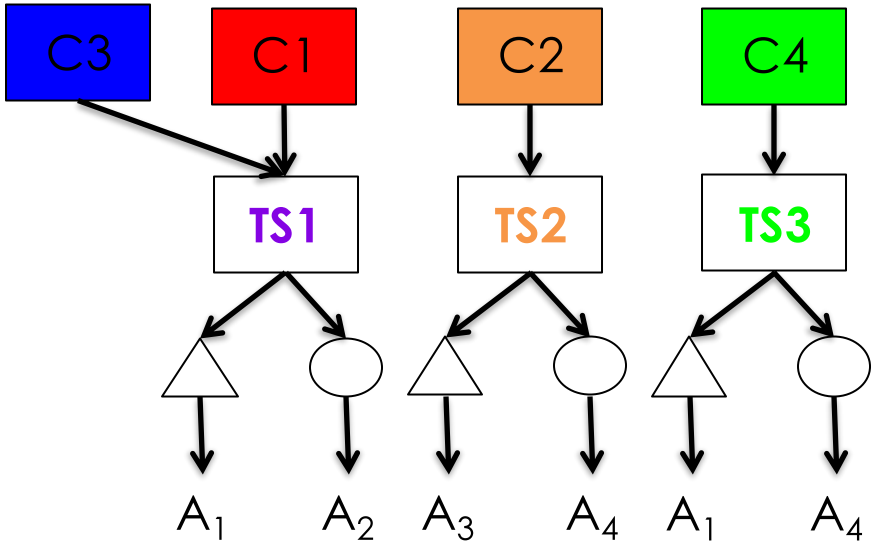
Complexification

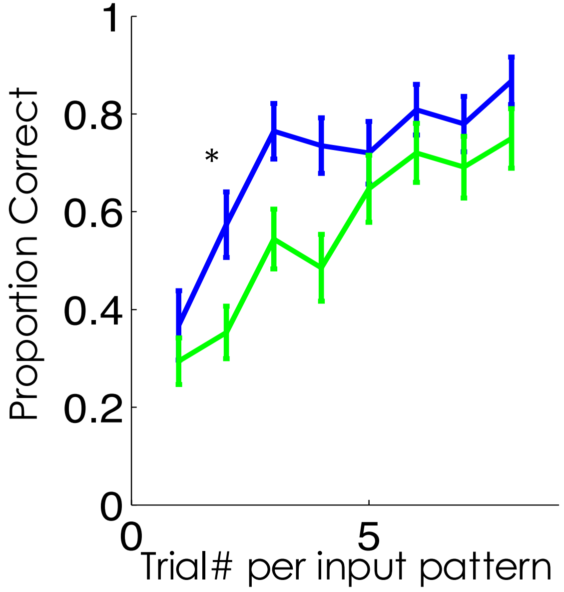


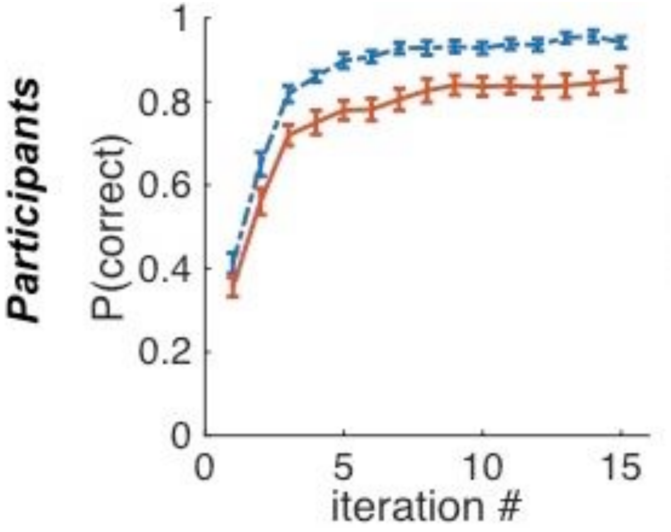


Yu et al 2008









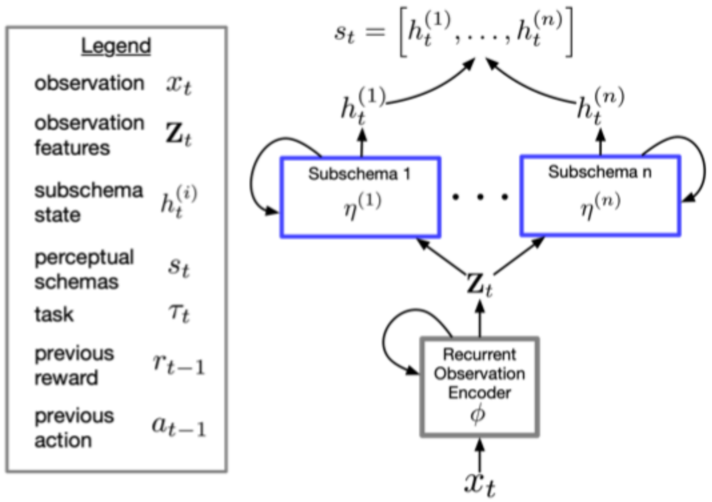
- Complex structure is a costly default
- Structure enables transfer and generalization.

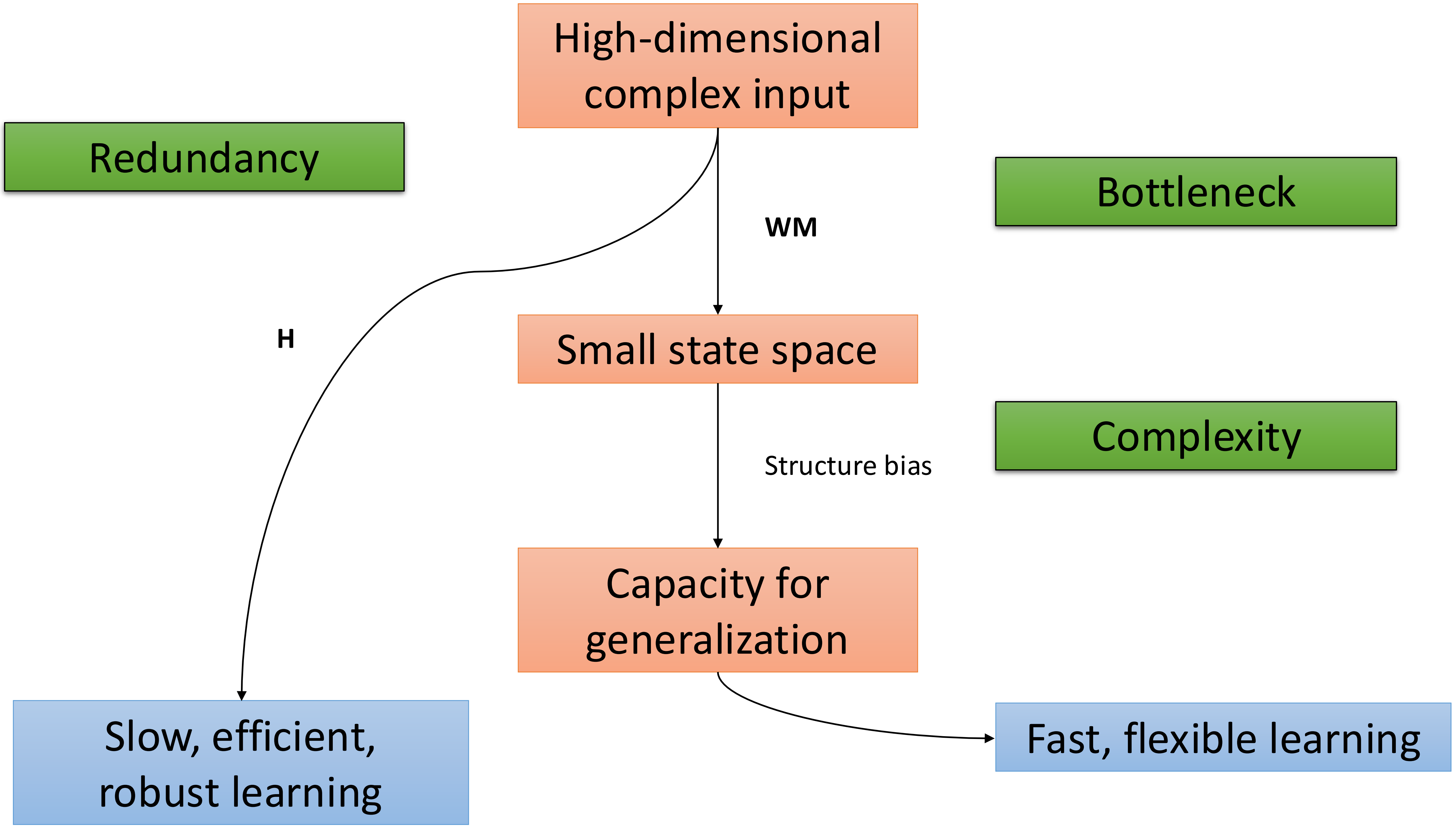
Collins, JoCN, 2017; Collins & Frank 2013, 2016

5. Complexity – (Strong structure representations)

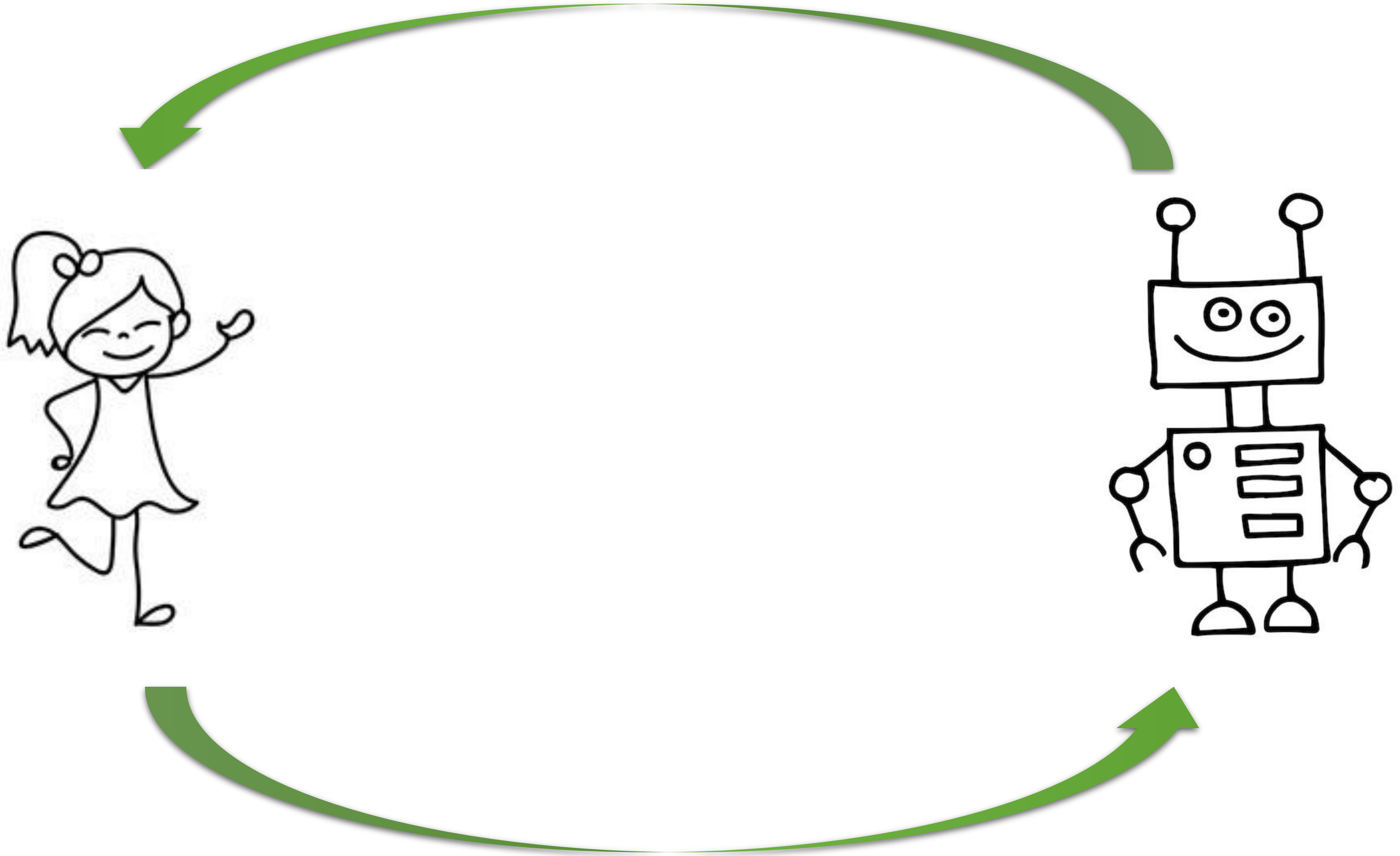
- Bug?
  - Accuracy and reaction time cost (e.g. Yu et al 2009, Collins 2018, …)
  - More complex representations with more memory requirements (E.g. Collins & Frank 2013, Collins & Frank 2016)

- Feature?
  - Inductive bias that enables transfer and generalization
  - May be resource optimal when working on low dimensional states
    - Divide and conquer approach
  - AI evidence for benefits of structured complexity
    - E.g. Composable Perceptual Schemas (Carvalho et al)




High-dimensional complex input
Redundancy
WM
Bottleneck
H
Small state space
Structure bias
Complexity
Capacity for generalization
Slow, efficient, robust learning
Fast, flexible learning

Computations supporting human intelligence: *Bugs or features?*



Thank you!

CCN lab (current and former)

CCN LAB

https://ccn.berkeley.edu
@Anne_on_tw
@ccnlab

Funding

- NSF SL-CN 1640885
- NSF 2020844
- NIMH R01-MH119383
- NIMH R21- MH057804
- NIMH R21-MH132974
- NIMH P50-056162

Many collaborators