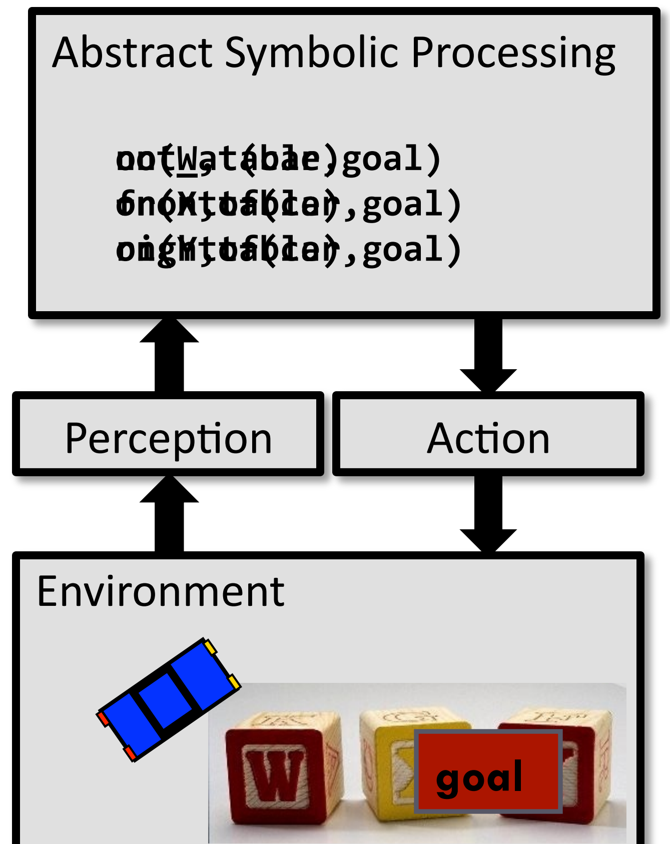
Using Imagery to Simplify Perceptual Abstraction in Reinforcement Learning Agents

Sam Wintermute, University of Michigan

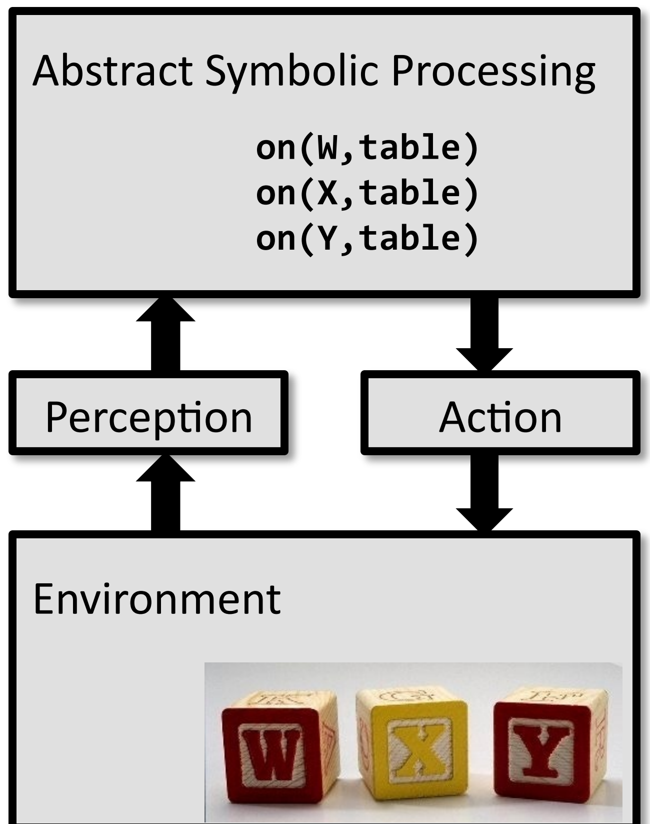
Perceptual Abstraction

- We are trying to build a task-independent cognitive architecture
- Different tasks require different abstract perceptual properties
  - For spatial properties, there does not seem to be a universal set
  - Related to the poverty conjecture (Forbus et al.)
- An architecture must use the same perception system in all tasks
- Some tasks are difficult to abstractly characterize

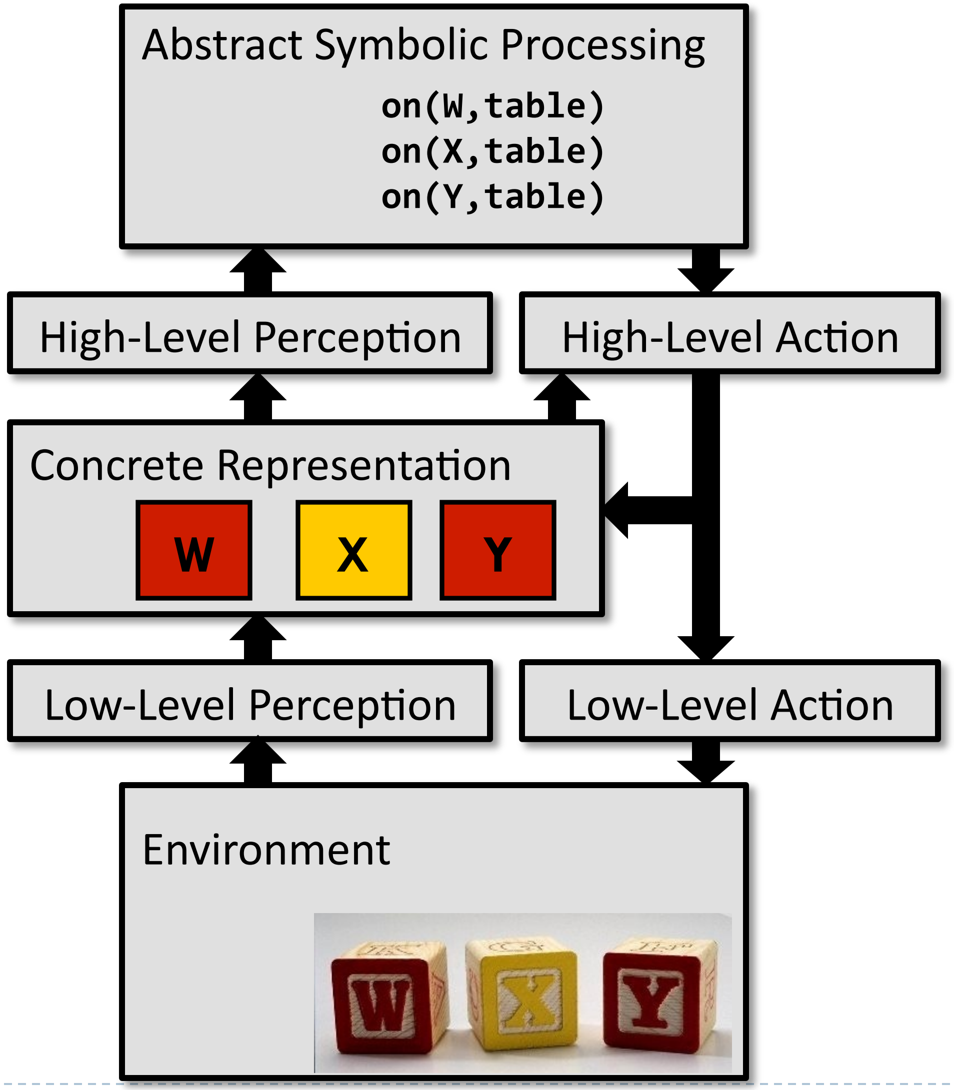


- *Perceptual Abstraction Problem:*
  - How can an agent construct appropriate abstract perceptual properties such that actions can be chosen?

Architecture

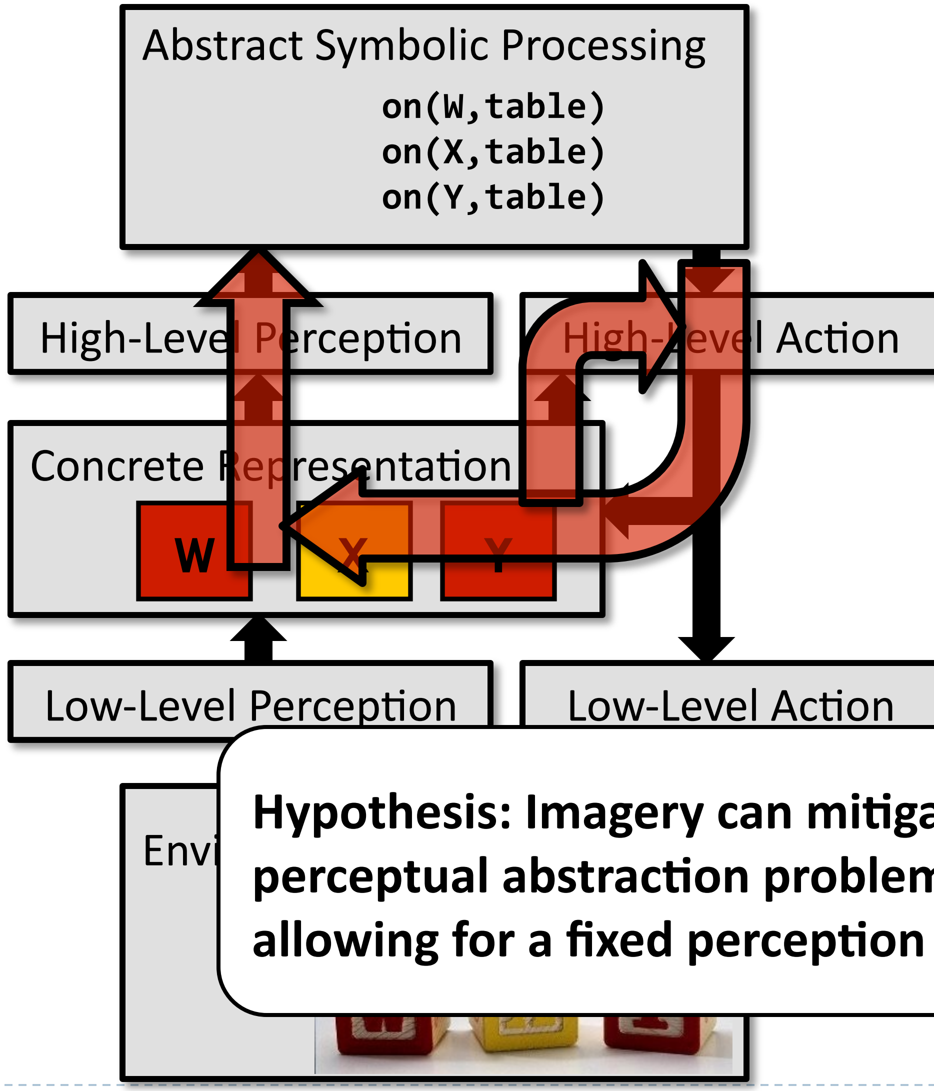


Imagery Architecture



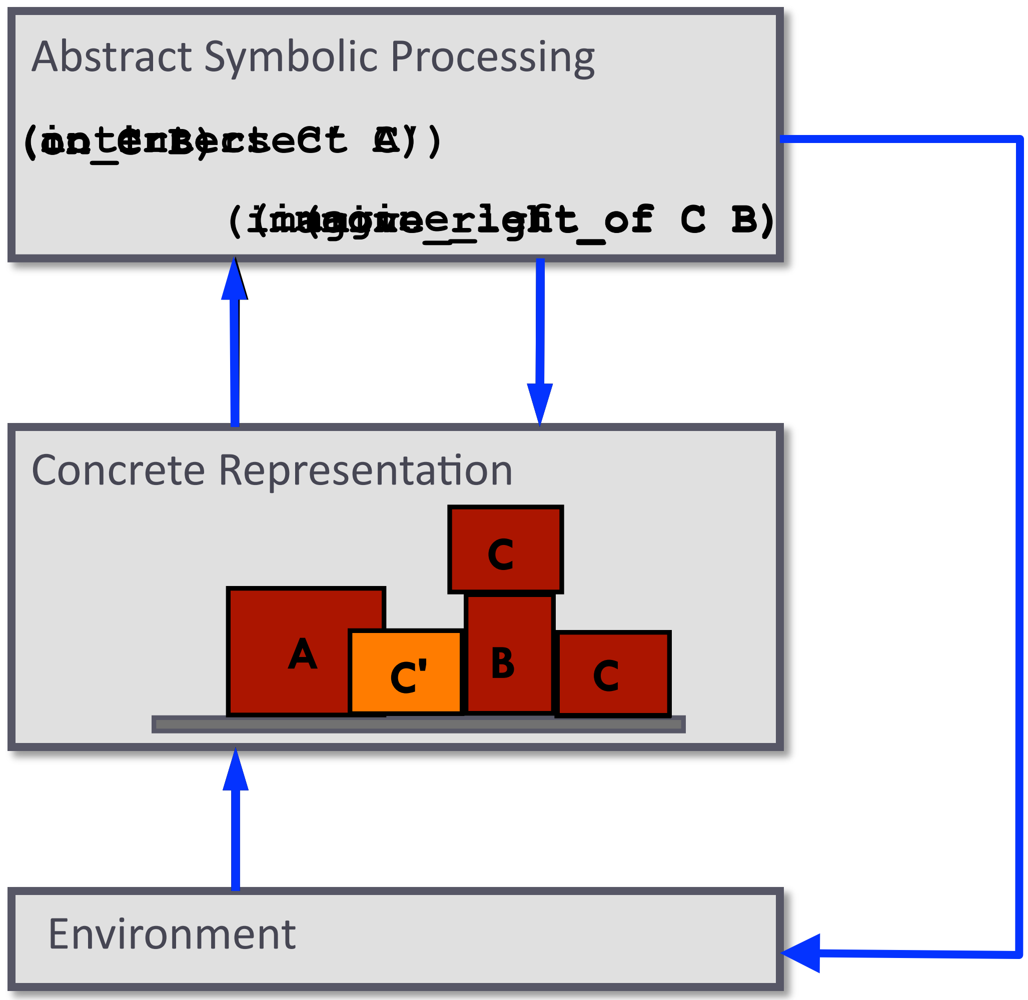
Imagery Architecture

Imagery

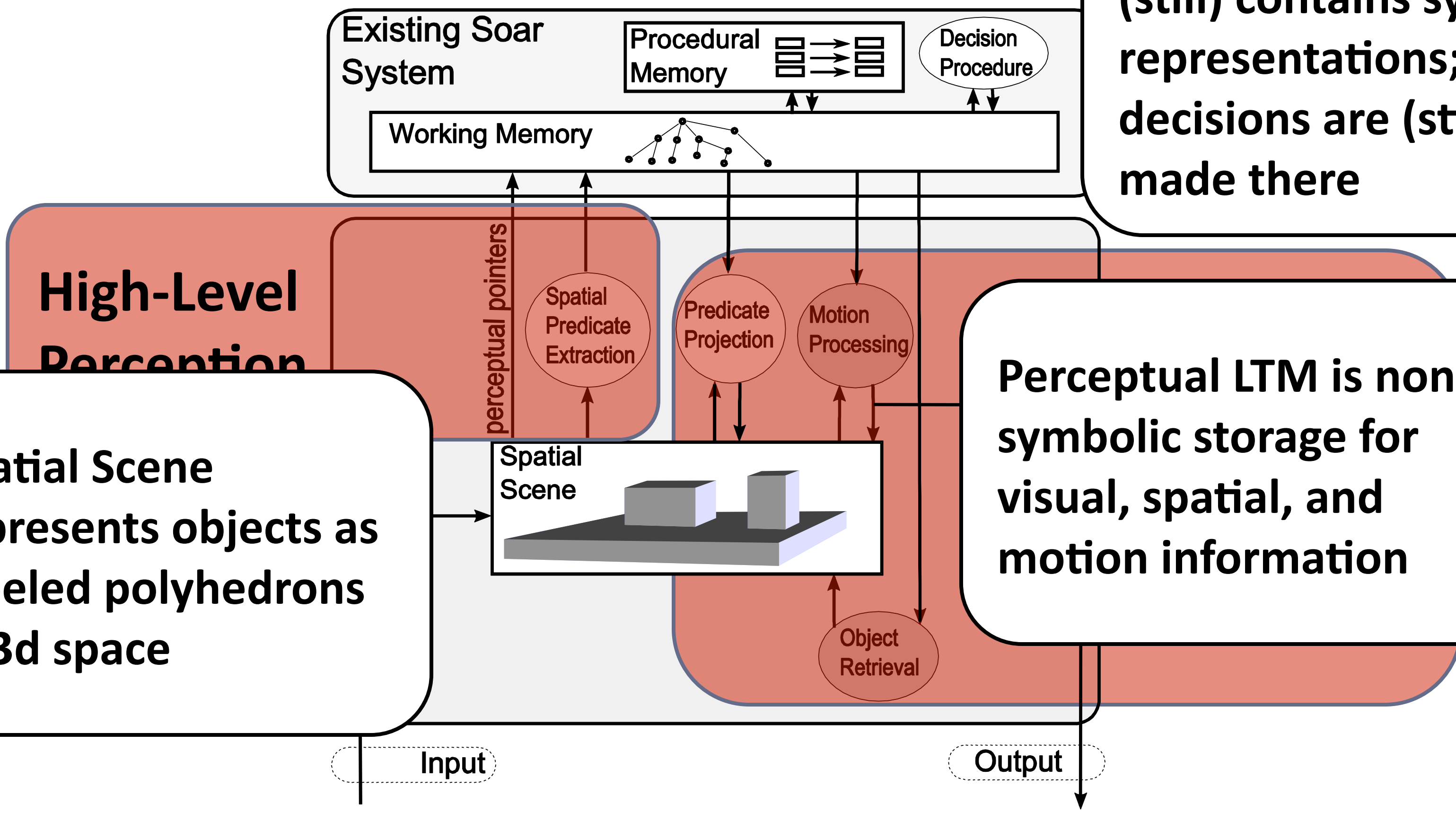


Hypothesis: Imagery can mitigate the perceptual abstraction problem while allowing for a fixed perception system

Imagery Example



Imagery in Soar



Working Memory (still) contains symbolic representations; decisions are (still) made there

Spatial Scene represents objects as labeled polyhedrons in 3d space

Perceptual LTM is non-symbolic storage for visual, spatial, and motion information

Imagery and Perceptual Abstraction

- Imagery allows an agent to represent information at multiple levels of abstraction
- *Predictions* can be made at a concrete level, *decisions* at an abstract level

- Hypothesis:
  - Imagery can increase the power of a high-level perception system to infer abstract states that are more useful across more problems

Motivating Example

- Frogger II

Perceptual Abstraction in Frogger II

Concrete Representation

Abstract Symbolic Representation

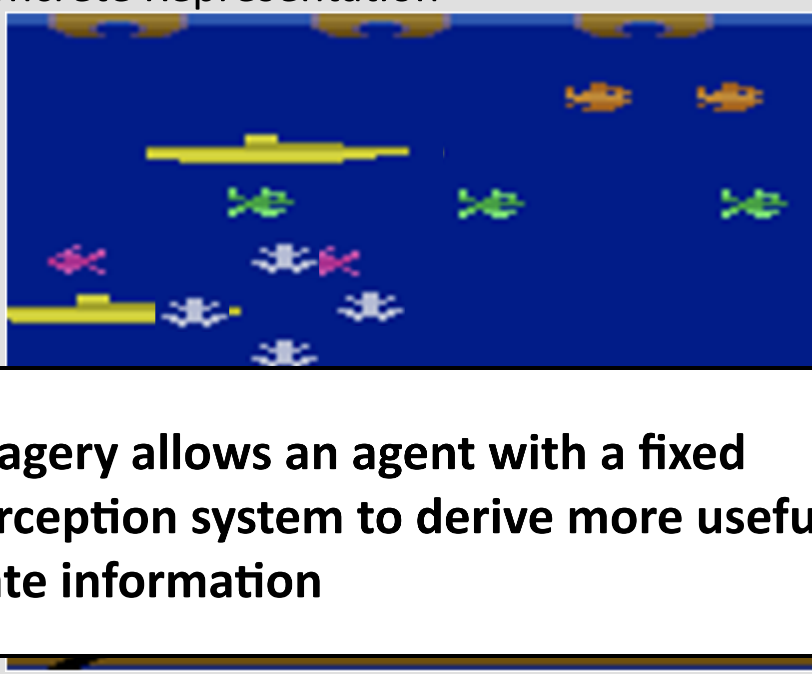
```
collision(false)
nearObstacle(left)
position(row5,middle)
```

`move(up)`

Imagery in Frogger II

- Imagery can predict action consequences
- High-level perception can be applied to imagined states

Concrete Representation



Imagery allows an agent with a fixed perception system to derive more useful state information

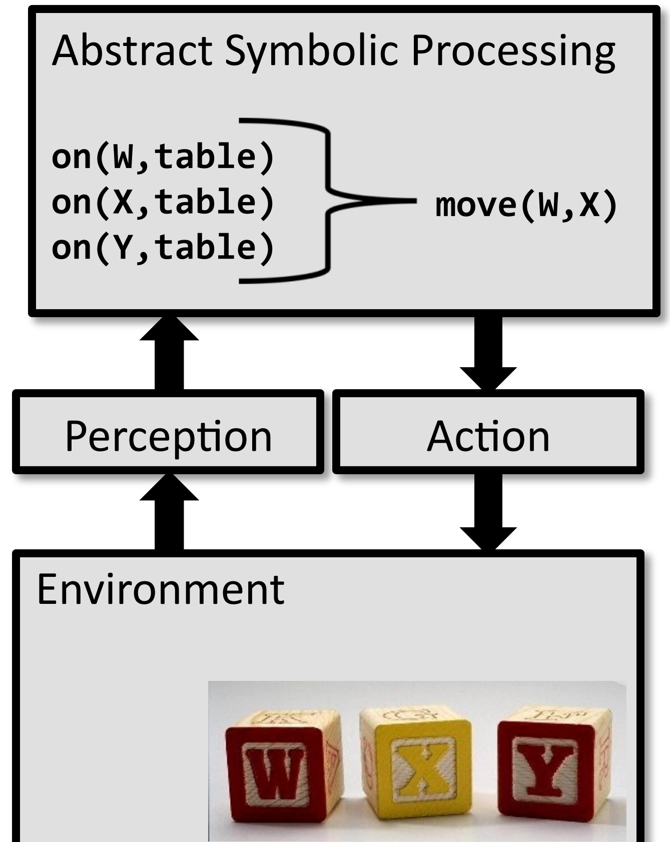
Abstract Symbolic Representation

```
right:  collision(false)
        nearObstacle(above)
        position(row5,middle)
up:     collision(true)
        nearObstacle(above,right)
        position(row6,middle)
left:   collision(true)
        nearObstacle(left)
        position(row5,left)
down:   collision(false)
        nearObstacle(none)
        position(row4,middle)
```

move(right)

Reinforcement Learning, Abstraction, and Imagery

- Imagery evaluation needs to be less ad-hoc
- RL can be used
  - Less human programming
  - Less arbitrary judgment of representation quality
- Perceptual abstraction is a means to a more compact learning problem
- The perceptual abstraction problem, manifested in RL:
  - How can an agent's perception system induce a compact representation that preserves the underlying problem?



Imagery in Frogger II

- Imagery can predict action consequences
- High-level perception can be applied to imagined states

Concrete Representation

Abstract Symbolic Representation

```
right:  collision(false)
        nearObstacle(above)
        position(row5,middle)
up:     collision(true)
        nearObstacle(above,right)
        position(row6,middle)
left:   collision(true)
        nearObstacle(left)
        position(row5,left)
down:   collision(false)
        nearObstacle(none)
        position(row4,middle)
```

Imagery for Soar-RL

```
sp {imagery-rl-production
  (state <s>   { [ right:   collision(false)
                            nearObstacle(above)
                            position(row5,middle)
                   up:      collision(true)
                            ...ve.right)
                            position(row6,...
                   left...  ...llision(tru...)
                            ...T)
                            position(row5,le...)
                   down...  ...llision(fal...)
                            ...e)
                            position(row4,midd... ] } )
  (<s> ^operator <o> +)
  (<o> ^name move
       ^direction right)
-->
  (<s> ^operator <o> = 0)
}
```

Theoretical Aside

- To meet theoretical assumptions and guarantee convergence, table-based RL in Soar requires the same sets of RL rules always fire together (match against a common state)
- What if individual rules match different aspects of state?
  - Rules might fire in multiple states with different competing rules
- Convergence of Q-Learning is guaranteed when:
  - Only one RL rule per operator is matched
  - Immediate reward for the operator is always predictable based on the RL rule
  - RL rules that will match in the next state are predictable based on the RL rule for the current operator
- "Predictable": no relevant state information was abstracted away
- For details, see the paper

Algorithm

- Reinforcement Learning with Abstraction and Imagery (ReLAI) algorithm was developed
- Imagery is used to simulate next state, perceptual abstraction is applied, Q values are learned based on predicted next abstract state

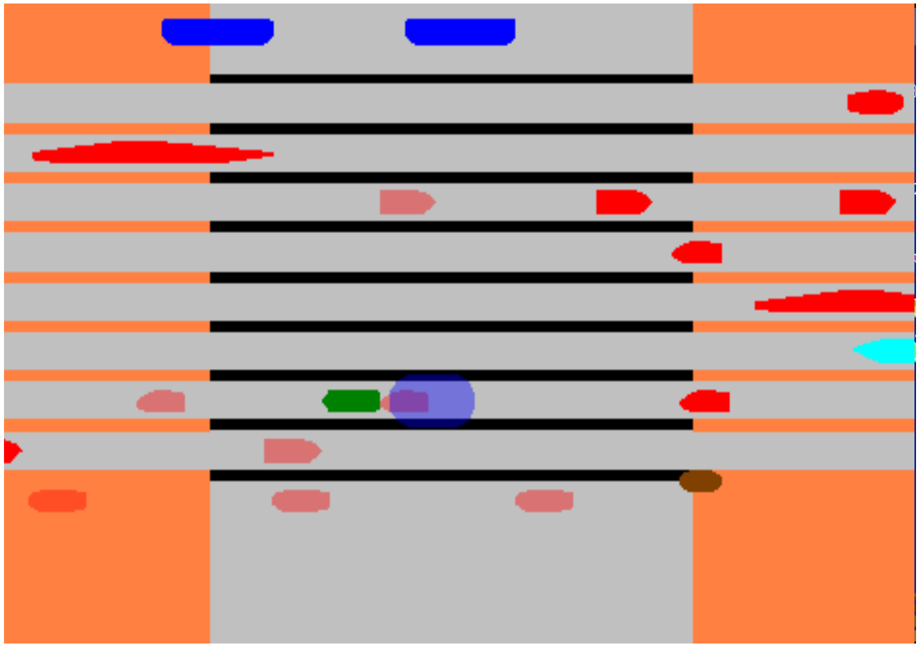
- Convergence conditions can be translated to prediction case
- Result: next abstract state can depend on aspects of the concrete state not captured by the abstract state
  - This differs from standard state abstraction, where abstract state must completely summarize concrete state

Experiments

- Assumptions are still very hard to meet, but robustness to abstractions where next state depends on details not abstractly captured gives empirical benefits

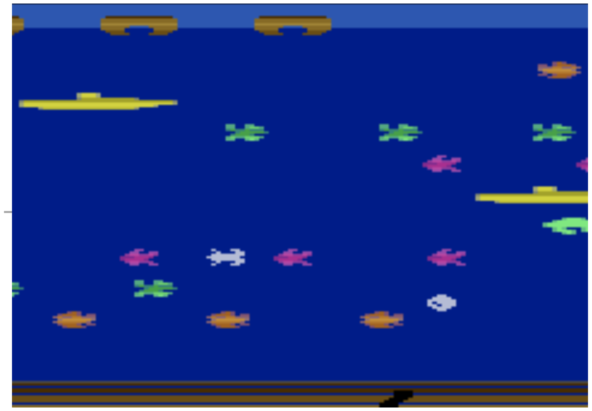
- Experiments were run to compare ReLAI to standard Q-learning with the same abstraction
- Interface between SVS and an Atari emulator was built
- Three games were tested

Frogger II Perception and Imagery

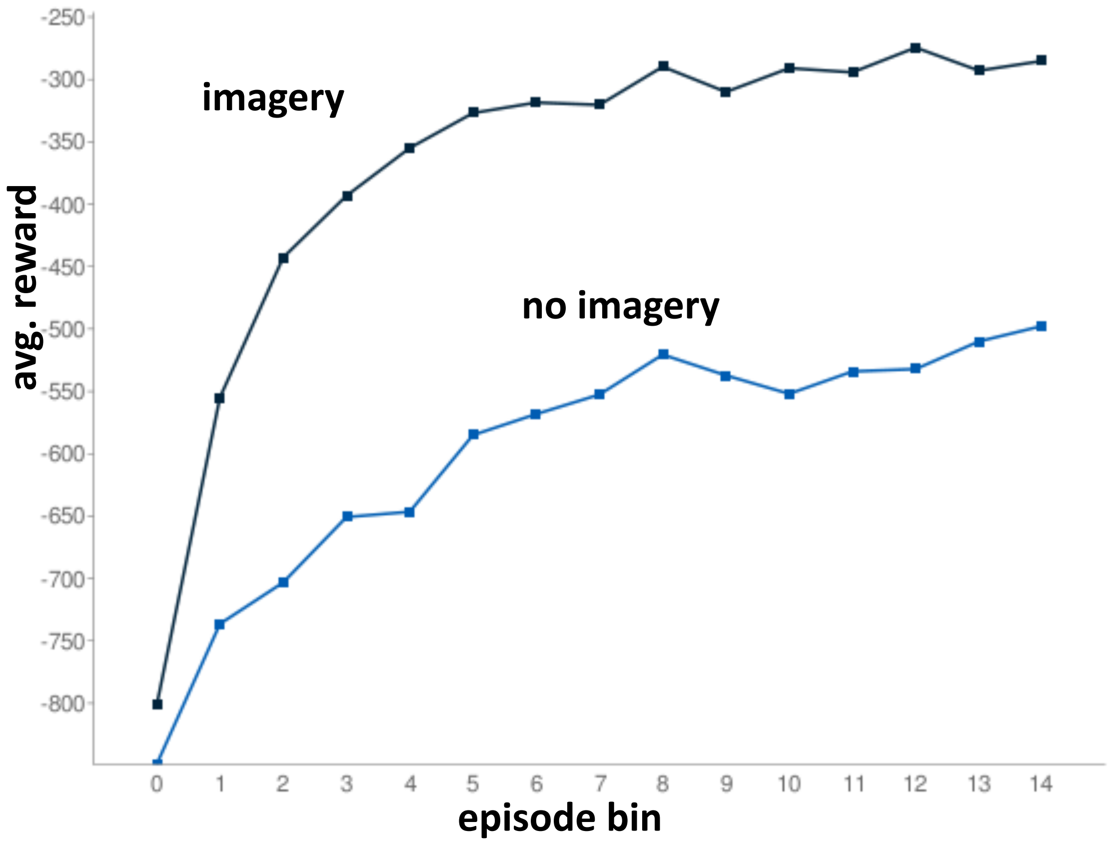



PerceptionCore
number:i45
raft:i34
raft:i35
raft:i36
fish3:i49
fish3:i37
boat:i41
fish1:i43
fish1:i44
fish2:i22
fish2:i24
fish2:i26
boat:i42
greenturtle:i53
fish2:i28
fish2:i30
fish2:i32
fish1:i50
fish1:i17
fish1:i19
fish3:i39
fish3:i40
frog:i47
bubble:i52
meter:i33
Soar Debugger in Java - remote svs_agent
File Edit Print Commands Debug Level Demos Layout Agents Kernel Help
Expand
Filters
Step Run Run 1 -p Stop Matches Print <s> Print <ts> Clear Watch 1 Watch 3 Watch 5 Init-soar Source cd Excise all Towers of Hanoi Water Jug RL Remote

Frogger II Results

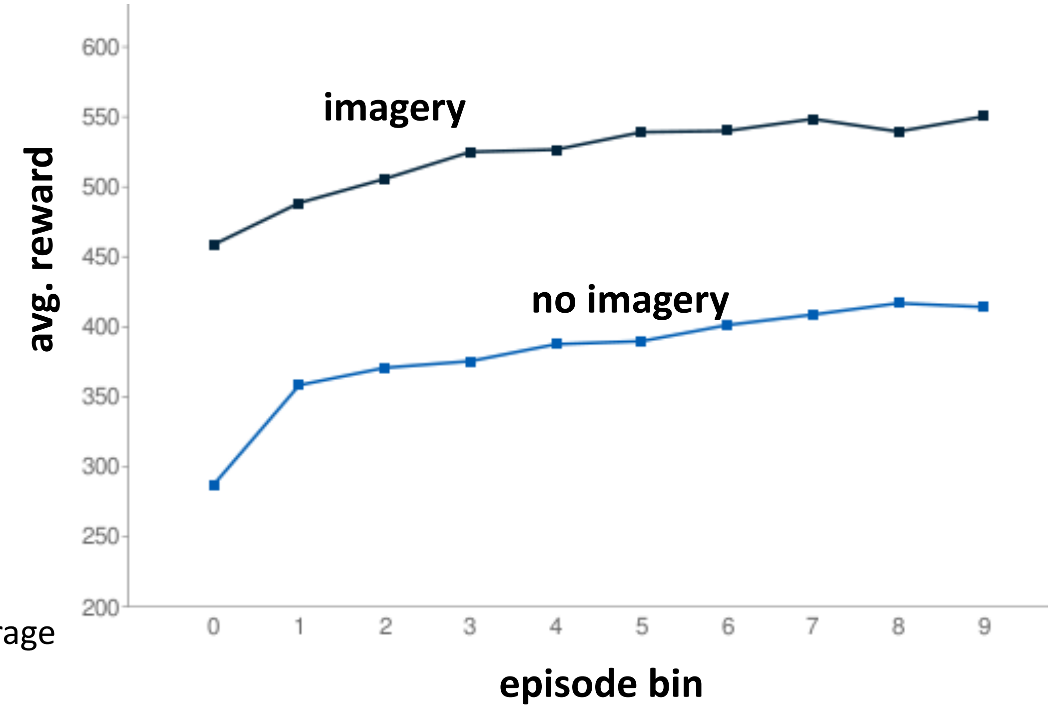


- State information:
  - Does frog collide with an obstacle?
  - What row is it in?
  - What horizontal region?
  - Is there an obstacle immediately to the left/right/above/below?
  - What was the previous action?
- Rewards:
  - +10 /-10 for moving up/down a row
  - 1000/-1000 for reaching top/dying
  - -1 at each time step
- 30 trials of 6,000 games were run
  - binned to groups of 400
  - each point is 12,000 games
- Final performance (without exploration)
  - Imagery agent won 70%
  - No-imagery won 45%

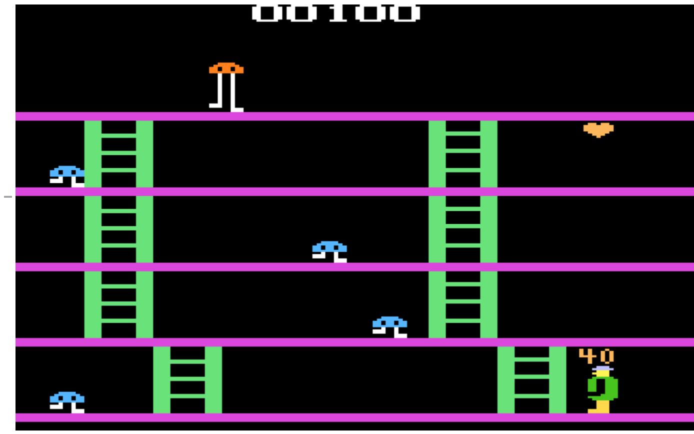


Space Invaders Results

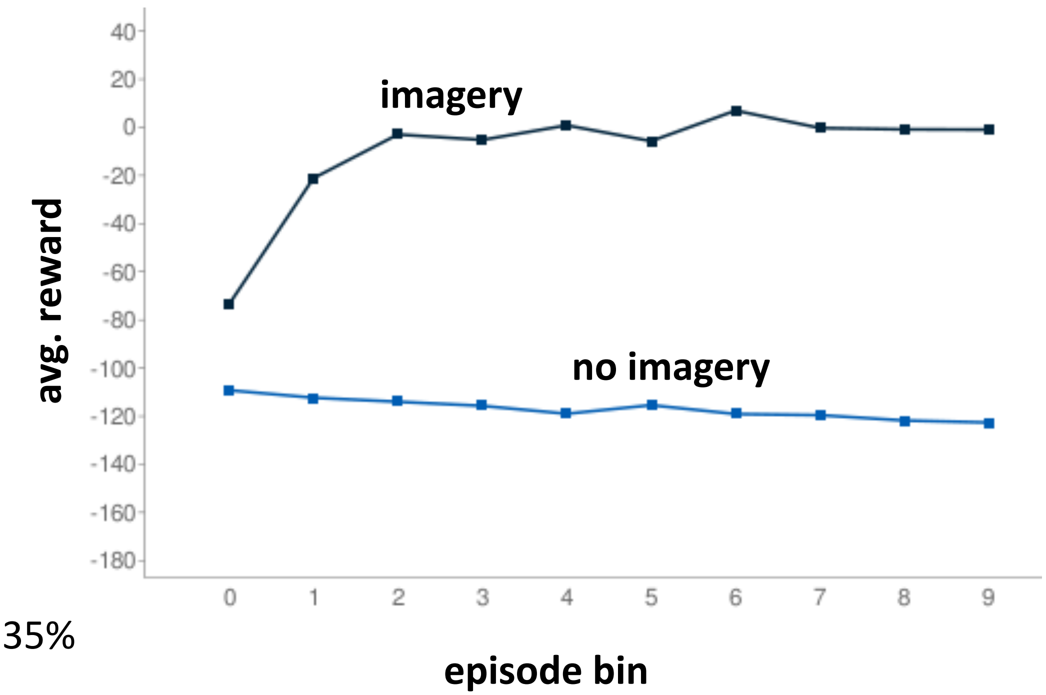
- State information:
  - discretized horizontal position [1-15]
  - is there an unblocked clear shot?
  - is there an unshielded bomb?
  - is there a missile (shot by the agent) in the air?
  - if so, is it aligned with an alien?
  - is there a falling bomb immediately to the left/right?
  - does the ship intersect a bomb?
- Rewards:
  - +50/-50 for killing an alien/dying
- 12 trials of 5,000 games were run
  - binned to groups of 500
  - each point is 6,000 games
- Final performance (without exploration)
  - Imagery agent killed 13-14 aliens on average
  - No-imagery killed 9-10 aliens on average



Fast Eddie Results



- State information:
  - Is Eddie near an obstacle?
  - Does Eddie intersect an obstacle?
  - Did the last action reduce the distance to the closest heart?
  - Was a heart collected in the last action?
- Rewards:
  - 50/-100 for collecting a heart/dying
  - -1 at each time step
- 24 trials of 1,000 games were run
  - binned to groups of 100
  - each point is 2,400 games
- Final performance (without exploration)
  - Imagery agent won (collect 9 hearts) 35%
  - No-imagery agent never won



Conclusion

- Nuggets
  - Imagery can improve the ability for an agent with a fixed perception system to make relevant distinctions between states
    - Evidence that imagery mitigates the perceptual abstraction problem
  - Only minimal changes to SVS were needed
  - Theoretical work should aid other Soar-RL applications
- Coal
  - Theoretical assumptions are hard to match in real games
  - Using imagery for every possible action is slow
  - A few game-specific hacks for perception and imagery were necessary
  - SVS still needs to be reimplemented and released

References

- Wintermute, S. (2010) "Using Imagery to Simplify Perceptual Abstraction in Reinforcement Learning Agents," to appear in *Proceedings of AAAI-10*.
- Wintermute, S. (2009) "An Overview of Spatial Processing in Soar/SVS," Technical Report CCA-TR-2009-01, University of Michigan Center for Cognitive Architecture.