Episodic Memory in Soar

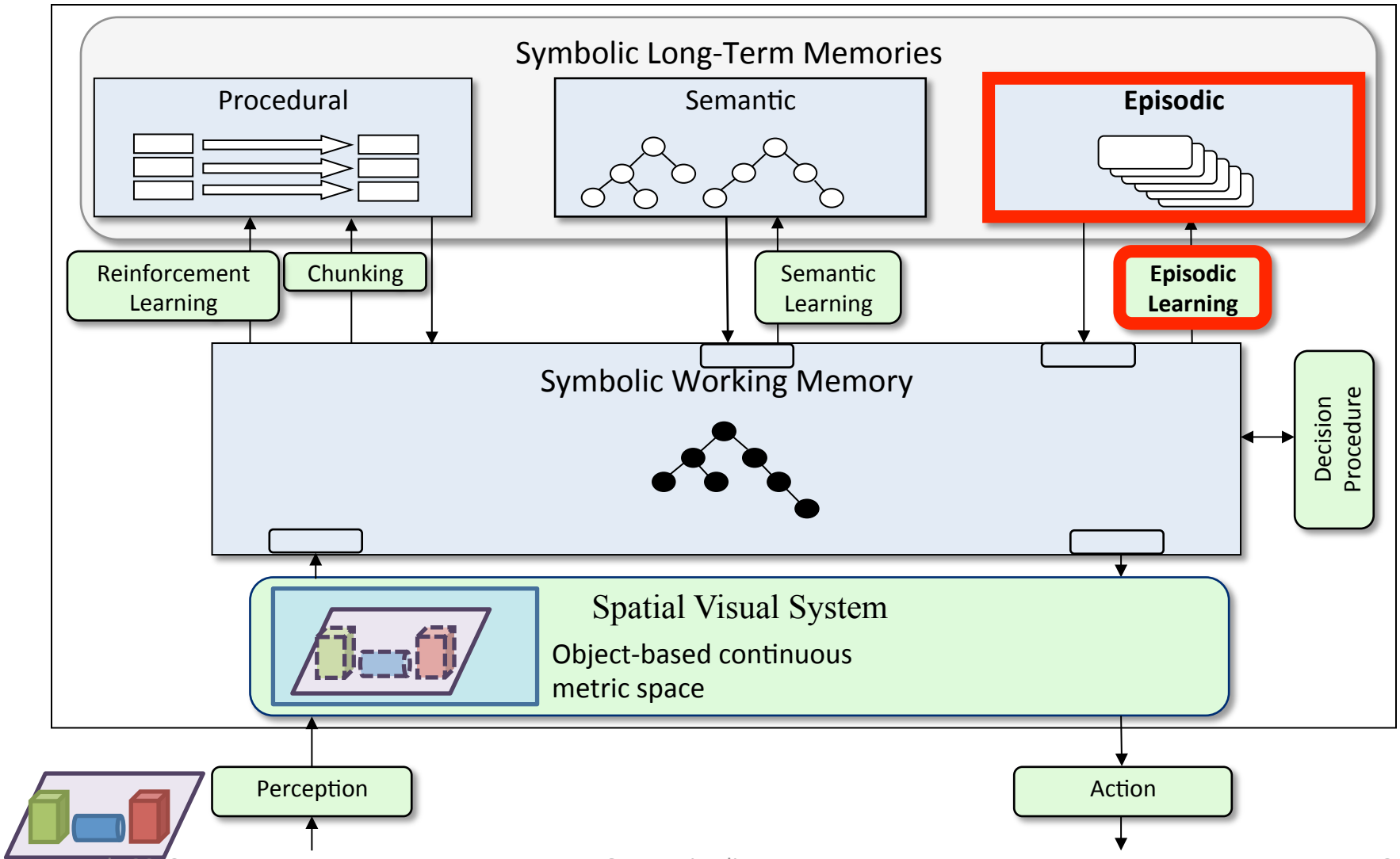
Nate Derbinsky

Disney Research, Boston

Agenda

- Episodic memory as a learning mechanism
- Basic usage
  - Example: TankSoar
- Scaling real-time performance
  - Multi-domain: linguistic, planning, robotics, games

Soar 9


Symbolic Long-Term Memories
Procedural
Semantic
Episodic
Reinforcement Learning
Chunking
Semantic Learning
Episodic Learning
Symbolic Working Memory
Decision Procedure
Spatial Visual System
Object-based continuous metric space
Perception
Action

Episodic Memory

Long-term, contextualized store of specific events (Tulving, 1983)

Action Modeling

Virtual Sensing

Episodic Memory in Soar

A weak learning mechanism

- Automatically captures, stores, and temporally indexes agent state
- Provides content-addressable agent interface to autobiographical prior experience
- Supports a general set of cognitive capabilities

Episodic Memory

Integration

Representation

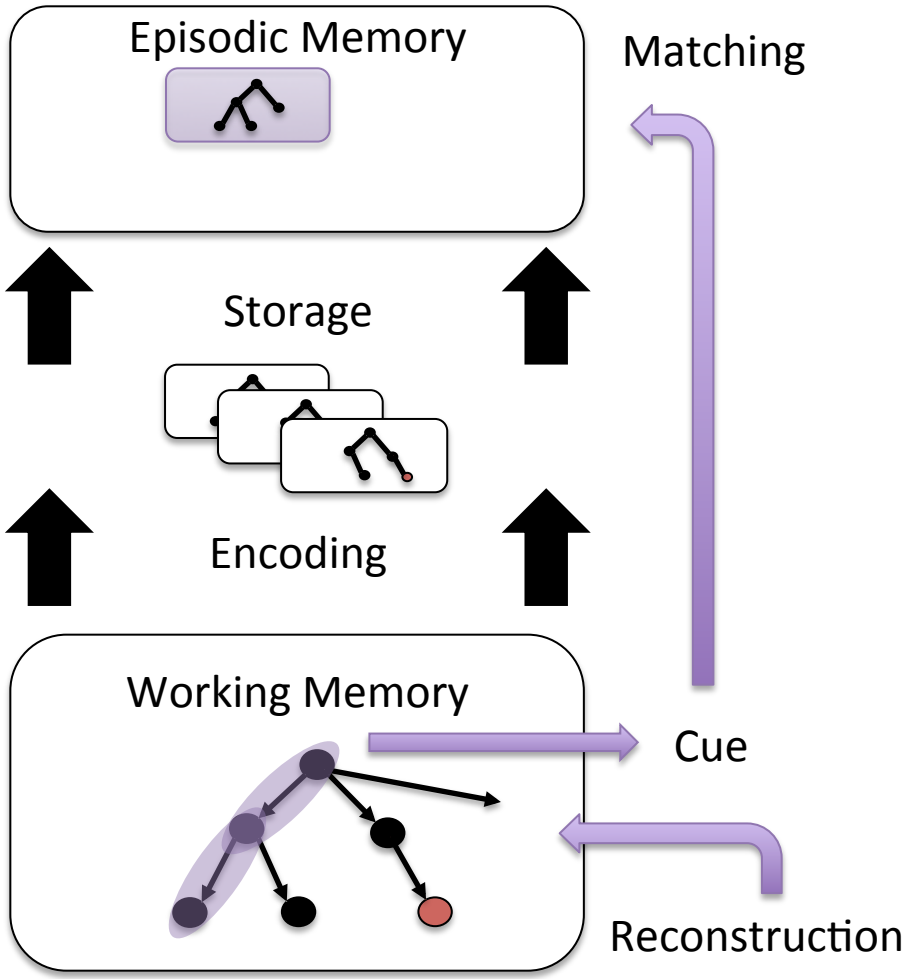
- Episode: connected di-graph
- Store: temporal sequence

Encoding/Storage

- Automatic
- No dynamics (e.g. forgetting, blending, …)

Retrieval

- Cue: acyclic graph
- Semantics: desired features in context
- Find the most recent episode that shares the most leaf nodes in common with the cue



Basic Usage

- Working-memory structure
- Episodic-memory representation
- Storing knowledge
- Retrieving knowledge

Working-Memory Structure

Soar creates an `epmem` structure on each state

Each `epmem` structure has specialized substructure

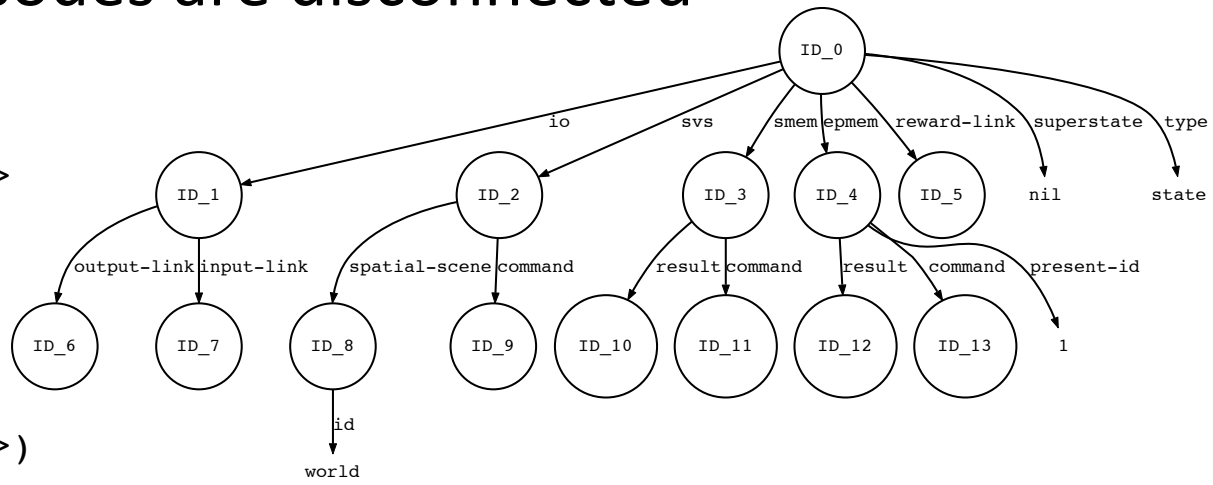
- `command`: agent-initiated actions
- `result`: architectural feedback
- `present-id`: current episode number (more later)

Episodic-Memory Representation

Similar to working memory: symbolic triples

- Structures within an episode are connected; separate episodes are disconnected

```
(<id0> ^epmem <id4>
       ^io <id1>
       ^reward-link <id5>
       ^smem <id3>
       ^superstate nil
       ^svs <id2>
       ^type state)
(<id1> ^input-link <id7>
       ^output-link <id6>)
(<id2> ^command <id9>
       ^spatial-scene <id8>)
(<id3> ^command <id11> ^result <id10>)
(<id4> ^command <id13> ^present-id 1 ^result <id12>)
(<id8> ^id world)
```



Storing Knowledge

- **What**. top state of working memory

- **Why**. task-independent trigger
  - `epmem --set trigger << dc output >>`
    - `dc`: decision cycle (default)
    - `output`: new augmentation of output-link

- **When**. at the end of a phase
  - `epmem --set phase << output selection >>`
    - `output` is default
    - `selection` may be useful for in-the-head agents

Retrieving Knowledge

Cue-Based

Find the episode that best matches a cue and add it to working memory

Temporal Progression

Replace the currently retrieved episode with the next/previously encoded episode

Non-Cue-Based (not covered)

Add an episode to working memory from episode #

Common Constraints (motivated by performance/reactivity):

- Only one per state per decision
- Processed during `phase` (slide 10)
- Only re-processed if WM changes to commands

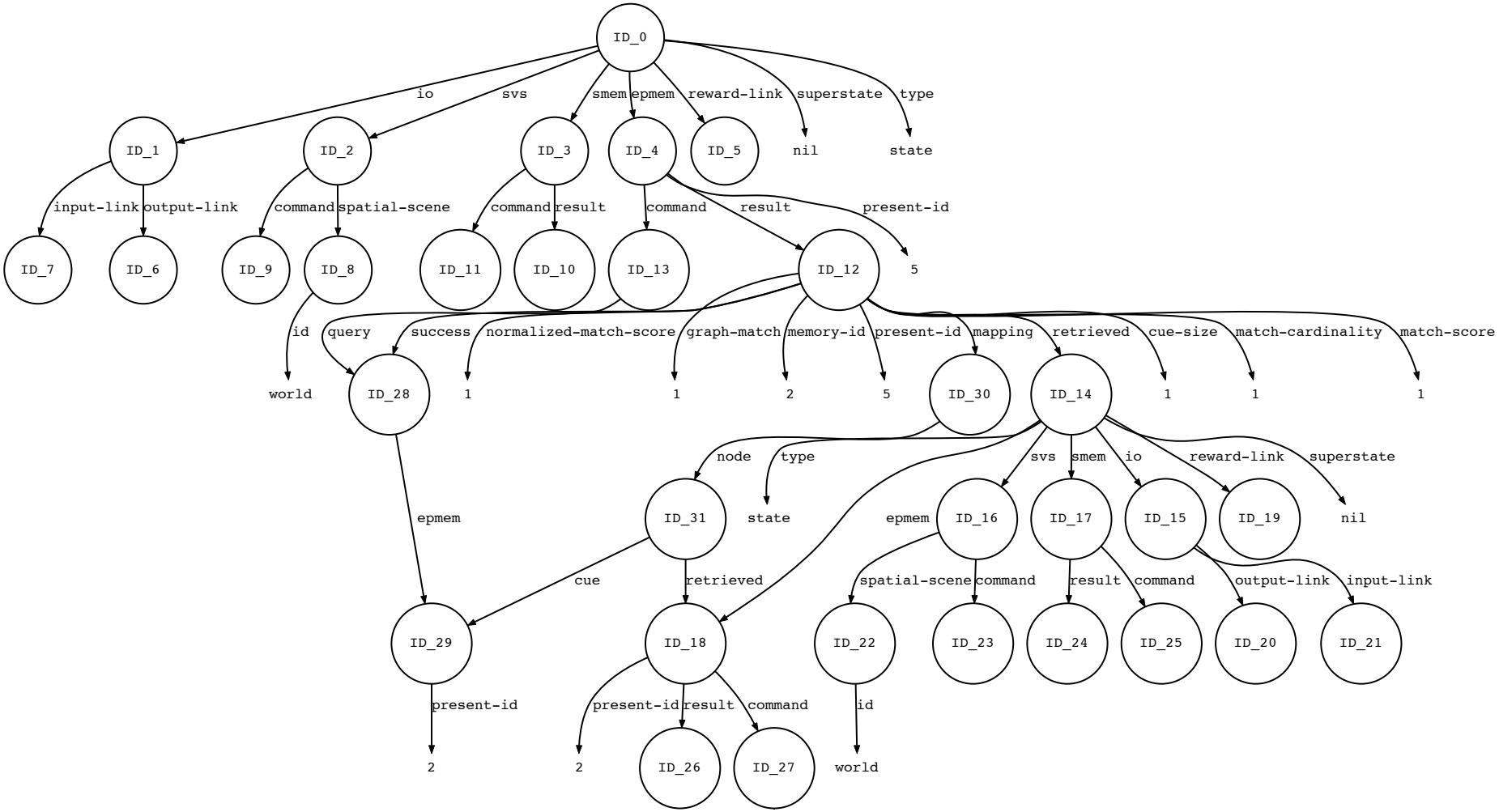
Cue-Based Retrieval: Syntax

```
(<epmem> ^command <cmd>)
(<cmd> ^query <q>
       ^neg-query <nq>)
```

- The `neg-query` is optional
- Cues must be acyclic
- The <q> and <nq> identifiers form the roots of episode sub-graph cues
  - `query` represents desired structures
  - `neg-query` represents undesired structures

Cue-Based Retrieval

Example



Cue-Based Retrieval

Optional Modifiers

```
(<cmd> ^before time-id)
(<cmd> ^after time-id)
(<cmd> ^prohibit time-id1 time-id2 …)
```

Hard constraints on the episodes that can be retrieved.

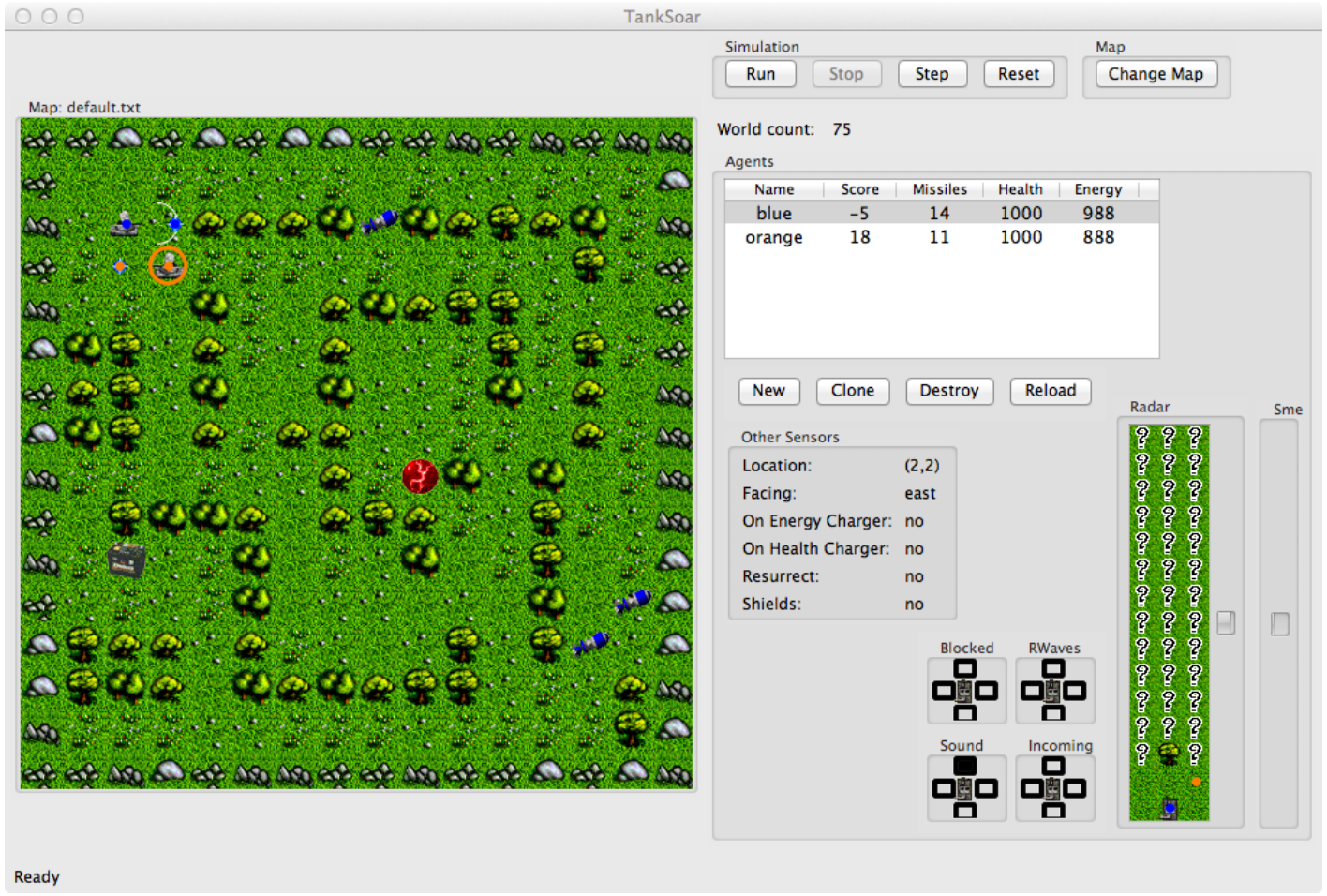
Temporal Progression

```
(<cmd> ^next <new-id>)
(<cmd> ^previous <new-id>)
```

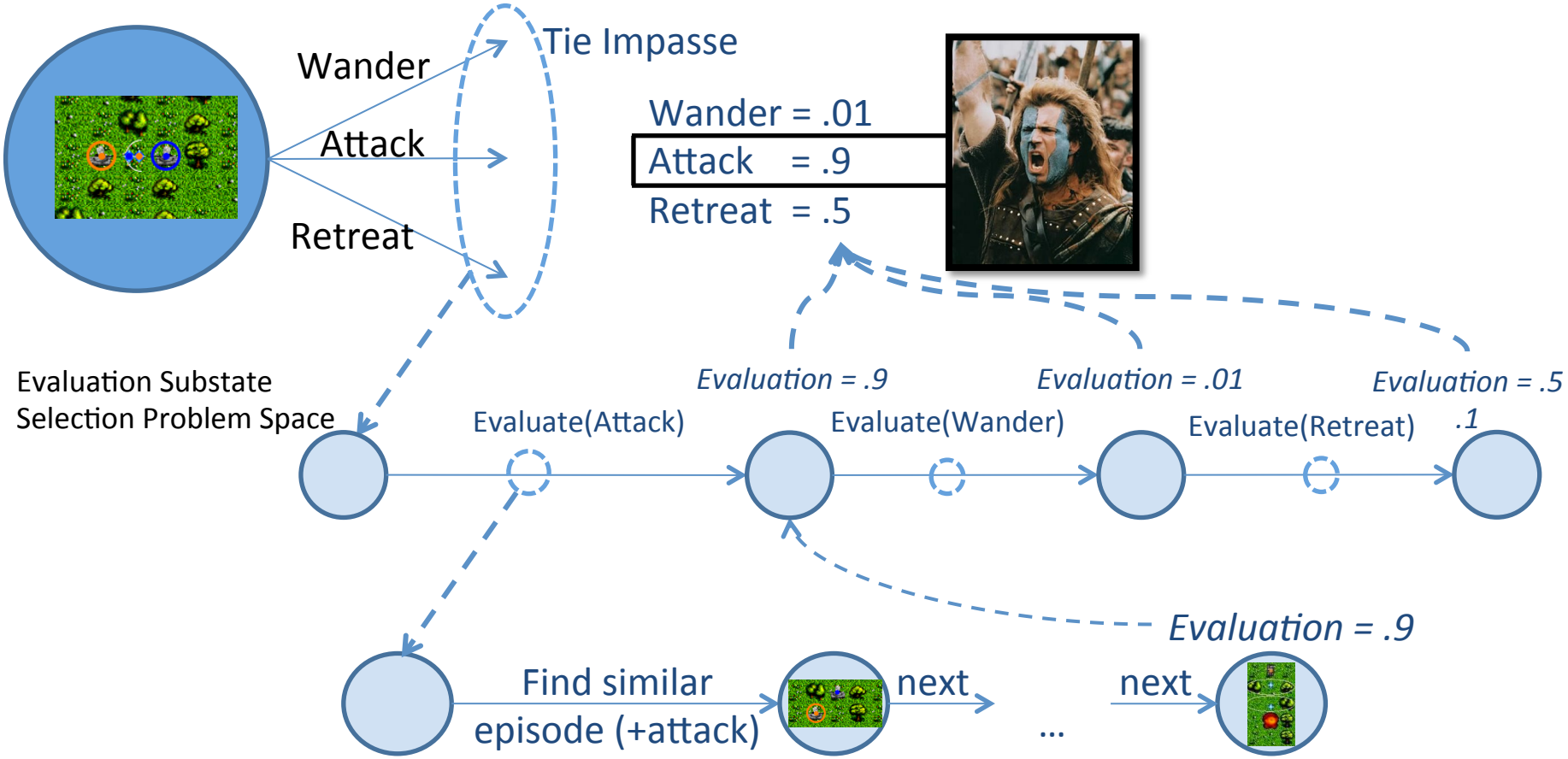
Retrieves the next/previous episode, temporally, with respect to the last that was retrieved

Example Task

TankSoar

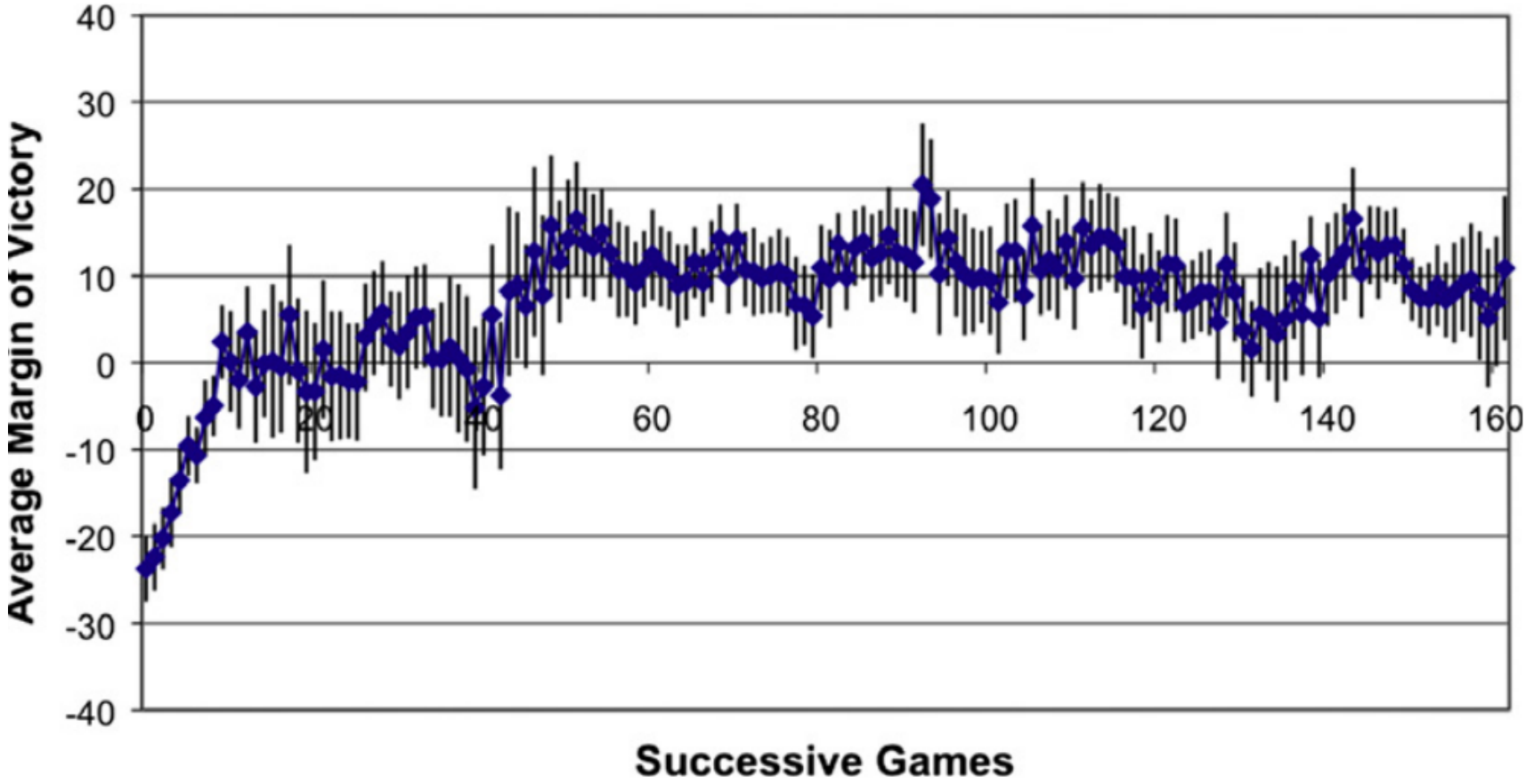


Using Episodic Knowledge for Subgoal Selection



Learning from Success & Failure

A. M. Nuxoll & J. E. Laird, 2012



Episodic Memory

Computational Challenges for Scaling RT Agents

Arbitrary, dynamic state

Scaling potential, agent…

- state (1000s nodes/edges)
- life (10^6-10^9 episodes ~ days)

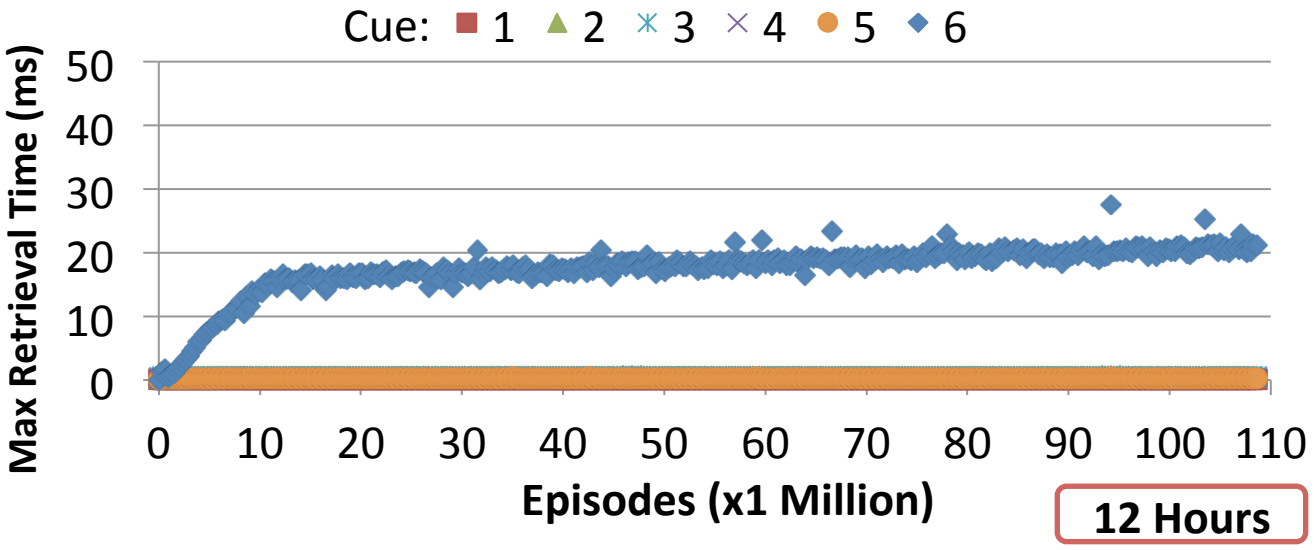
Cue-matching optimality

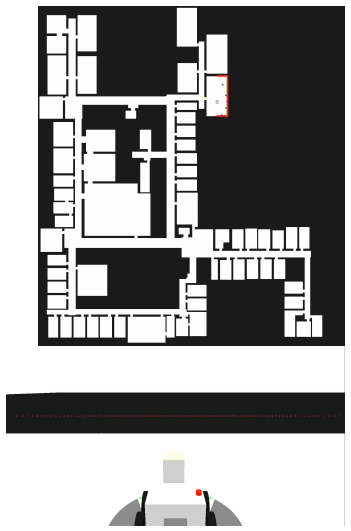
- Constrained subgraph isomorphism (NP-complete)
- Search: *O*(# episodes)

Multi-Domain Scaling Evaluation

Experimental Setup

- 49 domains: WSD, planning, robotics, games
- 10^5-10^8 episodes ~ days of real time, >100 cues





Thank You :)

Questions?