A Multi-Domain Evaluation of Scaling in a General Episodic Memory

Nate Derbinsky and Justin Li and John E. Laird
University of Michigan
2260 Hayward St.
Ann Arbor, MI 48109-2121
{nlderbin, justinnh, laird}@umich.edu

Abstract

Episodic memory endows agents with numerous general cognitive capabilities, such as action modeling and virtual sensing. However, for long-lived agents, there are numerous unexplored computational challenges in supporting useful episodic-memory functions while maintaining real-time reactivity. In this paper, we review the implementation of episodic memory in Soar and present an expansive evaluation of that system. We demonstrate useful applications of episodic memory across a variety of domains, including games, mobile robotics, planning, and linguistics. In these domains, we characterize properties of environments, tasks, and episodic cues that affect performance, and evaluate the ability of Soar's episodic memory to support hours to days of real-time operation.

Introduction

Prior work has shown that agents with episodic memory, a task-independent autobiographical store of experience (Tulving 1983), are more capable in problem solving, both individually (e.g. Nuxoll and Laird 2012) and collaboratively (e.g. Deutsch et al. 2008; Macedo and Cardoso 2004); are better able to account for human psychological phenomena, such as memory blending (Brom, Burkert, and Kadlec 2010) and emotional appraisal (Gomes, Martinho, and Paiva 2011); and are more believable as virtual characters (Gomes, Martinho, and Paiva 2011) and long-term companions (Lim et al. 2011).

However, relatively little research has examined the computational challenges associated with maintaining effective and efficient access to episodic experience as autonomous agents persist for long periods of time. Most approaches to storing and retrieving episodic knowledge are task-specific (e.g. Macedo and Cardoso 2004) and/or apply to temporally limited problems (e.g. Kuppuswamy, Cho, and Kim 2006). However, for autonomous agents that persist for long periods of time, we need to understand how episodic memory scales across a variety of tasks.

Research into the task-independent episodic memory that is part of Soar (Laird 2012) has focused on issues related to its architectural integration (Nuxoll and Laird 2012); demonstrations of the functional benefits for agents (Nuxoll and Laird 2012; Xu and Laird 2010); as well as real-time performance analysis, as evaluated for hours of operation in an action game (Derbinsky and Laird 2009) and simulated mobile robotics (Laird, Derbinsky, and Voigt 2011). This paper builds upon and extends this body of work with a focus on scaling. We subsume and expand prior evaluations along all pertinent dimensions: (i) we increase the number and diversity of problem domains, spanning video games, simulated mobile robotics, generalized planning, and linguistics; (ii) we extend agent runtime duration to days; and (iii) we exemplify new cognitive capabilities that agents can apply across domains by virtue of having an episode memory. Our results show that this mechanism supports useful memory operation across a variety of domains, while maintaining real-time behavior for agents that persist for days, accruing millions of episodes. We also analyze how general properties of environments, tasks, and retrieval cues affect mechanism efficiency, including cases when it does not scale.

Related Work

There have been limited evaluations of scaling of other episodic memories. For instance, Ubibot (Kuppuswamy, Cho, and Kim 2006) was tasked in a single 2D simulation that lasted 7 min. with fewer than 50 episodic memories.

Tecuci and Porter (2007; 2009) applied EM, the generic memory module for events, to planning, plan recognition, classification, and goal-schema recognition tasks in several domains. They presented evidence that in practice, their retrieval mechanism inspects far fewer events than are stored. However, their memory API is intended solely for events and must be specialized for each new domain. Also, they have not published retrieval-timing data, and thus it is unclear whether EM is applicable to real-time agents.

The case-based reasoning (CBR; Kolodner 1993) community studies issues closely related to episodic-memory research, including the use of past experience with present reasoning and learning, as well as scaling to large case libraries (Smyth and Cunningham 1996). However, most CBR systems optimize performance for a specific task, and specialize case format, storage, retrieval, and adaptation. By contrast, this paper focuses on a task-independent mechanism applied in a variety of domains.

Episodic Memory in Soar

Soar is a cognitive architecture that has been used for developing intelligent agents and modeling human cognition (Laird 2012). We first describe pertinent architectural mechanisms and processes, and then convey how episodic memory is efficiently integrated.

The Soar Cognitive Architecture

In Soar, the agent's current state is represented in a symbolic, short-term *working memory*. It is here that perception, goals, retrievals from long-term memory (including episodic), external action directives, and structures from intermediate reasoning are jointly represented as a connected, directed graph – an encoding that is sufficiently expressive for a wide variety of tasks. *Procedural* long-term memory holds the agent's knowledge of when and how to perform actions, both internal, such as querying long-term memories, and external, such as initiating robotic actuators or virtual game character actions. Procedural knowledge is represented as if-then rules: the conditions test patterns in working memory and the actions add and/or remove working-memory structures. Agents retrieve knowledge from either of the declarative long-term memories (*semantic* and *episodic*) by constructing a cue in working memory; the intended long-term memory interprets the cue, searches its store for the "best" match, and reconstructs the associated knowledge in working memory. Soar has other memories and processing mechanisms; however, they are not pertinent to this paper and are not discussed further.

Agent reasoning in Soar consists of a sequence of *decision cycles* to select *operators*, which perform actions in service of the agent's goal(s). The time to execute the decision cycle, which, in practice, primarily depends on the speed with which the architecture can retrieve knowledge from long-term memory, determines agent reactivity. Thus, the degree to which long-term memory operations can scale to large amounts of knowledge over long lifetimes directly affects the ability of the agent to act in real time.

Episodic Memory

This section summarizes the functional commitments (Nuxoll and Laird 2012) of Soar's episodic memory, as well as its implementation (Derbinsky and Laird 2009).

Functional Integration

Episodic memory comprises three phases: (1) *encoding* agent state; (2) *storing* this information as episodic knowledge; and (3) supporting *retrieval* at a later time.

During each decision cycle, Soar's episodic memory automatically encodes the contents of working memory as a connected di-graph. This information, as well as the time of encoding, is stored in episodic memory, where it remains without modification for the lifetime of the agent.

To retrieve an episode, rules fire to construct an episodic cue: a directed, connected, acyclic graph that specifies task-relevant relations and features. The cue-matching process identifies the "best" matching episode, defined as the most recent episode that has the greatest number of structures in common with cue leaf nodes. Episodic memory then reconstructs this episode in working memory.

This formulation of cue matching commits to two algorithmic properties that affect scaling. The process returns an episode if one exists that contains at least one feature in common with a cue leaf node. The mechanism also returns the "best" episode with respect to cue structure, leaf nodes, and temporal recency. Given these commitments, in the worst case, the encoding, storage, and retrieval operations scale at least linearly with the number of state changes. The implementation takes advantage of regularities in state representations and dynamics to reduce the frequency of this worst case.

Efficient Implementation

Soar's episodic memory exploits two assumptions about agent state, both of which relate to those that have been successfully applied in the rule-matching literature. The first is *temporal contiguity*: the world changes slowly, and thus changes to agent state, from episode-to-episode, will be few relative to the overall size of state. The second is *structural regularity*: for agent knowledge to generalize, it must reuse representational structure, and thus, over time, the number of distinct structures will be much smaller than the total number of experienced structures.

Episodic encoding and storage draw directly from these assumptions. Soar represents episodic knowledge in two data structures: (1) a global structure index, termed the *Working Memory Graph* (WMG), which captures all distinct graph structures that have been encoded, and (2) a set of temporal intervals that capture when each edge of the WMG was added to/removed from working memory. These data structures comprise a dynamic-graph index.

Cue matching is a constrained form of sub-graph isomorphism: to score an episode, the mechanism must compare two rooted, directed, connected graphs (where the cue is acyclic). To avoid this potentially combinatorial comparison, Soar's implementation utilizes a two-phase matching process (Forbus, Gentner, and Law 1995): a relatively cheap *surface* match identifies, in reverse temporal order, each episode that contains *all* cue leaf nodes *independently* and submits it to a more expensive *structure* match. Structural unification is implemented as graph matching with standard heuristics (e.g. MCV). The surface matcher, however, attempts to exploit the aforementioned assumptions and indexing.

First, the WMG indexes to only those temporal intervals that refer to cue features. The endpoints of these intervals are then walked in order of decreasing recency to identify and evaluate only those episodes during which pertinent features *changed*: temporal contiguity predicts that existing structures will tend to persist, and thus examining structure *changes* should require much less effort than examining individual episodes. This process scales with the *temporal selectivity* of the cue: in the worst case it must examine all episodes sharing features with the cue, which may be all episodes in the store, but it may be a much smaller proportion. This endpoint-walking process terminates when an episode unifies structurally.

Surface match, with respect to a cue, can be formulated as computing satisfaction of a disjunctive-normal-form (DNF) Boolean statement, where variables map to cue nodes, clauses map to root-to-leaf paths, and literals map to feature existence within episodes. In order to efficiently evaluate DNF satisfaction between discrete graph changes, a discrimination network, termed the DNF Graph, is used to maintain state between episode evaluations, and selectively propagate changes in the cue, similar to a Rete network (Forgy 1982). Given a DNF Graph and a feature change, evaluating an episode scales with the *structural selectivity* of the feature: in the worst case it must examine all structures in the episode that could map to the feature in the cue, which may be the size of the episode, or any fraction thereof.

Once the cue-matching process selects an episode to retrieve, the system uses a relational interval tree (Kriegel, Pötke, and Seidl 2000) to efficiently extract all features and relations of the episode from the dynamic-graph index.

Empirical Evaluation

Our goal is to understand the degree to which Soar's episodic memory supports useful operation across a variety of domains while scaling to long agent lifetimes.

Evaluation Metrics

In order to evaluate episodic-memory scaling, we measure two classes of computational-resource usage during agent runs: execution time and storage requirements.

The time it takes for Soar to complete a decision cycle dictates the rate at which it can respond to environmental change, and is thus a direct measure of agent reactivity. We instrumented Soar to directly measure time required for encoding/storing episodes, as well as performing cue matching (i.e. the time for retrieval, without reconstructing episodes in Soar's working memory). We report maximum time: whereas average time can mask momentary computation "surges," the maximum captures the agent's ability to respond under algorithmically stressful circumstances. We compare this metric to a reactivity threshold of 50 msec., a response time that is sufficient for real-time control in games, robotics, and HCI tasks.

Since memory becomes an important factor for long runs of agents, we measure the amount of memory used by episodic memory. We also relate this measure to the average size of and changes to working memory.

To reliably measure cue-matching timing data, we instrumented Soar v9.3.1 [http://sitemaker.umich.edu/soar] to perform this operation 100 times for each cue at regular intervals across the lifetime of the agent. Storage timing data, however, only captures a single operation, and is thus noisier and we can only extract qualitative trends. All experiments were performed on a Xeon L5520 2.26GHz CPU with 48GB RAM running 64-bit Ubuntu v10.10.

Evaluated Capabilities

For each evaluation domain, we developed a specialized set of cues that implemented a set of *cognitive capabilities*, or high-level functionalities supported by episodic memory (Nuxoll and Laird 2012). The following are the full set of capabilities that we include in this evaluation:

Virtual Sensing. An agent retrieves past episodes that include sensory information beyond its current perceptual range that are relevant to the current task.

Detecting Repetition. An agent retrieves past episodes that are identical (or close to identical), possibly indicating a lack of progress towards goal(s).

Action Modeling. An agent retrieves an episode of performing an action and compares that episode to one or more episodes that followed to model action consequences.

Environmental Modeling. An agent retrieves an episode and compares that episode to episodes that followed to model world dynamics, independent of its own actions.

Explaining Behavior. An agent replays episodes of its behavior to explain its behavior to itself or others.

Managing Long-Term Goals. An agent retrieves goals that were initiated in the past but are not currently active, to determine if they should be active in the current context.

Predicting Success/Failure. An agent replays episodes to estimate the value of an action with respect to task goals.

Empirical Results

For each evaluation domain, we describe the properties of the task and agent, related work, the set of cues developed for the task, and empirical results.

Word Sense Disambiguation

An important problem for any agent that uses natural language is word sense disambiguation (WSD) – the task of determining the meaning of words in context. We extend prior work that explored the degree to which memory-retrieval bias was beneficial in WSD (Derbinsky and Laird 2011). In this formulation, the agent perceives a <lexical word, part-of-speech> pair, such as <“say”, verb>, and, after attempting to disambiguate the word, the agent receives all word meanings that were appropriate in that context. To measure the benefit of memory in this task, the agent perceives the corpus, in order, numerous times, and is evaluated on learning speed and accuracy.

For this task, we implemented an agent that represents the last *n* lexical-word inputs as an *n-gram*. The agent then uses a sequence of episodic cognitive capabilities to form a disambiguation: first, it cues episodic memory to detect a *repeated situation* (e.g. “when did I last perceive the 3-gram {Friday, say, group}?”); it then retrieves the next episode, forming an *environmental model* of feedback (e.g. “what happened when I replied ‘express a supposition’?”); and then disambiguates using this prior information, *predicting future success* based upon prior experience.

We evaluated the agent using SemCor (Miller et al. 1993), the largest and most widely used sense-tagged corpus. During its first exposure to the corpus, the agent can disambiguate 14.57% of words using a 2-gram representation, and 2.32% using 3-grams. In the next exposure, these performance levels improve to 92.82% and 99.47%, respectively. These results show the benefit of flexible access to a high-fidelity store of experience.

This domain is small, on average requiring 234 bytes of memory to store the working-memory changes in each episode. However, as with all natural-language texts, there are some words that appear more often than others in SemCor, and so this task exemplifies the effects of temporal selectivity and cue-feature co-occurrence.

To evaluate scaling, we selected two 3-word phrases from the corpus and used a set of cues that represented all 1-, 2-, and 3-gram contexts for these phrases (11 cues total, as one word was common; see Table 1). We ran the agent five times across SemCor (4.6M episodes). We measured the storage and retrieval performance every 50K episodes.

Table 1. WSD: occurrence and cue endpoints for SemCor.

n-Gram	Occurrence	Endpoints	Proportion
{group}	1,333	1,333	~ 0%
{say}	1,005	1,005	~ 0%
{Friday}	18	18	~ 0%
{well}	150	150	~ 0%
{be}	8,400	8,400	~ 0%
{say, group}	6	2,338	0.21%
{Friday, say}	1	1,023	0.55%
{Friday, say, group}	1	2,356	1.27%
{be, say}	69	9,405	0.07%
{well, be}	27	8,550	0.17%
{well, be, say}	1	9,555	5.16%

All operations met our reactivity criteria (<50 msec.). Maximum storage time was essentially constant, with a maximum of 0.5 msec. The maximum query time, across all 11 cues, was 22.05 msec. We regressed a model that predicts cue-matching time in msec., as a linear factor of the number of interval endpoints walked (r^2>0.999): 0.0024x + 0.0647. This model predicts that retrieval time, in this task, is dependent almost exclusively on interval walking; we estimate scaling by computing the number of endpoints walked when the function value equals 50 msec. (20,806). If we assume one word per episode, Soar’s episodic memory can perform cue matching that examines 11.23% of SemCor (total = 185,269 words). We now examine how this scaling limit compares to the space of possible cues, and the evaluation cues we used in this task.

In SemCor, only two words occur more frequently than in 1% of inputs: “be” (4.53%) and “person” (3.61%). Thus, Soar’s episodic memory can reactively respond to any individual feature as a cue. However, as cue size increases, the number of potential endpoints to walk increases additively with each word, while co-occurrence frequency, the number of times the n-gram occurs within the corpus, can only stay constant or decrease. For instance, consider the following two phrases used for our cue evaluation: {Friday, say, group} and {well, be, say}. The endpoint and co-occurrence frequency data of all 1-, 2, and 3-grams of these phrases is in Table 1, where the final column is (endpoints/occurrence) divided by the size of SemCor, estimating the proportion likely to be examined for each cue (assuming uniform distribution of occurrence). For 1-grams, Soar achieves constant-time cue matching, independent of this data, since it concludes cue matching after the first match. For those n-grams with a co-occurrence of 1, cue-matching time exhibits saw-tooth patterns, where peaks are once-per-corpus exposure, since the number of endpoints to examine increases until the n-gram is re-encountered. For non-zero co-occurrence, we see more frequent, non-uniform heights in the data, as the n-grams are encountered through the corpus. Soar’s episodic memory can perform this task reactively for 1-grams, 2-grams, and 3-grams, as SemCor proportion for all cues of these lengths is below 11.23%. However, of the more than 184,000 distinct 4-grams in this corpus, there are 368 that examine more than 11.23% of SemCor, and thus a 4-gram is the scaling limit.

Generalized Planning

In WSD, temporal selectivity of cues was the primary factor affecting performance, whereas cue size and structure had little effect. To evaluate cue complexity, we extended prior work that used episodic memory as a source of action-modeling knowledge for planning (Xu and Laird

2010). In this evaluation, we used 12 planning domains, common in competitions (*Logistics*, *Blocksworld*, *Eight-puzzle*, *Grid*, *Gripper*, *Hanoi*, *Maze*, *Mine-Eater*, *Miconic*, *Mystery*, *Rockets* and *Taxi*) and made 44 problem instances by varying domain parameters (e.g. number of blocks in Blocksworld). The agent's state captures a set of objects and relations, and the agent has rules that encode the actions it can take. At each episode, the agent randomly selects an action, exploring the state space over time.

Our first experiment explored whether episodic memory could *detect repeated states*. For each problem instance, we extracted a random problem state as our evaluation cue. We then ran the agent for 50K episodes, and measured performance every 1K episodes as it explored the state space. This experiment evaluates Soar's episodic memory while stressing the dimension of structural selectivity: in these domains, the cue is relatively large and the agent state is structurally homogenous, and thus cues match multiple structures in most other episodes.

Soar's episodic memory reactively stored episodes in all problem instances (maximum < 12.04 msec.). Memory consumption in each domain was strongly correlated with the number of working-memory changes (r^2=0.86): storage ranged from 562 bytes per episode to 5454, averaging 1741. Using the full 48GB of RAM on our evaluation computer, we could thus store between 9 and 91 million episodes, with more than 29M on average. In summary, storage is not a scaling concern in this set of domains.

Of the 44 problem instances, there were 12 in which cue matching remained reactive for the full 50K episodes, all of which were instances of the Miconic, Maze, Hanoi, and Gripper domains. These problem instances did not exhibit growth in their cue-matching time, while the remaining problem instances grew rapidly and became unreactive in fewer than 10k episodes. When we explored the data for explanatory factors, we found that retrieval time within each domain strongly correlated with the number of episodes searched and working-memory size (r^2=0.85). The 12 problem instances that did scale have the smallest average working-memory sizes, as well as relatively small state search spaces (yielding small, bounded interval searches). The remaining instances were either too structurally unselective, too temporally unselective (due to a large state space), or both. These results characterize an upper bound in cue complexity for reactive retrievals.

Our second experiment explored whether episodic memory could be used to detect *analogous* states. We used the same setup as in the previous experiment, but removed all grounded features in the evaluation cues. However, this had the effect of making the cues less structurally selective (i.e. each cue feature could match more structures when compared to an episode). As a result, Soar's episodic memory could not scale on any problem instance, primarily due to frequent and expensive structural matching. These findings suggest that Soar's episodic memory is not appropriate for direct analogical mapping.

Our final experiment explored whether episodic memory could be used to detect analogous states *if* the agent had knowledge of important schemas at the time of encoding. This experiment relates to prior work showing that experts are able to encode memories that can be relationally retrieved (Gentner et al. 2009). We encoded the cues from our second experiment, those without grounded features, as rules that would place a flag in working memory whenever the pattern appeared, a feature episodic memory would automatically encode and could be queried for directly. Soar's episodic memory could scale to 50K episodes in all problem instances given this task formulation (max. < 0.08 msec.), suggesting agents can perform limited analogical reasoning over large stores of prior experience, while remaining reactive, by joining task-dependent recognition knowledge with a task-independent episodic memory.

Video Games & Mobile Robotics

The previous evaluations focused on specific aspects of episodic-retrieval cues: WSD stressed temporal selectivity, while the planning domains stressed structural complexity. We found that Soar's episodic memory has scaling limits that depend on domain structure and dynamics, knowledge representation, and cues. Here we examine the degree to which these limitations apply to domains in which agent actions impact its future perceptions of the world. We describe the domains, and then present combined results.

TankSoar. TankSoar is a video game that has been used in evaluating numerous aspects of Soar, including episodic memory (Derbinsky and Laird 2009; Nuxoll and Laird 2012). In TankSoar, the agent controls a tank and moves in a discrete 15x15 maze. The agent has numerous sensors, including path blockage and radar feedback, and it can perform actions that include turning, moving, and controlling its radar. The agent we use, *mapping-bot*, explores the world and populates an internal map, stored in working memory, which averages 2734 structures.

This task is interesting for episodic-memory evaluation due to a large working memory with relatively few changes (~23 per decision). However, most perceptual structures change frequently and many are highly selective, both temporally and structurally.

We used 15 cues in TankSoar, many developed in prior work (Derbinsky and Laird 2009). These cues implemented virtual-sensing, detecting-repetition, and action-modeling capabilities. For example:

1. "When did I last sense a missile pack on my radar?"
2. "When was I last at this (x, y) position on my map?"
3. "What happened last time I rotated left and turned on my radar while I was blocked in the forward direction?"

Cues that referred to a map cell (e.g. #2) were structurally unselective, as they could refer to any of the 225 entries.

Table 2. Empirical results for video games and mobile robotics.

	Storage			Cue Matching (Max. Time in msec.)	
	Max. Time (msec)	Avg. Bytes/Episode		High Selectivity	Low Selectivity
TankSoar	18.66	1,035		4.77	18.31
Eaters	1.39	813		0.71	-
Infinite Mario	55.01	2,646		1.66	40.43
Mobile Robotics	3.17	113		0.75	27.50

The temporal selectivity of cues relating to perceptual structures was typically reduced as the cue size increased, due to non-overlap in feature co-occurrence (e.g. in #1 there were episodes when the agent used the radar, but did not sense a missile pack). We ran mapping-bot for 3.5M episodes, which is >48-hours of simulated real-time (SRT: 50 msec./episode), and measured performance every 50K.

Eaters. Eaters has also been used in previous episodic-memory evaluations. Eaters is a video game, similar to PAC-MAN, where the agent controls an "eater," which moves through a 15x15 grid-world, eating different types of food. The agent senses a 2-cell radius, and can move in any of the four cardinal directions. The agent we use, *advanced-move*, prioritizes movement based on food types.

This task is interesting as a contrast to TankSoar. The agent's working-memory size is drastically smaller (230 structures), but changes are comparable (~19 per decision). We used 7 cues that exemplified virtual sensing, detecting repetition, action modeling, and explaining behavior. For instance: "What happened the last time there was normal food to the east of me and bonus food to the west of me?" Examining the episode following this retrieval supports the agent explaining its own preferences regarding the relative desirability of these food types, informing predictions of its own future decisions. The agent state is sufficiently simple such that no evaluation cue was unselective, either structurally or temporally. We ran advanced-move for 3.5M episodes (>48 hours, SRT) and measured every 50K.

Infinite Mario. Infinite Mario is a video game used in the 2009 Reinforcement-Learning (RL) Competition and is based on Super Mario. The allocentric visual scene comprises a two-dimensional matrix (16x22) of tiles and the agent can take actions that include moving, jumping, and increasing speed. We use an agent that applies an object-oriented representation and hierarchical RL to quickly improve performance in the task (Mohan and Laird 2011) and collect data on game level 0.

Several aspects of this game are interesting for evaluation. The working memory is large (~2000) and contains a variety of representational patterns, including flat features that are both symbolic (e.g. Mario is "small," "big," or "fiery") and real-valued (e.g. distance to enemies); hyper-edges (e.g. rows in the visual scene); and relational structures (e.g. relating hierarchical state representations to perception). Also, due to side-scrolling, a relatively large percentage of the visual scene in changes between episodes (avg. 73), stressing temporal contiguity.

We evaluated 14 virtual-sensing and action-modeling cues. For example, the following cue combines perceptual features with those derived from task knowledge: "What did I do when last I encountered a winged, downward-flying 'Goomba' that was a *threat*?" Cues that virtually sensed visual-scene cells were structurally unselective. We ran the agent for 3.5M episodes (>48 hours, SRT) and measured performance every 50K.

Mobile Robotics. For this evaluation, we used an existing mobile-robotics platform that has been applied to simulation and physical hardware (Laird, Derbinsky, and Voigt 2011). The agent perceives both physical perception data, including real-valued abstractions of laser range-finder data, as well as symbolic representations of objects, rooms, and doorways. The task is to explore a building with 100 offices, and then execute a fixed-patrol pattern. While performing these tasks, the agent builds an internal map, which it uses for path planning and navigation. The average working-memory size is 1235, with 2 changes.

We evaluated 6 cues for virtual sensing and goal management. Consider the following cue: "When was my desired destination doorway #5?" The agent could examine episodes that followed to recall progress made towards that goal. However, as the agent accumulated more distinct goals, this cue became less temporally selective. We did not evaluate any structurally unselective cues in this task. We ran the agent in simulation for 12 hours of real time, measuring performance every 300K episodes (~2 min.).

Results. The left half of Table 2 presents storage results, grouped by domain. Soar stored episodes in less than 50 msec. for all domains except Infinite Mario, where infrequent spikes in perceptual changes, caused by Mario dying and restarting the level, defied the temporal-contiguity assumption. The storage cost across domains correlated with working-memory changes (r^2>0.93).

The right half of Table 2 presents cue-matching results, grouped by domain and cue selectivity (temporal or structural). Soar maintained reactivity across all domains. With one exception, retrieval time did not meaningfully increase with time. The growth rate for goal management in mobile robotics (see Figure 1) depended upon the properties of the robot's mission: when behavior shifted from exploration to patrol (~10M episodes), new goal

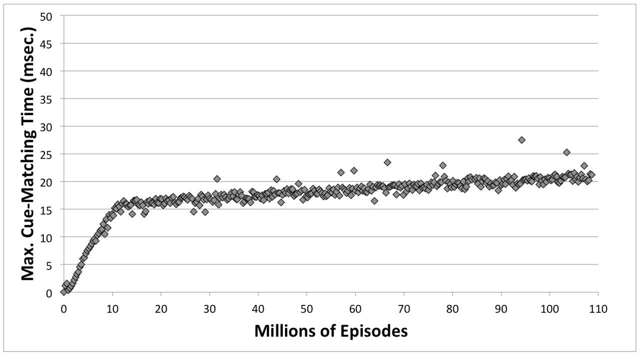

Figure 1. Mobile-robotics timing data for goal-management cue.

locations were encoded less frequently, and temporal selectivity decreased at a smaller rate. At its original rate, cue-matching time would grow beyond 50 msec. after 34M episodes; afterwards it would last nearly 634M (>3 days).

Discussion

Our goal in this paper was to evaluate the degree to which Soar's episodic memory supports useful operation across a variety of domains while scaling to long agent lifetimes. We presented evidence that in linguistics, planning, video games, and mobile robotics, it supports many useful capabilities while maintaining reactivity. However, we have also shown that the functional commitments of Soar's episodic memory lead to a mechanism that is not immune to properties of domains and cues. For example, the WSD and mobile-robotics domains illustrate how the temporal selectivity and co-occurrence of cue features can lead to searches that scale linearly with time. Also, attempting to match temporally unselective cues in structurally homogenous domains, such as planning problems, can cause cue matching to scale with cue and/or state size. These findings will inform the development and evaluation of future episodic-memory implementations.

Acknowledgments

We acknowledge the funding of the Air Force Office of Scientific Research, contract FA2386-10-1-4127, and the Office of Naval Research, contract N00014-08-1-0099.

References

Brom, C., Burkert, O., Kadlec, R. 2010. Timing in Episodic Memory for Virtual Characters. In *Proc. of the IEEE 2010 Conf. on Computational Intelligence and Games*, 305-312.

Derbinsky, N., Laird, J. E. 2009. Efficiently Implementing Episodic Memory. In *Proc. of the 8th Intl. Conf. on Case-Based Reasoning*, 403-417.

Derbinsky, N., Laird, J. E. 2011. A Functional Analysis of Historical Memory Retrieval Bias in the Word Sense Disambiguation Task. In *Proc. of the 25th AAAI Conf. on Artificial Intelligence*, 663-668.

Deutsch, T., Gruber, A., Lang, R., Velik, R. 2008. Episodic Memory for Autonomous Agents. In *Proc. of the 5th Intl. Conf. on Human System Interaction*, 621-626.

Forbus, K., Gentner, D., Law, K. 1995. MAC/FAC: A Model of Similarity-based Retrieval. *Cognitive Science* 19 (2): 141-205.

Forgy, C. L. 1982. Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem. *Artificial Intelligence* 19 (1): 17-37.

Gentner, D., Loewenstein, J., Thompson, L., Forbus, K. D. 2009. Reviving Inert Knowledge: Analogical Abstraction Supports Relational Retrieval of Past Events. *Cognitive Science* 33: 1343-1382.

Gomes, P. F., Martinho, C., Paiva, A. 2011. I've Been Here Before! Location and Appraisal in Memory Retrieval. In *Proc. of the 10th Intl. Conf. on Autonomous Agents and Multiagent Systems*, 1039-1046.

Kolodner, J. 1993. *Case-Based Reasoning*. Morgan Kaufmann.

Kriegel, H., Pötke, M., Seidl, T. 2000. Managing Intervals Efficiently in Object-Relational Databases. In *Proc. of the 26th Intl. Conf. on Very Large Databases*, 407-418.

Kuppuswamy, N. S., Cho, S., Kim, J. 2006. A Cognitive Control Architecture for an Artificial Creature Using Episodic Memory. In *Proc. of the 3rd SICE-ICASE Intl. Joint Conf.*, 3104-3110.

Laird, J. E. 2012. *The Soar Cognitive Architecture*. MIT Press.

Laird, J. E., Derbinsky, N., Voigt, J. R. 2011. Performance Evaluation of Declarative Memory Systems in Soar. In *Proc. of the 20th Behavior Representation in Modeling & Simulation Conf.*, 33-40.

Lim, M. Y., Aylett, R., Vargas, P. A., Ho, W. C., Dias, J. 2011. Human-like Memory Retrieval for Social Companions. In *Proc. of the 10th Intl. Conf. on Autonomous Agents and Multiagent Systems*, 1117-1118.

Macedo, L., Cardoso, A. 2004. Exploration of Unknown Environments with Motivational Agents. In *Proc. of the 3rd Intl. Joint Conf. on Autonomous Agents and Multiagent Systems*, 328-335.

Miller, G. A., Leacock, R., Tengi, R., Bunker, R. T. 1993. A Semantic Concordance. In *Proc. of the Workshop on Human Language Technology*, 303-308.

Mohan, S., Laird, J. E. 2011. An Object-Oriented Approach to Reinforcement Learning in an Action Game. In *Proc. of the 7th AAAI Conf. on Artificial Intelligence and Interactive Digital Entertainment*, 164-169.

Nuxoll, A. M., Laird, J. E. 2012. Enhancing Intelligent Agents with Episodic Memory.*Cognitive Systems Research* 17-18, 34-48.

Smyth, B., Cunningham, P. 1996. The Utility Problem Analysed: A Case-Based Reasoning Perspective. In *Proc. of Advances in Case-Based Reasoning, 3rd European Workshop*, 1560-1565.

Tecuci, D., Porter, B. 2007. A Generic Memory Module for Events. In *Proc. of the 20th Florida Artificial Intelligence Research Society Conf.*, 152-157.

Tecuci, D., Porter, B. 2009. Memory Based Goal Schema Recognition. In *Proc. of the 22nd Florida Artificial Intelligence Research Society Conf.*, 111-117.

Tulving, E. 1983. *Elements of Episodic Memory*. Oxford: Clarendon Press.

Xu, J. Z., Laird, J. E. 2010. Instance-Based Learning of Deterministic Relational Action Models. In *Proc. of the 24th AAAI Conf. on Artificial Intelligence*, 1574-1579.