Spontaneous Retrieval for Prospective Memory: Effects of Encoding Specificity and Retention Interval

Justin Li and **John Laird**
University of Michigan
2260 Hayward Street
Ann Arbor, MI 48109-2121
{justinnh, laird}@umich.edu

Abstract

This paper explores the role of spontaneous retrieval in prospective memory, in an agent implemented in the Soar cognitive architecture. At goal initiation time, spreading activation causes the goal to be the most activated element in long-term memory, at which point it is spontaneously retrieved into working memory and pursued. We show that goal encoding specificity increases prospective memory performance, while a lengthier retention interval decreases performance if the percepts are differentially presented; both trends qualitatively resemble results described in psychology literature. However, a large space of possible spontaneous retrieval implementations remain unexplored, and much work remains to be done before spontaneous retrieval in a cognitive architecture can be fully understood.

Keywords: spontaneous retrieval; prospective memory; encoding specificity, cognitive architecture; Soar

Introduction

Prospective memory is the ability to remember to do something in the future, often while other activities are being performed during the delay. Such tasks are common in everyday life: from passing a message to a colleague, to taking medication before bed, these tasks all require the subject to perform particular actions (giving the message, swallowing a pill) under particular conditions (when the colleague is in sight, at bedtime). One of the main research questions in prospective memory is how people recall that they have an action to perform when the goal conditions are met. Two general classes of strategies have been suggested: monitoring, where someone deliberately checks if the conditions are satisfied, and spontaneous retrieval, where the need to act somehow "pops" into mind. Cognitive architectures are well-suited to create models of monitoring, since agents created in that framework have fine deliberate control over the use of memory. The same, however, cannot be said of spontaneous retrieval strategies, as they require automatic memory mechanisms, which have thus far received little attention in cognitive architectures.

This paper presents a preliminary exploration of the use of spontaneous retrieval for prospective memory. We implemented an automatic, uncued, activation-based retrieval mechanism in the Soar cognitive architecture, and demonstrate that the mechanism provides agents with a robust prospective memory ability. In an abstract domain that presents agents with randomly-generated goal conditions, the use of spontaneous retrieval allows the agent to achieve its prospective goals across a wide range of environmental and agent parameters. Furthermore, the performance of the agent changes with encoding specificity and retention interval length in ways that qualitatively resemble those of people. This serves as one step in building a complete model of how people perform prospective memory tasks, and the factors that must be taken into consideration when selecting between strategies.

Background

Prospective Memory

Although prospective memory has gotten increasing attention from psychologists in the last twenty years, the capability is only defined as a "fuzzy set" of intuitions around "remembering to *do something* at a particular *moment (or time period) in the future*" (emphasis in original) (McDaniel & Einstein, 2007). For clarity, we define a prospective memory task as represented by the *target* — the conditions under which the goal is applicable — and the *action*, which the agent must take to achieve the goal. Within this framework, previous literature has identified the five stages of completing a prospective memory task (Ellis, 1996). To use message-passing as an example, the stages are:

Encoding The goal is created and stored in long-term memory; this occurs when the message is given to the agent and asked to be passed on to the colleague.

Retention This stage is the delay between the storage of the goal and when the target conditions are met, such as between when the message was received and when the colleague is seen.

Initiation The target conditions of the goal are fulfilled, and the goal must be retrieved from long-term memory to working memory. In the example, the colleague is in sight.

Execution The action of the goal is taken; in this case, the colleague is given the message.

Completion Long-term memory must be changed such that the goal will not be repeated; that is, when the colleague is next seen, another attempt to pass on the message would not be made.

The crux of prospective memory is during the initiation stage, which hides a knowledge dependency problem (Li & Laird, 2013a). Since people cannot directly act on knowledge in long-term memory, the goal must be retrieved for the sight of the colleague to be considered significant; at the same time, since retrieval from long-term memory can only be done deliberately, the goal is not retrieved without a recognition of significance in the first place.

The psychology literature identifies two classes of human strategies to avoid this dependency problem. The first is *monitoring*, which is characterized by the continual expenditure of attentional resources (Smith & Bayen, 2004); this can be modeled by the agent periodically retrieving and checking the relevance of goals that are related to its current situation, what is called a *preemptive strategy* in prior work (Li & Laird, 2013b). This class of strategies breaks the dependency cycle by retrieving goals without determining their relevance. The second category of strategies breaks the cycle the other way, by removing deliberation in the retrieval of goals; these require the long-term memory system to signal the agent in some way. One possibility is to signal that there is a relevant goal to prompt a deliberate memory search (what is called a *noticing-plus-search* strategy), while an alternative is to have the knowledge of the goal be spontaneously retrieved into working memory (a true *spontaneous retrieval* strategy).

These two classes of strategies are not mutually exclusive, but complimentary. Although early work on prospective memory debated whether monitoring or spontaneous retrieval is the better description of human behavior, recent work has shifted towards determining the factors that influence which strategy is used for a particular prospective memory task (Einstein et al., 2005). Monitoring strategies are preferred when the goal conditions are non-focal or when the goal is important, while spontaneous strategies are preferred when the delay is long, when working memory resources are low, or when the interim task is cognitively demanding (McDaniel & Einstein, 2007). This suggests that spontaneous retrieval is more effective than monitoring at completing prospective tasks when these properties are present in the environment and the goal. Both strategies are also affected by the encoding of the goal which, following the encoding specificity principle, must match the percepts at the time of initiation (Einstein & McDaniel, 2010).

Although computational models of prospective memory have been built, they tend to sidestep initiation and focus on retrieving the correct goal from long-term memory. One model explored the Intention Superiority Effect (ISE), which states that unachieved goals are retrieved more quickly than achieved goals (Lebiere & Lee, 2002). Another model looked at different accounts of finding the correct goal, and correlates the timing results to human data (Elio, 2006). Crucially, both models assume that the agent knows that a goal must be retrieved, while the difficulty of the initiation stage lies in how that fact is recognized. Since both models require deliberate use of memory, they more closely resemble monitoring strategies. Modeling spontaneous retrievals would require an automatic memory mechanism, which is discussed below.

Spontaneous Retrieval

Spontaneous retrieval from long-term memory has been acknowledged since the first studies of memory (Ebbinghaus, 1913). In contrast to deliberate or voluntary retrieval, which requires executive functions for search control, spontaneous retrieval is an associative process that requires little to no cognitive effort, often resulting in memories that overlap in features with the current situation (Berntsen, 2010). There is often a distinction between retrieval from semantic memory (i.e., retrievals of facts) and retrievals from episodic memory (i.e., retrievals of experiences) (Kvavilashvili & Mandler, 2004; Berntsen, 2008); although it is possible for prospective memory to use either mechanism, here we focus on retrievals from semantic memory.

Computationally, designing a spontaneous retrieval mechanism requires answering two questions: When is a memory retrieved? And which memory is retrieved? The answers to these two questions define a space of spontaneous retrieval mechanisms, a more thorough exploration of which can be found elsewhere (Li & Laird, 2015). Here we only note that there are additional constraints on the second question, namely, that the retrieved memory should be relevant to the current situation. In most cognitive architectures, retrieval from long-term memory requires a description of the features of the desired memory element. This description ensures that the retrieved element can be used for further reasoning. With spontaneous retrieval, however, the agent cannot deliberately create this description; a different mechanism for ensuring relevance must be used. One solution is to use a spreading activation mechanism, such that the knowledge in working memory influences which long-term memory elements are highly activated and are thus more likely to be retrieved. Again, the full space of spreading activation mechanisms is beyond the scope of this paper.

Implementation in Soar

Soar (Laird, 2012) represents all declarative knowledge as edge-labeled directed graphs. Knowledge in *working memory* is matched by *procedural if-then rules*, which in turn modify working memory. In addition to buffers that represent the perceptual input and motor output of the agent, working memory also contains buffers that allow agents to access long-term memory. In particular, a Soar agent can *store* a single element (a graph node plus all its outgoing edges) into *semantic memory*. Before the current work, the only mechanism for *retrieval* from semantic memory was for the agent to (deliberately) create a *cue* — a set of features of the desired memory element. Semantic memory then finds all elements that contains the entire set of features, and then places the element with the highest *activation* into working memory. This semantic memory element activation is *boosted* when the element is stored or is the result of a retrieval (similar to ACT-R), and *decays* over time as controlled by a *decay rate* parameter.

Spontaneous retrieval extends the capabilities of Soar's semantic memory. From the perspective of the agent's interface to memory, the biggest change is that whenever there is no deliberate retrieval, semantic memory automatically selects an element to be placed into the semantic memory buffer. As with the bias from deliberate cued retrieval, semantic memory selects the most highly activated element, with the caveat that it skips over any element that is already in working memory; this ensures that spontaneous retrieval is not attempting to retrieve knowledge that the agent has already retrieved.

To ensure that the spontaneously retrieved memory is relevant, spreading activation is used to boost the activation of elements related to the contents of working memory. Our implementation of spreading activation is different from the spreading activation in ACT-R (Anderson, 2007). In the latter, spreading is only a term in determining the bias for retrieval, but has no long-term effect on a long-term memory element's base-level activation. This is undesirable, since this means spreading is ahistoric — spreading makes an element more likely to be retrieved at this current time step, but has no effect on whether it is likely to be retrieved at the next time step.

Instead, we created a spreading activation mechanism that directly changes base-level activation. The mechanism is defined by two parameters: what triggers the initial activation boost occurs, and which elements are affected by the spread. This strictly subsumes the original activation mechanism of Soar, which can be cast as a "spreading" mechanism in which the initial boost is triggered by long-term memory storage and retrieval, and in which no other element is affected. Additionally, we added a third trigger: whenever a rule puts a long-term memory element into working memory, that element also receives an activation boost. In effect, this allows input percepts to cause changes in long-term memory activation. Whereas before only the element itself is boosted, now all graph-neighbors of that element, and the neighbors of those elements and so on, also receive a boost in activation, to some parameterized depth d. Note that these boosts are identical, as though those elements were themselves retrieved into working memory. This is not ideal — a boost which decreases with distance may be more intuitive — but it serves as an initial attempt at a useful spontaneous retrieval mechanism.

Given this mechanism, a Soar agent that uses spontaneous retrieval for prospective memory works as follows:

Encoding The goal and its targets and actions are stored into semantic memory. The goal receives a boost in activation as a result, but this has little impact on its retrieval later.

Retention The goal is forgotten from working memory. The semantic memory activation of the goal also decays, but remains retrievable. When external percepts coincidentally (partially) overlap with the target, the goal will receive a boost in activation due to spreading, but in general, the activation of the goal is low, and is not spontaneously retrieved. Even if the goal is spontaneously retrieved, the agent will discover that the goal conditions are not met, and the goal is ignored.

Initiation At initiation time, all the goal target conditions are matched. Each individual condition causes a boost in the activation of the goal; together, these significantly increase the goal's activation, causing it to be spontaneously retrieved. A rule then matches the retrieved goal, which verifies that the conditions of the goals are satisfied. The agent can then choose to pursue the goal.

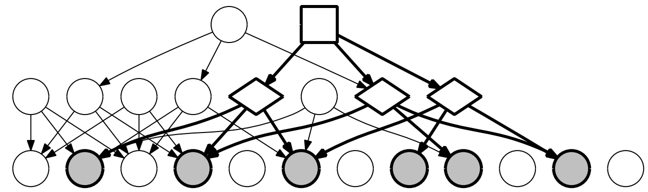

Figure 1: An example random knowledge hierarchy. See text for description.

Execution The action of the goal is then performed.

Completion Finally, once the goal is fulfilled, the agent removes the goal from semantic memory. The goal will never be spontaneously retrieved, and the agent never pursues that instance of the goal again.

Crucially, this strategy requires the goal to be within spreading distance of the agent's perceptions. This may not always be the case — the conditions of the goal may be described in abstract terms that do not directly correspond to perception. There are several ways to prevent this from occurring. One possibility is to increase the depth limit of spreading activation; due to the branching factor of semantic memory, however, this is exponentially costly. Another possibility is for the agent to encode the goal such that the conditions more closely match perceptual input; in other words, to increase encoding specificity. Finally, the agent may also have additional rules that elaborate on perceptual input (*elaboration rules*), building up to to the goal conditions; in essence, this is generalizing the percepts, and can be thought of as the flip side of encoding specificity. As we demonstrate below, these three factors are not independent, and together they determine whether the goal is boosted and retrieved.

Prospective Memory Domain

In order to evaluate the use of spontaneous retrieval for prospective memory, a domain was created that represents prospective memory tasks in the abstract.

To simplify our analysis, we restrict our work to where the structure of knowledge in long-term memory forms a recognition hierarchy, or equivalently an ontology with only `has-a` relations. For example, the agent may recognize that an object with four legs and a back is a chair, and that because there are multiple chairs and multiple tables, that the location is a classroom. For this domain, we randomly generate such hierarchies from the bottom up, where the creation of each lower-level feature has a probability of resulting in a feature one level up. This process continues until a specified number of features at a specified height is created; for example, Figure 1 shows a hierarchy of width 2 and height 3. Note that the hierarchy is not a connected graph: lower-level features may not be part of any higher-level feature.

Within the prospective memory domain, the knowledge hierarchy determines both the target conditions of goals and the percepts of the agent. The only input the agent perceives is from the lowest level of the hierarchy, while a subset of higher nodes are designated as goals. For example, in Figure 1, if the square node is a goal, then the agent should perform the goal action when all the percept-level descendants of that goal are perceived (shaded). To generate a particular trial for the agent, the percepts for goals are first inserted into the percept sequence, with the remaining percepts interpolated using a noisy random walk.

Before each trial, the knowledge hierarchy is inserted into the semantic memory of the agent, but *without* knowledge of which features are the goals. At each time step, the agent is presented with features from the lowest level of the hierarchy, on which elaboration rules would match to create higher-level features. In addition to percepts, the agent is also presented with goals and their features (if the goal is the square node in Figure 1, its features are the diamond-shaped nodes). It is up to the agent to store the goal into memory, where it may also encode the goal more specifically by expanding the intermediate-level features into percept-level features (for example, linking the goal with the shaded nodes instead).

Within this domain, we are interested in the proportion of prospective memory tasks completed by an agent using a spontaneous retrieval strategy. We are interested in several environmental and agent parameters:

- (*) The specificity of encoding by the agent.
- The maximum spreading depth in semantic memory.
- The highest level of perceptual elaboration.
- (*) The length of the retention stage.
- The average number of conditions in a goal.
- The decay rate of semantic memory.

The parameters marked with asterisks are known to have an effect on human prospective memory. For encoding specificity, it is expected that prospective memory performance increases when the goal encoding matches that of percepts. As for the length of the retention stage, the longer the interval, the more likely that a spontaneous strategy is chosen. It is assumed that this is due to the increased cost of monitoring for long periods, but that does not preclude the possibility of spontaneous retrieval performance also changing as a function of this parameter.

Results

In general, spontaneous retrieval provides a robust prospective memory ability, allowing an agent to complete an average of 81.5% (and a median of 90%) of its goals across a range of parameter settings. We examine the effects of encoding specificity and retention interval length below.

Encoding Specificity

Encoding specificity, in this case, refers to the target conditions of the goal that is stored in long-term memory. Instead of directly storing the features of the goal, the agent instead stores the goal with its lower-level features; in Figure 1, this means the goal (the square node) is stored with the shaded nodes as its conditions instead of the diamond-shaped nodes. This encoding means the goal is now connected to the knowledge hierarchy at a lower level than it would be otherwise. Since the goal is often at the top of the knowledge hierarchy, we denote the specificity of an encoding by how many levels below the goal it is linked to; in this example, the encoding specificity would be 2.

Given this definition of encoding specificity, we perform initial analysis to determine whether a goal could be spontaneously retrieved. For goal at knowledge level g, elaboration rules that create features up to level e, and a spreading depth of d, the goal must be encoded at specificity level s that satisfies the following relationship:

$$d \geq g - e - s + 1 \tag{1}$$

That is, the spreading depth must be able to reach from the highest-level elaborated features to goal conditions (plus an extra level to spread from the conditions to the goal itself). Note that the agent cannot complete any goals when $d = 0$, since the goal would never receive an activation boost from spreading (since no spreading occurs). We can additionally calculate the maximum number of boosts a goal will receive, assuming the knowledge hierarchy has branching factor b:

$$\sum_{i=\max(1,\, s-d+1,\, g-e)}^{\min(g,\, s+d-1)} b^i \tag{2}$$

That is, every feature in the levels indicated by the index would boost the goal, a number which is exponential in the branching factor. This classification allows us to group agents across a large parameter space for comparison. The results here are from exploring $1 \geq g \geq 3$, $0 \geq e \geq 3$, $1 \geq s \geq 3$ and with $d \in \{1,2\}$ and branching factor of 3.

A number of parameter settings within this space fail in completing any goals. Upon closer examination, these are settings where the goal is at least two steps away from the elaborated features — for example, if elaborations provide features of level 3 and the goal conditions are encoded at level 4, thus requiring a two-level spread from elaboration to condition to the goal. Equivalently, this is when $e + s < g$, or where the right hand side of Equation (1) is two or more. In these cases, the lower-level features are activated more frequently, causing them to have higher activation than the goal and preventing the goal from being retrieved. This suggests that the activation boosting of a goal is not sufficient to guarantee its completion.

All other parameter settings allow the agent to complete goals. The parameter that is most correlated with higher performance is the specificity of the encoding: every increase in specificity results in a higher proportion of goals being completed. The results in the table below are typical; the numbers represent the proportion of goals that the agent completed. In retrospect, this is not surprising: more specifically encoded goals are linked to more features, which means that there are more opportunities for the activation of the goal to be boosted.

Table 1: Representative results demonstrating the effects of encoding specificity. The numbers reported are the proportion of goals completed.

Elaboration Level	**Encoding Specificity** 1	2
0	0%	75%
1	70%	80%
2	65%	80%
3	65%	80%

Neither the level of elaboration nor the depth limit for spreading activation have uniform effect on the agent's performance. Although these parameters also effect the number of times a goal is boosted, the problem is that they also boost the activation of all *other* goals in addition to the goal that is being initiated. As with the low-level features from above, it is a high *relative* activation that allows a goal to be spontaneously retrieved. More specific encodings provide a large enough boost at initiation for the single goal to be retrieved, while these other parameters do not.

Overall, these results agree with the psychology literature: the best goal encoding should match both environmental parameters (such as how abstract the goal is) and agent parameters (such as the limit to spreading activation), but that more specific encodings in general lead to better performance.

Retention Interval

In our initial experiment, none of the retention interval length, the decay rate of semantic memory, nor the number of goal conditions had any individual effect; whether a trial lasts 2,000 or 10,000 time steps, or have between 1 and 20 conditions, the agent performs equally well. Learning from the experiments with encoding specificity, however, we suspect that this is due to the "density" of percepts to goals. The features that an agent perceives during the retention interval are randomly selected, and may coincidentally be one of the conditions for a goal; that goal would then receive a small boost in activation. Since all percepts are equally likely, all goals would receive roughly equal numbers of activation boosts, meaning no single goal is particularly highly activated (or particularly un-activated either).

We can frame this idea into one of "resting activation" — activation that a goal would have during the retention interval, which is determined by an equilibrium formed by the increase in activation due to spreading from random input and the decrease in activation due to decay. Changes in either would move the resting activation value; if the decay rate is increased, or if there is less activation from random input (as would be the case if the input did not contain target conditions at all), the resting activation value would decrease. Again, it is not the resting activation that directly determines the performance of the agent, but the relative activation of a goal at initiation time; this is why the decay rate has no effect, since it affects the activation of all goals. Conversely, if a goal has low resting activation compared to other goals, the activation spread from its target conditions may not be sufficient to make it the most activated element, preventing its spontaneous retrieval.

To demonstrate this, we modified the domain such that during the retention stage, the conditions of a single goal are never presented to the agent until initiation. We call this the Leave One Out percept sequence, as opposed to the Normal percept sequence. For that goal, there should be much less activation boosts from spreading as compared to other goals, leading to a lower resting activation level. In this case, a longer retention length (as activation decays after the goal is initially stored) should leave the goal uncompleted.

As expected, the Leave One Out sequence results in much more variance in the activations of goals. The least activated goal in a Normal percept sequence is 1.44 standard deviations away from the mean, while with a Leave One Out sequence, the least activated goal is 12.1 standard deviations away (the standard deviations were calculated without the outlier). This leads to the goal not being retrieved for completion as the retention interval increases, as show in the table below:

Table 2: The effect of different percept sequences. The numbers reported are the proportion of goals completed.

Mean Retention Interval (time steps)	**Sequence Type** Normal	Leave One Out
105.7	96.7%	58.3%
125.0	96.7%	55.0%
142.7	97.5%	50.0%
166.2	96.7%	50.0%
194.1	97.5%	43.8%
214.0	96.7%	38.3%
246.3	97.6%	37.6%

This result could be interpreted in two ways. On one hand, for random percepts, using spontaneous retrieval for prospective memory suffers no degradations in performance, which suggests that it may be preferable to monitoring strategies. On the other hand, goals for which the conditions are never encountered outside of initiation are unlikely to be retrieved under the current mechanism, which run counter to the trends described in psychology literature. We do not know of any studies which look at the baseline frequencies of goal conditions, nor of studies which examine human prospective memory performance where performing the goal require satisfying multiple disjoint conditions. It is possible that human performance exhibit similar patterns under such situations; alternately, a better model may be a hybrid strategy where occasional monitoring-like retrievals prevent the activation of any goal from dropping too low.

Discussion and Conclusion

This paper presented a strategy for completing prospective memory tasks, by using spontaneous retrieval to bring the goal into working memory at the right time. This strategy proves robust across a number of parameters. In particular, the change in performance over two parameters qualitatively matches human data: the increased performance when the goal is encoded more specifically, and the decreased performance when the retention interval is lengthened (where the conditions of the goal are presented differentially). These results crucially depend on the idea that the goal must have higher *relative* activation compared to other knowledge in order for the prospective memory task to succeed. This explains why other memory parameters have no effect, as they alter the activation of all goals on an absolute scale, but leave relative differences unchanged.

At the same time, a major shortcoming of this work is the unexplored space of both the spreading activation and spontaneous retrieval mechanisms, as well as in the structure of knowledge in memory. These results only hold when spontaneous retrieval is based on activation, when spreading activation has a hard limit on depth, and when long-term memory is a hierarchy. It is easy to imagine alternatives: where spontaneous retrievals are based on analogical mapping, where the size of the activation boost decays over graph distance as it spreads, or where long-term memory is a more complicated graph. None of these parameters can be easily enumerated and tested, and each requires significant evaluation on its own to determine the conditions under which they best match human data or are most useful to artificial agents.

Spontaneous retrieval is an important mechanism for cognitive architectures: it is necessary to fully model human prospective memory, and it also serves as a heuristic for when memory-search guidance knowledge is lacking in artificial agents. While this work is one step in understanding such a mechanism, much work remains to be done, and given that many algorithmic details affect the utility of spontaneous retrieval, these effects may be better explored using simpler models before being implemented in a cognitive architecture.

Acknowledgments

The authors acknowledge the funding support of the Office of Naval Research under grant number N00014-08-1-0099.

References

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* Oxford University Press.

Berntsen, D. (2008). Involuntary autobiographical memories: Speculations, findings, and an attempt to integrate them. In J. Mace (Ed.), *Involuntary memory* (pp. 20–49). Blackwell Publishing.

Berntsen, D. (2010). The unbidden past: Involuntary autobiographical memories as a basic mode of remembering. *Current Directions in Psychological Science*, *19*(3), 138–142.

Ebbinghaus, H. (1913). *Über das gedächtnis (Memory: A contribution to experimental psychology)* (H. A. Ruger & C. E. Bussenius, Trans.). Columbia University Press.

Einstein, G. O., & McDaniel, M. A. (2010). Prospective memory and what costs do not reveal about retrieval processes: A commentary on Smith, Hunt, McVay, and McConnell. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *36*(4), 1082–1088.

Einstein, G. O., McDaniel, M. A., Thomas, R., Mayfield, S., Shank, H., Morrisette, N., & Breneiser, J. E. (2005). Multiple processes in prospective memory retrieval: Factors determining monitoring versus spontaneous retrieval. *Journal of Experimental Psychology: General*, *134*(3), 327–342.

Elio, R. (2006). On modeling intentions for prospective memory performance. In *Proceedings of the 28th annual conference of the Cognitive Science Society (CogSci)* (pp. 1269–1274).

Ellis, J. (1996). Prospective memory or the realization of delayed intentions: A conceptual framework for research. In M. A. Brandimonte, G. O. Einstein, & M. A. McDaniel (Eds.), *Prospective memory: Theory and applications* (pp. 1–22). Lawrence Erlbaum.

Kvavilashvili, L., & Mandler, G. (2004). Out of one's mind: A study of involuntary semantic memories. *Cognitive Psychology*, *48*(1), 47–94.

Laird, J. E. (2012). *The Soar cognitive architecture*. MIT Press.

Lebiere, C., & Lee, F. J. (2002). Intention superiority effect: A context-switching account. *Cognitive Systems Research*, *3*(1), 57–65.

Li, J., & Laird, J. E. (2013a). The computational problem of prospective memory retrieval. In *Proceedings of the 12th international conference on cognitive modeling (ICCM)* (pp. 155–160).

Li, J., & Laird, J. E. (2013b). Preemptive strategies for overcoming the forgetting of goals. In *Proceedings of the 27th AAAI conference on artificial intelligence (AAAI)* (pp. 1234–1240).

Li, J., & Laird, J. E. (2015). Spontaneous retrieval from long-term memory for a cognitive architecture. In *Proceedings of the 29th AAAI conference on artificial intelligence (AAAI).*

McDaniel, M. A., & Einstein, G. O. (2007). *Prospective memory: An overview and synthesis of an emerging field*. Sage.

Smith, R. E., & Bayen, U. J. (2004). A multinomial model of event-based prospective memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *30*(4), 756–777.