A memory for goals model of sequence errors

J. Gregory Trafton
(greg.trafton@nrl.navy.mil)
Naval Research Laboratory
Washington, DC 20375 USA

Erik M. Altmann
(ema@msu.edu)
Michigan State University
East Lansing, MI 48824 USA

Raj M. Ratwani
(raj.ratwani@nrl.navy.mil)
Naval Research Laboratory
Washington, DC 20375 USA

Abstract

We propose a model of routine sequence actions based on the Memory for Goals model. The model presents a novel process description for both perseveration and anticipation errors, as well as matching error data from a previously collected dataset. Finally, we compare the current model to previous models of routine sequential action.

Keywords: routine sequential actions; errors, cognitive modeling

Introduction

Several researchers have described classes of errors that people make as they perform routine sequential actions (Norman, 1981; Reason, 1990; Baars, 1992). Most of the categorization for these errors comes from either diary studies (Reason, 1990) or from neurologically damaged patient studies (Schwartz et al., 1998).

Sequence errors occur during routine action and consist of perseverations, omissions/anticipations, and intrusions (Reason, 1984). Perseveration errors are repeats of a previous action and come in two forms (Sandson & Albert, 1984). Continuous perseveration errors occur when an action is performed over and over. Recurrent perseveration errors occur when a previously completed subtask is performed again, usually with one or more intervening subtasks. For example, putting cream in a cup of coffee multiple times is a perseveration error. Omissions are skipped steps, while anticipation errors are skipped steps that are quickly rectified. For example, an omission error would be completely forgetting to put cream in a cup of coffee, while an anticipation error would be attempting to pour from an unopened container. It can be quite difficult to differentiate omission and anticipation errors (Cooper & Shallice, 2000). Intrusion errors (sometimes called capture errors) occur when an action comes from a different, usually related, task. For example, a capture error would occur when attempting to make coffee a person gets distracted by a tea bag and instead makes tea.

There are other types of errors that occur during routine action, but this report will focus on sequence errors.

Previous models of sequential behavior

There are two computational models of routine sequential behavior: the interactive activation network (IAN) model (Cooper & Shallice, 2000; Cooper & Shallice, 2006) and the simple recurrent network (SRN) model (Botvinick & Plaut, 2004; Botvinick & Plaut, 2006).

In the IAN model, different schemas compete for activation. Activation comes from triggers (environmental or context) and source-schemas (related schemas), but a schema will not be activated if it is not over a specific threshold. Thus, while working on a routine task, the selection of a schema is influenced by the current schema and the state of the world. The IAN model suggests that errors are caused by a lack of attentional resources or distraction in normal populations (Norman & Shallice, 1986; Cooper & Shallice, 2000). Variability in attentional resources is instantiated in IAN by noise. In the case of sequence errors, noise has two major effects. First, noise in the system can cause variability in the ordering of schemas that do not have ordering constraints. Second, noise can cause variability in the selection of which schema is selected when multiple schemas are applicable. Both these forms of variability can cause various sequence errors.

The SRN model has a set of input units that are activated by features of the environment. Activation is passed along the input units to a set of hidden units, which receive recirculated activation. The hidden units then pass activation to a set of output units that then perform an action (fixating an object, pouring an object, etc.). The connection weights encode series of sequential attractors which the trained model tends to follow (Cooper & Shallice, 2006). Errors in the SRN model are made by increasing the noise, which in turn can cause the network to drift to a related task sequence (i.e., a sequential attractor) whose internal representation resembles the next step. Thus, an error is made by the SRN model not because an attentional operation has been omitted, but because the model's internal representations have resulted in a loss of information about a previous or current state (Botvinick & Plaut, 2004; Botvinick & Bylsma, 2005).

The Memory for Goals model

A different model of routine behavior is the memory for goals model (MFG) which is an activation-based model that has been used in the study of interruptions and goal-related tasks (Altmann & Trafton, 2002; Trafton, Altmann, Brock, & Mintz, 2003; Altmann & Trafton, 2007).

The MFG is based on the hypothetical construct of activation of memory items—in particular, activation as construed in the ACT-R (Adaptive Control of Thought-Rational) cognitive theory (Anderson & Lebiere, 1998). A basic processing assumption in this theory is that when central cognition queries memory, memory returns the item that is most active at that instant. Activation thus represents relevance to the current situation. To capture the relevance of any particular item, the memory system computes that item's activation from both the item's history of use and from its associations to cues in the current mental or environmental context. In Bayesian terms, the logic is that history of use and current context together serve to predict the current relevance of that item (Anderson, 1990). In functional terms, the implication is that the cognitive system should be able to exploit the predictive computations of the memory system to overcome decay and keep certain information active for use in the future.

Two main constraints determine goal activation: strengthening and priming. The strengthening constraint suggests that the history of a goal (i.e. frequency and recency of sampling) will impact goal activation such that a subgoal that is retrieved more often or the most recently retrieved subgoal will have a higher activation value than others with less history. The priming constraint suggests that associated cues in the mental or environmental context can provide activation to a pending goal. For example, particular information in a task interface may prime a subgoal, allowing the subgoal to be retrieved over competing subgoals. In addition, each procedural step is associatively linked to the next step within the task hierarchy; thus, previously completed tasks are a source of associative activation (Altmann & Trafton, 2007).

The model incorporates the assumption that cognitive control is mediated at a fine-grained by episodic codes passed between different processes (Altmann & Gray, 2008). Applied to cognitive control here, in the context of routine sequential behavior, the assumption is that action preparation and action execution are separate processes, with the first retrieving a procedural step from semantic memory, then communicating with the second by creating an episodic code that represents the retrieved task. The communication between these two processes can be disrupted if some other cognitive operation (e.g., an interruption) occurs after the first process has executed but before the second has started.

All three models have different process explanations and capabilities for accounting for sequence errors.

Perseveration Errors

The IAN model does occasionally repeat steps, resulting in a continuous perseveration error. This occurs when, due to too much self-activation or lack of inhibition, a schema is not deselected at the appropriate time, causing a schema to be repeatedly selected. The IAN model can not, however, account for recurrent perseveration errors because once a goal is completed it is "ticked off" and not applicable for later selection (Cooper & Shallice, 2000; Botvinick & Plaut, 2004).

The SRN model does make both continuous and recurrent perseveration errors. However, one interesting aspect of the original SRN model was that virtually all errors were due to capture errors but had different manifestations. For example, with a small amount of noise, the network would occasionally drift to a similar sequential attractor (a capture process) and repeat a step (a perseveration error) (Botvinick & Plaut, 2004; Cooper & Shallice, 2006). While it is interesting that the SRN model can elicit so many error types, the capture process that causes those errors to occur has been questioned by some (Cooper & Shallice, 2006).

The MFG model can make both types of perseveration errors, though the process explanation is the same for both continuous and recurrent perseveration. The reason that MFG makes perseveration errors rests primarily on the interference level. Perseveration errors may occur when the wrong subgoal is retrieved to direct behavior. Occasionally, the difference in activation levels between previous subgoals and the target subgoal may be quite small and noise in the cognitive system may result in the retrieval of an incorrect subgoal. The constraints of the memory for goals theory suggest that when an incorrect subgoal is retrieved, it should be in close temporal proximity to the target subgoal. Recency suggests that the subgoal just completed will have a relatively high activation level and associative activation from the most recently retrieved subgoal will provide activation to neighboring subgoals. Occasionally, then, the cognitive system may retrieve the wrong subgoal to direct behavior. This will occur especially when there are relatively few environmental cues so that priming has less of an impact. Interestingly, the MFG model predicts that errors should be proximate to the next correct action. Not only should the most common error action be to retrieve the subgoal just completed, other error actions should be to subgoals that are temporally close to the next correct action. Recency suggests that the last few steps prior to the next correct action will have relatively high activation levels. The farther away the subgoal is from the correct action, the less likely this step should be retrieved. Thus, the general prediction is that when perseveration errors are made, most of the error actions should be localized to within a few steps of the correct action in a graded fashion.

Anticipation and Omission Errors

The IAN model also makes anticipation and omission errors. Omission errors could occur because a schema may not have a high enough activation due to low self-activation or poor environmental influences. Anticipation errors occur for a similar reason, but are not able to be executed because a precondition was not satisfied (e.g., a container still has its top attached).

The SRN model occasionally makes anticipation and omission errors, primarily through the capture process described before.

The MFG model also suggests that anticipation and omission errors will occur. In fact, MFG suggests that there are two possible explanations for skipping a goal. First, the primed retrieval component of the theory suggests that future steps receive activation in a decreasing graded fashion (Altmann & Trafton, 2007). Second, the model suggests that action preparation and action execution are separate processes. If communication between these two stages gets disrupted, an anticipatory error may occur. Because the primed retrieval model is not yet implemented in ACT-R, the separate-stages explanation will be focused on in the remainder of this report.

While all three models can account for the majority of error types, neither IAN nor SRN makes strong predictions about which types of errors should be more prevalent in this type of task. MFG, however, makes a strong prediction that perseveration errors should occur more often than any other type of sequence error. Additionally, MFG makes a nuanced prediction that errors should be proximate and graded from the correct step, especially with respect to perseveration errors.

Experiment

There are very few datasets that can be used to constrain or reject different models (Botvinick & Plaut, 2006). One of the issues is that the when a task is routine, people generally make very few errors, making statistical analysis difficult. Thus, different researchers have examined errors in non-routine tasks (Ruh, Cooper, & Mareschal, 2005), made the task difficult to remember (Giovannetti, Schwartz, & Buxbaum, 2007; Ruh, Cooper, & Mareschal, 2008) or interrupted participants during the routine task (Botvinick & Bylsma, 2005). We used an interruption paradigm because interruptions have been shown to increase error rates even on well-learned tasks (Li, Blandford, Cairns, & Young, 2008; Ratwani, McCurry, & Trafton, 2008). In addition, we provided no global placekeeping (Gray, 2000) such that the next step of the task could not be determined from visible cues.

Method

Participants. Fifteen George Mason University students participated for course credit.

Task and Materials. The primary task was a complex production task called the sea vessel task (based on Li et al., 2008; Ratwani et al. 2008). The goal was to fill an order for two different types of sea vessels by entering in order details through various widgets on the interface (Figure 1). Order information was provided in the middle of the screen on the "Navy Manifest." A correct sequence of actions is required to complete the order: (1) Enter Vessel Information, (2) Material, (3) Paint, (4) Weapons, and (5) Location. Before entering information into each widget, the widget must be "activated" by clicking the corresponding selector button (lower right hand corner of Figure 1). The procedure was arbitrary, but participants had no trouble learning it because (1) the information that was needed to fill in the widgets was available on the Navy Manifest; and (2) the order of the widgets was straightforward to remember due to a simple spatial rule, which we provided to participants.

After completing each widget, the participant must click "ok" and the information that was entered in the fields is no longer visible. This information was cleared from the fields because it may have served as an explicit cue indicating which steps in the task hierarchy have been completed. After entering information in each of the five widgets, the order must be processed by clicking the "Process" button. Once this button is clicked, a small pop-up window appears informing the participant of the total number of sea vessels that have been created. This pop-up window served as a false completion signal (Reason, 1990). Participants must click the "ok" button to acknowledge this window. Finally the "Complete Contract" button must be clicked to finish the order. The "Next Order" button is clicked to bring up a new order. Any deviation from this procedure was recorded as an error; any time an error was made, the computer emitted a brief auditory tone to alert the participant that an error was made. When a participant committed an error the participant had to continue with the task until the correct action was made.

The interrupting task required participants to answer addition problems with four single digit addends.

Design and Procedure. Each order on the sea vessel task constituted a single trial; participants performed twelve trials. Control and interruption trials were manipulated in a within-participants design; half of the trials were control with no interruption and half were interruption trials with two interruptions each. The order of trials was randomly generated. There were six predefined interruption points in the sea vessel task. There was a potential interruption point after clicking "ok" in each of the five widgets. The sixth interruption point was after the "Process" button was clicked. During the experiment there were a total of 12 interruptions (6 interruption trials x 2 interruptions in each trial); each lasting 15 seconds. Participants were instructed to answer as many addition problems as possible in this time interval. The interruptions were equally distributed among the six interruption locations. When returning to the primary task after the interruption, there were no visual cues on the task interface indicating where to resume (i.e. no global place keeping).

Before beginning the experiment, participants were given instructions about the two tasks they were going to have to perform and completed two trials as part of training; one had no interruptions and one had two interruptions. All participants were proficient at the task before beginning the actual experiment. The experiment was self-paced. A break was offered after six trials.

Figure 1: Screenshot of the ship production task

Description of Errors. Perseveration errors were any actions that repeated an action that had already been accomplished for that trial. Anticipation and omission errors were any actions that skipped one or more steps.

Because it was not possible to actually omit a step, all skipped steps were categorized as anticipation errors. Errors where participants failed to activate a particular module before working on the module (e.g. device initialization errors (Cox & Young, 2000)) were not analyzed.

Measures. Error rates were calculated for control and interruption trials by calculating percentages (actual errors/total error opportunities). Multiple incorrect actions in a sequence were counted as a single error for the purposes of calculating error rates. Error actions that occurred less than 500 ms from the previous action were excluded from all analyses as they were taken to be inadvertent mouse clicks; this accounted for less than one percent of the data.

Results and Discussion

Comparing Error Rates. Of the fifteen participants, eleven participants made at least one perseveration or anticipation error. Error rates were compared between the control trials and actions immediately after the interruption using a repeated measures ANOVA. Participants made more errors following an interruption (M = 9.3%) compared to the control (M = .9%), F (1, 14) = 5.8, MSE = 91.9, p<.05. Participants rarely made errors in the control trials, suggesting the task was well learned. The non-zero error rate on control trials also matches studies showing that people do make errors on well-learned tasks (Reason, 1990).

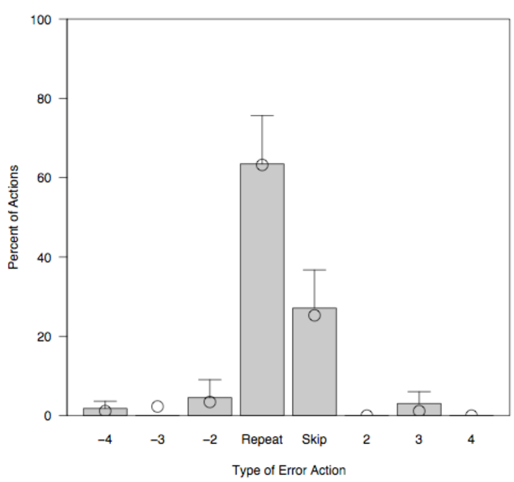

Figure 2: Distribution of errors during a sequential action task. Bars are empirical data; circles are model fits.

Pattern of Error Actions. Next, we focused on the pattern of error actions. In order to compare error actions at different points in the task hierarchy, the error actions were coded relative to the correct action at that point in the task hierarchy. Recall that the correct order of actions was Vessel, Material, Paint Scheme, Weapons, Location, Process and Complete Contract. If the next correct action is to work on the "Weapons" subtask and the participant made the error of working on the "Paint" subtask, this error action was coded as a "-1". If instead the participant clicks the "Process" button this was coded as a "2". Based on this coding scheme, a "-1" represents a repeat of the just completed action and a "1" represents skipping the next correct action. All errors were coded using this scheme.

The distribution of error actions is illustrated in Figure 2. A visual inspection of this graph suggests that both perseveration and anticipation errors occur relatively frequently. Additionally, the number of errors seems to be proximate to the next correct action in both directions, though this effect is not strong in this dataset. To determine whether the error action of retrieving the subgoal just completed and performing this action again was the most common error action, a repeated measures ANOVA was conducted to compare error actions at this position to all other error actions. There was a significant difference among the different error positions, F(7,70) = 12.8, MSE = 434.2, p<.0001. Tukey HSD post-hoc comparisons revealed that participants were significantly more likely to repeat the subtask just completed (M = 63.5%) than to make any other action (all p's<.05).

Model Description

An MFG model was written in the ACT-R cognitive architecture.

High Level Description of the MFG model

There are five model components that are critical for routine sequential skill and errors that occur during execution of a routine task: the need for well-learned knowledge; the encoding of an episodic trace; the strengthening constraint, the priming constraint, and the interference level.

Well-Learned knowledge There are several ways to represent well-learned knowledge in ACT-R. We provided the model with declarative knowledge about the task such that it always knew the sequence of steps it should follow.

Encoding of an episodic memory When the model knows which step it should perform, it encodes an episodic memory. A separate ACT-R module (goal-style), called episodic was created for this purpose. An episodic memory in this task is an extremely lean memory item that contains the current goal and a unique identifier. This unique code helps differentiate an episodic memory from a semantic one. All episodic memory items are created with a slightly higher initial activation so that they can be retrieved later. This mechanism is very similar to other models (Altmann & Trafton, 2002; Altmann & Gray, 2008); we propose that people encode and retrieve episodic memories during interactive routines. This episodic trace is later retrieved to guide action; retrieval is biased by the strengthening constraint, the priming constraint, and the interference level.

Strengthening constraint Which episodic memory element is retrieved depends in part on the strengthening constraint. The strengthening constraint suggests that the most recent episode will have the highest activation.

Priming Constraint When the model attempts to retrieve an episodic memory element, activation spreads from the focus of attention to related elements, of which the relevant episodic memory element is one. Thus, the mental context provides context to facilitate the retrieval of the correct episodic trace. The environmental context could also provide priming, but that aspect is not implemented in the current model.

Interference Level When the model attempts to retrieve an episodic trace (or any other memory item, for that matter), there is interference from other similar memory items. Interference occurs because a memory request is made that does not contain a perfect cue for retrieval. Since there may be several items that match the memory request, the system retrieves the most active memory element. Transient noise (sampled from a zero-mean logistic density function) can cause older elements to be retrieved. Thus, interference can lead to retrieval of an incorrect episodic memory item.

For all models, we kept most of the ACT-R parameter defaults. Specifically, we enabled several parameters with typical ACT-R values, including the maximum associative strength parameter which is priming (from nil to a typical value of 3), activation noise (from nil to .03), and the randomize-time parameter, which allows some perceptual and motor actions to have a small amount of variability in their timings (we kept the default value of 3). The base level learning parameter was set at the default of .5.

A sample experimental model run

To provide a match to the experimental procedure, 15 models (15 participants) were run. An abstracted interface was used for model runs. The model did not perform the post-completion step (Byrne & Bovair, 1997).

Normal processing The first thing that the model does in an experimental trial is to prepare to make a step. In order to do this, it retrieves from declarative memory the first step to perform (well-learned knowledge). Next, the model encodes an episodic memory of that step (encoding of the episodic memory). This retrieval and encoding is the preparation component of the model. Next, the model must execute the action. The execution component of the model begins with an immediate attempt to retrieve that episodic memory. Because the current mental context primes the episodic memory (priming constraint) and it is the most recent (strengthening constraint), the correct episodic memory is highly likely to be retrieved. After retrieving an episodic memory, that action is executed, the next step in the procedure is retrieved (well-learned knowledge), and the whole process repeats. Note that as the model completes one action, it starts to prepare for and encode the next step. This interleaving of motor and mental actions has been shown to occur in a variety of tasks and contexts (Salvucci & Taatgen, 2008).

Interruption processing When the model notices there was a screen change, it starts working on the interruption. The interruption effectively clears out all state information from the primary task. According to the model, the two most important aspects of the interruption are that (1) state information from the primary task is cleared and (2) decay occurs during the interruption. In the current model, only cursory model processing occurs during the interruption and all state information (e.g., focus-of-attention and problem representation) is cleared.

Resumption processing After an interruption completes, the model notices the screen change and attempts to remember the last task-relevant episodic memory. If it is unable to recall an episodic item, the model executes a random action. This rarely happens in the current model, given the brief interruption duration. If the model is able to retrieve an episodic memory, it assumes that the retrieved element was the last completed action so retrieves the next step and continues in the task.

Error behavior Most of the time, due to the strengthening and priming constraints, the correct episodic memory is retrieved and the procedural task is executed flawlessly. During normal execution, however, the model will rarely (when transient noise of an older episodic trace is greater than strengthening and priming), retrieve an incorrect episodic trace (interference level). When an error is made, the model suggests that the most likely memory element to be retrieved will be the one with the next highest activation.

The model makes perseveration errors because the episode that was just completed is likely to have a relatively high activation. Thus, the model makes perseveration errors in a graded fashion away from the correct action.

The model makes anticipation errors because sometimes the model pre-encodes a particular episodic action before it gets completed (e.g., it encoded an episode but got interrupted before it could complete that action). When this pre-encoding / interruption occurs, the episodic element with the highest activation is likely to be the next (uncompleted) action upon resumption and therefore selected, leading to an anticipation error. Note that when the model makes an anticipation error, it is a simple skipped step and can not skip more than one step.

As in the empirical data the model very rarely makes an error during non-interrupted trials. These errors occur because the wrong episodic memory was retrieved: noise in the interference level overcomes the strengthening and priming constraints of the correct episode.

The role of noise Greater noise in the system increases the number of errors the system makes because there is a greater probability that a different episodic memory will have a higher activation than the correct one. Additionally, a greater noise increases the "spread" of applicable episodes. So, increasing noise increases both the number and spread of errors.

Model fit

As is evident in Figure 1, the model matches the data quite well; $R^2 = .99$ and RMSD = 1.3.

General Discussion

The current paper presents an experiment and model of sequential actions. The experiment used an interruption paradigm, increasing the rate of errors enough to see emergent patterns from the data. The model used a memory for goals model that describes the process people go through both during error-free behavior and when they make errors. In general, errors occurred because the wrong episodic memory was retrieved. Perseveration errors occurred because a recent episodic memory had a high enough activation that, with noise, it was retrieved instead of the correct memory. Anticipation errors occurred because the communication between the preparation and execution of an action gets disrupted for some reason.

The MFG model shares both similarities and differences to the other two models of sequential routine action, IAN and SRN. MFG focuses on perceptual and memorial processes rather than schemas (IAN) or distributed representations (SRN). However, it is interesting that all three models use noise as one of the primary explanatory constructs for why errors are made.

The current MFG model does have several limitations. First, it only accounts for sequence errors; it does not account for intrusions, capture errors, etc. Second, while both IAN and SRN attempt to model both normal and patient populations, the MFG model only addresses normally functioning individuals. Third, the model-task is quite simple, and a more complete task description is needed to expand the coverage of this model. Finally, the MFG model does not model the learning of the task itself.

The experiment reported here and the MFG model itself do, however, have several strengths. First, the experimental paradigm used here allows errors to be studied in the lab with normal populations. This data and other like it should be able to constrain current models of sequential actions, as Botvinick and Plaut (2006) suggest. Second, the MFG makes both qualitative and quantitative predictions about the error pattern for this task. Both the IAN and SRN models have been critiqued for the way they make perseveration errors. Finally, the model makes episodic memory an aspect of its normal processing, so errors arise out of normal processing of routine behavior.

Acknowledgments

This work was supported by ONR under funding document N0001409WX20173 to JGT and N000140310063 to EMA. The views and conclusions contained in this document should not be interpreted as necessarily representing the official policies of the U. S. Navy.

References

Altmann, E. M., & Trafton, J. G. (2002). Memory for goals: An activation-based model. *Cognitive Science*, *26*(1), 39-83.

Altmann, E. M., & Trafton, J. G. (2007). Timecourse of Recovery from Task Interruption: Data and a Model. *Psychonomic Bulletin & Review*, *14*(6), 1079-1084.

Altmann, E. M., & Gray, W. D. (2008). An Integrated Model of Cognitive Control in Task Switching. *Psychological Review*, *115*(3), 602-639.

Anderson, J. R. (1990). *The adaptive character of thought.* Hillsdale, NJ: Erlbaum.

Anderson, J. R., & Lebiere, C. (1998). *Atomic components of thought.* Mahwah, NJ: Erlbaum.

Baars, B. J. (1992). *Experimental slips and human error: Exploring the architecture of volition.*

Botvinick, M., & Plaut, D. C. (2004). Doing without schema hierarchies: A recurrent connectionist approach to routine sequential action. *Psychological Review*.

Botvinick, M. M., & Bylsma, L. M. (2005). Distraction and action slips in an everyday task: Evidence for a dynamic representation of task. *Psychonomic Bulletin and Review*.

Botvinick, M. M., & Plaut, D. C. (2006). Such stuff as habits are made on: A reply to Cooper and Shallice (2006). *Psychological Review*.

Byrne, M. D., & Bovair, S. (1997). A working memory model of a common procedural error. *Cognitive science*.

Cooper, R., & Shallice, T. (2000). Contention scheduling and the control of routine activities. *Cognitive Neuropsychology*.

Cooper, R. P., & Shallice, T. (2006). Hierarchical schemas and goals in the control of sequential behavior. *Psychological review*.

Giovannetti, T., Schwartz, M., & Buxbaum, L. (2007). The Coffee Challenge: A new method for the study of everyday action errors. *Journal of Clinical and Experimental Neuropsychology*, *29*(7).

Gray, W. D. (2000). The nature and processing of errors in interactive behavior. *Cognitive Science*, *24*(2), 205-248.

Li, S. Y. W., Blandford, A., Cairns, P., & Young, R. M. (2008). The effect of interruptions on postcompletion and other procedural errors: An account based on the activation-based goal memory model. *Journal of Experimental Psychology: Applied*, *14*(4), 314-328.

Norman, D. A. (1981). Categorization of action slips. *Psychological review*.

Norman, D. A., & Shallice, T. (1986). *Attention to action: willed and automatic control of behaviour*. Paper presented at the Consciousness and Self-Regulation: Advances in research and theory, New York.

Ratwani, R. M., McCurry, J. M., & Trafton, J. G. (2008). Predicting postcompletion errors using eye movements. *CHI*.

Reason, J. (1984). Absent-mindedness and Cognitive ControI. *Everyday Memory Actions and Absent-mindedness: Actions and.*

Reason, J. (1990). Human Error. *Cambridge University Press*.

Ruh, N., Cooper, R. P., & Mareschal, D. (2005). The time course of routine action. *Cognitive Science 2005*.

Ruh, N., Cooper, R. P., & Mareschal, D. (2008). The Hierarchies and Systems that Underlie Routine Behavior: Evidence from an Experiment in Virtual Gardening. *Cognitive Science 2008*.

Salvucci, D. D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*.

Sandson, J., & Albert, M. L. (1984). Varieties of perseveration. *Neuropsychologia*.

Schwartz, M. F., Montgomery, M. W., Buxbaum, L. J., Lee, S. S., TG,. (1998). Naturalistic action impairment in closed head injury. *Neuropsychology*.

Trafton, J. G., Altmann, E. M., Brock, D. P., & Mintz, F. E. (2003). Preparing to resume an interrupted task: Effects of prospective goal encoding and retrospective rehearsal. *International Journal of Human Computer Studies*, *58*(5), 583-603.