Human Error as an Emergent Property of Action Selection and Task Place-Holding

Franklin P. Tamborello, II, Cogscent, LLC, Houston, TX, and
J. Gregory Trafton, U. S. Naval Research Laboratory, Washington, DC

Objective: A computational process model could explain how the dynamic interaction of human cognitive mechanisms produces each of multiple error types.

Background: With increasing capability and complexity of technological systems, the potential severity of consequences of human error is magnified. Interruption greatly increases people's error rates, as does the presence of other information to maintain in an active state.

Method: The model executed as a software-instantiated Monte Carlo simulation. It drew on theoretical constructs such as associative spreading activation for prospective memory, explicit rehearsal strategies as a deliberate cognitive operation to aid retrospective memory, and decay.

Results: The model replicated the 30% effect of interruptions on postcompletion error in Ratwani and Trafton's Stock Trader task, the 45% interaction effect on postcompletion error of working memory capacity and working memory load from Byrne and Bovair's Phaser Task, as well as the 5% perseveration and 3% omission effects of interruption from the UNRAVEL Task.

Conclusion: Error classes including perseveration, omission, and postcompletion error fall naturally out of the theory.

Application: The model explains post-interruption error in terms of task state representation and priming for recall of subsequent steps. Its performance suggests that task environments providing more cues to current task state will mitigate error caused by interruption. For example, interfaces could provide labeled progress indicators or facilities for operators to quickly write notes about their task states when interrupted.

Keywords: computational modeling, human error analysis, cognitive modeling, human performance modeling, human systems integration

Address correspondence to Franklin P. Tamborello, II, PhD, CSP, Cogscent, LLC, 4402 Cheena Dr., Houston, TX 77096, USA; e-mail: frank.tamborello@cogscent.com.

HUMAN FACTORS
Vol. 59, No. 3, May 2017, pp. 377–392
DOI: 10.1177/0018720816672529

INTRODUCTION

Error is common in everyday working life. Studying human error is important not only for what it reveals about the normal operation of cognitive mechanisms but also because with increasing capability of our technological systems (e.g., transportation, power generation), the amount of damage that can result from error is magnified. With increasing complexity of those systems, error, once committed, is often more difficult to diagnose and correct (Reason, 1990). Studying human error is difficult because of the variability of error behavior. Furthermore, error often arises from the dynamic interactions of several cognitive processes that normally perform very reliably.

The goal of this project is to understand the cognitive mechanisms that underlie our ability to perform sequential procedures and that also lead to certain error classes. To that end, we developed a computational model to account for multiple types of error using mechanisms presumed to operate during normal (non-error) cognition. We will present the model together with results of tests we conducted comparing model predictions to human data collected in several laboratory studies, one of which was performed specifically as part of this research and others of which were collected and published previously.

Theories of Action Selection and Error

A theory explaining error must also explain the correct action execution behavior that is much more common in the milieu of human task performance. Furthermore, as error by definition occurs in contexts in which people are trying to perform some other intended action, a theory of error behavior should explain error as something that arises out of the operation of the same mechanisms that enable correct action execution. What follows is a brief review of relevant theories.

Working memory capacity. Byrne and Bovair (1997) studied a particular error type termed *postcompletion error* (PCE), which is the act of neglecting to perform some "clean-up" step after having accomplished the main goal of a task. They explained PCE as a function of a theoretical working memory construct. Here, working memory holds an active mental representation of the task and its current state. This representation is important for mental operations, such as recall of what action to perform next. Working memory has a limited capacity to hold these representations, and this capacity varies between people.

Byrne and Bovair studied working memory load as well as individuals' differences of working memory capacities. Their Collaborative Activation-based Production System (CAPS) model (Just & Carpenter, 1992) assumed a hierarchical goal representational structure. It propagated activation necessary for retrieval of step representations from the task supergoal, held in working memory, to retrieve subgoals. Subgoals must maintain their activations above a certain threshold for them to remain accessible. Crucially, the main goal of the procedure would be satisfied before it was time to perform the postcompletion step. The presence of other information to maintain in an active state, in this case a three-back memory task, taxed the system to capacity such that it failed to maintain the postcompletion subgoal.

Memory for goals. Another account of systematic error, Memory for Goals (Altmann & Trafton, 2002), posits that we encode episodic traces of our goals as we complete tasks. Each goal is encapsulated in an episodic memory, which sparsely represents a behavioral context at the time of its encoding. The strength of these memories decays over time such that it may be difficult to remember a past task context. Memory for Goals provides a process-level theory for why certain types of errors are made during a well-learned task as a consequence of retrospective, episodic memory (Altmann & Trafton, 2007; Ratwani & Trafton, 2010; Trafton, Altmann, & Ratwani, 2009).

The Remember-Advance Model. Altmann and Trafton (2015) developed a formal model of Altman, Trafton, and Hambrick's (2014) UNRAVEL sequence task, describing it as a two-phase retrieval process. The model carried over no task context from step to step in any sort of buffers or working memory. Instead, at the beginning of each step it retrieved an episodic encoding of the last action it performed. It then used that memory as the cue for an associative retrieval from long-term memory of the action to perform for the current step of the task. Perseverations, inappropriate repetition of an action, occurred due to interference in the retrieval of the episodic codes during the first retrieval phase. Omissions, skipping a step, were a consequence of associative interference during the action selection process.

Interruptions

Being interrupted increases people's error rates by 5% to 50% (Monk, Trafton, & Boehm-Davis, 2008; Trafton et al., 2011). After an interruption, people will frequently perseverate, or they may omit a step. Sometimes these errors are irritating (e.g., ruining a meal by leaving out a crucial ingredient), but sometimes they can have disastrous consequences (e.g., administering medicine twice or not configuring airplane flaps for takeoff). For these reasons, we find the interruption paradigm to be both useful for eliciting error behavior from subjects in empirical studies as well as an important topic of study in its own right.

REMEMBER-ADVANCE PROCESS MODEL

We developed our computational process model using the ACT-R 6 cognitive architecture (Anderson, 2007; Anderson et al., 2004), and all source code is available for download from https://github.com/tamborello/postcompletion-error and https://github.com/tamborello/UNRAVEL, as well as in the online supplementary material. ACT-R represents a claim that cognition is modular. Each module can operate independently. A module may have one or more buffers with which it may communicate with the other modules. The procedural module is a rules-based execution system that matches "if-then" rules to conditions presented by the set of buffers and their contents and then executes the rules' actions. These actions are typically to move information between buffers, which may result in their respective modules' performing some action, such as "retrieve from long-term memory the sum of three and four" or "move the mouse cursor to the location where I'm currently looking." We implemented our model using a cognitive architecture because we wanted to use the decades of research underlying the architecture to constrain the choices we made in model development. All of the model mechanisms are ACT-R's mechanisms, which have the support of vigorous testing from an active community of researchers (ACT-R Research Group, 2013).

Our model predicts error to manifest as different types according to whether it occurs during an action selection process or a deliberate rehearsal process. During the action selection process, the model uses a set of limited-capacity buffers containing moment-to-moment goal and problem state (or "imaginal") representations to spread retrieval activation to long-term memory. Retrieval activation varies according to strength of associative priming (Anderson et al., 2004). Since all memory activations decay, the model can engage a deliberate rehearsal process to increase probability of future retrospective retrieval of the rehearsed memory. Here, decay serves to suppress the activation of older memories so that they are less likely to interfere with newer ones (Altmann, 2002). Table 1 enumerates the major ACT-R theoretical constructs used by this model, their operation within ACT-R, and their implications for this model.

Normal Task Execution

Table 1 described the pieces of the model and theoretical motivation for borrowing them from ACT-R. The following sections explain how those pieces work together to produce correct and error behaviors.

Selecting the next step. We conceptualized action selection as a prospective memory task, using a representation of the current task context to associatively prime retrieval of a memory representation of the next step. These task context representations are simply the goal memories that currently reside in the active buffer contents. Memory activation spread from the model's active buffer contents to memories residing in long-term memory, and it did so as a function of the strength of association between an item j in a buffer to memory i in long-term memory (Anderson, 2007; Anderson et al., 2004). We assumed these patterns of association strength were learned during training and followed the sequential co-occurrences of the actions in the task environment that these memories represented (Botvinick & Plaut, 2004). This gave the model a way to adjust its behavior according to context.

At the beginning of each simulation run, our model set strengths of association from each step's representation to the next according to $m \div (i - j)$. Association strengths remained static for the duration of each model run. Here, j is the serial position within the task of the step encoded by a memory representing the model's current context, namely, the action it just performed. Item i is the serial position within the task of the step encoded by an associated memory for an action in long-term memory. The scalar m is a global parameter to set the maximum association strength. For example, if m were 5 and the model had just performed the first step of a task, the association strength to the memory encoding the second step would be 5. The strength of association to the third step would be 2.5. This enabled associative chaining from the model's current context to the next procedure step while also producing a graded representation that decreased in strength with increasing psychological distance, a feature borrowed from Altmann and Trafton (2007). As will be shown in the error behavior section, this graded representation is a crucial feature of this model.

People tend to remember their actions because we typically form episodic memories of events we experience. Each time the model performed the action specified by the memory it had retrieved, it copied a reference to that action from the active buffer contents to an episodic buffer (Altmann & Trafton, 2002). The contents of that episodic buffer, once established, were then removed from the buffer and relegated to long-term memory. As such, the model generated a sequence of episodic memories epiphenomenal to the process of task execution, with sequence retroactively indicated by relative activation strength of the episodic memories, the most recent being the most highly active.

Interruption and Resumption

When the model was interrupted, it immediately tried to retrieve the last action it executed, which was encoded in one of these episodic

TABLE 1: ACT-R Theoretical Constructs and Their Implications for the Remember-Advance Process Model

ACT-R Construct	Operation in ACT-R	Model Implications
Chunk	A single atomic unit of representation, such as a memory or percept. One chunk may and often does refer to a handful of other chunks.	Task state information, such as current position within the sequence of a task, is encoded in chunks, one task step per chunk.
Spreading activation	Primes retrieval from long-term memory.	A chunk held within certain buffers will propagate activation to other chunks in long-term memory.
Activation source	Spreading activation provided by a buffer, divides evenly among all the chunks referenced from the buffer chunk.	The same buffers used to maintain a working, internal representation of the task state also prime retrieval of memories encoding subsequent states.
Buffer contents	The contents of active buffers form an internal representation of task state, percepts, and motor plans. Any chunks held there may be accessed by cognitive modules other than just the one to which that buffer belongs.	The imaginal buffer contains a single chunk encoding the current task step. When it is time to remember the next step, all of the imaginal buffer's activation source propagates to chunks in long-term memory according to strength of association. The goal buffer also contains a chunk encoding the current task step, and so it also contributes some spreading activation. However, because the goal buffer also contains some intra-step control information (e.g., "retrieve the next step" or "respond"), the goal buffer only provides half as much spreading activation as does the imaginal buffer.
Association strength	Association strength encodes how strongly one chunk may prime retrieval of another chunk. Chunks only prime retrieval when they are in a buffer that provides retrieval activation source.	One chunk encoding a task step associates strongly to the chunk encoding the next task step. Thus, when such a chunk is retrieved from long-term memory and then copied to the imaginal and goal buffers, it associatively primes retrieval of the chunk encoding the subsequent task step.
Base-level activation (BLA)	BLA provides an estimate of future need of a chunk based on past use. Chunks in long-term memory retrieved more frequently or more recently are more highly active.	Rehearsal, which the model performs during interruption, keeps the rehearsed chunk active by periodically retrieving it while allowing other similar, competing chunks' activations to decay.

(continued)

TABLE 1: (continued)

ACT-R Construct	Operation in ACT-R	Model Implications
Retrieval count	Retrieving a chunk increases its BLA.	When rehearsing, the model implements a strategy of maintaining retrieval availability of a memory by increasing the retrieval count of a memory.
Decay	Time elapsing since last retrieval decreases BLA. Decay is a power law function of time.	Besides increasing retrieval count, rehearsal works against decay of the rehearsed chunk's BLA.
Activation noise	Retrieval from long-term memory is a stochastic process. ACT-R adds transient activation noise, with a mean of 0 and standard deviation set by ACT-R's ANS parameter (0.3 for this model), to every chunk's activation when it is evaluated for retrieval.	Activation noise underlies all errors, with the manifested error type a result of what kind of retrieval is attempted, for example, a large quantity of noise occurring during prospective retrieval may lead to prospective interference, which may lead to an omission error.
Condition-action matching	At its heart, ACT-R is a production rule system wherein a production rule matches to a condition consisting of the buffer contents (multiple buffers may match). One rule matches and fires at a time. The rule then specifies some action, such as copying an alphabet letter retrieved from memory to a manual buffer and requesting a move of the hand to that letter's position on a computer keyboard.	Production rules perform all the actions of the model, such as requesting long-term memory to retrieve a memory. They are the means by which the model transitions from one state to the next.

memories. Retrieval provided a boost of activation to the retrieved memory, making it more likely to be retrieved during a future retrieval attempt. Upon resumption, once the model had retrieved one of these episodic memories, it then used that episodic memory to bootstrap its task context representation. Altmann and Trafton (2007) demonstrated that this process occurred gradually, and so the model at first copied the retrieved episodic memory to only the imaginal buffer and not also to the goal buffer. The consequence is that there will be relatively less activation source to spread to long-term memory for retrieval of the next step, making retrieval take longer and be more likely to result in retrieval of an incorrect memory for the given task context.

Error Behavior

We followed ACT-R's architecture to develop our own theory of the causes of error during normal task execution. Errors arise out of the interaction of noise with the processes of normal task execution (Figure 1). Each of the two processes, action selection and deliberate rehearsal, function differently, and so the effects of their combinations with retrieval activation noise produce the two different sequence error types, omissions and perseverations. Note that

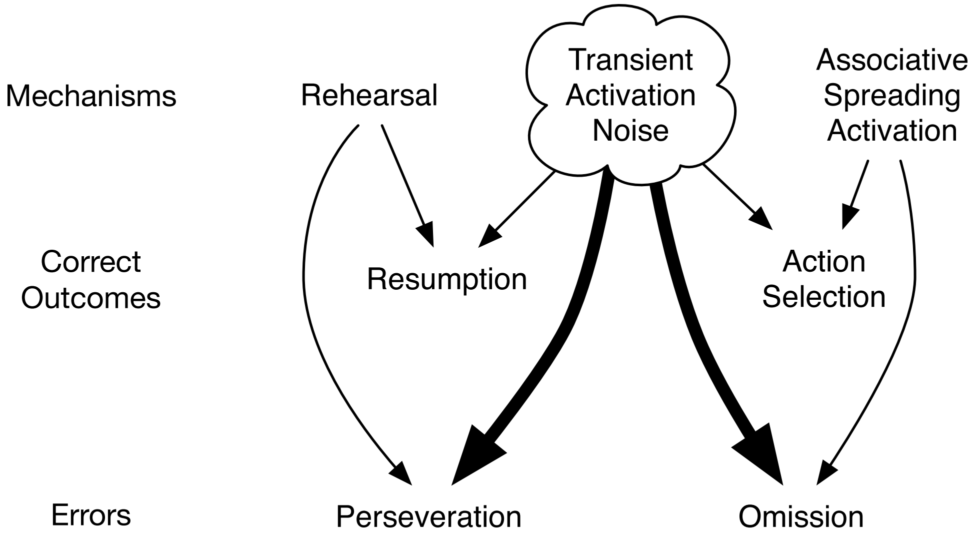

Figure 1. The role of noise in the model's memory processes: Associative spreading activation is the prospective memory process underlying selection of correct actions. When transient activation noise, a fundamental property of human memory, spikes during prospective retrieval, it can lead to an omission of one or more steps. The model implemented retrospective memory with an explicit rehearsal strategy that it threaded with the interrupting task. Spikes in transient activation noise during retrospective retrieval sometimes caused perseverations of actions from one or more steps previous.

the three modeled tasks made no affordance for any action that was not part of the experiment procedure. Therefore, any action performed was either the correct action or one of the procedure's actions out of sequence and so was by definition an omission or perseveration by one or more steps.

Omission. Because of the model's graded associative representation—namely, that sequence memory chunks each associated most strongly to their immediate successor but also less strongly to more distantly future successors—it occasionally omitted a step when transient activation noise was such that it simultaneously suppressed activation of the correct next step and enhanced activation of an immediately subsequent item. In conditions of normal task execution, the model occasionally (≈1%) omitted one or two steps.

As in Altmann and Trafton's (2007) work, the model gradually rebuilt its task context representation during the course of resuming its normal task execution cycle. Once it retrieved an episodic encoding of a past action, the model held task context representation in only one of two buffers that it normally used during task execution. It needed more time to build more task context representation, and meanwhile that representation was weak. For the model, this meant that it had less retrieval activation available to spread for its first action selection attempt after the interruption. With the proportion of activation provided by buffer contents smaller in this case while the amount of noise remained the same, the model was relatively more likely to retrieve the representation for an action that should come one or two more steps hence.

PCE. Some tasks, such as Byrne and Bovair's (1997) Phaser Task, and Ratwani and Trafton's (2011) Stock Trader Task, exhibit hierarchical goal structure. For instance, in the Phaser Task, procedure steps are physically arranged on the interface into functional groups, for example, actions having to do with a "charge phaser" subtask all took place within one cluster of interface elements. Subjects were trained to consider those steps as part of one cohesive subgoal, requiring completion in its entirety, as part of the main goal of firing the phaser. However, some subtasks were only a single step long. They were part of the main procedure but may have had no obvious connection to other steps. Such structurally isolated steps tend to exhibit much higher rates of error than other steps appearing within groups (Reason, 1990).

Postcompletion steps are both structurally isolated and tend to appear late in a procedure sequence. The model treats PCE as a special case of omission error. We assume that in tasks with a hierarchical goal structure, people retrieve a representation of the main task goal multiple times during the course of executing that task once. We adapted this assumption from the Anderson, Bothell, Lebiere, and Matessa (1998) model of sequence memory, although it is also congruent with Byrne and Bovair's (1997) model. As in rehearsal, each time a goal is retrieved, its activation is boosted. If such a memory's activation is already strengthened by repeated retrievals, and it also happens to be associated to the current task context (because it will appear again soon in the task sequence), then it has both some undecayed base-level activation and some associative spreading activation. These two sources of activation coming together in the one memory makes the model even more likely than in the case of typical omissions to retrieve the memory of the main goal rather than the memory of the postcompletion step. This is why postcompletion steps, when present, elicit greater rates of omissions than do other steps.

Perseveration. The episodic memory of the most recently performed step has the highest activation because it was referenced most recently. However, the next most recently referenced step still has a high, albeit less so, memory activation level. Noise can temporarily make the memory of the next most recently performed step more active than the memory of the most recently performed step. Typically, this happens at interruption onset, when the model begins its rehearsal, because by this time the episodic memories from recent previous trials have only decayed somewhat and so are still retrievable. Retrieval noise can temporarily increase the activation of an episodic memory from a previous but recent trial while simultaneously suppressing the activation of the episodic memory encoding the trial that was just completed. Then the model rehearses an incorrect but recent action, namely, from one or two steps back.

EMPIRICAL STUDIES

Our goal in selecting the following studies was to use well-documented, well-controlled studies of sequence errors to guide development of our model. We sought insight into perseverations, omissions, and PCE because they are common error classes and because we could find suitable human data corpori about them.

The Stock Trader Task

We used a version of Ratwani and Trafton's (2011) Stock Trader Task. It is a kind of computerized, interactive, single-page, form-filling task in which participants must follow a specific procedure (Figure 2).

Task and materials. The spatial layout of the interface (working from top to bottom down the left column and then the right column) and the operations required to perform the task were quite intuitive. The spatial layout of the task grouped steps by proximity. This encouraged use of an intuitive heuristic ("go down the column") as well as having an isolated "clean-up" step at the end. This format followed the form of other tasks shown by GOMS analysis to lead to subgoaling (e.g., Byrne & Bovair, 1997). After entering information in each module, the participant clicked the Complete Order button (upper right corner). Clicking the Complete Order button was the postcompletion step, and failing to click the Complete Order button constituted a PCE.

Design and procedure. Twenty-five George Mason University undergraduate students participated for course credit. Each order on the Stock Trader Task constituted a single trial. Control and interruption trials were manipulated in a within-participants design; participants performed 12 trials. Half of the trials were control trials with no interruption, and half were interruption trials with two interruptions each. The order of trials was randomly generated, and participants did not have prior knowledge as to which trials would be control or interruption trials.

There were eight possible interruption points in the Stock Trader Task. These points occurred

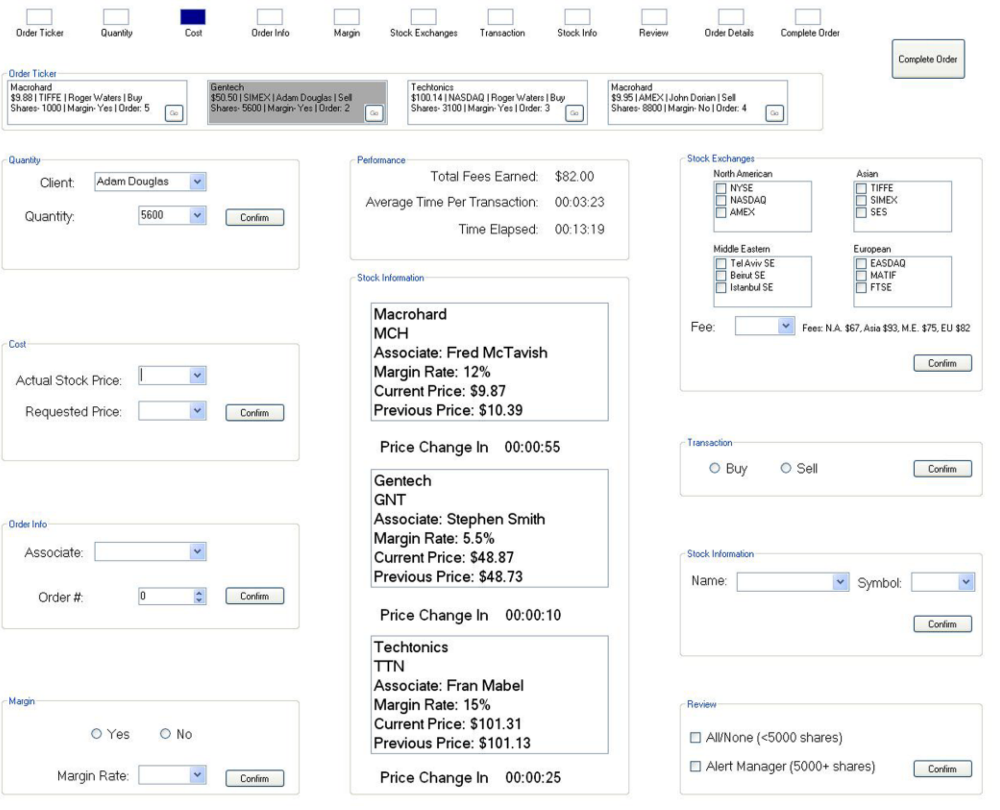

Figure 2. The Stock Trader Task interface resembled a web form.

after clicking the Confirm button following the first seven modules, including just prior to the postcompletion step. The location of the interruptions on a trial-by-trial basis was randomized with the constraint that exactly two interruptions occurred just prior to the postcompletion step and at least one interruption occurred at each of the other seven possible locations. There were 12 postcompletion error opportunities, 1 during each trial. Six of these opportunities were during control trials with no interruptions, 2 opportunities were immediately following an interruption, and 4 opportunities were during interruption trials where an interruption occurred at a point that did not immediately precede the postcompletion step.

After the experimenter explained the Stock Trader Task and interrupting task to the participant, the participant completed two training trials (one trial with and one trial without interruptions) with the experimenter. Following these two training trials, participants had to perform two consecutive randomly selected trials on their own without making a postcompletion error before the participant could begin the experiment. Forcing participants to perform two consecutive error-free trials ensured participant proficiency at the task before beginning the actual experiment. Each participant was instructed to work self-paced. When performing the interrupting task, participants were instructed to answer the addition problems as soon as the solution was known and to answer as many addition problems as possible in the time interval. Upon resumption of the Stock Trader Task, there was no information available on the interface to indicate where to resume.

Results. Interruption had a much larger effect on postcompletion error rates than on simple omission rates (Figure 3). Mean interruption PCE rate was 0.34 (i.e., subjects committed the PCE one-third of the time they encountered the PC step), while mean PCE rate in the control

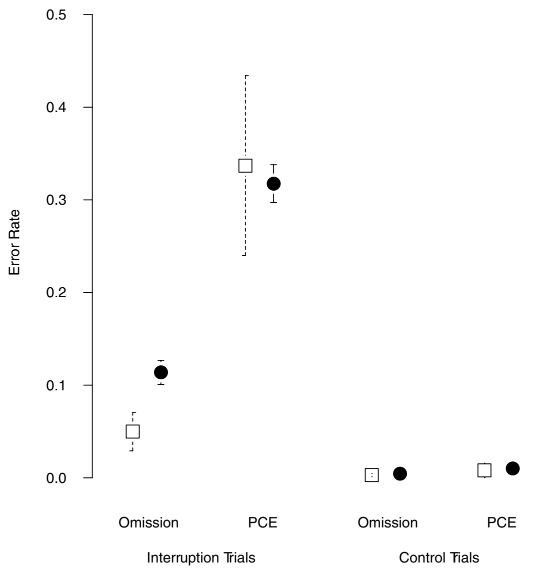

Figure 3. Mean error rates from the Stock Trader Task from human subjects (open squares) and model (solid circles). Error bars display the 95% confidence interval of the mean.

condition was only 0.01. Mean interruption omission was 0.05, and mean omission rate for the control condition was 0.003. In terms of effect size, Cohen's *d* for interruption's effect on the postcompletion step was 0.67, a large effect, compared to 0.20 for interruption's effect on non-postcompletion steps, a small effect size. Looking at it the other way, the effect of being interrupted was much more severe for subjects working on the postcompletion step (d = 0.54) than for subjects working on a non-postcompletion step (d = 0.05).

Model data collection and performance. Because our simulation was stochastic, we could not run it once to produce the true prediction of the theory that the simulation implements. But because we wished to treat the simulation as theory and the simulation's data as the theory's true predictions, we wanted to run the simulation enough times to produce a sample yielding stable predictions of performance for conditions of interest. To the extent that effect size and the number of model runs produced high statistical power, we could be confident that the data set produced by the model runs was usefully close to its true predictions (Ritter, Schoelles, Quigley, & Klein, 2011). For the effects we model, typically 1,000 runs more than suffices.

The mean PCE rate for the Stock Trader model's control condition was 0.0101 (*SD* = 0.0301, SEM = 0.0010) and 0.3175 (*SD* = 0.3302, *SEM* = 0.0104) for the interruption condition, for a difference of mean rates of 0.3074 (pooled *SD* = 0.2345), yielding an effect size of 0.9265. With effect size and *N* this large, power is effectively 1 for any sensible significance test (Howell, 2002). The model's rates of PCE and omission for control and interrupted trials closely matched those of participants, r = .976, root mean square deviation (RMSD) = .0334.

The model retrieved each subsequent step using the prospective memory process described previously. As described previously in the PCE section, functionally isolated steps like the postcompletion step both immediately followed and preceded retrieval of the task's main goal, and so such steps were subject to much greater degrees of retrieval interference. Furthermore, at resumption, the interference effect was exacerbated by the degraded context representation described in the previous "Omission" section.

PCE's distinction from simple omission is illustrated by comparison of their rates. If PCE were simply a matter of an omission error happening to fall at the last step, then PCE and omission rates should be identical. However, Figure 3 shows clearly that the two error types are different. What makes PCE unique is that it is a product of the simultaneous convergence of all of these factors: (1) goal activation decay below that of a competing goal's, (2) a competing goal's relatively high memory activation because it happens to have also been retrieved recently, (3) working memory structures with limited capacity to spread activation to long-term memory retrieval, and (4) some context representation was not immediately available upon resumption.

The Phaser Task

We applied our model to Byrne and Bovair's (1997) Phaser Task from their second experiment. Like the Stock Trader Task, the Phaser Task was designed to elicit postcompletion errors. Unlike the Stock Trader Task, it did so by

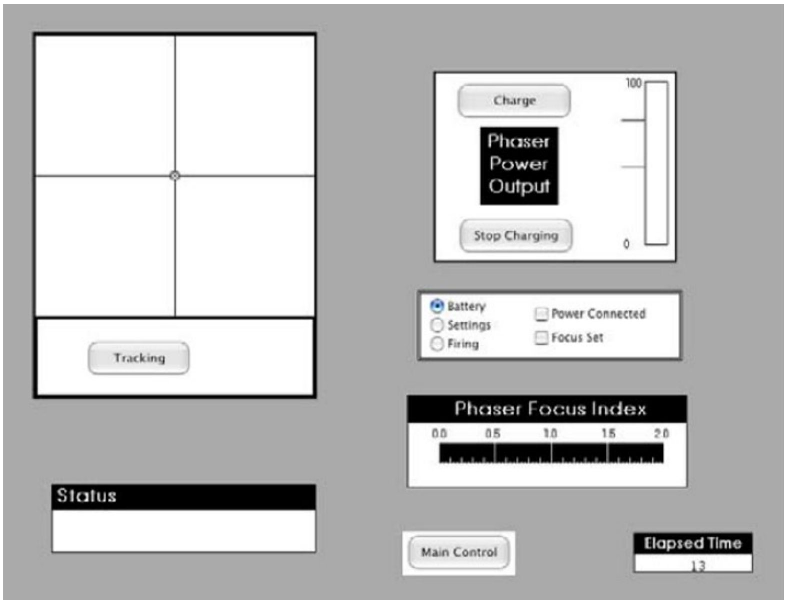

Figure 4. Example of Phaser Task interface from Byrne and Bovair (1997).

loading subjects' working memory rather than by interrupting subjects.

Task and materials. The authors of that study collected data from 64 undergraduates at the Georgia Institute of Technology. They used an interactive, single-page computer task that implemented a kind of video game procedure to arm and fire a starship phaser from the fictional *Star Trek* franchise (example depicted in Figure 4). As in the Stock Trader Task, participants strictly followed a procedure. The spatial layout of the task grouped steps by proximity. This encouraged use of an intuitive heuristic ("do all the items in the cluster") as well as having an isolated "clean-up" step at the end. Byrne and Bovair's own Goals, Operators, Methods, and Selection (GOMS) analysis of their Phaser Task resulted in a hierarchical task representation that they used in their CAPS cognitive model of the task.

Design and procedure. Participants learned to perform the Phaser Task during a training phase and then returned later to perform during a testing phase. During some test trials, subjects performed a concurrent three-back memory task intended to increase their workload. Furthermore, Byrne and Bovair (1997) administered a working memory span test. They used the results of this span test to group subjects into high and low working memory capacity groups, split on the median working memory span score.

Model data collection and performance. As for the Stock Trader Task, we executed the model simulation 1,000 times. For each working memory capacity condition, we took the difference of postcompletion error rate of the two load conditions. The difference of effect on model PCE rate of the capacity condition difference scores was 0.4339. This is a very large effect, 1.36 times the pooled standard deviation. And as in our modeling study of the Stock Trader Task, our power was effectively 1.

The model inherited a limited capacity working memory construct from ACT-R. This means that the model has a limited pool of memory activation to spread to retrieval from long-term memory. Occupying buffer space with the phaser's additional memory task or by adjusting a parameter related to individual differences in working memory capacity, source activation available to the imaginal buffer (0.8 for high capacity subject, 0.3 for low, fit empirically) had the same ultimate effect on the model's prospective retrievals as did the interruption-resumption process of the Stock Trader Task. It restricted the amount of retrieval spreading activation

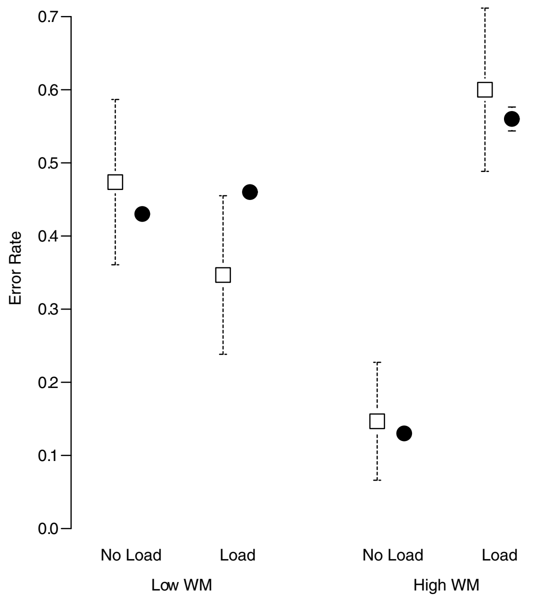

Figure 5. Mean postcompletion error (PCE) rates, human data from Byrne and Bovair's Experiment 2 Phaser Task (open squares) and model (solid circles), as a function of memory load and capacity. Error bars display the 95% confidence interval of the mean. Participants in the low working memory group did not reliably differ in PCE rate as a function of load condition (Byrne & Bovair, 1997).

available to the prospective retrieval process. This is why the model's PCE rate varied according to working memory load and working memory capacity, following the pattern observed in Byrne and Bovair's (1997) subjects.

The model's implementation of Byrne and Bovair's working memory capacity theory of PCE replicated the capacity and working memory loading factors in the Phaser Task's subject data (Figure 5), $r(4) = .920$, RMSD = .0657, for mean PCE rates (Tamborello & Trafton, 2013). The key is the model's dynamic interaction of a limited capacity to spread memory activation with the base-level activation feature of the structurally isolated step. Because of the working memory load imposed by the three-back memory task, the model has less memory activation to spread to long-term memory. But because the main goal of the task immediately follows the postcompletion step, it also receives some memory activation because of the graded spreading activation described previously

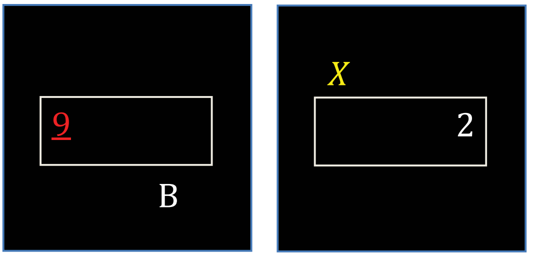

Figure 6. An example UNRAVEL display. Left: The underlined numeral 9 is displayed in red. Right: The italicized letter X is yellow in this example.

in the "Selecting the Next Step" and "Omission" sections. Because the main goal often gets retrieved during the course of task execution, it has a relatively elevated activation strength in addition to its receipt of some spreading activation. This makes the memory for the main goal a particularly strong competitor against a postcompletion step memory that is weakened by a combination of relatively less marginal spreading activation received from active buffer contents relative to competing goals and a longer time elapsed since its last retrieval.

The UNRAVEL Task

In contrast to the Stock Trader and Phaser Tasks, the UNRAVEL Task was designed to study not postcompletion error but omission (forgetting an action) and perseveration (repeating an action). Furthermore, in contrast to the single-page interactive form-filling paradigm, it used a simple, continuous, stimulus-response format.

Task and materials. The UNRAVEL Task (Altmann et al., 2014) is a sequential memory task in which subjects perform a two-choice decision regarding features of a simple alphanumeric display (Figure 6). *UNRAVEL* was an acronym for the stimuli features subjects responded to, such as that one item is Underlined or italicized, or that the letter in the display is Near to or far from the beginning of the alphabet, and so on. The UNRAVEL acronym specifies the order in which subjects must make these decisions, one decision per trial.

Each decision in the UNRAVEL sequence had only two possible options, and each of the

14 possible options were uniquely indicated by a single letter to be pressed on the keyboard as the subject's response. Thus, when subjects erred, it was possible to infer which of the task's steps they thought they were performing and to then count by how many procedure steps forward or backward they erred.

Design and procedure. Subjects were to remember which step of the UNRAVEL sequence they were currently on and to respond to the stimulus as appropriate for that step. For example, a subject seeing Figure 6 (left) and having just completed the U step would now perform the N step. Since the letter B is near to the beginning of the English alphabet, N would be the correct response. As soon as subjects pressed a key indicating 1 of the 14 potential responses, the experiment advanced to the next trial. In this example, the next correct trial would be an R action, to indicate whether the highlighted character is red or yellow.

Subjects performed approximately 320 (depending on randomized inter-interruption interval) UNRAVEL trials, completing one sequence of UNRAVEL steps after another. The experiment interrupted subjects 10 times per each of four trial blocks, at random once every three to six trials. After interruption, the interface provided no cue that might aid subjects' recall of their position within the UNRAVEL task sequence.

The interruption task was to transcription type a randomized sequence of the 14 response letters. Each time subjects were interrupted, they typed one, two, or three of these sequences before returning to the main task. These three interruption duration conditions lasted approximately 13, 21, and 30 seconds, respectively, depending on how quickly subjects could perform the transcription typing task.

Results. Without interruption, the "baseline" trial type, subjects hardly erred, only omitting one step 1% of the time and hardly ever omitting more than one step (Figure 7). Post-interruption, subjects erred much more but tended to do so mainly one step backward (–1) or forward (+1). Human resumption performance trended slightly toward chance as interruption duration increased.

Model data collection and performance. We executed our UNRAVEL model 1,000 times. We correlated its mean sequence error rates per trial type and interruption duration with those of Altmann et al.'s (2014) subjects. That is, model run means and human subject means each contributed 36 data points to Pearson correlation. With $r = 0.933$ and model $N = 1{,}000$, power was effectively 1 (Howell, 2002). We found that overall the model's data predicted human data quite well, $R^2 = .87$, $F(1, 34) = 227$, $p < .001$ (Tamborello, Trafton, & Altmann, 2015).

The model's graded task context representation reproduced participants' tendency to omit one step in 1% of the "baseline" (non-interrupted) condition trials. The graded task context representation, coupled with the manner in which the model gradually rebuilt its task context representation at resumption, led to greater rates of omission at this phase of the task, with gradually decreasing rates for increasing numbers of steps skipped. The increased omission rate is a direct result of the model's degraded context representation at resumption while the decrease in omission rate for increased number of steps skipped is a result of the graded association strength from context to action memories.

For perseverations at resumption, the graded effect of step distance is due to the gradual decay of episodic memory activation strength. At interruption onset, episodic encodings of trials one or two steps ago still have some memory activation, though not as much as for the episodic memory encoding the step that was just completed. Because retrieval is probabilistic, there is some small chance that one of these recent but wrong episodes could be recalled for rehearsal instead of the episodic memory encoding the action that was just completed.

Furthermore, the model's rehearsal algorithm provides an explanation for an interruption duration's deleterious effect on resumption performance. Each subsequent rehearsal has a chance of retrieving an incorrect memory. When an incorrect memory is retrieved, it receives the additional base-level activation because it was retrieved in place of the correct memory. This means that at the next rehearsal phase, it will have an even greater chance of being retrieved in place of the correct memory.

The UNRAVEL model's maximum association strength—the maximum amount by which one chunk may associate to another—was 7 versus 2.2 for the two PCE models (fit empirically).

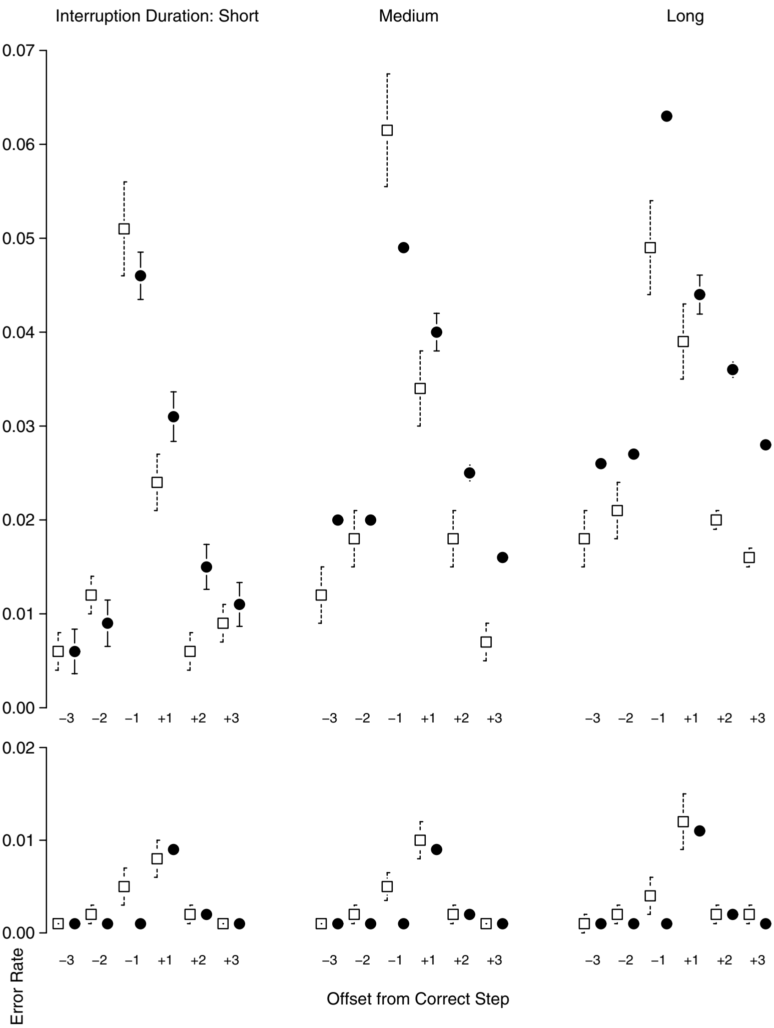

Figure 7. Human (open squares) and model (solid circles) sequence error rates for the UNRAVEL Task, post-interruption trials (top) and baseline trials (bottom). Error bars represent the 95% confidence interval of the mean.

Likewise, the amount of memory activation propagated from its imaginal buffer was 1.1.

DISCUSSION

The handful of processes comprising the process model, interacting dynamically, are sufficient to explain omissions, postcompletion error, and perseverations. These processes are synthesized from prior work in this field into a unified model. We speculate that the particular combination of processes explaining PCE will also explain omission errors in other structurally-isolated procedure steps.

Comparison With Byrne and Bovair's Working Memory Model

Like Byrne and Bovair's (1997) CAPS model, this model also explains PCE in terms of limited-capacity working memory. It does this both by working memory loading and by varying capacity, implemented in the model as varying activation source available from the imaginal buffer. However, this model also addresses errors with causes beyond available working memory capacity. Its omissions rely on the temporal co-occurrence basis of the strengths of association from one action memory to subsequent action memories. Its perseverations rely on the gradual decay of episodic memories' base-level activations over time.

Comparison With Memory for Goals

Altmann and Trafton (2002) make the prediction that priming is critical to correct performance of postcompletion steps. The Remember-Advance Process Model reliably performs postcompletion steps only when it has its greatest amount of priming available. One priming mechanism they suggest is "deliberate cognitive operations . . . [such as] rote associative (procedural) learning—through temporal co-occurrence, the step that precedes a postcompletion action will eventually come to serve as a cue for the action itself" (pp. 64–65). In the Remember-Advance Process Model, associative temporal priming was in fact the default method of action selection, one goal priming retrieval of the next. The strengths of association followed other theories of learned co-occurrence (e.g., Botvinick & Plaut, 2004). Furthermore, when goals must be suspended, as during interruption, the model implemented a deliberate goal-strengthening strategy to enable resumption—rehearsal.

Comparison With Remember-Advance Formal Model

The Remember-Advance formal model claims that for normal task execution people perform the same two-phase retrieval that they use for resumption. This means that for each step people must recall what they did on the last step. The implication here is that people do not retain a current task context representation in any sort of working memory–like buffer but instead must recreate it with each step of the task.

The process model somewhat simplifies assumptions underlying task execution relative to the Remember-Advance formal model. The process model uses two-phase retrieval sparingly because time-wise, it is expensive, and even small-scale costs of time matter (Gray & Boehm-Davis, 2000). Instead, for normal task execution it is a simpler explanation and provides for more efficient task execution for the model to retain some task context representation in a working memory capacity, a buffer, which it has available anyway. In fact, retaining such a task context in active buffers is critical for action selection in normal task execution because this task context representation serves as the source of retrieval from long-term memory of the next action to be performed. Furthermore, the capacity of this working memory construct is quite limited in the process model. It is for this reason that the process model engages in a cognitively threaded (Salvucci & Taatgen, 2008) rehearsal of episodic memory while executing the interrupting task. Furthermore, the limited capacity nature of the process model is critical for its explanation of the Byrne and Bovair (1997) data.

Explicit Rehearsal Strategies

The process model incurs the expense of rehearsal due to two necessary factors: (1) It must preserve access to state information over a longer duration than what decay would allow, and (2) it does not have the working memory capacity to retain this information and simultaneously accomplish its interrupting task. Our theory, as implemented by the process model, is that people use a cognitive workaround to preserve reference to a past task context representation in spite of the limited nature of working memory. At interruption onset, people pack away task state information into a form that can be retrieved later (an episodic memory), using a minimum of cognitive resources to rehearse throughout the interruption. At resumption, they attempt to retrieve that episode and then use it to reload the task context information to the active buffers.

Interruption duration impacts resumption performance because with every rehearsal iteration, there is a chance that an incorrect episodic memory could be retrieved. By ACT-R's base-level learning mechanism, every time a memory is retrieved, its activation is strengthened. But as time elapses from the last retrieval, the memory's activation decays. Typically, this manifested in the model's behavior when the model would, at rehearsal onset, retrieve by mistake an episodic memory from one or two trials ago rather than from the just completed trial.

Anecdotally, we noticed that incorrect episodic retrievals at resumption typically stemmed from an incorrect retrieval that occurred at the onset of rehearsal. Thus, the model would rehearse the wrong episodic memory. However, occasionally the model would retrieve an incorrect episode later during retrieval or at resumption, and chances of this happening increased with increasing interruption duration.

Example Application: Task State Cuing

The model's performance of all three discussed tasks depends critically on its ability to use a representation of the current task state to prime recall of subsequent procedure steps. This suggests priming as a means to mitigate post-interruption error. If a task interface could provide a sort of progress bar with specific named steps corresponding with the exact terminology describing the procedure that is known to the human operator, it should provide effective retrieval priming.

Or, where it may be unfeasible for the interface to track the task state, it could instead provide a convenient note-taking facility. This could provide human operators with a quick "brain dump," a verbal record of the person's internal task state that could be referenced later. Provided operators have at least a few seconds warning when an interruption occurs, they could write quick descriptions of the last action performed, the next action to perform, and any other pertinent details, analogous to how office workers often use Post-It notes to help them and their coworkers process paperwork.

CONCLUSION

The process model demonstrates one manner in which complex behavior arises from the interaction of a handful of processes. Associative spreading activation from active buffer contents to long-term memory drives selection of the next action. However, it sometimes combines with retrieval noise and graded association of task context to cause omission. These processes of omission, combined with strengthening of a main goal memory, drive up omission rates of structurally isolated steps such as the postcompletion step. Meanwhile, during interruption, decay imperfectly prevents retroactive interference; older episodic memories can intrude on newer ones, leading to perseveration. At resumption, people gradually bootstrap from their rehearsed episodic memory to the task context representation they require for normal task execution. But this ability to reconstruct a past task context representation comes with some cost of higher omission rates from the degraded interim task context representation.

In this process model, separate memory systems each provide fairly reliable performance for the prospective and retrospective retrievals necessary for many daily tasks. But it is their dynamic interaction, along with the system's own stochasticity, from which error behaviors emerge. The model makes a strong claim that some amount of human error is inevitable in routine procedures. System designers should plan for it and make systems resilient to it wherever and however possible.

The model is relatively broad and is the first quantitative model that we know of that can account for multiple error types in data from multiple labs. With some further refinement, for a given task it may be able to predict a priori how much aggregate error will result both in baseline conditions and in conditions of aggravating factors like interruption and high workload.

ACKNOWLEDGMENTS

This research was performed while the first author held a National Research Council Research Associateship Award at the United States Naval Research Laboratory in Washington, DC, USA. The authors thank the anonymous reviewers for their helpful suggestions.

KEY POINTS

- Certain classes of human routine procedural error are demonstrably inherent to the cognitive processes underlying routine procedural behavior.
- Omissions and postcompletion errors (PCEs) arise out of prospective memory for task sequences.
- PCE is a special case of omission error. It is due to a conjunction of goal structure, associative spreading activation, and memory strengthening.
- Perseverations occur because at the onset of task place rehearsal, such as at interruption onset, memories of actions from one or two sequence items ago are still slightly active. Transient noise may make one of these memories more active at rehearsal onset than the correct encoding of the last action performed.

- Model source code is available at https://github.com/tamborello/postcompletion-error and https://github.com/tamborello/UNRAVEL.

SUPPLEMENTARY MATERIALS

Supplemental material for this article is available with the manuscript on the *Human Factors* website.

REFERENCES

ACT-R Research Group. (2013). *ACT-R: Theory and architecture of cognition*. Retrieved from http://act-r.psy.cmu.edu

Altmann, E. M. (2002). Functional decay of memory for tasks. *Psychological Research*, *66*(4), 287–297. doi:10.1007/s00426-002-0102-9

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 and Review*, *14*(6), 1079–1084.

Altmann, E. M., & Trafton, J. G. (2015). Brief lags in interrupted sequential performance: Evaluating a model and model evaluation method. *International Journal of Human-Computer Studies*, *79*, 51–65. doi:10.1016/j.ijhcs.2014.12.007

Altmann, E. M., Trafton, J. G., & Hambrick, D. Z. (2014). Momentary interruptions can derail the train of thought. *Journal of Experimental Psychology: General*, *143*(1), 215–226. doi:10.1037/a0030986

Anderson, J. R. (2007). *How can the human mind exist in the physical universe?* New York, NY: Oxford University Press.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036–1060. doi:10.1037/0033-295X.111.4.1036

Anderson, J. R., Bothell, D., Lebiere, C., & Matessa, M. (1998). An integrated theory of list memory. *Journal of Memory and Language*, *38*, 341–380.

Botvinick, M., & Plaut, D. C. (2004). Doing without schema hierarchies: A recurrent connectionist approach to normal and impaired routine sequential action. *Psychological Review*, *111*(2), 395–429. doi:10.1037/0033-295X.111.2.395

Byrne, M. D., & Bovair, S. (1997). A working memory model of a common procedural error. *Cognitive Science*, *21*(1), 31–61.

Gray, W. D., & Boehm-Davis, D. A. (2000). Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of Experimental Psychology: Applied*, *6*(4), 322–335. doi:10.1037//1076-898X.6.4.322

Howell, D. (2002). *Statistical methods for psychology* (5th ed.). Pacific Grove, CA: Duxbury.

Just, M. A., & Carpenter, P. A. (1992). A capacity theory of comprehension: Individual differences in working memory. *Psychological Review*, *99*, 122–149.

Monk, C. A., Trafton, J. G., & Boehm-Davis, D. A. (2008). The effect of interruption duration and demand on resuming suspended goals. *Journal of Experimental Psychology: Applied*, *14*(4), 299–313. doi:10.1037/a0014402

Ratwani, R. M., & Trafton, J. (2010). A generalized model for predicting PCEs. *Topics in Cognitive Science*, *2*(1), 154–167. doi:10.1111/j.1756-8765.2009.01070.x

Ratwani, R. M., & Trafton, J. G. (2011). A real-time eye tracking system for predicting and preventing PCEs. *Human-Computer Interaction*, *26*(3), 205–245. doi:10.1080/07370024.2011.601692

Reason, J. (1990). *Human error*. Cambridge, UK: Cambridge University Press.

Ritter, F. E., Schoelles, M. J., Quigley, K. S., & Klein, L. C. (2011). Determining the number of simulation runs: Treating simulations as theories by not sampling their behavior. In S. Narayanan & L. Rothrock (Eds.), *Human-in-the-loop simulations: Methods and practice* (pp. 97–116). London: Springer-Verlag. Retrieved from http://acs.ist.psu.edu/papers/ritterSQKip.pdf

Salvucci, D. D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*, *115*(1), 101–130. doi:10.1037/0033-295X.115.1.101

Tamborello, F. P., II, & Trafton, J. G. (2013). A long-term memory competitive process model of a common procedural error, part II: Working memory load and capacity. In *Proceedings of the International Conference on Cognitive Modeling* (pp. 245–250). Ottawa, Ontario, Canada: Carleton University.

Tamborello, F. P., Trafton, J. G., & Altmann, E. A. (2015). Memory processes of sequential action selection. In *Proceedings of the 37th annual Cognitive Science Society meeting* (pp. 2338–2343). Austin, TX: Cognitive Science Society.

Trafton, J. G., Altmann, E. M., & Ratwani, R. M. (2009). A memory for goals model of sequence errors. In *Proceedings of the 9th International Conference of Cognitive Modeling* (pp. 140–145). Ottawa, Ontario, Canada: Carleton University.

Trafton, J. G., Altmann, E. M., & Ratwani, R. M. (2011). A memory for goals model of sequence errors. *Cognitive Systems Research*, *12*, 134–143.

Franklin P. Tamborello, II, PhD, CSP, is chief scientist at Cogscent, LLC, where he consults on projects involving cognition, software development, or safety. He obtained his PhD in human-computer interaction from the Department of Psychology at William Marsh Rice University in 2009.

J. Gregory Trafton, PhD, is head of the Intelligent Systems Section at the Navy Center for Applied Research in Artificial Intelligence. He obtained his PhD in psychology from Princeton University.

Date received: December 12, 2015
Date accepted: September 11, 2016