Evaluating Systematic Error Predictions in a Routine Procedural Task

Jennifer Tsai[1] and Michael D. Byrne[2]
[1]University of Illinois at Urbana-Champaign, Urbana, IL
[2]Rice University, Houston, TX

Systematic errors in routine procedural tasks present an important problem for psychologists who study interactions between humans and technological systems. This paper details an experiment designed to examine systematic error patterns and evaluate error predictions made by a notable psychological theory and industry-standard usability tools when performing multiple routine procedural tasks on a single highly visual interface. Participants completed three dynamic, computer-based routine procedural tasks involving execution of multiple steps. Differences were found in error frequencies at particular steps between the three tasks, a result that is consistent with predictions derived from Altmann and Trafton's (2002) activation-based model of memory for goals, but contrary to those of usability guidelines. Error patterns were reminiscent of several familiar types of systematic error.

INTRODUCTION

History and Motivation

Execution of routine tasks can be periodically marred by the commission of non-random errors. Virtually everyone has experienced these lapses, whether in the form of forgetting to remove the original after making a photocopy, or finding oneself driving down a habitual route when the intended destination requires a different path. Spanning a multitude of domains and tasks, such procedural errors—or errors in the execution of an otherwise routine procedure—are often mere nuisances demanding little in the way of remediation or prevention. But a nontrivial number can have as high a penalty cost as human life (Casey, 1998). Despite this, research on general error prediction and mitigation is still not particularly common.

In his (1990) book *Human Error*, James Reason defined error as “all those occasions in which a planned sequence of mental or physical activities fails to achieve its intended outcome, when these failures cannot be attributed to the intervention of some chance agency.” Both Reason’s approach and most in the literature on human error consist of attempts to classify errors into various types and forms. However, this exclusive devotion to categorization has left a void where prediction and mitigation are concerned. Often the best of current practice is to simplify prediction into a probabilistic measure based on analyses of the tasks and interfaces. No method is able to provide a comprehensive account for the specific cognitive components and processes leading to erroneous behaviors or the conditions that foster them. Furthermore, once a potential error is identified by such techniques, nothing is said for how the evaluator or designer should go about generating a solution.

A theory of error capable of supporting prediction and based upon a solid understanding of human cognition would greatly advance development of error remediation techniques. Construction of such a theory would require an extensive amount of comprehensive data from both the laboratory and the field. While this is not immediately in sight, computer interfaces have proven to be an outstanding medium for showcasing and capturing this element of human behavior. Because of the heavily visual nature of most computer-based tasks today, simple alterations of the visual design of an interface can induce operators to commit particular errors and affect overall error patterns in interesting and often unexpected ways (Byrne, Maurier, Fick, & Chung, 2004). From such findings and further study of classes of systematic errors, it is possible to extrapolate design principles grounded in cognitive and perceptual theory, enabling designers to create safer, less error-prone visual interfaces.

Systematic Errors

Given the abundance of evidence suggesting humans will use whatever means available to them to reduce cognitive workload and effort (Gray, 2000; Altmann & Trafton, 1999), it is not surprising that operators often reuse former interaction patterns when interacting with new devices (Besnard & Cacitti, 2005). While under certain circumstances this strategy results in successful performance, it can also lead to mistakes when a previous pattern is unsuitable for a new device. Numerous documented cases exist where changes in work environments caused accidents when an operator interacted with a novel device that was erroneously perceived as familiar (Casey, 1998). Despite fundamental differences in system structure, similarities in form and surface features of two dissimilar problems or devices can drive inappropriate assumptions of parallelism, resulting in these transfer errors.

A number of control systems employ modes as their underlying structure. Modes are an architecture commonly used for grouping several machine configurations under one label (Degani et al., 2000). Goal operators for behaviors in each mode are often enabled by heavily overlapping interface components of the control system. The ensuing mode ambiguity (Thimbleby, 1982), where the identical user input performs two different operations in two different modes, can result in mode errors—in essence, carrying out an intention when the situation is falsely classified (Degani et al., 2000). An action may be appropriate for the perceived situation, but inappropriate for the real one, just as an operation may be the correct step in one mode, but erroneous in another.

Research suggests that some common procedural errors occur when environmentally triggered behaviors override planned actions (Norman, 1981). These capture errors reflect the notion that forces of habit can exert profound influence on behavior. They are especially likely when there are strong associations between specific contexts and certain behaviors, as in the case of commonly performed or even partially automated actions. Any number of factors could serve as a triggering environmental cue, such as the visual context of an interface. Roberts et al. (1994) also established that these errors could be induced by taxing working memory with a secondary memory-loading task. The results of this research have been used to suggest that working memory capacity is the driving force behind individual differences in everyday errors.

Modeling Approaches

Many assumptions behind theories of procedural error rest on the concept of hierarchical control structures, which suggests that a complicated task with an overarching goal can be simplified and broken down into smaller units, each a component subtask with its own subgoal and steps. In previous studies by Byrne and Bovair (1997) and Byrne and Davis (2006) involving procedural tasks with these structures, participants reliably generated several different types of procedural errors at particular steps both within subtasks, as well as within the larger task framework. Cognitive modeling work by Kieras, Wood, and Meyer (1997) has also provided strong evidence to suggest that even task or domain experts never abandon task hierarchies. These goal-based processing strategies traditionally rely on a "task-goal" stack model (Young et al., 1989) that somewhat dubiously forecasts perfect memory for old goals.

Hierarchical control structures and goal management are key tenets of major cognitive task modeling approaches in popular use today, including Hierarchical Task Analysis, which deconstructs task goal structures into subcomponents (Kirwan & Ainsworth, 1992), and Goals Operators Methods and Selection Rules (GOMS), which offers a means to represent knowledge required for correct task performance (John & Kieras, 1996). However, these and many other techniques suffer from two main weaknesses: they rely on the individual skill and subjective judgment of analysts, which can lead to compromising inter- and intra-rater reliability problems, and they posit weak or nonexistent accounts for the environment, lacking such vital components as stress and noise (Stanton, 2004). This neglect of factors outside the head can prove especially problematic for error prediction in highly visual interfaces, a context for which current methods are not well-equipped.

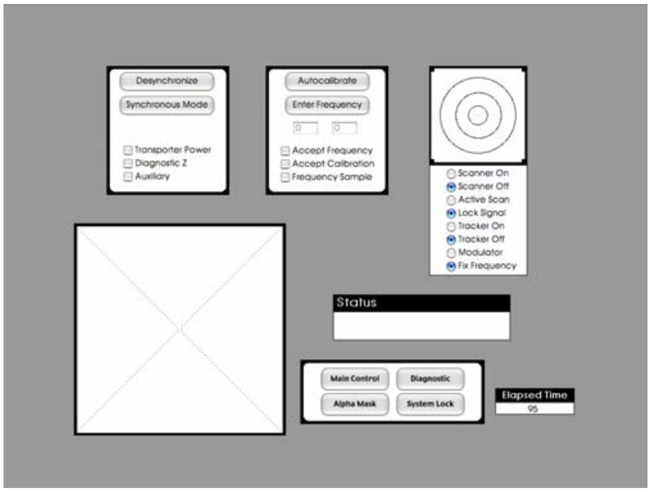
Altmann and Trafton's (2002) activation-based model of memory for goals (MAGS) offers an alternative theory that may be able to account for some of the internal and external factors omitted by standard task modeling. MAGS views task goals as memory elements that must remain active and overcome interference from other goals. Memory and the environment are substituted for a goal stack, and task goals are considered as ordinary memory elements requiring the usual encoding and retrieval processes. For example, retrieval cues from the environment can reactivate suspended or decayed goals. Alternatively, ambiguous cues can lead to interference and activation of incorrect goals.

Previous Studies

GOMS predicts that isomorphic tasks should have identical error profiles at matched steps in the procedures. Byrne, Maurier, Fick, & Chung (2004) evaluated this claim with a set of heavily visual dynamic tasks and found that operators produced significantly different error rate profiles at several steps. These differences were attributed to dissimilar perceptual components of the interfaces. This suggests an important role for the match between a task's structure and the visual layout of its interface, a factor clearly omitted by extant GOMS-class models, and not fully accounted for by MAGS. A similar vein of experiments has shown that minor visual layout modifications can greatly affect task execution (Byrne, Chung, & Fick, 2004), while intuitively large manipulations can inspire little to no change (Byrne, 2005), results that are in part both consistent and inconsistent with standard industry tools and MAGS. In short, no model currently exists that can wholly account for this array of findings. Additional work is necessary to further outline the influence of visual layout properties on task performance.

EXPERIMENT

The paradigm is a group of routine procedural computer tasks set against a backdrop of the fictional Star Trek world. They are the Transporter, Tactical, and Navigation tasks from Byrne and Bovair (1997), the Medical task from Chung and Byrne (2004), and a new Jammer task. All are comprised of a sequence of predetermined steps to be executed on different visual interfaces, each equipped with assorted buttons, sliders, and text entry fields (Figure 1). For the Transporter, Jammer, and Medical tasks, subjects must execute sets of procedures that are hierarchical in nature, consisting of series of subgoals and their composite steps.



□ Transporter Power
□ Diagnostic Z
□ Auxiliary

□ Accept Frequency
□ Accept Calibration
□ Frequency Sample

○ Scanner On
● Scanner Off
○ Active Scan
● Lock Signal
○ Tracker On
● Tracker Off
○ Modulator
● Fix Frequency

Figure 1: Transporter/Jammer interface.

The Transporter and Jammer are performed on the same interface. Two versions of the Jammer were developed: the New Jammer, with steps consisting of buttons that are unused in the Transporter (no overlap of controls), and the Mix Jammer, with steps consisting of a mix of used and unused Transporter buttons (some overlapped controls) (Figure 2). The Transporter and both versions of the Jammer have isomorphic subgoals and step procedures.

Step #	**Transporter**	**New Jammer**	**Mix Jammer**
First Subgoal			
1	Scanner On	Tracker On	Scanner On
2	Active Scan	Modulator	Modulator
3	Lock Signal	Fix Frequency	Fix Frequency
4	Scanner Off	Tracker Off	Scanner Off
Second Subgoal			
5	Enter Frequency	Calibration	Calibration
6	<type, left field>	<type, right field>	<type, left field>
7	Accept Frequency	Accept Calibration	Accept Calibration
Third Subgoal			
8	Transporter Power	Auxiliary Power	Transporter Power
9	Synchronous Mode	Desynchronize	Desynchronize
10	<track-and-click>	<track-and-click>	<track-and-click>

Figure 2: Transporter, and New and Mix Jammer task subgoals and steps.

Error Predictions

Standard industry modeling tools take into account only the underlying task structure, without consideration for any perceptually-driven interface properties. This being the case, they essentially predict no differences in performance of isomorphic tasks such as the Transporter and Jammer. In other words, one would expect to see identical error patterns for both tasks.

Appealing to MAGS, pairing execution of the Transporter with the Jammers is expected to adversely affect operators' performance on both tasks. A pairing of the Transporter with the Mix Jammer is predicted to be most detrimental, with error-prone steps at transition boundaries between areas of button overlap and no-overlap (e.g. steps 2, 4, 5, 8, and 9). Because mental models formed of the two tasks would partially overlap, the pairing encourages a large amount of interference in working memory. Participants must keep in mind the particular task—Transporter or Jammer—to be performed at the moment, requiring that step and goal traces for the current task have sufficient activation to be able to overcome a high retrieval threshold, elevated by the strong goal trace of the opposing task and a visually ambiguous interface.

It is predicted that a pairing of the Transporter and New Jammer will similarly affect performance, though not to as great an extent (e.g. error-prone first steps of each subgoal—1, 5, and 8). Participants must again keep in mind the particular task they are performing, but in this case the two mental models are discrete, precluding much of the interference created with the previously discussed Transporter and Mix Jammer pairing. The step and goal traces for the current task must still possess sufficient activation to be able to overcome a retrieval threshold elevated due to the ambiguous interface, but the opposing task goal trace is considerably weaker, stemming from the lack of overlapping steps.

With a pairing of the Transporter and Medical tasks, performance on one should remain unaffected by the other regarding the aforementioned factors. The tasks and interfaces were designed to be different in structure and form so as to preclude interference arising from these variables.

METHODS

Participants

Forty-six undergraduate students from Rice University, ages 17-22 years, participated in this experiment for course credit in a psychology course. Additionally, cash prizes were awarded to the top three performers ($25, $15, and $10).

Design

The study used a two-factor within (step of procedure) and between (task assignment) participants design. There were three conditions:

Control. Transporter paired with Medical, which used a different display and shared no steps with the Transporter task.

No-overlap. Transporter paired with New Jammer.

Overlap. Transporter paired with Mix Jammer.

Participants were randomly assigned to a condition, all of which included the Tactical and Navigation tasks serving as fillers. The dependent measure was the frequency of errors made at individual steps within the Transporter and Jammers, out of total number of opportunities.

Procedure

Participants were run in two sessions spaced one week apart. The first served as a training session in which paper manuals were consulted for task execution instructions. Participants were required to train to criterion on a task by logging four error-free trials, before being allowed to continue onto the next task. Errors resulted in a warning beep, ejection back to the main control screen with a message detailing the error and what the right step was, and forced restarting of the task. This measure ensured that participants could not complete training without having performed a task four times correct in its entirety. Order of task training was randomized.

The second session consisted of thirty-eight test trials. Participants completed twelve trials each of the Transporter and its paired task dependent on condition, and seven trials each of the Tactical and Navigation tasks. Presentation order of all trials was randomized. For this session, the program also emitted beeps upon error commission, but trials were otherwise continued until completion. A concurrent three-letter span auditory working memory task was implemented to promote generation of errors at high enough frequencies to be systematically studied. Participants were encouraged to work quickly and accurately by means of a scoring system, timer, and cash incentives.

RESULTS

Two participants, one from the No-overlap condition and one from the Overlap condition, had extreme error frequencies exceeding three interquartile range values from the 75% quartile, and were thus removed as outliers. Of primary interest for analysis were the mean frequencies of errors at each step of the Transporter and Jammer tasks, across groups.

Transporter Error Frequencies

For the Transporter, post-hoc tests were conducted on steps 1, 2, 8, and 9 to determine if there were performance differences between conditions (Figure 3).

Step 1 errors were reliably different, with the No-overlap condition faring worse then the Overlap and Control conditions, $F(2, 41) = 7.95$, $p = 0.001$. The No-overlap and Control conditions fared better than the Overlap condition for steps 2, $F(2, 41) = 20.1$, $p < 0.001$, and 9, $F(2, 41) = 4.41$, $p = 0.018$. For step 8, the Control condition fared worse than the No-overlap and Overlap conditions, $F(2, 41) = 4.22$, $p = 0.022$.

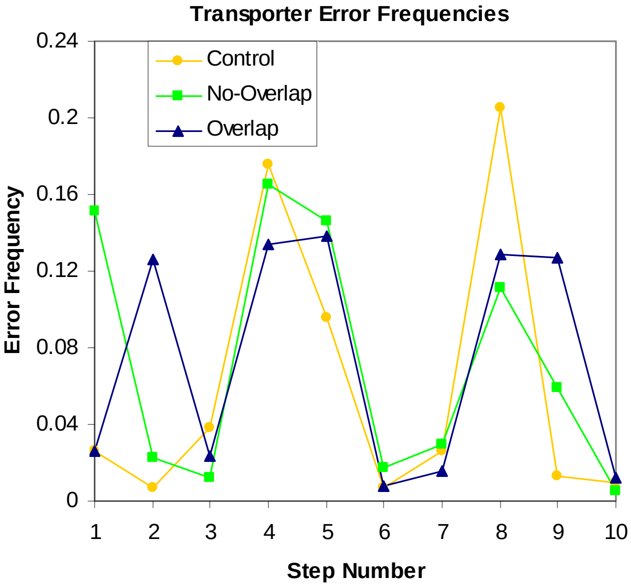


Figure 3: Transporter Error Frequencies by Step and Condition.

Jammer Error Frequencies

For the Jammers, post-hoc tests were conducted on steps 1, 2, 5, and 8 to determine if there were any performance differences from effect of condition (Figure 4).

Step 1 error frequencies were reliably different, with the Overlap condition faring better than No-overlap, $F(1, 27) = 32.6$, $p < 0.001$. The opposite was found for step 2, with the No-overlap condition faring better than Overlap, $F(1, 27) = 7.12$, $p = 0.013$. No differences were found for steps 5 and 8.

Given the divergent error frequency values for step 1, coupled with the observation that these values are reversed in direction from the general trend of the remaining steps, a test for main effect of condition was conducted with step 1 removed. The resulting analysis missed significance at $F(1, 27) = 2.64$, $p = 0.116$, offering some weak suggestion for overall performance being worse on the Mix Jammer (Overlap condition) than the New Jammer (No-overlap condition).

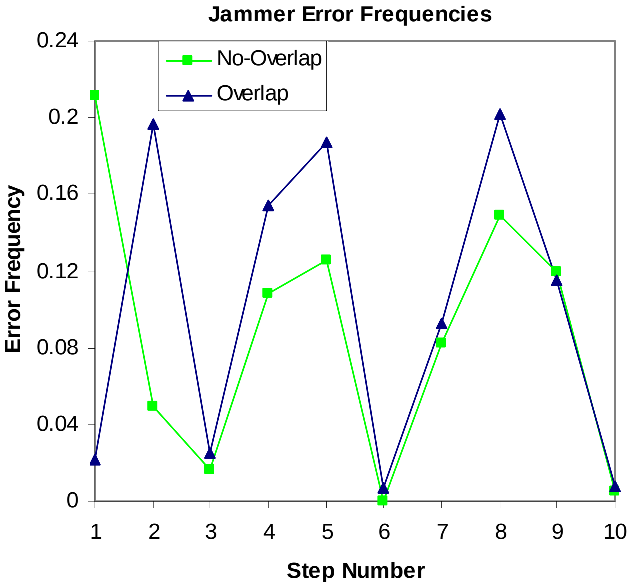


Figure 4: Jammer Error Frequencies by Step and Condition.

DISCUSSION

Transporter Errors

Transporter steps 1, 2, 8, and 9 differed by condition. The No-overlap condition being more error-prone for step 1 is in line with the notion that performance suffers for steps executed with a different button for the Transporter than the Jammer, especially when the step starts off a subgoal. Steps 2 and 9 for the Overlap condition were more error-prone for the same reason, though 2 was not the start of a subgoal. Both also occurred at a point where there is a switch from the two tasks sharing the same step procedure button, to them not sharing the same button.

Step 8 is surprising to note in that the Control condition, where the Transporter task was paired with the Medical task, actually fared worse than both of the other conditions where the Transporter was paired with the two Jammers. There is intuitively no reason for this. Since properties of the Medical task and interface should not impair Transporter performance in any way, the implication is that the presence of the paired Jammer tasks somehow facilitates correct execution of this Transporter step.

Jammer Errors

Jammer steps 1 and 2 differed by condition. The No-overlap condition was more error-prone than the Overlap condition for step 1, with the pattern being reversed for step 2. Again, these results are presumably due to the two matched procedure steps being executed with a different button for the Transporter than for the Jammer on step 1 of the No-overlap condition, and step 2 of the Overlap condition. Step 1 is also the start of a subgoal for the No-overlap condition, while step 2 is at a point where there is a switch from the two tasks sharing, to not sharing the same step procedure button for the Overlap condition.

Conclusions

From general guidelines or current error identification methods, there is little to suggest why there should be differences in error frequencies at any of the steps between the Transporter and Jammers. The implication is that a key factor responsible for the different profiles is the identical visual interface of the matched task.

Visual aspects of an interface can greatly influence error patterns in computer-based routine procedural tasks, an effect that has mostly remained elusive to psychological and human factors models in terms of prediction. The results of this study provide support for some of the activation and interference based predictions of error generated by MAGS, and contradict heuristic and evaluation techniques that suggest no differences between the experimental tasks.

These findings demonstrate the inherent complexity of error prediction in highly visual, dynamic computer-based tasks. More extensive evaluation is necessary not only to test for the accuracy of conventional rules and models, but also to establish more explicit analytical tools that are applicable to a variety of interfaces, and capable of producing accurate predictions and effective design solutions for errors in all kinds of interactive systems.

ACKNOWLEDGEMENTS

This research was supported by the Office of Naval Research under grants #N00014-03-1-0094 and N00014-06-1-0056. The views and conclusions expressed are those of the authors and should not be interpreted as representing the official policies or endorsements, either expressed or implied, of the ONR, the U.S. government, or any other organization.

REFERENCES

Altmann, E. M., & Trafton, J. G. (1999). Memory for goals: An architectural perspective. *Proceedings of the Twenty-first Annual Conference of the Cognitive Science Society* (pp. 19-24). Hillsdale, NJ: Erlbaum.

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

Besnard, D., & Cacitti, L. (2005). Interface changes causing accidents: An empirical study of negative transfer. *International Journal of Human-Computer Studies, 62(1),* 105-125.

Byrne, M. D. (2005). *Execution of isomorphic routine procedures: Errors and models.* Presented at the ONR Workshop on Attention, Perception, and Modeling for Complex Displays, Arlington, VA, May 2005.

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

Byrne, M. D., Chung, P. H., & Fick, C. S. (2004). *Mitigating errors in the execution of routine procedures.* Presented at the ONR Workshop on Attention, Perception, and Modeling for Complex Displays, Newport, RI, May 2004.

Byrne, M. D., & Davis, E. M. (2006). Task structure and postcompletion error in the execution of a routine procedure. *Human Factors, 48,* 627–638

Byrne, M. D., Maurier, D., Fick, C. S., & Chung, P. H. (2004). Routine procedural isomorphs and cognitive control structures. In C. D. Schunn, M. C. Lovett, C. Lebiere & P. Munro (Eds.), *Proceedings of the Sixth International Conference on Cognitive Modeling* (pp. 52-57). Mahwah, NJ: Erlbaum.

Casey, S. (1998). *Set phasers on stun.* Santa Barbara, CA: Aegean Publishing.

Chung, P. H., & Byrne, M. D. (2004). Visual cues to reduce errors in a routine procedural task. *Proceedings of the Twenty-Sixth Annual Conference of the Cognitive Science Society.* Austin, TX: Cognitive Science Society.

Degani, A., Shafto, M., & Kirlik, A. (2000). Modes in human-machine systems: Constructs, Representation, and classification. *The International Journal of Aviation Psychology, 9,* 125-138.

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

John, B. E., & Kieras, D. E. (1996). The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction, 3(4),* 320-351.

Kieras, D. E., Wood, S. D., & Meyer, D. E. (1997). Predictive engineering models based on the EPIC architecture for multimodal high-performance human-computer interaction task. *ACM Transactions on Computer-Human Interaction, 4*, 230-275.

Kirwan, B., & Ainsworth, L. K. (Eds.). (1992). *A guide to task analysis*. London: Taylor & Francis.

Norman, D. A., (1981). Categorization of action slips. *Psychological Review, 88,* 1-15.

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

Roberts, R. J., Jr., Hager, L. D., & Heron, C. (1994). Prefrontal cognitive processes: Working memory and inhibition in the antisaccade task. *Journal of Experimental Psychology: General, 123,* 374-393.

Stanton, N. A. (2004). Human error identification in human-computer interaction. In J. Jacko (Ed.), *Handbook of Human Factors and Ergonomics* (2nd ed.). New York: John Wiley.

Thimbleby, H. (1982). Character level ambiguity: Consequences for user interface design. *International Journal of Man-Machine Studies, 16,* 211-225.

Young, R. M., Barnard, P., Simon, T., & Whittington, J. (1989). How would your favorite user model cope with these scenarios? *ACM SIGCHI Bulletin, 20,* 51-55.