TOWARDS A COMPREHENSIVE COMPUTATIONAL THEORY OF HUMAN MULTITASKING: ADVANCING COGNITIVE MODELING WITH DETAILED ANALYSES OF EYE MOVEMENT DATA AND LARGE-SCALE EXPLORATION OF TASK STRATEGIES

by

YUNFENG ZHANG

A DISSERTATION

Presented to the Department of Computer and Information Science
and the Graduate School of the University of Oregon
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

June 2015

DISSERTATION APPROVAL PAGE

Student: Yunfeng Zhang

Title: Towards a Comprehensive Computational Theory of Human Multitasking: Advancing Cognitive Modeling with Detailed Analyses of Eye Movement Data and Large-Scale Exploration of Task Strategies

This dissertation has been accepted and approved in partial fulfillment of the requirements for the Doctor of Philosophy degree in the Department of Computer and Information Science by:

Anthony Hornof	Chair
Allen Malony	Core Member
Michal Young	Core Member
David Kieras	Core Member
Ulrich Mayr	Institutional Representative

and

Scott L. Pratt	Dean of the Graduate School

Original approval signatures are on file with the University of Oregon Graduate School.

Degree awarded June 2015

DISSERTATION ABSTRACT

Yunfeng Zhang

Doctor of Philosophy

Department of Computer and Information Science

June 2015

Title: Towards a Comprehensive Computational Theory of Human Multitasking: Advancing Cognitive Modeling with Detailed Analyses of Eye Movement Data and Large-Scale Exploration of Task Strategies

Designs of human-computer systems intended for time-critical multitasking can benefit from an understanding of the human factors that support or limit multitasking performance and a detailed account of the human-machine interactions that unfold in a given task environment. An integrated, computational cognitive model can test and provide such an understanding of the human factors related to multitasking and reveal the dynamic interactions that occur in the task at the level of hundreds of milliseconds. This dissertation provides such a detailed computation model of human multitasking, built for a time-critical, multimodal dual task experiment and validated by the eye tracking data collected from the experiment. This dissertation also develops new approaches to conducting cognitive modeling, which enable efficient and systematical exploration of multitasking strategies, as well as principled model comparisons.

The dual task experiment captures many key aspects of real-world multitasking scenarios such as driving. In the experiment, the participant interleaved two tasks: one requires tracking a constantly-moving target with a joystick, and the other requires keying-in responses to objects moving across a radar display. Peripheral visibility and auditory conditions of the experiment were manipulated to assess the influence of peripheral visual information and auditory information on multitasking performance. Detailed eye tracking data were collected, and this dissertation presents a detailed analysis of this set of data, which provides the bases for model development and validation.

The cognitive model presented in this dissertation, built based on the EPIC (Executive Processes-Interactive Control) cognitive architecture, accurately accounted for the eye movement data and other behavioral data of each participant using systematic explorations of task strategies and parameters configured for each individual participant. A parallelized cognitive modeling system was developed to accommodate the much increased computational demand of strategy exploration and individualized model building. New model comparison techniques were proposed to determine which strategy best accounts for the empirical data. Payoff analyses were applied, and they revealed people's tendency to locally optimize task performance based on task payoff as well as instantaneous feedback. The results point to new approaches for building *a priori* models that predict multitasking performance.

This dissertation includes previously published coauthored material.

CURRICULUM VITAE

NAME OF AUTHOR: Yunfeng Zhang

GRADUATE AND UNDERGRADUATE SCHOOLS ATTENDED:

University of Oregon, Eugene, Oregon
Beijing Normal University, Beijing, China

DEGREES AWARDED:

Doctor of Philosophy, Computer and Information Science, 2015, University of Oregon
Master of Science, Computer and Information Science, 2013, University of Oregon
Bachelor of Science, Information Science and Technology, 2007, Beijing Normal University

AREAS OF SPECIAL INTEREST:

Human-Computer Interaction
Cognitive Science

PROFESSIONAL EXPERIENCE:

Research Intern, IBM T. J. Watson Research Center, May 2014–September 2014

Research Intern, Palo Alto Research Center, May 2013–December 2013

Graduate Research Assistant, University of Oregon, 2008–Current

Research Intern, Beijing Key Lab of Applied Experimental Psychology, 2007–2008

GRANTS, AWARDS AND HONORS:

J. Donald Hubbard Family Scholarship, University of Oregon, 2012

Henry V. Howe Scholarship, University of Oregon, 2012

J. Donald Hubbard Family Scholarship, University of Oregon, 2010

Graduate Research Fellowship, University of Oregon, 2009 to 2015

Graduate Teaching Fellowship, University of Oregon, 2008 to 2009

PUBLICATIONS:

Zhang, Y., Paik, J., & Pirolli, P. (2015, to appear). Reinforcement learning and counterfactual reasoning explain adaptive behavior in a changing environment. To appear in *Topics in Cognitive Science*.

Zhang, Y., Bellamy, R. K. E., and Kellogg, W. A. (2015, to appear). Designing information for remediating cognitive biases in decision-making. To appear in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*.

Zhang, Y., Paik, J., & Pirolli, P. (2014). Reinforcement learning and counterfactual reasoning explain adaptive behavior in a changing environment. In P. Bello, M. Guarini, M. McShane, & B. Scassellati (Eds.), *Proceedings of the 36th Annual Meeting of the Cognitive Science Society* (pp. 1850–1855). Qubec City, Canada: Cognitive Science Society.

Zhang, Y., & Hornof, A. J. (2014). Understanding multitasking through parallelized strategy exploration and individualized cognitive modeling. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (CHI '14), 3885–3894.

Zhang, Y., & Hornof, A. J. (2014). Easy post-hoc spatial recalibration of eye tracking data. In *Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA '14* (pp. 95–98).

Zhang, Y., & Hornof, A. J. (2013). Using model tracing and evolutionary algorithms to determine parameter settings for cognitive models from time series data such as visual scanpaths. In *Proceedings of the 12th International Conference on Cognitive Modeling*, 433–438.

Zhang, Y., & Hornof, A. J. (2012). A discrete movement model for cursor tracking validated in the context of a dual-task experiment. In Proceedings of the 56th Annual Meeting of the Human Factors and Ergonomics Society, 1000–1004.

Zhang, Y., & Hornof, A. J. (2011). Mode-of-disparities error correction of eye-tracking data. *Behavior Research Methods, 43*(3), 834–842.

Hornof, A. J., & Zhang, Y. (2010). Task-constrained interleaving of perceptual and motor processes in a time-critical dual task as revealed through eye tracking. In *Proceedings of the 10th International Conference on Cognitive Modeling*, 97–102.

Hornof, A. J., Zhang, Y., & Halverson, T. (2010). Knowing where and when to look in a time-critical multimodal dual task. In *Proceedings of ACM CHI 2010: Conference on human factors in computing systems*, 2103–2112.

ACKNOWLEDGEMENTS

This research was supported by the Office of Naval Research under Grant N00014-06-10054 and by the National Science Foundation through Grant IIS-1017593 to the University of Oregon with Anthony Hornof as the principal investigator.

I would like to thank my advisor Anthony Hornof for always being there and providing me with intellectual, moral, and financial support. Thank you for encouraging me to always strive for perfection.

Thank you to my family for their support and encouragement, and especially to my wife, Xu Zhao, for her understanding and patience.

TABLE OF CONTENTS

Chapter Page

Chapter Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

LIST OF TABLES

Table Page

CHAPTER I

INTRODUCTION

Today, in many different scenarios enabled by powerful digital devices, people engage in multiple tasks at the same time. However, many contemporary hardware and software user interfaces (UIs) such as touchscreens and smartphone software interfaces are not specifically designed to support multitasking. Such UIs can dramatically impair multitasking performance and pose a serious risk of harm to personal and public safety (Olson, Hanowski, Hickman, & Bocanegra, 2009; Richtel, 2009).

Designing UIs to support multitasking is difficult considering that the field of human-computer interaction (HCI) still lacks a good understanding of the fundamental capabilities and limitations of human multitasking. Though theories exist (e.g., Wickens, 2002), there are many controversies around several important issues such as whether people can truly select responses for multiple tasks at the same time (see Meyer & Kieras, 1997a). Furthermore, there are no definitive conclusions regarding how to effectively reduce interference between two tasks. For example, in driving research, many studies show that visual displays can distract drivers and suggest that speech-based interfaces are better (e.g., Wierwille, 1993), but some other studies suggest that speech-based user interfaces for certain tasks may introduce more interference than visual displays (e.g., Lee, Caven, Haake, & Brown, 2001). Perhaps the source of this lack of consistency is that multitasking scenarios are often highly dynamic, and when and how task interference occurs may depend a lot on the task context.

Computational cognitive modeling can potentially help resolve the theoretical controversies and tackle the complex dynamics of multitasking. A computational cognitive model is, in the context of this research, a computer program that approximates how humans process information and carry out tasks at the level of sensorimotor and cognitive processes. Computational cognitive models implement the key psychological theories of human information processing, and these implementations enable researchers to (a) validate theoretical claims by comparing a model's concrete numeric predictions to empirical data, and (b) predict performance under various task contexts by running the model through a simulated task environment that reproduces the actual task procedure. Just as computational modeling has transformed research and development in other scientific and engineering disciplines (Hamming, 2003), computational cognitive modeling has the potential to bring cognitive science and its related applied domains such as HCI to a new era.

Decades of research on cognitive modeling have made significant progress towards this goal, and have resulted in several cognitive architectures (software frameworks that implement a set of perceptual, motor, and cognitive processes for model-building) that unify psychological theory on human information processing. Two major cognitive architectures often used in the HCI domain are ACT-R (Atomic Components of Thoughts-Rational, Anderson & Lebiere, 1998) and EPIC (Executive Processes-Interactive Control, Kieras & Meyer, 1997). These architectures embrace the notion of embodied cognition (Newell, 1973) and provide a set of software modules that simulate perceptual, motor, and cognitive processes as well as the interactions among these processes. With cognitive architectures, researchers no longer have to "reinvent the wheel", i.e., to write the computer code to simulate some well-understood human information processes, but instead can focus on the programming of higher levels of abstractions such as task strategies.

Recent studies demonstrated the effectiveness of cognitive modeling in helping design efficient user interfaces (e.g., Foyle & Hooey, 2008; Gray & Boehm-Davis, 2000; Kieras, Wood, & Meyer, 1997), but many challenges still prohibit its widespread adoption. One challenge is that a cognitive model often has many free parameters that (a) prohibit predictive modeling due to too many unknown factors, and (b) make it difficult to falsify and thus improve an architecture or a model because of excessive degrees of freedom. Addressing these issues imposed by free parameters may have a great positive impact on research that employs cognitive modeling.

Another challenge in employing cognitive modeling to inform the design of user interfaces is to efficiently and comprehensively explore a wide range of task strategies. Cognitive modeling researchers are well aware that for different task environment and different individuals, many parameters of a cognitive architecture need to be adjusted to appropriately reflect the characteristics of a task and an individual person's information processing capabilities (e.g., Anderson, Taatgen, & Byrne, 2005; Byrne & Anderson, 2001; Fleetwood & Byrne, 2006; Meyer & Kieras, 1997b). Few modeling studies, however, have explored the effectiveness of task strategies in explaining empirical phenomena (though see Kieras et al. (1997), Hornof and Halverson (2003), Altmann (2007) for studies that did focus on using strategies to explain data). As demonstrated by several empirical studies (e.g., Gray & Boehm-Davis, 2000; Gray, Sims, Fu, & Schoelles, 2006; Maloney & Zhang, 2010), humans clearly adapt their strategies to the changes in the task environment. To accurately predict task performance, cognitive modeling research should also take into account a wide range of plausible task strategies. Such strategy exploration might sometimes have an added benefit of revealing why people choose one strategy over others (e.g., Gray et al. (2006) shows that in their task, people adapted strategies to minimize task completion time), which can provide useful guidance for narrowing down the strategies that need to be considered in modeling a general class of tasks. However, to efficiently and comprehensively explore strategies in cognitive modeling requires addressing many issues, including: effectively generating the modeling code for a myriad of strategies, efficiently running many models that span across a large strategy and parameter space, rigorously analyzing and comparing the goodness-of-fit of the numerous models, and determining the best-fitting strategy and parameter settings from the numerous configurations. Sometimes, such exploration might find multiple best-fitting strategy and parameter settings, especially when the strategies and parameters are convolved in fitting the empirical data. This research aims to tackle all of these issues, and as a result, this dissertation ventures into unexplored research frontiers in cognitive modeling and provides the first effective solutions to some of these issues.

The goal of this dissertation is to advance our understanding of human multitasking through cognitive modeling and to advance modeling techniques. To achieve this goal, this dissertation work (a) analyzed the behavioral data, including eye movement data, of a highly-interactive, time-critical multimodal dual task, (b) developed a methodological framework to rigorously explore a large space of models with different parameter and strategy settings, (c) built models for each individual participant to capture individual differences, (d) conducted extensive modeling analyses to reliably reveal different multitasking strategies across individuals, and (e) hypothesized on the reasons that such strategies were adopted. The results reveal (a) fundamental human multitasking capabilities and limitations, (b) the nuanced strategies that people use when facing multitasking scenarios, and (c) a promising approach to construct predictive models in other multitasking settings. This research pushes forward the “theory” of cognitive modeling by inventing new methods and combining previous methods to address the challenges posed by free parameters and flexible task strategies. In particular, I present a software framework that automatically generates various models with different parameter and strategy settings, and distributes and run the models on a parallel computer cluster. Such a framework will, in the future, likely prove critical for predicting human behavior.

The remainder of the dissertation is organized into five chapters: Chapter II reviews existing theories on multitasking and recent advances in modeling methodologies. Chapter III presents the multimodal dual task experiment and an analysis of the behavioral data, including eye movement data. Chapter IV introduces the cognitive models I developed for the dual task, and the new modeling framework that can harness the power of the computer cluster. Chapter V compares modeling predictions with human data, presents the best-fitting model for each participant, and discusses how experimental payoff may have influenced the participants' strategies. Chapter VI discusses the implications of this research, including how it improves our understandings about what invariable factors and task strategies affect multitasking performance, sheds light on how people optimize task performance, contributes to improving the usability of multitasking user interfaces, and sets a new standard for model development and validation [1].

[1] Chapters III and IV include published co-authored materials.

CHAPTER II

REVIEW OF RELEVANT RESEARCH

Theories of Multitasking

Previous psychological theory on human multitasking can be largely divided into two groups: One group concerns the invariable human factors that lead to interference among multiple tasks, and the other concerns the role of executive control in multitasking. Both are necessary for understanding and modeling multitasking performance. This section reviews theories in each group and discusses the theoretical basis of this dissertation research.

Invariable Factors in Multitasking

The current predominant multitasking theory in cognitive science and HCI is the multiple resource theory (Navon & Gopher, 1979; Wickens, 2002). This theory postulates that human perceptual, cognitive, and motor capabilities can be thought of as individual resource pools, each with some divisible capacity, and tasks that use the same resource pools would interfere with each other. For example, this theory predicts that two auditory tasks presented at the same time (e.g., dichotic listening) would generate greater interference than an auditory task combined with a visual task.

To further formalize the multiple resource theory, Wickens (2002) proposed that the various human information-processing resources can be categorized based along four dimensions: processing stages, perceptual modalities, visual channels, and processing codes. Each dimension is divided into two components: the processing-stage dimension is divided into perceptual-cognition stage and responding stages; the perceptual modalities are divided into visual and auditory modalities; the visual perceptual modality (visual channel) is further divided into focal and ambient vision; and the processing codes are divided into spatial and verbal codes. Although there are still many debates around this four-dimension taxonomy system, it servers as a good framework to navigate through a plethora of structural human factors concerning multitasking performance. The rest of this section discusses those factors along each dimension and how the two major cognitive architectures in HCI—ACT-R and EPIC—implement them.

The Perceptual Modality and the Visual Channel Dimensions

The perceptual modality dimension is divided into visual and auditory modalities, and the visual channel dimension sits within the visual perceptual modality. Many studies have shown that concurrent cross-modal perceptual processing—simultaneous use of multiple perceptual modalities (e.g., auditory and visual)—can be done more efficiently than concurrent unimodal processing (e.g., visual and visual). For example, Rollins and Hendricks (1980) showed that messages presented only through auditory stimuli are more difficult to process than messages presented partly through visual stimuli and partly through auditory stimuli. The fact that humans have dedicated organs (eyes and ears) and brain regions for processing the two types of sensory stimuli also suggests that there should be less interference across the two perceptual modalities Kandel, Schwartz, and Jessell (2000). The question, though, is what interference exists for information delivered within the same perceptual modality.

Parts of the within-modality interference arises from the physical constraints of the sensory organs. For vision, the main constraint is the limited size of the foveal region. While certain information (e.g., color and motion) can be perceived in the periphery, other information (e.g., small text) requires an eye movement to center the object of interest into foveal vision. Because an eye movement typically takes about 100 ms to initiate and 10–100 ms to complete (Duchowski, 2007), this lag alone can cause considerable interference between visual tasks (Meyer & Kieras, 1997b). For hearing, one of the physical constraints is that sounds with similar frequencies stimulate the same set of mechanoreceptors in the ears (B. C. J. Moore, 1986), and so sounds at the same frequencies can cause strong interference.

Selective attention is another source of the within-modality interference. Selective (or covert) visual attention can bias processing towards one part of the visual periphery (T. Moore & Armstrong, 2003; Sperling & Melchner, 1978), and selective auditory attention can bias processing towards one side of the listening space (Cherry, 1953). It is, however, still not clear whether raw sensory information is filtered by attention (also known as an early selection model, Broadbent, 1958) or by higher-level sensory analysis such as semantic analysis (also known as a late-selection model, Deutsch & Deutsch, 1963), but the common implication of these two possible hypotheses is that information from the stimuli that has not been attended to may be lost.

The cognitive architectures ACT-R and EPIC implement the highly efficiency of cross-modal perceptual processing as well as the interference of within-modality processing. To implement parallel cross-modal processing, both architectures employ separate processors for handling visual and auditory stimuli, and these processors can work concurrently with no interference. To implement the within-modality interference, however, the two architectures have taken different paths. ACT-R relies on selective attention to access information from the outside world; since this attention mechanism can only be directed to one location at a time, an ACT-R model cannot simultaneously perceive multiple stimuli of the the same perceptual modality. By contrast, EPIC's implementation of within-modality interference focuses on the physical properties of the perceptual modalities; for example, visual information from the stimuli is filtered by the graded resolution of the retina, and is selected for further inspection by means of eye movements rather than internal attention. This implementation thus instantiates the multiple resource theory's differentiation between focal and ambient vision. EPIC does not implement selective visual attention because in most real world tasks, visual attention is tightly coupled with eye movements (Findlay & Gilchrist, 2003).

Currently, neither ACT-R nor EPIC implements the frequency interference that occurs in the auditory modality, but Prof. David Kieras is actively working on this area for EPIC (personal comment).

The Code Dimension

The code dimension of Wickens' taxonomy consists of spatial and verbal codes. These two codes represent two hypothesized types of memory storage employed in human information processing (Wickens, 2002). Under the assumption of separate spatial and verbal codes, the multiple resource theory predicts that two tasks utilizing the same codes would incur strong interference, whereas a combination of a spatial task and a verbal task would incur little interference. This prediction has been verified by a few studies. For example, Brooks (1968) found that when recalling information about a line diagram, a task that requires spatial thinking, it takes much longer for a participant to point at the response in a spatial array than to speak the response. By contrast, when recalling information about a sentence, it takes longer to speak the response than to point at it. Research on distracted driving (Recarte & Nunes, 2000) shows that spatial-imagery tasks greatly reduce a driver's visual scan area, whereas verbal tasks do not. These results suggest that spatial thinking and verbal thinking may indeed operate based on different resources.

Although there is some evidence to support the separation between spatial and verbal codes at the perceptual and cognitive stages, it is questionable whether this separation should be extended to the response stage. Wickens argued that the strong interference between responses within the same modality is due to the sharing of the same code. However, this interference could also be explained as the sharing of the same motor modality resource such as manual motor or vocal motor. The latter explanation is arguably more parsimonious and easier to verify because it represents physical and biological separations that can be observed instead of an abstract mental construct—processing codes—that cannot be easily tested.

Preferring a seemingly more parsimonious theory, EPIC adopted the modality hypothesis rather than the processing-code hypothesis in explaining interference that occurs at the response stage. ACT-R then followed suit, deriving its implementation from EPIC's. Their implementation of the response modalities largely follows Rosenbaum (1980)'s motor programming framework. In this framework, motor processing—including manual, ocular, and vocal processing—generally goes through a preparation stage and an execution stage. Each stage only allows processing of one movement at a time. This movement-production bottleneck is the key to explaining the interference between concurrent responses. For example, it predicts strong interference between concurrent manual responses because both hands share the same manual motor processor (Kieras & Meyer, 1997), and the execution or preparation of the next movement has to wait until that of the previous movement is completed. A manual response and a vocal response, however, can be processed more efficiently because they are handled by separate processors and can be executed in parallel without interference.

EPIC and ACT-R's implementation of motor programming can explain many effects that were attributed by the multiple resource theory to the spatial-verbal dichotomy, and can even predict when and how the interference may occur. For example, Martin-Emerson and Wickens (1997) conducted an experiment that consisted of a tracking task and an arrow-discrimination task. The arrow-discrimination task requires the participant to press a key in response to a left or right arrow that periodically appeared above the tracking task. The study found a considerable interference between the two tasks, and both tasks' performance deteriorated as the separation between the two tasks' displays increased. Martin-Emerson and Wickens attributed the interference to the sharing of the spatial processing codes. However, Kieras, Meyer, Ballas, and Lauber (2000) found that a model with just the movement production bottleneck can also explain the effect. Their model shows that the two tasks, though both fall into Wickens' category of spatial tasks, do not always interfere with each other on all three processing stages. In fact, the response selection stage of the arrow-discrimination task was done while tracking was in progress. The multiple resource theory cannot make such detailed inference, nor can it predict how exactly the interaction unfolds in such multitasking scenarios.

The Stage Dimension

In the stage dimension, Wickens' taxonomy separates resources between the response stage and the perceptual-cognitive stages, but it does not separate resources between the perceptual and cognitive stage. EPIC and ACT-R, however, each went a step further to also assert that perceptual and cognitive processing stages are also separated. In each architecture, the cognitive processor and the perceptual processors operate independently without interference. This separation is supported by many brain imaging and brain lesion studies (see Kandel et al., 2000, for a review), which show that decision-making primarily uses the prefrontal cortex, whereas perception uses other brain structures.

The prior discussion concluded that concurrent perceptual processes and concurrent responding processes would each cause interference if they share the same modalities, but it remains to be answered whether there is interference among concurrent cognitive processes. ACT-R and EPIC take different positions on this issue: ACT-R assumes that the cognitive processor can only execute one rule within a cognitive cycle (asserting the cognitive bottleneck theory), whereas EPIC assumes no limitations (asserting cognitive parallelism). This theoretical difference has yet to be resolved, and in many cases both assumptions seem to predict similar human performance (Byrne & Anderson, 2001; Meyer & Kieras, 1997b).

Overall, EPIC and ACT-R are mostly consistent with the multiple resource theory, and in many areas, the two architectures offer more concrete and verifiable explanations. Because EPIC and ACT-R are computational cognitive architectures, they can automatically take into account a variety of human multitasking characteristics such as the the capacity of visual processing over the entire visual field, and the interference between concurrent motor responses. These implementations are vital for studying complex multitasking scenarios that cannot be easily examined through traditional qualitative analysis.

The next section discusses how the above invariable human factors are put together within a cognitive architecture, and how task strategies, a more flexible factor, affects multitasking performance.

The Role of Strategic Control

Many recent studies (Brumby, Salvucci, & Howes, 2009; Hornof & Zhang, 2010; Meyer & Kieras, 1997b; Monsell, 2003; Zhang & Hornof, 2014) demonstrate that different multitasking strategies can impact task performance. These strategies typically concern when and how to switch tasks. For example, in a dual task experiment, Schumacher et al. (2001) showed that just by instructing the participants to take a “daring” strategy to execute Task 2’s response as soon as possible, the so called psychological refractory period—a period in which the participant is assumed not able to process a second task—can be shortened and even completely eliminated.

Cognitive modeling is particularly suitable for studying task strategies because most cognitive architectures are equipped with a production system that can formally describe and reason about strategies. Production systems were first proposed as a formal method to describe humans’ problem solving processes (Newell & Simon, 1972). Under such a system, task strategies are written by an analyst (a person who builds cognitive models in the context of this research) in a form of if-then statements called production rules. During each cognitive-processor cycle, which lasts a simulated 50 ms in most cognitive architectures, every production rule’s conditions are tested to determine if they can be satisfied by the model states (including memory and processor status). If a rule’s conditions are matched, the actions of the rule are then executed. This characterization of human decision processes as a collection of low-level stimulus-response (if-then) pairs seem to capture well the capabilities and constraints of human decision making.

The rest of this section reviews research on multitasking strategies in the context of cognitive modeling. The ability to precisely specify strategies in a formal language has greatly advanced the research of strategic control.

Multitasking Executive Process

Models for multitasking typically consist of two types of processes: task processes and executive processes. Task processes are responsible for performing a single task, whereas executive processes are responsible for coordinating task processes to achieve multitasking. Executive processes themselves generally do not have enough information to perform tasks, but they maintain task priorities, and control when to suspend and resume a task. In ACT-R, executive processes are usually embedded in task processes, meaning that the production rules used for executing single tasks often also carry the role of managing task switches. The executive processes in EPIC, however, are often decoupled from the task rules, and because EPIC can fire multiple rules in each cycle, the model can perform executive processes simultaneously with task processes. This decoupling gives EPIC an unique advantage for studying different forms of executive processes.

In their seminal work, Meyer and Kieras (1997a, 1997b) applied EPIC to study the psychological refractory period (PRP), an important multitasking phenomenon that may underly the fundamental limitations of human multitasking. In PRP experiments, two choice-reaction tasks are performed concurrently, with the second task's stimuli appearing slightly after (typically from 50 ms to 2 s) the first task's stimuli. Typically, participants are required to respond to Task 1 before Task 2. It is found that in this dual task situation, participants take longer to respond to Task 2 than they do when they perform Task 2 separately. This additional reaction time was initially thought to be caused by a hypothetical psychological refractory period in which no processing can be done but, more recently, it is believed to be caused by a bottleneck processing stage that can only process one subcomponent of a task at a time.

Figure 1 illustrates one widely accepted bottleneck theory that can explain many PRP results: the movement-production bottleneck hypothesis (Keele, 1973). This hypothesis assumes that each motor modality can only execute one movement at a time, and if both tasks require the same motor responses, one of them has to be delayed. The theory predicts that as the stimulus onset asynchrony (SOA, the time interval between S1 and S2) increases, the slack time before movement production would compress, which reduces the effect of the movement production bottleneck. This is indeed observed in many experiments (e.g., De Jong, 1993; Pashler, 1989). However, another competing theory, the response-selection bottleneck hypothesis, also explains these observations well. More experiments were conducted (e.g., Pashler, 1989; Schumacher et al., 1999, 2001), and evidence for and against either one theory was found, leaving the issue unsettled.

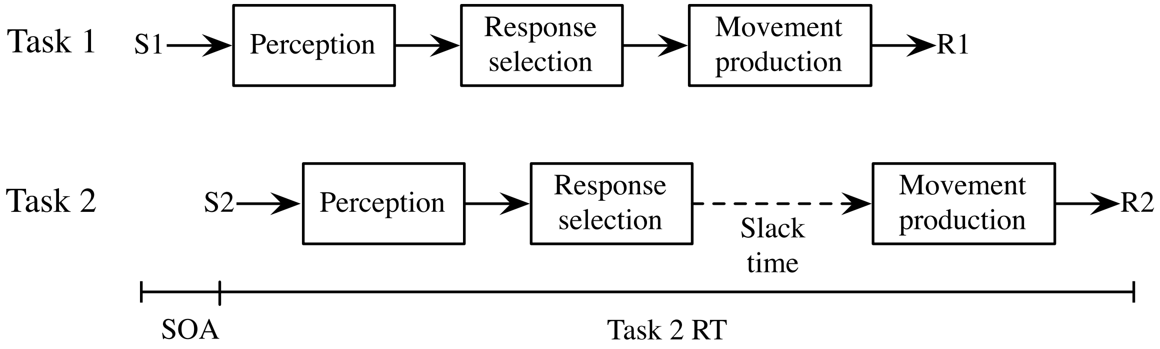


FIGURE 1. A stage model that illustrates the movement-production bottleneck hypothesis. The movement production of the second task is delayed because only one response can be executed at a time by the same motor processor. SOA (stimulus onset asynchrony) is the time between the appearance of the two tasks' stimuli.

Using EPIC, Meyer and Kieras (1997a, 1997b) showed that with the assumption of the movement-production bottleneck and a set of carefully constructed executive processes, the seemingly conflicting results observed in many PRP experiments can all be reconciled. The EPIC cognitive architecture does not assume a response-selection bottleneck, but as discussed earlier, its implementation of the motor processors is consistent with the movement-production bottleneck theory. For the executive processes, Meyer and Kieras proposed a so-called *strategical response-deferment* model, which caches the results of response selection for Task 2, defers Task 2 movement-production if necessary, and then resumes Task 2 after Task 1 finishes.

Figure 2 illustrates the strategical response-deferment executive process. At the beginning of a trial, the executive process enables both task processes, and allows perceptual processing and response selection for both tasks to be proceeded without holdup. Task 1 also directly proceeds to motor processing, while Task 2's motor processing might be delayed. Once Task 1 completes, the executive process then permits Task 2 responses.

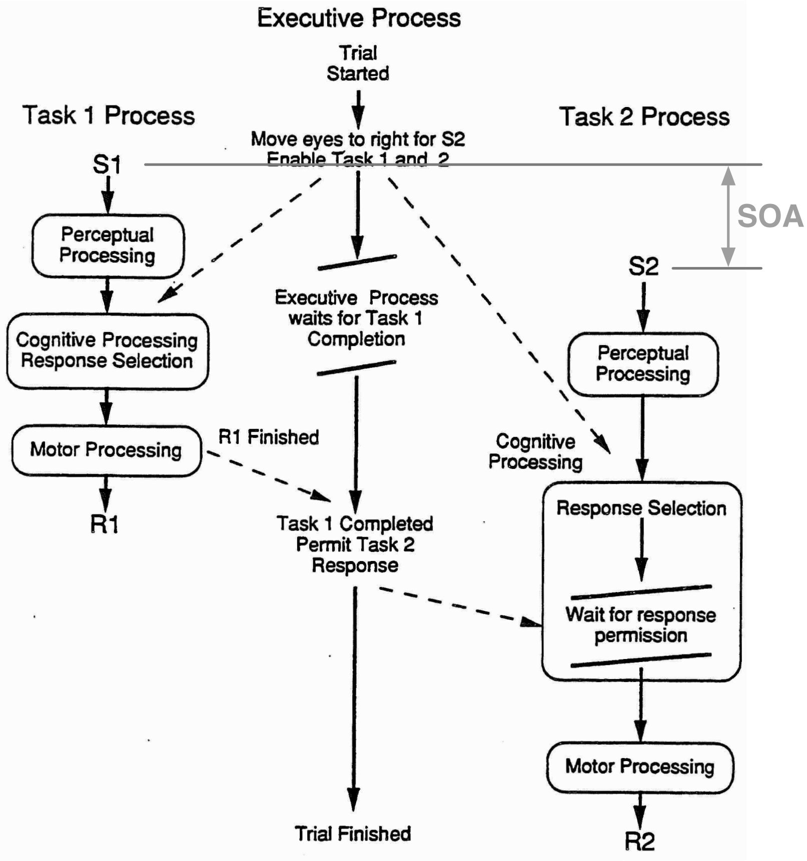


FIGURE 2. The strategic response-deferment executive process and the two task processes. Response execution for Task 2 is delayed until Task 1 is finished. Image adapted from Meyer and Kieras (1997a).

Depending on the length of the stimulus onset asynchrony (SOA) and the duration of the perceptual processing and response selection stages, however, the model may take different execution paths. For example, if Task 1 finishes before Task 2's response selection finishes, the executive process will switch Task 2 from a deferred mode to an immediate mode such that Task 2 motor processing would follow response selection immediately without the caching and retrieval processes required by the deferred mode. It is this flexibility and the detailed implementation of the perceptual and motor processes that enabled the EPIC models to predict various PRP results.

Meyer and Kieras' study on PRP was one of the first multitasking studies that put emphasis on task strategies. Many modeling studies then followed this path and exploited the formalism of production systems to examine various plausible multitasking strategies.

Sequential-Ordering vs. Partial-Overlap

Besides the PRP tasks, EPIC was also applied to other multitasking situations where a continuous task such as tracking is performed in parallel with a choice-reaction task. A continuous task can cause many resource conflicts because of the task's constant demands on perceptual-motor resources. One continuous task commonly used in laboratory settings is the tracking task, which requires participants to constantly monitor a tracking target and adjust the position of a tracking cursor, such as with a joystick. When performed concurrently with other tasks that also require foveal vision and manual motor control, the tracking task is inevitably interrupted.

To model such multitasking scenarios, Kieras et al. (2000) explored two types of executive processes, each interleaving tasks in a different way. The first type of executive imposes a strictly sequential order on the continuous task and the other task, whereas the second type of executive allows as much overlap as possible. Kieras et al. modeled Martin-Emerson and Wickens (1997)'s dual task using each of these two different executives. As discussed previously, this dual task consists of a tracking task and an arrow-discrimination task. In the sequential-ordering model, the tracking task is immediately suspended when the arrow appears on the screen, and is resumed only after the arrow is responded to. In the partial-overlap model, the tracking task is resumed at an earlier point in time, just after the visual information of the arrow has been acquired; after resuming, the model then performs the tracking task while simultaneously selecting and preparing the manual response for the arrow-discrimination task, which greatly reduces interruption to the tracking task. The results show that the partial-overlap model fits the empirical data better than the sequential-ordering model, suggesting that participants likely used the overlapping strategy to improve performance on both tasks.

General Executive Processes

In addition to the executive processes discussed above—which are tailored to individual experiments—Kieras et al. (2000) also explored *general executives* that can be readily applied to different task contexts. Inspired by the way computer operating systems manage processes, Kieras et al. proposed two general executives, a conservative executive and a liberal executive. The conservative general executive manages motor resources using a first-come, first-serve algorithm. It assumes that task processes are "impolite" and grab resources without asking for the general executive's permission. Once a task acquires a resource, the executive simply blocks further access to that resource. By contrast, the liberal general executive works with "polite" task processes. A polite task process always requests the executive before using a resource and would not proceed until permission is granted. This schema gives the general executive greater control such as by enabling it to impose priorities. Kieras et al. examined these two types of general executives by applying them to model Martin-Emerson and Wickens' dual task. The results show that both types of general executives can fit the empirical data very well, and in particular the liberal general executive's reaction time for the arrow-discrimination task had only 7.8% mean absolute prediction error. These results show that even without task-specific knowledge, the general executives can still fit the data reasonably well. Kieras et al. then further improved the goodness of fit by adding some task-specific knowledge to the model such as when to preallocate motor resources, because it is reasonable to assume that, with practice, participants would learn the task structure and anticipated tasks. Thus, the general executives not only serve as the basis for building specialized executives, but perhaps also for modeling some aspects of how novices transform to experts.

Threaded Cognition in ACT-R

Threaded cognition (Salvucci & Taatgen, 2008) is a general multitasking theory that has been developed and used in ACT-R. Threaded cognition is similar to Kieras et al.'s general executives in that it also assumes an operating-system style process management. In threaded cognition, task processes compete for resources in a greedy, polite manner. That is, a task process requests resources when needed and releases them immediately when no longer required. If a task process requests a resource that is already being used, this request is put on hold until the resource is released. If multiple task processes wait for the same resource, threaded cognition favors the task that least recently used the resource. This conflict resolution policy is certainly not the only way to interleave tasks, but it is a simple, preliminary solution for balancing processing among tasks.

To implement this conflict resolution policy, threaded cognition was built into the ACT-R cognitive architecture as a separate module. This is different from how general executives are implemented in EPIC, which are constructed as modularized subsets of production rules, just like other task strategies. Although embedding threaded cognition into the architecture potentially reduces the programming effort for an analyst, the implementation is less flexible and thus makes it harder to use threaded cognition to explore various other task-interleaving strategies.

Summary

This section reviewed the existing theories on the cognitive and sensorimotor factors involved in multitasking, and several hypotheses of how executive processes might function in multitasking. This section discussed how the two major cognitive architectures used in HCI have accounted for most of the cognitive and sensorimotor factors, and that one architecture, EPIC, was also used to explore different forms of executive processes. Because EPIC permits explorations of executive processes, and because of its more sophisticated implementation of the perceptual and motor processes, it is used in this research for modeling.

The expressiveness of production rules, especially when any number of them can fire in parallel, enables a broad exploration of task strategies. Though this flexibility is generally good for modeling many different tasks, it also exacerbates the problem of empirically validating the assumptions of cognitive architectures and task strategies. This is because when there are many different plausible task strategies and especially when the empirical data do not provide enough constraints for modeling, it is likely for analysts to find a set of strategies that happen to fit the data but are actually different from the participants' real strategies. It is also likely to find a set of strategies that overfit the noise in the data rather than the more general behavioral characteristics. To address such problems, several recent studies attempted to improve modeling methodologies to provide constraints and guidance for strategy and parameter exploration. The next section reviews these studies.

Recent Methodological Advances in Model Development and Validation

As previously discussed, one challenge in cognitive modeling is that a model often has too many free parameters that prohibit it from making useful predictions, and which make it difficult to falsify or improve the model. The parameters in cognitive models represent aspects of human information processing that might change for different individuals and task environments. For example, Fitts' law uses a logarithmic equation to predict the duration of a pointing movement (such as touching an object or moving a mouse cursor to a button) given the target width and movement distance. This logarithmic equation has two parameters that can change depending on the device (Card, English, & Burr, 1978) or limb (Langolf, Chaffin, & Foulke, 1976). A cognitive model can have many such parameters because the model usually consists of several modules (such as visual-perceptual, cognitive, and manual-motor) and each with several parameters of its own. Though researchers can reasonably "guesstimate" many parameter settings based on the psychological literature, it is often necessary to explore (or fit the data) for a new, specific task (such as the Fitts' law parameters for a new gestural interface).

Besides numeric parameters, many cognitive models, especially those built using a cognitive architecture, also have a special set of "parameters" in the task strategies. Different tasks often require different strategies, and even for the same task different people may also use different strategies (e.g., Howes, Lewis, & Vera, 2009). Determining plausible hypotheses of the task strategies used by participants is often more difficult than determining appropriate numeric parameter settings for perceptual, memory, and motor processes, because the effects of task strategies are often nonlinear and indirect. Changing a task strategy is similar to changing the equation of a mathematical model as opposed to changing the equation parameters.

Given the variability of parameters and strategies, it is important for a cognitive modeling study to report how the parameters were searched and what strategies were considered, and yet this is rarely done (though see Kieras et al. (2000), Hornof and Halverson (2003), and Hornof and Zhang (2010) for examples that presented models with alternative strategies and showed how those models were ruled out). Many studies (e.g., Byrne & Anderson, 2001; Salvucci, 2006) only report the best-fitting models' parameter and strategy settings without mentioning what other settings were considered, and thus it is impossible to know whether there could be better parameter and strategy configurations. Also, as pointed out by Roberts and Pashler (2000) and illustrated by Figure 3, a model is best evaluated when the model's entire prediction space resulted from all parameter and strategy configurations is known. This is because if multiple models can fit the same data, it is still possible to decide that the model with the narrowest prediction space is the best model since a narrow prediction space indicates that a good fit is less likely due to a specific parameter and strategy configuration, but more likely because of the underlying model assumptions. Without knowing alternative parameter and strategy settings and their predictions, it is impossible to evaluate a model comprehensively.

Because of this neglect on reporting how models were explored, the methods for developing and validating cognitive models have not been improved until recently. In the last decade, several studies focused on the methodological issues involved in cognitive modeling, in which many approaches were proposed and formally examined. This section next reviews these studies, and discusses the issues each method addresses as well as how this dissertation further improves upon these methods.

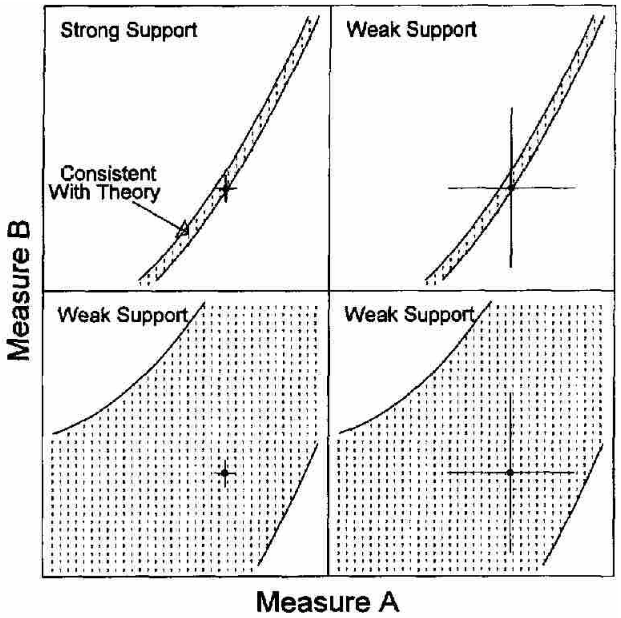


FIGURE 3. A sketch from Roberts and Pashler (2000) to show the four possible relationships between theory (model) and data. Measures A and B are measures of behavior, such as reaction time and accuracy for a memory recall task. In each panel, the cross indicates the range of the observed data, and the dotted area indicates the ranges of the model's predictions. In all four panels, the model's predictions always cover the observed data, but the model in the top-left panel gains the strongest support from the data.

Measures of Goodness-of-fit

One hurdle in evaluating models is to decide on a measure of goodness-of-fit. Schunn and Wallach (2005) thoroughly examined the following commonly used measures in cognitive modeling:

- Pearson's r. The correlation coefficient between model predictions and the empirical data.

- r^2. The variance of the data accounted for by the model. In this dissertation, to follow the conventional definition in cognitive modeling research, r^2 is defined as Pearson's r squared (see Schunn & Wallach, 2005).

- *MAD*. The mean absolute deviation between model predictions and the empirical data.

- *RMSD*. The root-mean-squared deviation between model predictions and the empirical data. Sometimes also referred to as RMSE (root-mean-squared error).

The first two measures compare the trends in the model predictions with those in the human data, and the second two reflect the absolute deviations of the model predictions from the human data. Shunn and Wallach argue that r^2 is better than r in capturing trends because r^2 can better separate models that have strong correlations with the data. They also conclude that RMSD is better than MAD in measuring absolute deviations because by squaring the deviations, RMSD deemphasize minor changes in the data that are likely caused by noises, and put more weight on fitting substantial changes. Schunn and Wallach (2005) also proposed new goodness-of-fit measures that scale the deviations between model predictions and human data by the standard error of the human data. They argued that because these scaled measures of deviations are unitless, they facilitate comparisons across different types of data (such as reaction time and error rates). However, they also point out that these measures can be problematic for data points with a small sample size and for data with zero variances. Because the research on the measures of scaled deviations is still ongoing, I chose to use RMSD for this dissertation research.

Another measure, *likelihood*, is potentially a useful general goodness-of-fit measure. Likelihood measures how probable the observed data are to be generated by a model given the assumptions built into the model. This measure is particularly useful for evaluating discrete behavioral data that cannot be quantified by r^2 or RMSD. For example, our visual search model (Zhang & Hornof, 2013) has to predict which of the many on-screen objects a participant is likely to look at. The prediction error in this case (when the model mispredicts the object a participant was looking at at a particular point in time) cannot be quantified by r^2 or RMSD, but can be measured by likelihood if the model assigns a probability to the fixation on each object. The drawback of the likelihood measure is that the model needs to assign a likelihood to every possible prediction. In other words, the entire distribution of the predictions needs to be predetermined, which would typically require many runs of the model to generate an estimate of the distribution. For many studies, this is impractical and the gain is very small. Thus, for this dissertation research, I still use RMSD and r^2 as the primary measures of goodness-of-fit.

Systematic Exploration of Parameter Settings and High-Performance Computing

A systematic search over a model's parameter space can uncover the best-fitting parameter settings in an efficient and replicable manner, and can also help generate the model's entire prediction space for comprehensive model evaluation. There are several algorithms for searching through a parameter space, and perhaps the easiest to implement is grid sampling. This method defines a finite search range for each parameter, and then samples several settings at equal intervals over the range. Different parameters are sampled independently such that the number of parameter configurations grow combinatorially as the number of settings sampled for each parameter grows. Grid sampling tends to work well when free parameters are few, but when free parameters are many, the number of samples needed to cover the entire search space increases dramatically. Grid sampling was applied in Zhang and Hornof (2014), and it helped us find accurate parameter settings for each participant.

Grid search works well for parameters that have a linear effect on model predictions; when the effect is nonlinear, some regions of the model prediction space may change more steeply than other regions, requiring more samples to closely approximate how the parameters affect predictions. To determine where to place the samples, Gluck et al. (2010) applied an algorithm that places samples based on how close the predictions are to the human data. Regions in which predictions are closer to the human data receive more samples, effectively increasing the resolution as the search moves closer and closer to the best-fitting settings.

When the sampling space is very large, evolutionary algorithms may be used to efficiently find an approximately best-fitting model. Evolutionary algorithms work similarly to Gluck et al. (2010)'s method in that it also places more samples at the regions of the search space that are more likely to have the best-fitting setting (global maximum). But evolutionary algorithms also typically implement heuristics that help avoid local maxima. My previous study (Zhang & Hornof, 2013) applied evolutionary algorithm to optimize a visual search model with 12 parameters. The algorithm was able to find good parameter settings within 360,000 runs, whereas a grid sampling algorithm that tests 10 settings for each parameter would need a trillion runs.

No matter which algorithm is applied, the parameter search has to go through many different models, and this may require enormous computational power unavailable to a single desktop computer. One solution to this problem is to use high-performance computer clusters. Gluck et al. (2010) is one of the first modeling studies that take this approach to explore parameters. My previous research (Zhang & Hornof, 2014) takes this approach further by also enabling search of task strategies. This dissertation continues to use the framework, which will be discussed in Chapter IV.

Sensitivity Analysis

As discussed previously, many cognitive modeling studies (e.g., Byrne & Anderson, 2001; Fleetwood & Byrne, 2006; Salvucci, 2006; Taatgen, Van Rijn, & Anderson, 2007) often only report the best-fitting model and does not provide enough evidence to rule out other possible alternative models. This practice may lead to incorrect interpretations, such as that all the specific parameter and strategy settings of the best-fitting model are necessary to explain the data. But it is possible that a model is insensitive to the changes in some parameters, and the best fitting may be achievable by a range of settings as opposed to just one particular parameter configuration. To determine whether a model is affected by changes in free parameters or strategies, a sensitivity analysis is needed.

One way to conduct an informal sensitivity analysis is to create visualizations that show how the predictions change as the parameters change. Figure 4 shows an example of such visualizations taken from Gluck et al. (2010). The graph shows a model's goodness-of-fit, measured in RMSE (root mean squared error), as a function of two ACT-R architectural parameters, alpha and egs[1]. From the graph, it can be seen that the model

[1] Both parameters determine the utility of a production rule that is used by ACT-R's reinforcement learning (RL) mechanism. Alpha is the learning rate parameter for RL, and egs controls the amount of noise associated with the production utility

is insensitive to changes in alpha, and sensitive to changes in egs when egs is around 0.99. This shows that the model's goodness-of-fit does not depend on alpha, which could be a useful piece of information for other analysts who want to model a similar task as the analysts can just focus on finding the best-fitting setting for egs.

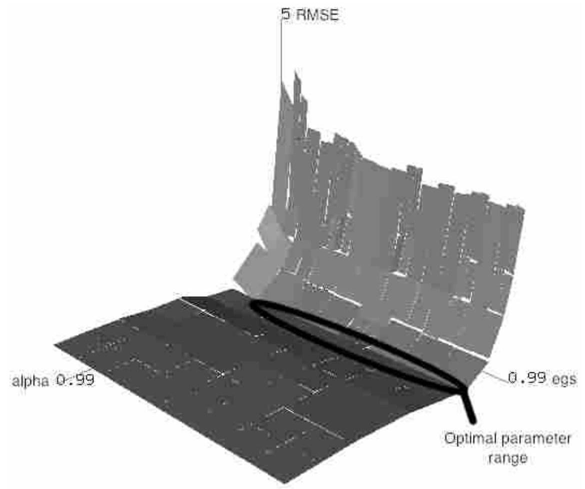


FIGURE 4. A surface plot of a model's goodness-of-fit, measured by RMSE, as a function of two ACT-R architectural parameters. Lower RMSE corresponds to better fit to the human data. The graph shows that the model is insensitive to alpha because the surface does not change along the alpha axis, but the model is sensitive to egs because the surface takes a sharp turn around $egs = 0.99$. The graph is taken from Gluck et al. (2010).

In this dissertation research, I developed a more formal way of conducting sensitivity analysis which examines the model's sensitivity to parameters as well as strategies. Merely relying on visualizations for sensitivity analysis is insufficient, especially when there are too many free parameters to visualize all together. My method combines several statistical analysis techniques with visualizations to help determine decisively which strategy settings are necessary for fitting the data and which are not. The method will be introduced in detail in Chapter V.

Parameter Calibration

If a model has many free parameters, it is typically easy to fit the model to the human data because the free parameters can be adjusted to produce a wide range of predictions (Roberts & Pashler, 2000); reducing the number of free parameters can thus narrow the prediction space and make it easy to falsify or improve a model. Free parameters can be reduced through *parameter calibration* (Howes et al., 2009), which determines parameter settings based on parts of the empirical data that are not used to evaluate the model's goodness of fit and that are closely related to the processes that the parameters directly affect. For example, many tasks involve pointing movements such as moving a mouse cursor to click a button. For these pointing movements, the Fitts' law parameters do not have to take on the default setting or be varied to fit the overall task performance. Rather, they can be determined with some carefully extracted mouse movement data. Such calibration procedures will reduce the variability of the model when fitting the evaluation data, and hence increase the support for the model if a good fit is found. This research adopted this approach and reduced the number of free parameters in the model to just one, which would seem to improve the validity of the results.

Use Eye Movement Data to Constrain the Model

Another method to constrain parameter and strategy variations is to use more detailed data that are collected at the intervals of tens of milliseconds to fit the model. When fitting to high-level data such as the trial completion time, which typically lasts more than a few seconds for HCI tasks, numerous factors are involved, and many combinations of parameter and strategy settings may fit the high-level measures. By contrast, more detailed data such as those contained in mouse movement or eye movement data provide more probe points to the task processes, which then provide tight constraints for model fitting.

Our previous research (Hornof & Zhang, 2010) demonstrated how eye movement data lead to more accurate understandings about task strategies than do reaction time data alone. In that study, we built a series of models to explain the average performance of a multimodal dual task, the same task that will be introduced in Chapter III. The first model used a hierarchical-sequential strategy, which performs only one task at any point in time. This initial model fit the reaction time data well, but an examination of the eye movement data revealed that the model predictions about the time course of the eye movements were in fact off by 91%. Based on the eye movement patterns, a new model is developed that partially overlaps the two tasks. This model produced better fit with reaction time and a much better fit with the eye movement data. The eye movement data thus helped refute the hierarchical-sequential strategy, and supported the more flexible, partially overlapping multitasking strategy. This dissertation research continues to use eye movement data, and demonstrates in more details how eye movement data help reveal participants' strategies.

Cognitive Bounded Rational Analysis

Cognitive bounded rational (CBR) analysis (Howes et al., 2009) is another approach to narrowing a model's prediction space, and it is distinctly different from the above approaches in that it does not seek to use empirical data to constrain the predictions. Instead it relies on one assumption: People are bounded rational such that they tend to adopt strategies that maximize the expected utility, the perceived usefulness of a good or experience, achievable under the constraints of their perceptual, cognitive, and motor abilities. Under this assumption, only the strategies that lead to the best payoff should be considered because they are the strategies that participants would use.

Following the CBR analysis, if the model with the best-payoff strategy fits the human data, then the analyst should conclude that the model is correct; otherwise, the analyst should conclude that either the utility function (which maps performance to utility such as monetary rewards) or the model's underlying assumptions about human information processing is incorrect. This analysis thus dramatically reduces the number of plausible models and makes it easy to falsify basic model assumptions.

Several studies demonstrated the effectiveness of this approach. Howes et al. applied this analysis to modeling Schumacher et al. (1999)'s psychological refractory period experiment, and found that indeed the strategies that maximize each individual's payoff, as determined by the same payoff evaluation function used in the original experiment, also fit the reaction time data well. In a similar line of work, Gray et al. (2006) showed how strategies that minimize the time cost of a task accurately predict the observed performance. For more real-world tasks, Brumby et al. (2009) showed that when asked to dial a ten-digit phone number while driving, the most commonly used strategy—entering digits in chunks of 3-3-4—leads to a better balance between driving and dialing performance than does chunking the digits in any other way. These studies demonstrated that cognitive bounded rational analysis is potentially an effective approach for predictive modeling as researchers could potentially just use the model with the highest payoff to predict participants' performance.

Despite the above success, the assumption of bounded rationality is a strong one that may be easily violated. For example, Fu and Gray (2006) shows that in an information-seeking task, people seem to use a local-maximization decision rule that eventually leads to suboptimal overall performance. This result leads to Gray et al. (2006, p. 463)'s following claim:

> The soft constraints hypothesis predicts optimal performance only in tasks where maximizing the expected gains and minimizing the expected costs of interactive routines (i.e., over 1/3 to 3 sec) is congruent with an optimal strategy at the global task level. In environments that violate this property, the soft constraint hypothesis predicts persistently suboptimal performance.

Here, Gray et al. proposes a weaker form of bounded rationality: People tend to optimize locally with respect to short interactive routines that last about 1/3 to 3 seconds. This local optimization does not necessarily lead to globally optimal performance, because globally optimal performance in some tasks may require people to temporarily sacrifice performance in parts of the task such as spending more time exploring better options than settling with the current best.

This dissertation research found evidence that supports this local maximization hypothesis, which suggests that this might be a more reliable assumption for predictive modeling than the cognitive bounded rational analysis.

Summary

This section reviewed a range of methods used by previous research to improve the rigorousness and validity of modeling studies. These methods contribute to a methodological framework for principled model explorations. This dissertation combined these methods, and developed new ones to expand model exploration to task strategies. By demonstrating these approaches with a time-pressured, multimodal dual task experiment, I show that principled model exploration is achievable even when modeling complex cognitive tasks.

CHAPTER III

THE EXPERIMENT AND THE EMPIRICAL DATA

This chapter is developed based on Hornof, Zhang, and Halverson (2010), which was published in the Proceedings of the 2010 SIGCHI Conference on Human Factors in Computing Systems. My coauthors conducted the experiment, and I performed the data analysis. I also assisted in writing the original manuscript of the paper.

The multimodal (auditory and visual) dual task experiment was originally designed by Ballas, Heitmeyer, and Pérez-Quiñones (1992) at the Naval Research Laboratory in Washington, D. C.. It has proven useful for the development of detailed computational cognitive models of multitasking (Kaber & Kim, 2011; Kieras, Ballas, & Meyer, 2001; Kieras et al., 2000). Our lab, primarily my advisor Professor Anthony Hornof and his former Ph.D. student Dr. Tim Halverson, replicated the experiment but with several important extensions, including:

- Eye movements are recorded to capture a more detailed account of how people interleave the two subtasks.
- A gaze-contingent experimental paradigm is introduced. Specifically, in some conditions, to simulate two displays that are separated by a substantial visual angle, eye tracking data are used in real-time to hide objects on the display that is not currently being looked at.
- As in the original experiment, auditory cues are spatialized such that they appear to come from the locations in a three-dimensional space where the stimuli reside. However, the transformation scheme that is used to map the visual stimuli to 3D auditory locations is different than that of the original experiment (see Hornof et al., 2008).

- Participants are rigorously trained, financially motivated, and given extensive feedback so that their performance approaches that of an expert.

This section describes the experimental setup and presents the important empirical data, including eye movement data, that reveal more than just the effects of the experimental condition manipulations, and that permit me to tell a rich story of the cognitive strategies that people develop in complex multimodal multitasking scenarios.

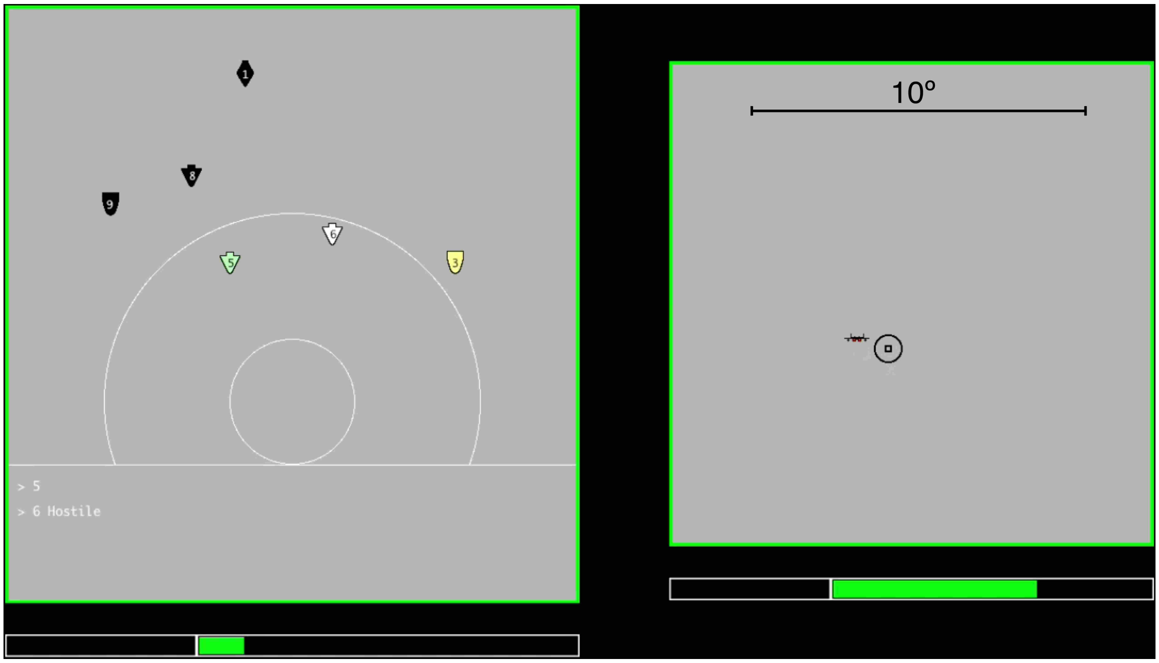
Experimental Setup

Two tasks were performed in parallel: a tracking task and a radar classification task. Figure 5 shows the visual displays for each of the two subtasks, which were presented side-by-side on a single computer monitor, the radar on the left and the tracking on the right. The tracking task is considered the primary task because it requires continuous visual attention and manual responses, like steering a car, whereas the classification task is considered secondary because it permits intermittent visual monitoring and requires just fifty-seven responses across each eight- or nine-minute session.

Task Procedure

Radar Classification Task

In the classification task, small icons referred to as *blips* appeared at random locations in the top half of the radar display, and moved slowly down the display. The participant's task was to key in, for each blip, the single numerical digit on the blip along with a single-key classification of hostile or neutral. Participants were trained to use the keypad without looking.



```
> 5
> 6 Hostile
```

FIGURE 5. A screenshot of the multimodal dual-task display. Radar blips were black before they were ready for classification; red, green, or yellow when ready-to-classify; and white after classification. Progress bars below the displays indicate that the participant is doing well on tracking (with lots of green), but needs to work harder on classification.

There were three types of blips—fighter aircraft, missile sites, and cargo airplanes—represented by three shapes: an arrowhead, a bullet, and a diamond. While on the display, each blip maintained a constant speed and direction, but went through several changes in appearance. Every blip started as black. While black, a blip's hostility classification could not be keyed-in but could be determined based on its shape, speed, and direction, according to the following rules:

Fighter aircraft Hostile if heading for the inner circle of the radar screen; otherwise neutral.

Missile site Hostile if its trajectory intersects the outer circle (the missile sites only move vertically); otherwise neutral.

Cargo airplane Hostile if moving fast; neutral if moving slowly.

After a blip was on the screen for four to twenty-nine seconds ($M = 13$ s, $SD = 5$ s), each blip changed from black to red, green, or yellow, at which point it was ready to be classified, and response time started. Two of the three colors directly identify blip hostility: Red indicates hostile and green neutral. For yellow blips, however, the participant must apply the rules stated above to classify the blips. After classification responses were keyed in for a blip, the blip turned white. All blips disappeared ten seconds after they became ready to classify, whether or not they were classified.

In some conditions, spatialized auditory cues were used to signal a blip's initial appearance and color change. Blip appearance was cued with a (0.1 s) woodblock sound. The color change was cued with an alarm that also indicated the blip's color: Red was cued with seven pulses of 740 Hz (which together lasted 1.1 s); green with one pulse of 385 Hz (0.5 s); yellow with three pulses of 520 Hz (1.5 s). Auditory cues were mapped to visual locations using the most effective transformation discussed in Hornof et al. (2008). All cue volumes were normalized before spatialization.

Blips appeared in *waves* containing 1, 2, 4, 6, or 8 blips. Each session included sixteen waves: three with one blip, five with two, four with four, two with six, and two with eight. Small waves (of one or two blips) had at least 4.5 s between blip color changes. Large waves (with four, six, or eight blips) had a blip changing color roughly every 2.7 s, with least 2 s between blip color changes. Waves were separated by short periods in which no blips appeared on the radar display. Most of these tracking-only periods lasted for 1 s, but in each session two tracking-only periods with no blips to classify were extended to 10 s to measure tracking performance in single task condition. The ordering of waves and blips was varied across sessions.

Extensive visual and auditory feedback motivated good performance. Every time a blip was correctly identified, a pleasant "cha-ching" sound was played. Every time a blip was incorrectly identified or the participant entered an invalid key (e.g., the number for a blip not currently on the screen), an annoying buzzer sounded. Other errors (e.g., entering the blip hostility more than 750 ms after the blip number) resulted in a distinct but less annoying "bloop" sound. Financial incentives for the classification task were as follows: Each blip carried a bonus of up to six cents. Until it was classified or disappeared, one cent was lost per second. Every time a blip was incorrectly classified, all bonus plus an additional five cents was lost. Other error cost one cent. A status bar below the radar display indicated how much money the participant earned with the previous ten blips.

Tracking Task

In the tracking task, the participant moved a joystick to keep the circular tracking cursor over a moving target. As in the original experiment, the joystick affects both the acceleration and velocity of the tracking cursor. The task demanded constant attention because the target moved continuously in an unpredictable manner.

Tracking error, the distance from the center of the tracking cursor to the center of the target, was recorded at a frequency of 12 Hz, or every 83 ms. A summary statistic, root-mean-squared tracking error, was calculated for every session and presented to the participant at the end of each session.

Financial incentives for the tracking task were as follows: For tracking error calculated every 83 ms, if the error is small (less than 20 pixels), the participant earned 0.6 cents. If the error is large (greater than 40 pixels), the participant lost 0.6 cents. Visual feedback helped to motivate good performance: When the participant was making money, the circle was highlighted in green; when loosing money, in red. A status bar below the tracking display reflected the average tracking error of the past 40 seconds.

Experimental Design

Besides the stimulus factors involved in the classification task (blip type, blip color, and wave size), there are also two within-subject experimental factors concerning both the classification and tracking task: (a) peripheral visibility on or off, and (b) auditory cues present or absent. Each session is a unique combination of these two factors.

Peripheral visibility manipulated whether participants could see the contents of the display—radar or tracking—that they were not currently looking at. When the peripheral visibility is on, all visual information is available all the time. When it is off, a participant can only see the information (blips or tracking icons) on the display that he or she is currently looking at, with each display updated within 40 ms of the eyes (point of gaze) arriving or leaving. This simulates a task environment in which visual displays are separated by enough distance such that they cannot be monitored with peripheral vision.

Auditory cues present or absent manipulated whether sound alerts were played to indicate blip appearance and color change events.

Participants

Twelve participants completed the experiment, but two were excluded from analyses due to poor tracking performance. The ten remaining participants (six female), between the ages of 18 and 46 (M = 25.1), were from the University of Oregon and surrounding communities. Each participated on three consecutive days, for roughly one and a half hours per day, and completed four sessions per day. Each session lasted eight to nine minutes and presented a unique combination of the two factors of peripheral visibility and presence or absence of auditory cues; orderings were counterbalanced. Participants were trained to criteria for each of the two tasks individually before starting the first session. Rewards for each subtask were reported after each session. Participants earned a base payment of ten dollars per hour plus an average of eleven dollars in bonuses per day.

Apparatus

Figure 6 shows the experimental setup. Visual stimuli were presented on a 1280×1024 LCD display attached to a Dual 2.5GHz PowerMac G5 running OS X. Spatialized audio was generated using a VRSonic SoundSim Cube spatialized audio server and was delivered to the participants via Sennheiser HD250 headphones. Eye movements were collected using a 120 Hz LC Technologies dual-camera eye tracker. A chin-rest was used to maintain a constant eye-to-screen distance and a stable head position. The experimental source code was acquired from the original experimenter and rewritten (in C++ using Apple XCode) so that the experiment software could interact with the VRSonic SoundSimCube and the eye tracker system in real time via TCP/IP. Two technicians staffed the three systems during all data collection.

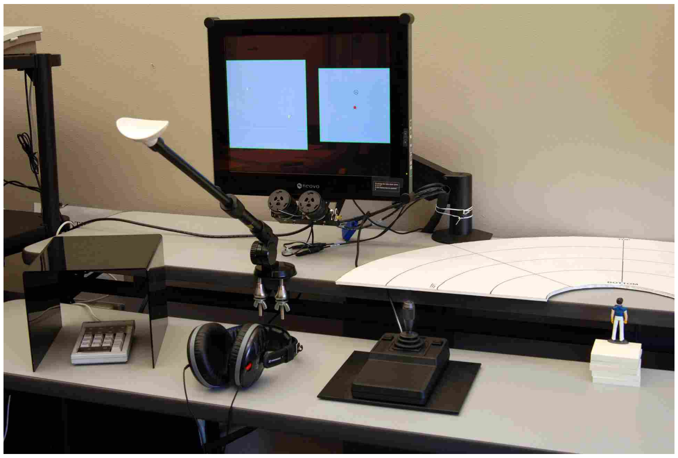


FIGURE 6. The experimental setup including the visual display, headphones, chin-rest, keypad, joystick, and a physical representation of the audio transformation scheme. Image from Hornof et al. (2008).

Experimental Results

I entered this project after data collection, and all subsequent work presented in this dissertation, including data analyses and modeling, is primarily my work.

Figures 7 and 8 show the average classification time and accuracy over three days. As can be seen, participants' performance improved substantially from day to day, but only reached 95% accuracy on Day Three[1]. Because novice behaviors are noisy and are typically hard to model, the following sections only present and model the empirical data from Day Three.

Top-Level Results

Figure 9 shows the average classification time of the correctly-classified blips on Day Three across blip color and the four sound and visibility conditions. The three

[1] Note that all error bars in the graphs shown in this chapter represent 95% confidence intervals (CI) of the mean across participants. In other words, these CIs were calculated by first averaging each participant's trial data into a single value, and then using the resulting ten data points, one per participant, to calculate the CI.

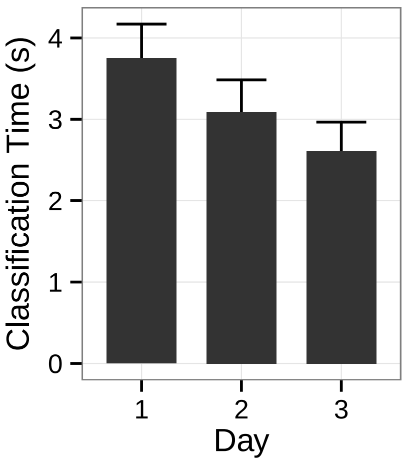


FIGURE 7. Mean classification time across three days. Error bars represent 95% confidence interval of the population mean.

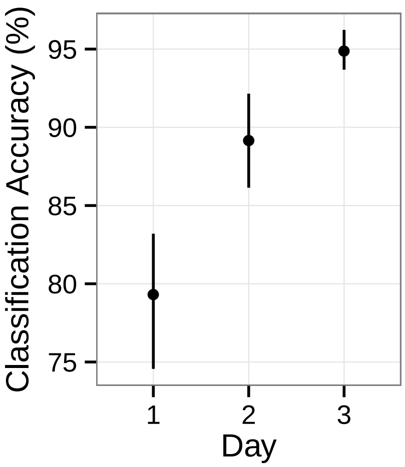


FIGURE 8. Mean classification accuracy across all three days. Note the non-zero *y* axis.

blip colors are grouped into two categories: (a) Red/Green blips, for which hostility is indicated by color and (b) Yellow blips, for which hostility needs to be determined by the participant based on shape, speed, and direction. There are roughly an equal numbers of blips in each category in each session (24 yellow and 23 red/green blips), so the results of these two categories are comparable. As can be seen, the yellow blips took participants longer to classify than the red/green blips, $F(1,9) = 31.44$, $p < .001$, $\eta^2 = .362$[2]. The peripheral-visible conditions are faster than the peripheral-not-visible conditions, $F(1,9) = 52.13$, $p < .001$, $\eta^2 = .455$. The sound factor, however, did not have a significant main effect on classification time, $F(1,9) = 2.36$, $p = .16$, $\eta^2 = .03$, nor a significant interaction with the peripheral-visibility factor, $F(1,9) = 1.56$, $p = .24$, $\eta^2 = .034$.

Figure 10 shows the root-mean-squared (RMS) tracking error (calculated over the entire tracking error data of each session) across the four sound and visibility conditions.

[2] Effect size η^2 measures the proportion of variance in the response variable explained by the factor. Repeated measures ANOVA was applied using R(R Core Team, 2015) to obtain the statistics.

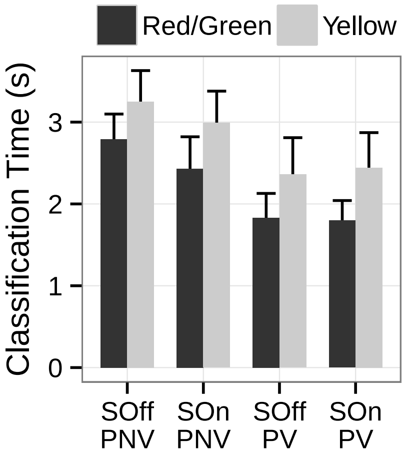


FIGURE 9. Classification time across blip color and the four sound and visibility conditions. SOff=Sound-Off; SOn=Sound-On; PNV=Peripheral-Not-Visible; and PV=Peripheral-Visible.

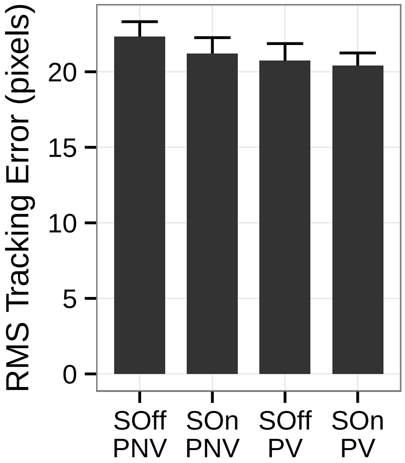


FIGURE 10. The root-mean-squared (RMS) tracking error per session across the four sound and visibility conditions.

Notable trends include: (a) Peripheral-visible conditions have smaller RMS tracking error than the peripheral-not-visible conditions, $F(1,9) = 22.03$, $p = .001$, $\eta^2 = .124$. (b) Sound-on conditions have smaller RMS tracking error than the sound-off condition, $F(1,9) = 27.78$, $p < .001$, $\eta^2 = .049$. The interaction of the two perceptual factors are nonsignificant, $F(1,9) = .61$, $p = .457$, suggesting that the effect of sound on tracking error stays the same across the peripheral-visibility conditions. Though both sound and peripheral-visibility had significant effects, the RMS tracking error only changes by less than 2 pixels across the conditions, which indicates that participants somewhat maintained their tracking performance in the more difficult conditions.

These results show that participants tended to perform better in both tasks when more perceptual information was available. Peripheral visibility assists with performance on both tasks, and sound assists with the tracking task. It seems that participants adapted to the different sound and visibility conditions and were able to find strategies that protected the classification performance without negatively affecting tracking by much.

This top-level analysis, however, does not clearly reveal any details of the strategies. The next section presents the eye movement data, which starts to reveal these details of strategies through careful analysis.

Eye Movement Data

Eye Movement Data Preprocessing

Eye movement data were preprocessed to remove bad eye tracking data and improve eye tracking accuracy. Blip waves that have more than 5% bad gaze frames, in which the eye tracker failed to find the participant's eyes, were removed from the eye movement analysis. Fixations were identified using the dispersion-based fixation detection algorithm (Karsh & Breitenbach, 1983). For the two parameters of the fixation detection algorithm (the minimum fixation duration and the minimum dispersion threshold), I tested a range of settings and found that a minimum fixation duration of 100 ms and a minimum dispersion threshold of 0.7° of visual angle appeared to correctly identify most of the fixations that I perceived based on studying visualizations of the moment-to-moment eye tracking data. After the fixations were identified, each fixation was then assigned to its nearest object that appeared on the screen at the same time as the fixation and that was within 2.5° from the fixation.

Eye tracking error, which are systematic deviations of the recorded gaze locations from their true locations, was found in the raw data, and was corrected using the post hoc eye movement data error correction technique developed as part of this dissertation work (Zhang & Hornof, 2013). The technique extends Hornof and Halverson's (2002) required fixation location (RFL) technique to (a) accommodate moving RFLs (the blips) and (b) incorporate multiple error signatures across locations and time. The error was reduced by more than 0.5° for half of the sessions, and by more than 1° for 30% of the sessions.

This section presents two sets of results pertaining to eye movements, one regarding when the eyes moved in response to the blips and one regarding where the eyes moved.

When to Look

Figure 11 shows the three eye movements needed to classify a blip and resume tracking: (a) move to the radar display, (b) move to the target blip, unless the first eye movement landed on it, (c) move back to the tracking display. These eye movements, plus the dwell time on the blip, divide the classification task into four stages—pre-radar, blip search, blip encoding, and post-radar—which will be described below in more detail. Participants classified sixty-four percent of the blips with this pattern of eye movements, while the remaining blips were classified in slightly different patterns (e.g., eyes move to the radar display again after moving back to the tracking display). Thus, these four stages, delimited by the eye movements, capture the participants' primary behavioral pattern, and the durations of these four classification stages potentially provide clues for inferring participants' strategies.

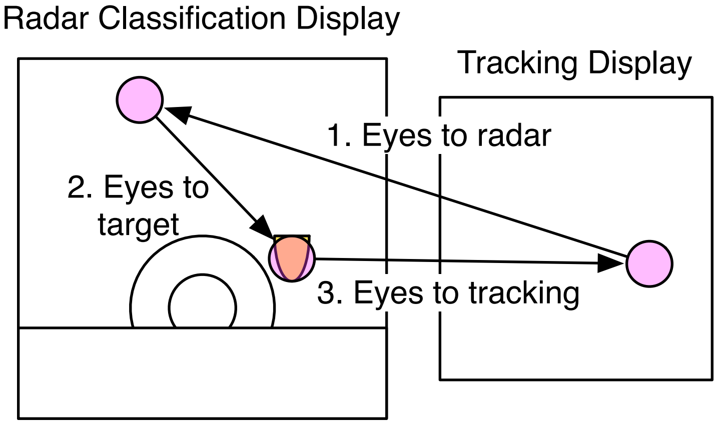


FIGURE 11. The three eye movements needed to classify a blip (purple circles represent fixations and the arrows represent saccades). The second eye movement is only needed if the first eye movement did not directly land on the blip. These eye movements, plus the dwell time on the blip, separate the classification task into four stages: pre-radar, blip search, blip encoding, and post-radar.

The pre-radar stage is the time interval between when a blip changes color and when the eyes move to the radar display to look for that blip. There are three primary events that motivate an eye movement to the radar: Participants (a) see the color-change in the periphery, (b) hear an auditory cue, or (c) just decide that it is time to move the eyes to the radar. Figure 12 shows the average duration of the pre-radar stage across the four sound and visibility conditions. As can be seen in Figure 12, participants responded much more quickly when color-change events were peripherally visible than when they were not, $F(1,9) = 37.8$, $p = .0002$. Sound enabled significantly faster responses ($F(1,9) = 5.06$, $p = .05$), particularly in the peripheral-not-visible condition. The visual task-switching in the sound-off peripheral-not-visible condition was self-paced, and thus produced the slowest pre-radar duration.

After the gaze arrived on the radar display, unless it landed directly on the target blip, the next step was to find the target. Figure 13 shows how long this search process took across the four conditions. As can be seen, the average search time is about 50 ms for peripheral-visible conditions. This time is so short presumably because the participants could see the active blip (the blip that changed color) before moving to the radar, and could often thus fixate it directly. Whereas for the peripheral-not-visible conditions, the search process took about 200 ms—just enough time to plan and execute a single eye movement after an initial fixation on the radar. The blip search time is longer for red and green blips than for yellow blips, $F(1,9) = 28.7$, $p = .0004$, suggesting that the participant might have prioritized the classification of yellow blips.

After locating the blip, the eyes stayed on the blip, for a period of time, to encode the visual information needed for classification. Figure 14 shows the average encoding time for red/green and yellow blips. As can be seen, the eyes stayed on the yellow blips significantly longer than on red or green blips ($F(1,9) = 24.8$, $p = .0007$), presumably

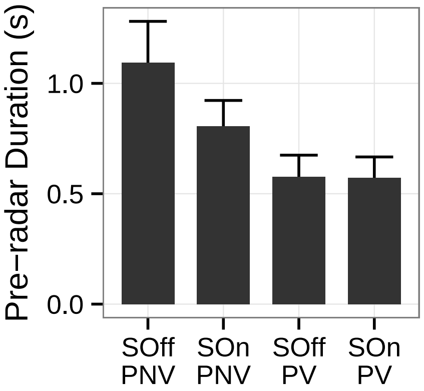


FIGURE 12. The pre-radar duration, the time interval between when a blip changes color and the eyes move to the rdar, across the four sound and visibility conditions.

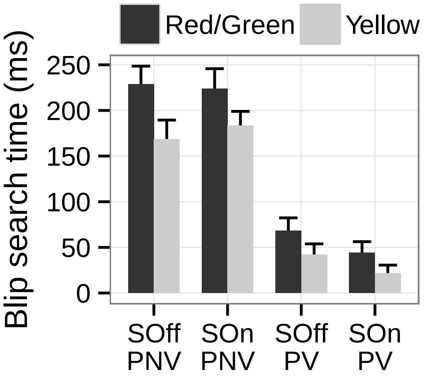


FIGURE 13. The blip search time, the time interval between eyes-to-radar and eyes-on-active-blip, across the four conditions and two blip-color classes.

because participants had to gather additional visual features such as shape, speed and direction in order to classify yellow blips.

The post-radar stage is the time interval between the eye movement that looks back to tracking and the first keystroke that enters the blip number. In 90% of the trials, the participants made the keystrokes after moving their eyes back to tracking, and quite often, based on the manual tracking data, they seemed to have made some tracking adjustments before keying-in the classification. These results suggest that the participants at least sometimes adopted an overlapping strategy which interleaved subtasks of tracking with classification. This overlapping is consistent with the multiple resource theory because the results suggest that visual processing is being done in parallel with both cognitive and motor processing. This overlapping also suggests that participants were attempting to optimize performance, and that good cognitive models of multitasking, at least for this task, need to be positioned to predict such overlapped and parallelized behaviors.

Figure 15 shows the effect of blip color and peripheral-visibility on the duration of the post-radar stage. The sound factor is not shown because it did not have a significant effect on post-radar duration. As can be seen, the keystrokes for yellow blips were more delayed than for red/green blips ($F(1,9) = 11$, $p = .009$), which again shows that yellow blips took longer to classify. The peripheral-visible conditions tend to have a shorter post-radar stage than the peripheral not visible conditions ($F(1,9) = 15$, $p = .0038$), suggesting that the task-switching process might be faster when the periphery display is visible.

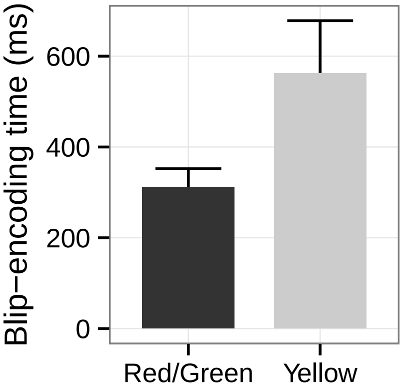


FIGURE 14. The blip encoding time, the average time spent fixating a blip, across blip color classes.

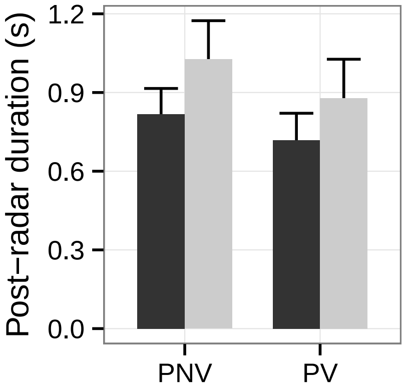


FIGURE 15. The post-radar duration, the time interval between eyes-to-tracking and key-in-classification, across the two peripheral-visibility conditions and two blip color classes.

The above results are for blips that were classified with the three prototypical eye movements, in which only one glance is needed to classify a blip; for some other blips, multiple glances are needed. Specifically, 6% of the red/green blips had repeated glances, and 29% of the yellow blips had repeated glances. This disparity suggests that the red/green blips could generally be classified with a single glance, but that it often took several glances to classify the yellow blips.

In summary, the above analysis shows how sound, peripheral-visibility, and blip color conditions affected performance. Peripheral visual information enabled faster responses to color changes (pre-radar duration), shorter blip search time, and shorter delay for keying-in the classification (post-radar duration). Sound seemed to only reduce the duration of the pre-radar stage when the peripheral display was not visible. These results provide some useful probes into task performance that, when explored with computational cognitive modeling, can provide strong insights into people's task strategies. However, to get a comprehensive view of the data, it is still necessary to examine where, in addition to when, the participants looked.

Where to Look

In this task, there was a clear performance benefit if, after a blip changed color, the eyes could move to that blip with a single movement. There are perhaps three ways for a participant to know the location of a blip that just changed color: peripheral vision, memory from earlier glances at the radar, and the acoustic location delivered by the spatialized auditory cues. This section presents evidence for the first two of these hypothesized substrategies.

To determine whether the participants used peripheral vision to know where the blips were, we can examine how often, when participants moved their eyes to the classification display, the eyes directly landed on a blip that just changed color. Figure 16 shows these results. Across all conditions, these direct fixation occurred more frequently for yellow blips than for red/green blips, again suggesting that the participants prioritized yellow blips. In the peripheral-visible conditions, the percentage is around 80%, suggesting that the participants could indeed see color changes in the periphery. Surprisingly, in the peripheral-not-visible conditions, the participants could also direct-fixate a quarter to a third of the blips, which suggests that participants, to locate the blips, either (a) recalled blip locations from early glances or (b) used acoustic location information. However, that the sound-on peripheral-not-visible condition has about the same performance as the sound-off peripheral-not-visible condition indicates participants were likely using memory rather than sound.

To determine whether the participants used memory from earlier glances to know the blip locations, it is beneficial to examine how often participants looked at blips while they were black, because these fixations could indicate that the participants were trying to memorize the blip locations. Figure 17 shows, for each condition, the proportion of black blips that participants looked at. As can be seen, a substantial percentage (more than 40%) of black blips were looked at across all conditions. These glances to black blips were likely to maintain some sort of "situational awareness" rather than to classify the blips because the average duration of these fixations was 244 ms, much shorter than the fixation duration of 562 ms on yellow blips. Thus, the objective of these glances was perhaps to see where the blips are so that later, when the blips changed color, they could be located more quickly. In summary, the above results suggest that participants used both peripheral vision (when available) and memory from earlier glances to know where to look at blips.

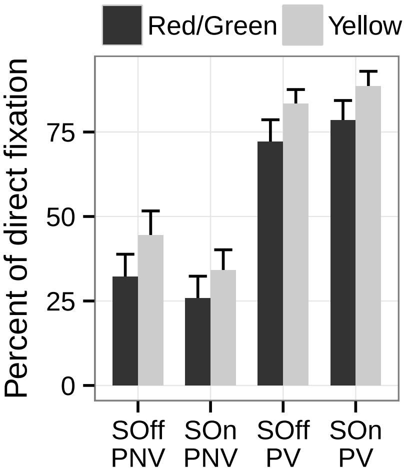


FIGURE 16. The percentage of tracking-to-radar eye movements that landed directly on blips that just changed color.

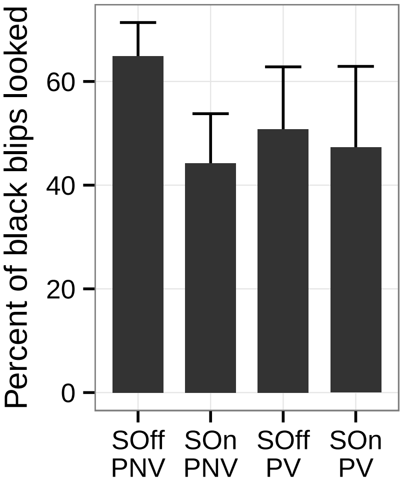


FIGURE 17. The percentage of black blips that were looked at across conditions.

Together, the analyses about when and where the participants looked show the effects of sound, peripheral visibility, and blip color on different stages of the classification process. These effects combine to cause the effects observed in the top-level measures of classification time and RMS tracking error, but are needed, on their own to fully understand the complexity of the multitasking behavior. These eye movement data suggest that the awareness of blip status brought by peripheral visibility is particularly important for multitasking, and that sound, even when spatialized, is not a sufficient substitute for peripheral visual information.

Summary

Our time-critical multimodal dual task experiment pushes the research on human multitasking to a more complex, real-world scenario than typically studied by psychological researchers. The experiment is in some ways similar to driving a vehicle while simultaneously completing other tasks such as making a phone call or selecting a song from a touchscreen entertainment system. Theory that is built based on the results from this experiment has implications for real-world multitasking problems.

By means of rigorous eye tracking data collection and analysis, we acquired a rich set of data that provide insight into people's multitasking strategies. We found that (a) making important task information available in peripheral vision can improve multitasking performance, (b) auditory cues can somewhat complement visual information, but not completely substitute it, and (c) people seemed to be able to sometimes simultaneously interleave multiple tasks. However, the results so far are inconclusive and incomplete, because the data only suggest some possible task strategies. To truly test a set of assumptions about how participants multitasked in this experiment, computational cognitive models that incorporate these assumptions need to be constructed in order to see whether such models can produce the observed behaviors. The next section discusses how I built models to explore various plausible multitasking strategies.

CHAPTER IV

INTRODUCTION TO THE MODELS

This chapter includes materials from my two previous publications Zhang and Hornof (2012, 2014). I was the primary author of those two papers, my coauthor provided editorial assistance.

Introduction to EPIC

The models presented in this thesis were built using the EPIC (Executive Process Interactive Control) cognitive architecture (Kieras & Meyer, 1997). EPIC provides a software framework (written in C++) for simulating humans interacting with a task environment. Figure 18 shows the components of the architecture as well as a simulated task environment. A human is modeled as an information processing system consisting of a cognitive processor and various perceptual and motor processors. The simulated task environment, typically needs to be programmed by the analyst for every new task, reproduces the task design using EPIC's device framework and provides symbolic information input to the model's perceptual processors, and receives and responds to the model's motor output. This section describes in detail what happens inside the model's cognitive, perceptual, and perceptual processors.

The Cognitive Processor, Memory, and Production System

The cognitive processor, so to speak, is the "central command" of the simulated human information processing system. It has access to (a) the long-term memory, (b) the working memory, and (c) preprogrammed task strategies stored in the production memory. EPIC's current implementation of long-term memory is somewhat minimal

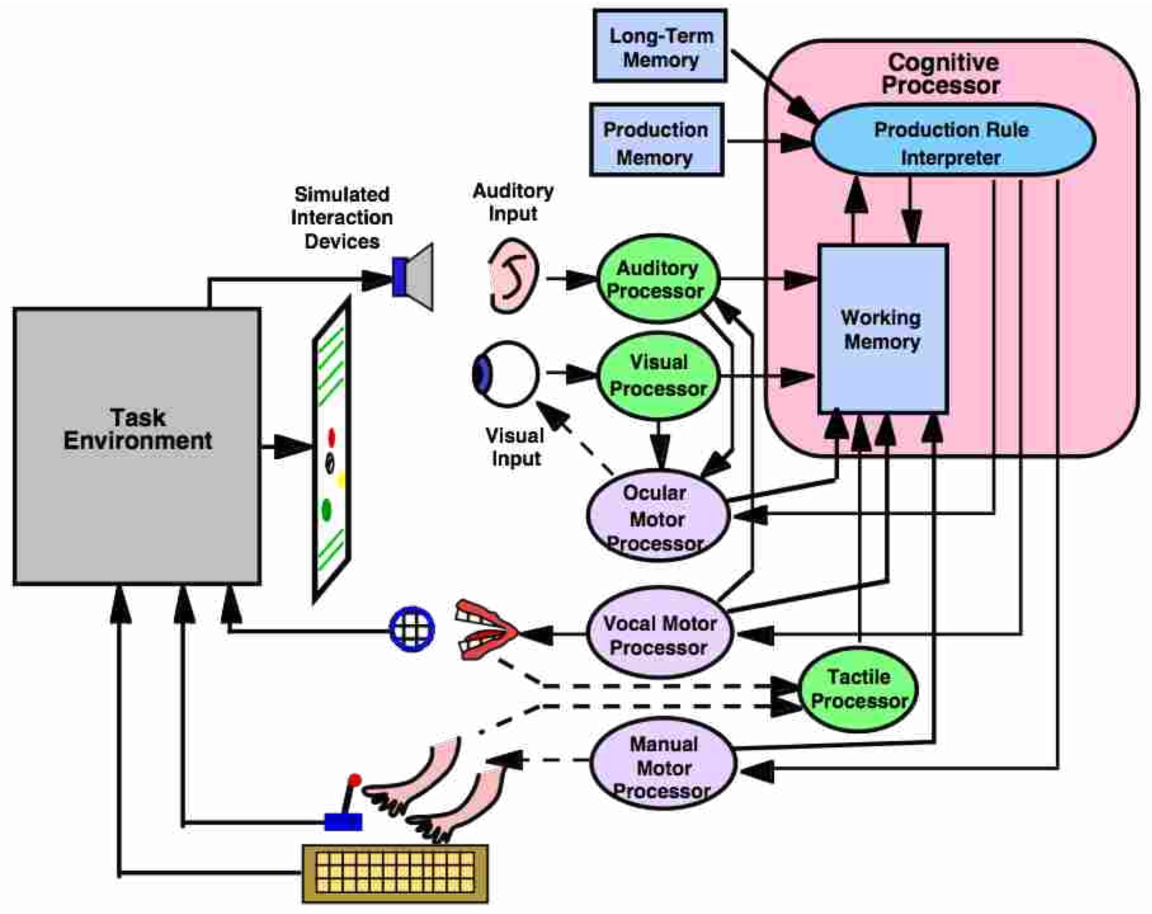


FIGURE 18. The simulated task environment and EPIC's various components. The perceptual and motor processors actual contain a series of processors and stores.

compared to ACT-R's implementation: In EPIC, information that needs to be stored in long-term memory is deposited to the long-term memory at the beginning of a simulation and, unlike in ACT-R, information in long-term memory does not decay. This is, however, not a serious problem for the types of tasks that EPIC is typically applied to, which generally emphasize perceptual and motor activities, and which rely more on working memory rather than long-term memory.

EPIC's working memory contains a variety of information such as perceptual information, motor processor status, task-control information, and tags that assign labels to other perceptual memory items. The perceptual information in the working memory decays, and the decay time varies depending on the perceptual modality and the perceptual properties that are encoded (e.g., color, shape, sound frequency, and sound timbre). The default decay time for visual and auditory properties is 500 ms, but this can be changed based on empirical evidence. The status of a motor processor signals whether a motor is busy processing some movement commands, and is updated at every cognitive cycle (50 ms). More details about the status memory will be discussed later in the motor-processor section. Memory related to task-control includes information such as goals and steps, which are only accessed and modified by the production rules to keep track of the task processes. Tags can be thought of as notes put in the memory by some task strategies in order to keep track of the relevant stimulus information. Task-control memory and tags do not decay.

The production memory stores task strategies in the form of IF-THEN statements called production rules. These are typically written by the analyst for a specific task. Figure 19 shows an example of a production rule that specifies the conditions and action needed to fixate the tracking target. The symbols with question marks in front of them are variables that get matched to objects in the working memory. The conditions of this production rule includes: (a) The current goal is to do the dual task, (b) an object with a cross-hair shape is visible, and (c) the ocular motor (for making eye movements) is free. The action statement commands the ocular motor to saccade to the tracking target. The ?tracking_target gets bound to the object that satisfies the specified shape and visibility status and used in the action as the target of the eye movement.

```
(Look_at_tracking_target

If(
    (Goal Do Dual_task)

    (Visual ?tracking_target Shape Cross_Hairs)
    (Visual ?tracking_target Status Visible)

    (Motor Ocular Modality Free)
)
Then
(
    (Send_to_motor Ocular Perform Move ?tracking_target)
))
```

FIGURE 19. An example of a production rule. The rule checks goals, the visibility of the tracking target, and the availability of the ocular motor. If all conditions are satisfied, the rule sends an eye movement to the tracking target.

The cognitive processor runs on a 50 ms cycle. On each cycle, it executes all the production rules whose conditions are satisfied by the contents in the working memory and long term memory. As discussed in Chapter V, this parallel execution of multiple production rules makes EPIC extremely suitable for exploring multitasking strategies, because the executive processes that manage task switching can be coded as an independent set of production rules that run alongside the task processes, and that can be easily modified to test different kinds of executive processes.

Perceptual Processors

EPIC's perceptual processors have a pipeline structure that goes through two stages: sensory processing and perceptual processing. Figure 20 shows this pipeline structure for visual processes. Information generated by the simulated task environment begins as symbols in the physical store, goes through the eye processor to the sensory store, and through the perceptual processor to the perceptual store, which then becomes a part of working memory accessible to the cognitive processor. The sensory store is similar to the concept of iconic storage in that the store can hold information briefly for deeper perceptual processing. For each visual property, there is a transduction time that determines the duration of sensory processing and an encoding time that determines the duration of perceptual processing. For example, for text, the default transduction time is 50 ms and the encoding time is 100 ms. The auditory processor has a similar pipeline structure.

EPIC's model of the eye includes a retina that determines what kind of sensory information is available based on the distance in visual angle between the object and the center of the gaze. A basic implementation of the retina specifies fixed availability zones for different visual properties. For example, text is available in the foveal zone (within 1°of the point of gaze), color in the parafoveal zone (within 7.5°), and position and luminance changes in the peripheral zone (within 60°). A more complex implementation of the retina uses psychometric functions to determine the probability of property availability. Developing this more complex implementation is part of our ongoing research (see Kieras & Hornof, 2014; Kieras, Hornof, & Zhang, 2015; Zhang & Hornof, 2013), and the models described here use the zone availability mechanism.

The implementation of the retinal availability functions and an ocular motor processor obviates a covert visual attention mechanism. In EPIC, if an object falls

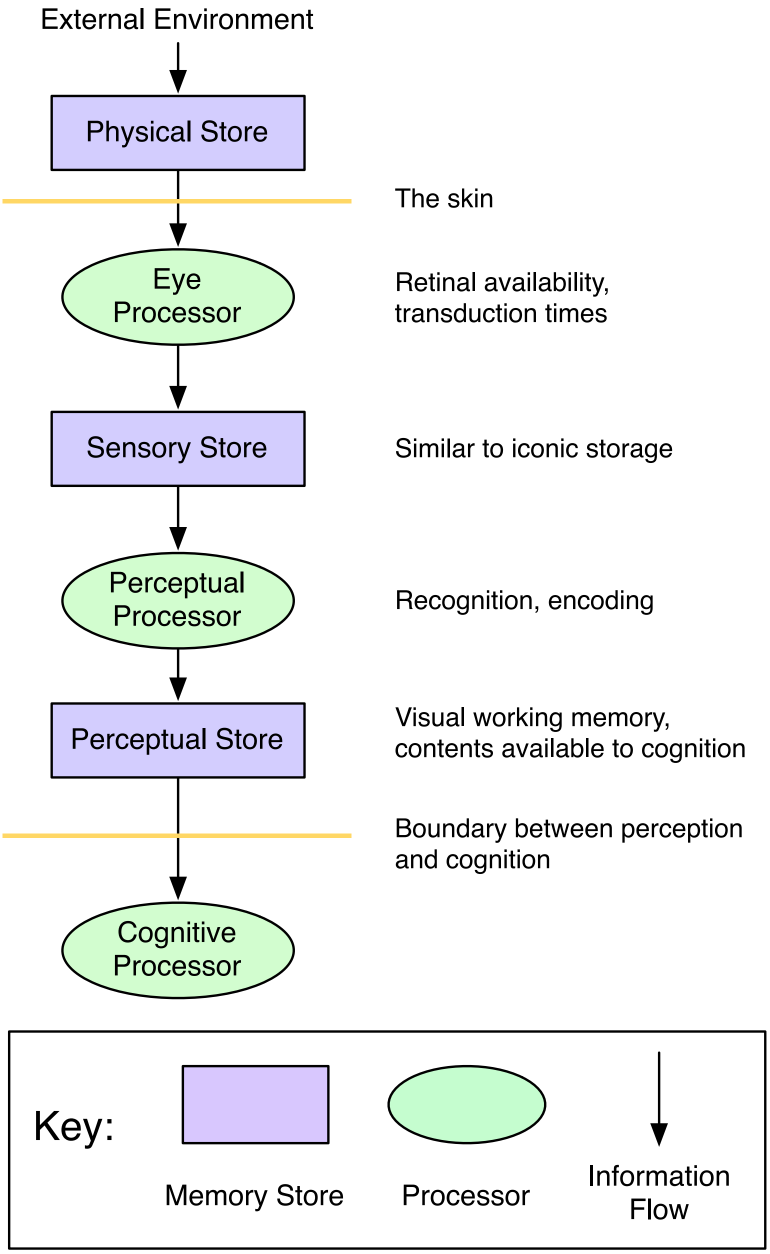


FIGURE 20. EPIC's visual perceptual processing pipeline. Information stems from the external environment and passes the eye and perceptual processors, eventually become available to the cognitive processor.

in the visual periphery and cannot be perceived, the model does not shift its covert attention to the object, but rather it orients the gaze point to the object to fixate it with foveal vision. This emphasis on the role of eye movements is consistent with the "active vision" hypothesis (Findlay & Gilchrist, 2003) which asserts that in real-world tasks, eye movements tend to occur much more frequently than shifts of covert visual attention. Thus, in EPIC, within-modality interference in the visual perception stage is a result of tasks competing for the foveal vision rather than competing for visual attention. This makes the theory, or the models constructed based on the theory, easier to validate because eye movements are observable, while shifts of attention are not. The implementation of a simulated retina and an ocular motor processor makes EPIC an excellent choice for modeling tasks that involve eye movements.

Motor Processors

EPIC's motor processing also goes through two stages: preparation and execution. Following Rosenbaum's (2009) research on motor control, movements are specified in terms of movement "features". For example, punching a key is specified with two features, one to specify the hand (left or right) that is used and the other to specify the finger. Each features takes 50 ms to prepare. However, no preparation is needed for aimed movements such as eye movements and tracking movements, as discussed in Kieras (2009). After preparation is completed, movements are executed with a standard delay of 50 ms. The execution time varies depending on the movements executed. For example, for pointing movements, the execution time is determined by Fitts' law (Fitts, 1954), which is a function of target width and movement distance.

Consistent with the multiple resource theory, a motor processor in EPIC can only process one movement at a time; however, EPIC's implementation of the motor processing bottleneck is more nuanced than that specified by the multiple resource theory. Specifically, the preparation stage and the execution stage can each process a movement at the same time. For example, if the manual motor processor is commanded to execute two keystrokes successively, such as in the classification task, the manual motor processor can execute one movement while preparing for the second. This streamlined processing can help explain a range of phenomena, such as how typing words is usually more efficient than typing letters individually (Kieras et al., 1997).

Our Extensions to the Architecture

We made two extensions to the EPIC architecture: (a) a new implementation for simulating tracking manual movements, and (b) a temporal processor, implemented by Tim Halverson, to determine from within the simulated human when a certain amount of time has elapsed. The implementation of the tracking movement will be discussed in the next section. The implementation of the temporal processor replicates that in the ACT-R cognitive architecture (Taatgen et al., 2007), which seems to accurately model people's performance in temporal estimation. The addition of this processor provides a mechanism that assists the dual task models in making self-paced periodic checks of the radar display in conditions with no peripheral visibility or auditory cuing.

Summary

In summary, the EPIC architecture embodies many aspects of the multiple resource theory, and is particularly well-suited for modeling multitasking. EPIC's implementation of the perceptual and motor processors offer concrete, computational accounts of how within-modality interference occurs within these stages of information processing. EPIC's cognitive processor adheres to cognitive parallelism, which facilitates exploration of different multitasking strategies. All these features make EPIC an excellent architecture for modeling multitasking.

The Dual Task Model

The model for our multimodal dual task has three components: the tracking processes, the classification processes, and the executive processes. This section discusses how each component is implemented and how the model parameters as well as task strategies were set and explored. Note that the model discussed here is developed for each individual participant, and thus the parameter calibration and strategy exploration were conducted for each participant separately.

Modeling the Tracking Task

The tracking task is implemented primarily with two production rules, one to keep the eyes on the tracking target, and the other to issue the manual tracking command[1]. The production rule for manual tracking, as outlined in Figure 19, fires when the model's gaze is on the tracking screen and the manual motor is not in use. Each tracking movement takes some time to complete and when a movement finishes, the manual tracking production rule could fire again to immediately initiate another tracking movement. This way, manual tracking is executed continuously to keep the tracking error as small as possible.

The production rule for manual tracking only needs to specify the tracking cursor and target; the trajectory and duration of the cursor movement are handled by the simulated device and the EPIC architecture. In a previous study that also used EPIC to model tracking (Kieras et al., 2001), a manual tracking movement was implemented as

[1] This section is developed based on Zhang and Hornof (2012)

a ballistic Fitts' law movement: Its direction is fixed when in action, and its duration is a logrithmic function of the moving distance (tracking error). This dissertation builds upon this previous tracking model, but with the assumption that tracking movements are non-ballastic rather than ballistic. Figure 21 illustrates how the non-ballistic tracking movement works (solid arrows and circles), as well as how the original ballistic movement works (dashed arrows and circles). As illustrated, because of the shifting of the tracking target, the ballistic movement ended up with a large tracking error, whereas the non-ballistic movement reduced the error. For the tracking task presented here, it was found that this assumption of non-ballistic movements predicts the root-mean-squared tracking error (RMS TE) better than the assumption of ballistic movements (Zhang & Hornof, 2012).

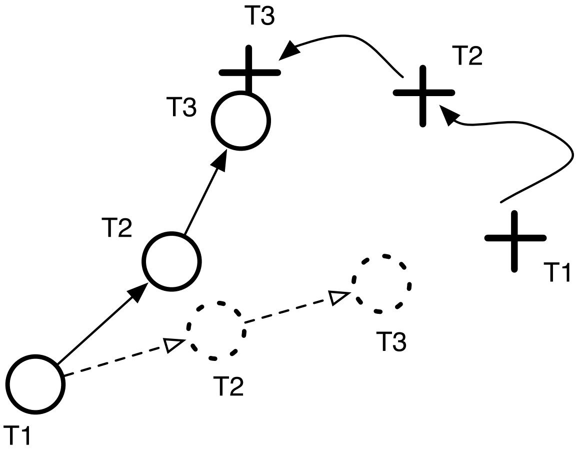


FIGURE 21. An illustration of how a ballistic tracking movement and a pursue tracking movement proceed. The cross represents the target, and the circle represents the cursor. The arrows mark the paths of the target and cursor, from time T1 to time T3. Dashed lines represent the ballistic movement, and solid lines represent the pursue movement.

As discussed above, the duration of a tracking movement is assumed to follow Fitts' law. To accurately model each participant's tracking task, the parameters were calibrated for each participant. The Fitts' parameters are the intercept *a* and slope *b* of the following equation:

$$MT = a + b\ log_2(A/W + 0.5) \tag{4.1}$$

They were estimated by running linear regression across movement data that were extracted from the tracking error data. The movements were extracted by isolating the periods in which (a) tracking started with an error that was greater than 20 pixels (which was always the case when the tracking cursor was not green) and (b) the tracking error dropped continuously by more than 5 pixels. The 5-pixel threshold made it less likely that a drop in tracking error was due to the random shifts of the tracking target. To ensure that the movement data used for calibration was not affected by the classification task (such as a tracking movement was interrupted by keying-in classification), only the movements in the between-wave periods in which there were no blips on the radar screen (each session had two between-wave periods, and each period lasted 10 seconds) were used to estimate the Fitts' parameters.

Using the non-ballistic tracking movement implementation and the calibrated Fitts' parameters, the model accurately predicted almost every participant's observed RMS TE of the between-wave, single-task periods. Figure 22 shows the root-mean-square deviation (RMSD) between each participant's observed RMS TE and the model's RMS TE after calibrating the tracking movements to that participant. As can be seen, except for P04 and P16, the predictions for the participants were all within 1 to 2 pixels. Given that the RMS TE is around 20 pixels for these between-wave tracking periods, the model seems to predict participants' tracking performance well, with less than 10% error. For P04 and P16, however, there were not enough movement data to accurately estimate the parameters. Therefore, for these two participants' tracking models, the slope parameter was adjusted until the models' predictions matched the RMS TE of the between-wave periods. After the adjustment, the RMSD for P04 was 1.55 pixels and for P16 was 1.28 pixels.

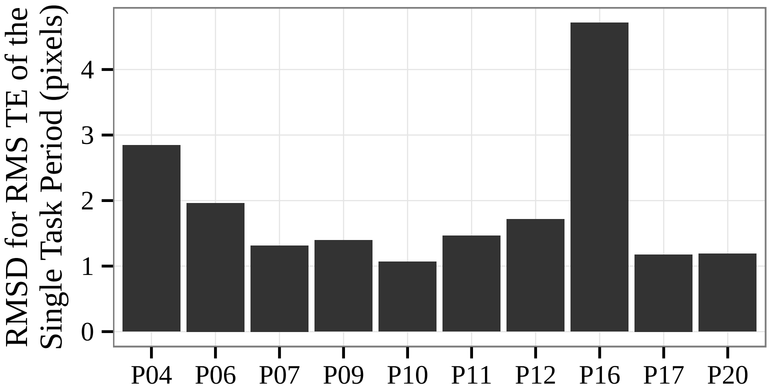


FIGURE 22. The root-mean-square deviation (RMSD) between each participant's observed RMS TE and the model's RMS TE.

Note that the tracking model presented here serves as a straightforward approximation of the tracking behavior observed in this task rather than a millisecond-level veridical representation of what truly happens in tracking. The two underlying assumptions—that tracking movements are non-ballistic and follow Fitts' law—still require further validation. For the purpose of modeling this dual task, however, the tracking model simulates enough details of tracking such that the interaction between tracking and classification can be predicted.

Modeling the Classification Task

The classification task is in essence a choice reaction task, but two aspects of the task make it more complex. First, in the sound-off peripheral-not-visible condition, there is no cue to signal the blip appearance or color change. This means that the model has to actively, periodically check the radar display for blips that changed color. Second, across all conditions, multiple blips can become active in close succession, which means that the model has to decide which blip to classify. This in itself creates a multitasking situation. To handle these two aspects of the task, the model for the classification task not only includes the typical perception, response selection, and movement production stages related to choice reaction tasks, but also a monitoring stage and a stimulus selection stage.

Monitoring and Selection

The monitoring stage detects active blips and initiates task switching from tracking to classification. It is the only component in the model that has different implementations for different experimental conditions. In the peripheral-visible conditions, the model monitors color-change events via a blip's luminance change because luminance change is accessible in the periphery (60°) whereas color change is only available in the parafovea (7.5°). In the sound-on conditions, the color change events were sonified and so the monitoring processes use the auditory alerts as the cues for the classification task.

For the sound-off peripheral-not-visible condition, the process for detecting active blips is more complicated. For this condition, there is no visual cue or sound cue to indicate color changes. Visualizations of the eye movement data show that in this condition, participants periodically stopped tracking and checked the radar display, and that they did this roughly every 1.5 seconds. Thus, the model implements a similar process: It uses the temporal processor to estimate how long the model has not looked at the radar display and, if the estimated time reaches 1.5 seconds, the executive processes would interrupt tracking and move the eyes to the radar display to look for active blips.

The time required to detect that a blip changed color depends on the experimental condition and the cognitive strategy being used for that condition. The following parameters come into play for different strategies: (a) A change in luminance is available to the cognitive processor 100 ms after a physical color-change event. (b) Auditory information is available 500 ms after a color-change event. (Sound is available after 100 ms, and the encoding time for frequency and timbre is set to 400 ms, the approximate duration of the first pulse of each color-change alert.) (c) The self-interruption interval (how often to check the radar display in the sound-off peripheral-not-visible condition) is set to 1.5 seconds, and so the average time to detect a color-change event is roughly 750 ms.

As discussed earlier, a blip-selection process is needed to deal with situations that have multiple active blips available at the same time. This selection process keeps track of all the blips that are active, and maintains a single target blip such that the subsequent perception, response selection and movement production stages are all directed to this target blip. After the target blip is classified, the stimulus selection process will randomly choose one of the remaining active blips (if there are any), and designate it as the new target blip. Maintaining a single target blip throughout the whole classification process precludes any possibility of overlapped blip classification. That is, the model cannot look at one blip while manually classifying another. The same behavior is observed in the human data, possibly because classifying many blips in a row would substantially impair the tracking performance.

Perception

The perception stage of the classification task involves the encoding of a blip's number and physical characteristics that indicate the blip's hostility. To recognize the blip number, because the text property is only available in the foveal vision, the model has to position its gaze directly on the blip. To acquire the hostility information, for red or green blips, the model only needs to encode the blip color; for yellow blips, however, the model also needs to encode the blip's shape, speed, and direction, which are only available in the foveal vision. The duration of these perceptual processes are controlled by parameter which are set either to the default values or calibrated from the eye movement data, as follows:

- *Text encoding time*. Calibrated to each participant's average fixation time on red and green blips. The resulting parameter settings, along with the settings for all other parameters introduced in this section, is summarized in Table 1.
- *Color encoding time*. Set to EPIC's default value, 50 ms.
- *Speed and direction encoding time*. Calibrated to each participant's average fixation time on yellow blips.

Response Selection

In the classification task, the response selection stage determines whether a blip is hostile or neutral. Because the participants almost always (in 90% of the trials) keyed in the classification some time after their eyes moved away from the target blip, this response selection stage, which should occur right before keying-in, is assumed to occur after the eyes left a target blip, i.e., in the post-radar stage. For red and green blips, response selection is assumed to only take one cognitive cycle (50 ms) because the blip hostility can be directly inferred from the blip color. For yellow blips, however, the classification rules are more complicated and, as shown in Figure 15, the duration of the post-radar stage for yellow blips is much longer than that for red or green blips. Also, more than 30% of yellow blips required multiple glances to be classified (the gaze moving back and forth between tracking and the target blips multiple times). To model these characteristics, I used EPIC's visual-perceptual encoding mechanism and a custom hostility-encoding function to simulate response selection for yellow blips. More specifically, response selection for yellow blips is modeled as a perceptual encoding process that happens after the encoding of physical features and before movement production.

To determine the hostility encoding time for yellow blips, however, a systematic search of the parameter value is needed. The hostility encoding time for yellow blips is a free parameter and cannot be calibrated from the data because there are no directly observable events that mark the beginning or the end of response selection. (The post-radar duration cannot be used as the encoding time because the participants sometimes performed tracking during this period.) To find the best-fitting setting in a rigorous manner, a grid sampling method is used in which the parameter space is sampled at equal intervals. Specifically, the parameter is systematically varied from 100 ms to 1600 ms at 100 ms intervals.

A custom hostility encoding function is developed to simulate how sometimes more than one glance is needed to classify a yellow blip. Specifically, hostility encoding is modeled as a Poisson process, in which each visit to the yellow blip has a certain probability of failing to encode the blip hostility. The Poisson process approximates the distribution of the observed number of glances on the yellow blips very well. For example, Figure 23 shows the distribution of number of glances (circles) for P06, P16, and P20, each fitted with a Poisson regression line (dashed lines). Most of the data points fall closely along the regression lines,

Movement Production

In the classification task, the movement production stage executes two keystrokes for each blip: one for entering the blip number and one for the hostility. Because these two keystrokes had to be entered within 750 ms (otherwise a time-out error and penalty occurred), participants typically made the two keystrokes in quick succession. The model

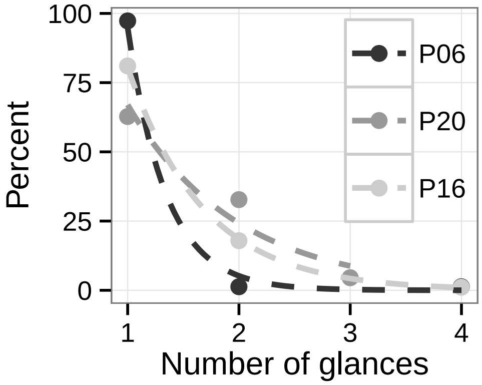


FIGURE 23. Distribution of number of glances across three participants. For these participants, at most 4 glances were used to classify a yellow blip. The *y* axis shows the percentage of yellow blips that was classified with 1, 2, 3, or 4 glances. The dashed lines are best-fitting Poisson regression lines.

issues the two commands similarly. Recall that EPIC's movement production consists of two stages, preparation and execution. Due to this streamlined motor processing mechanism, the preparation stage of the hostility keystroke can be done in parallel with the execution of the number keystroke. Figure 24 illustrates how the two keystrokes are thus processed by the manual motor processor.

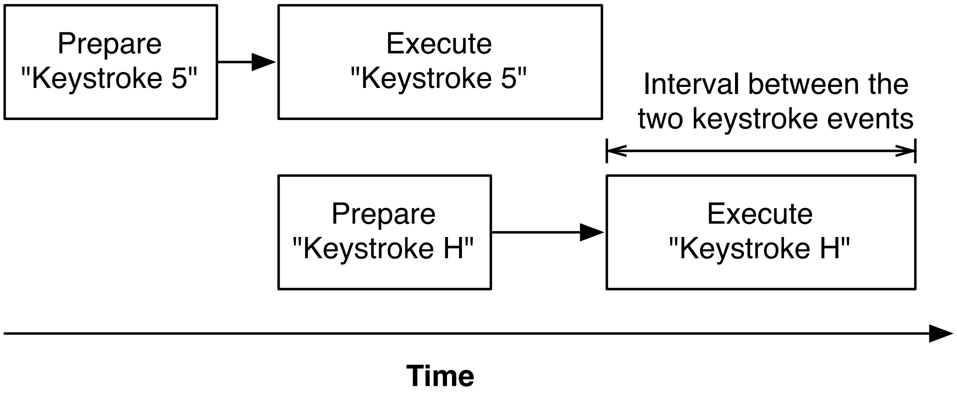


FIGURE 24. An illustration of how EPIC streamlines two consecutive keystrokes. The preparation of the second keystroke "H" is done in parallel with the execution of the first keystroke "5", and though the preparation for "H" is done before the execution of "5", the execution of "H" has to wait until "5" finishes.

The preparation time of a keystroke is set to the default 150 ms (3 features—hand, finger, and movement direction—each takes 50 ms to prepare), whereas the execution time is calibrated to the data. EPIC's default keystroke execution time is 280 ms, but this default execution time is estimated for typing on a regular keyboard. This dual task uses a numeric keypad and thus the parameter needs to be re-estimated. The parameter can be estimated using the time interval between the two keystroke events. This is because, as Figure 24 shows, if the two keystrokes are executed in succession, then the interval between the two keystrokes should be the same as the keystroke execution time. Thus, this interval is measured for every participant and is set as the execution time for both keystrokes.

Summary

The four stages—stimulus monitoring and selection, perception, response selection, and movement production—constitute the model for classification. Table 1 summarizes the parameters used in each of the four stages. The classification processes are interleaved with the tracking processes by a set of executive processes, discussed next.

The Executive Processes

The executive handles conflicts and manages transitions between tracking and classification. In EPIC, the only processors that cannot be shared among multiple tasks are the motor processors due to bottlenecks in motor processing. The multimodal dual task uses the ocular motor and the manual motor processors. The tracking task needs the ocular motor to fixate the tracking target, and the manual motor to make tracking movements. The classification task needs the ocular motor to fixate active blips, and the

TABLE 1. Parameters used in modeling the classification task. For parameters that were calibrated to individual participants, the mean and *SD* of the ten participants' calibrated parameters are shown. Other parameters were set to the same value across all participants.

Classification Stage	Parameter	Source	Setting
Monitoring and Selection	Luminance change detection time	EPIC default	100 ms
	Sound onset detection time	EPIC default	100 ms
	Sound timbre and frequency encoding time	Estimated based on sound duration	400 ms
	Self-paced glance interval	Estimated from human data	1500 ms
Perception	Text encoding time	Calibrated to each participant	*Mean*: 92 ms *SD*: 111 ms
	Color encoding time	EPIC default	50 ms
	Speed and direction encoding time	Calibrated to each participant	*Mean*: 240 ms *SD*: 217 ms
Response Selection	Hostility encoding time for red/green blips	EPIC default	50 ms
	Hostility encoding time for yellow blips	Free parameter	Sampled from 100 to 1600 ms
Movement Production	Keystroke preparation time	EPIC default	150 ms
	Keystroke execution time	Calibrated to each participant	*Mean*: 204 ms *SD*: 86 ms

manual motor to make the classification responses. Conflicts arise when both tasks want to use the ocular motor at the same time or the manual motor at the same time.

Figure 25 illustrates how the executive moves the eyes and shifts the manual motor between between the tracking and classification tasks. The executive consists of two independent sets of production rules, represented abstractly by the top and the bottom state transition diagrams in the figure. One set of production rules summarized in the figure passes control of the ocular motor processor back and forth between the two tasks, while another set of rules passes control of the manual motor processor back and forth. This independent interleaving of ocular and manual motor processing was found in Hornof and Zhang (2010) to explain performance better than a model that imposes strict serial ordering between the two tasks.

Unlike the tracking or classification task processes, which do not seem to lend themselves to many variations of task strategies, the transitions between the two tasks can be handled in many different ways. Figure 25 only shows the minimum requirements for the transitions to occur. For example, the transition of the manual motor from the tracking task to the classification task happens when a blip is ready to be keyed in. There are, however, several other conceivable ways in which this transition can unfold. For example, the transition does not have to happen immediately after a blip's hostility is identified. A participant could continue tracking until the tracking error is reduced to a money-making state (when the tracking cursor is green) before switching to the classification task. The task-switching strategies that our participants used could contribute to an understanding of how people manage multiple tasks. This aspect of the model is thus a central topic of this dissertation and will be discussed in the next section.

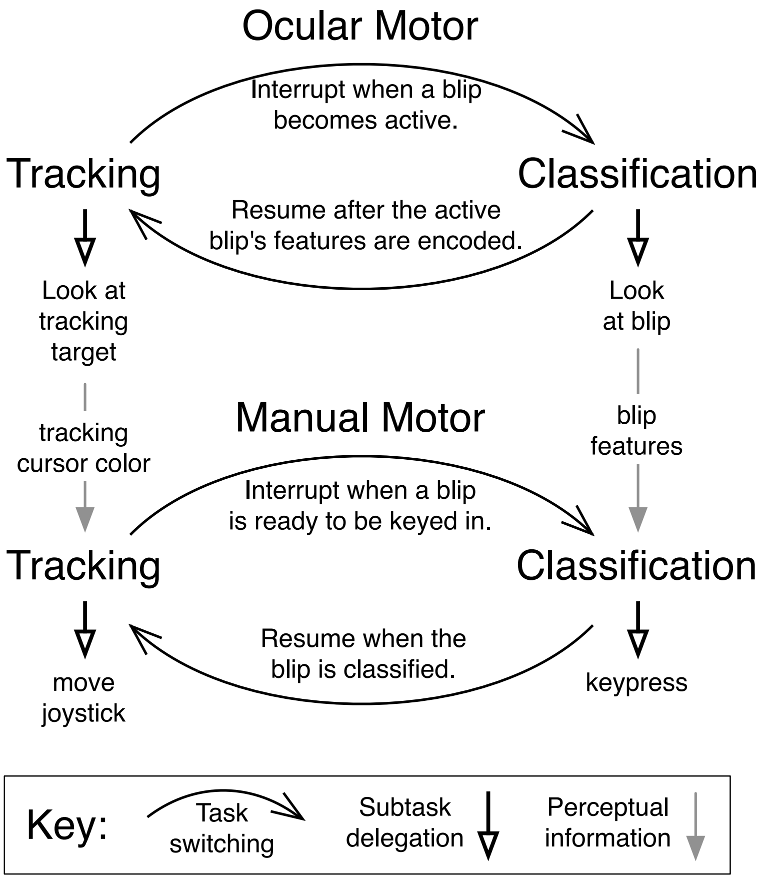


FIGURE 25. State transition diagrams showing the independent interleaving of the ocular motor and manual motor processing in the dual task model. Image adapted from Hornof and Zhang (2010) and appeared in Zhang and Hornof (2014).

Summary of the Dual Task Model

This section discussed in detail the many components of the dual task model. Like many other multitasking models, I started by modeling the single tasks. Accurate single-task models are the foundations for fitting a correct multitasking strategy and an accurate dual task model. Thus, great care was taken to ensure that the tracking model is correct, that the parameters needed for both tasks are accurate, and that the general approach for integrating the two tasks is plausible.

Many of the model parameters were calibrated based on careful analysis of the empirical data. This calibration reduced the model's degrees of freedom as later when fitting the model to other results, they are no longer varied. The single-task models are relatively straightforward, but the executive processes can have many variations. The next section discusses the variations of executive processes.

The Strategy Settings

Strategy Dimensions

The new and unique approach to exploring the broad range of integrated strategies that can be used in a multitasking scenario is to divide the executive processes into several decision points or *dimensions*. Each dimension represents one aspect of the task that can be completed using several alternative strategies. These dimensions then form a strategy space, which can be thought of as a multidimensional grid, with each point in the grid representing a specific combination of strategies from the different dimensions.

Table 2 shows the four strategy dimensions defined for the dual task executive processes. Each dimension affects a unique aspect of the executive processes. The letters T and R in the dimension names refer to Tracking and Radar. Dimensions whose name starts with T-R relate to transitions from tracking to radar, whereas dimensions whose name starts with R-T relate to transitions from radar to tracking.

Figure 26 further illustrates how each strategic dimension affects the executive processes. The T-R-Priority dimension controls whether to continue tracking before switching the ocular motor processing to classification. The T-R-Sound dimension controls, in the sound-on conditions, whether to switch the ocular motor processing to classification at every auditory alert, or only at the color-change alerts. The T-R-Location dimension controls whether to use memory to infer the active blip location. The R-T-Priority dimension controls whether to prioritize the manual processing of tracking or classification.

Zhang and Hornof (2014) had one more strategy dimension that explored how judiciously the participants tracked the target. One strategy in this dimension is to perform tracking whenever possible, and the other strategy is to only perform tracking when the cursor is not green. The results of that study suggested that nearly all participant did tracking whenever possible, and only two participants, P04 and P16, seemed to be only doing tracking when the cursor was not green. However, a closer examination upon the visualization of the tracking data showed that these two participants still tracked continuously, but as discussed previously, there were not enough tracking data for accurately estimating their tracking parameters. Thus in this dissertation research, I adjusted tracking parameters for these two participants to fit the tracking data instead of exploring the additional tracking strategy.

A Qualitative Analysis of the Effects of Various Strategies

The effects of the different alternative strategies on task performance can be analyzed qualitatively before running the model. Specifically, this section examines the

TABLE 2. The four strategic dimensions explored in the dual task model. Each dimension has two or three alternative strategies.

T-R-Priority When to move the eyes to the radar display after knowing a blip change color.

Immediate-Eyes-to-Blip (IEB) Move immediately.

Track-then-Eyes-to-Blip (TEB) Continue tracking until the tracking cursor is green, and then move.

T-R-Sound When to move the eyes to the radar display after hearing an auditory cue.

Eyes-to-All-Sounds (EAS) Move immediately for all sounds.

Eyes-to-Color-change-Sounds (ECS) Move only for color-change sounds.

T-R-Location In the peripheral-not-visible conditions, where to put the eyes in the radar display when switching to the classification task.

Look-Window-Center (LWC) Go to the center.

Look-prior-Blip-Location (LBL) Go to a black blip recalled from a previous visit.

R-T-Priority What to do with the hands after the model acquired the visual features of a yellow blip and moved the eyes back to tracking, but while waiting for the hostility classification to be encoded.

Keypad-Then-Joystick (KTJ) Wait for the encoding, key in the response, and then resume tracking.

Joystick-Then-Keypad (JTK) keep the tracking cursor in the green until a blip is ready to be keyed in.

Keypad-If-Green (KIG) Wait until the tracking cursor color is seen. If the tracking cursor is green, then do classification; otherwise do tracking.

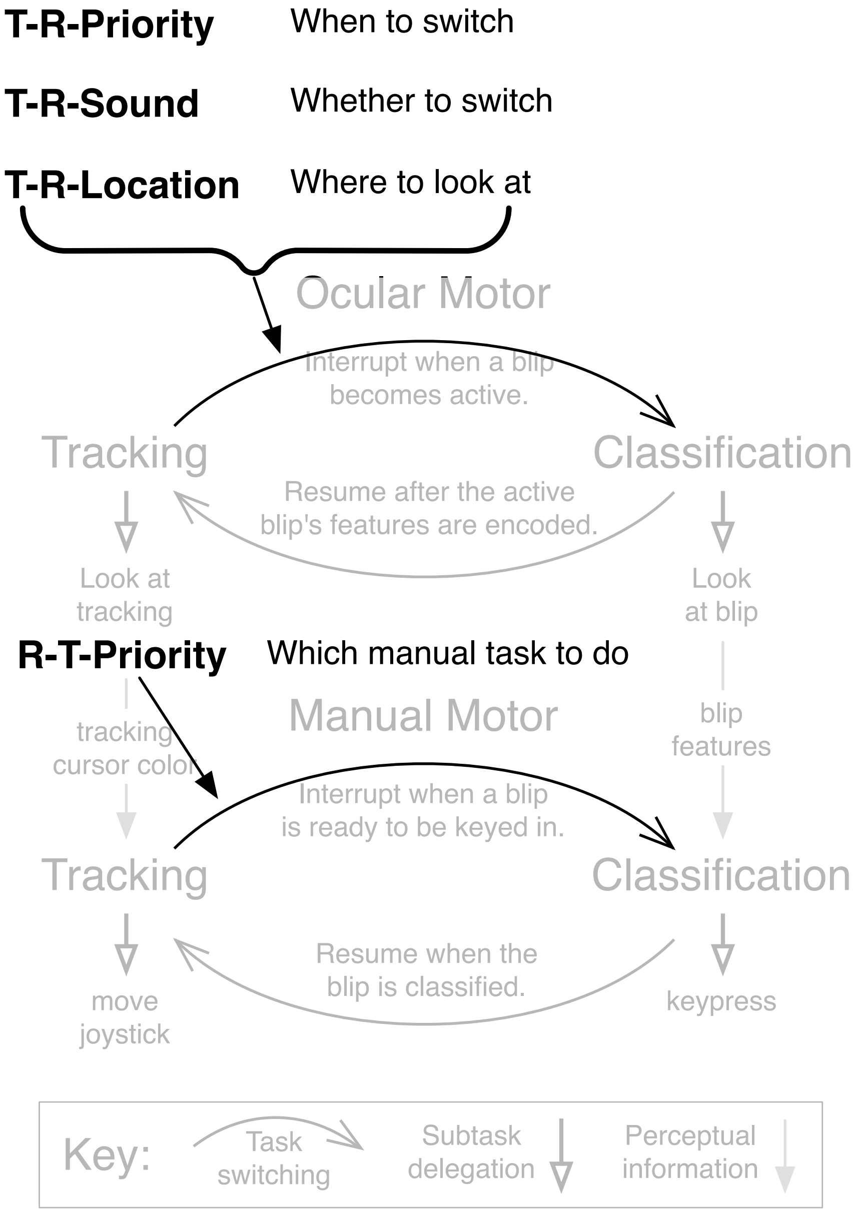


FIGURE 26. The influence of each strategy dimension on the executive processes.

possible effects of strategies and the free parameter of hostility-encoding time on three of the four classification stages defined earlier with eye movement data (see Figure 11): pre-radar, blip-search, and post-radar. The blip-encoding time is directly controlled by perceptual parameters, which are already calibrated to the eye movement data. This qualitative analysis is useful for understanding what quantitative effects the strategies might cause and where to look for these effects.

The effect of the first three strategy dimensions on task performance is straightforward. The T-R-Priority dimension affects when the gaze moves to the radar display, and hence the pre-radar stage. Specifically, the Track-then-Eyes-to-Blip (TEB) strategy takes extra time to do tracking before switching to classification and should thus produce longer pre-radar durations and classification times than the Immediate-Eyes-to-Blip (IEB) strategy, although it should also lead to smaller RMS tracking error than the IEB strategy.

The T-R-Sound dimension also affects the pre-radar stage because the Eyes-to-All-Sounds (EAS) strategy initiates the ocular-motor transition immediately after hearing a sound, whereas the Eyes-to-Color-change-Sounds (ECS) strategy waits until the frequency and timbre properties of the sound are perceived (to differentiate between the color-change and blip-appearance alerts). Therefore, ECS should cause longer pre-radar durations and classification times than the EAS strategy. ECS should reduce tracking error, however, because it only responds to color-change alerts and thus leads to fewer interruptions to the tracking task.

The T-R-Location dimension affects the blip-search stage in the peripheral-not-visible conditions. With the Look-prior-Blip-Location (LBL) strategy, the model uses memory to infer blip locations, and can sometimes fixate the correct active blip with the first fixation to the radar display. Thus, the LBL strategy likely leads to smaller blip search times than the Look-Window-Center strategy.

The R-T-Priority dimension has more complex effects on performance than the other dimensions, because its effects can vary depending on the hostility-encoding time as well as the strategies. As discussed earlier, R-T-Priority controls what to do with the hands after acquiring the visual features of a yellow blip, and there are three strategies in this dimension: Keypad-Then-Joystick, Joystick-Then-Keypad, and Keypad-If-Green. The effects of these strategies can vary depending on the hostility-encoding time. For example, using the Keypad-Then-Joystick strategy does not guarantee that the manual-motor processing needed for the classification task would be started before tracking, because if the hostility encoding time is long, the classification has to be postponed until the hostility is available. Much like in the PRP paradigm, the model can take on different execution paths during this delay. These execution paths are discussed next.

Figure 27 shows two possible execution paths for when the model selects a hostility response before it has been able to resume manual tracking. This early response-selection completion happens most likely for red and green blips, but also for yellow blips if the hostility-encoding time is short. In the graph, Path 1 immediately proceeds to manual classification upon selecting a response, whereas Path 2 postpones the response and instead does manual tracking first.

The three alternative strategies in the R-T-Priority dimension take different paths in the graph. The Keypad-Then-Joystick strategy takes Path 1 because it always prioritizes classification, while the Joystick-Then-Keypad strategy takes Path 2 because it always prioritizes tracking. The Keypad-If-Green (KIG) strategy, however, takes different paths in different circumstances. If the tracking cursor is green when the eyes move back to

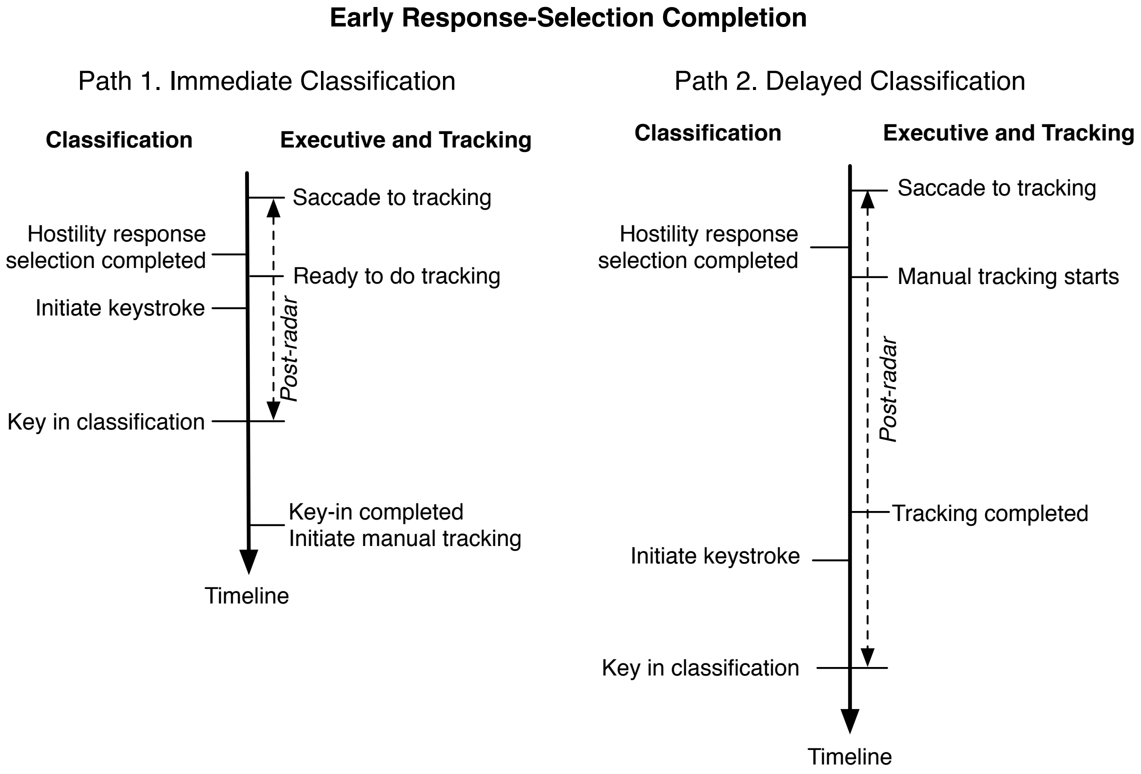


FIGURE 27. Two possible sequences of cognitive and motor events for the circumstances in which the eyes move back to tracking and in which the model selects a hostility response before resuming manual tracking. Path 1 does manual classification before manual tracking, whereas Path 2 does the contrary. The dashed line marks the post-radar stage.

the tracking display, the KIG strategy will prioritize classification and thus take Path 1; otherwise, the strategy will prioritize tracking and take Path 2.

Figure 28 shows the possible execution paths for when the model selects a response after resumes manual tracking, which likely occurs for yellow blips. In the graph, Path 3 interrupts tracking after selecting a hostility response, whereas Path 4 continues tracking until the tracking cursor becomes green. The Keypad-Then-Joystick strategy leads to Path 3 because it prioritizes classification, whereas the Joystick-Then-Keypad strategy leads to Path 4 because it prioritizes tracking. The Keypad-If-Green strategy could lead to either path depending on the color of the tracking cursor at the moment the hostility response is selected.

Figures 27 and 28 show that different execution paths lead to different post-radar durations. Because Paths 1 and 3 have shorter post-radar durations than Paths 2 and 4, the Keypad-Then-Joystick strategy should predict the shortest post-radar durations, whereas the Joystick-Then-Keypad strategy should predict the longest. The Keypad-If-Green strategy produces a mixture of paths in both the early and late response-selection circumstances, and should predict, on average, intermediate post-radar durations.

Figure 29 summarizes the above analyses about the influence of the free parameter and strategy dimensions on the durations of three classification stages: pre-radar, blip search, and post-radar. Notably, no more than two factors directly affect one stage. This graph suggests that it is possible to infer participants' strategies in each dimension based on how long they spent on each of the three classification stages. The next chapter will use this method to identify each participant's multitasking strategies.

The above analyses shows qualitatively how the model predictions might vary depending on the strategy and parameter settings, but quantitative analysis in the form of computational cognitive modeling is still needed to produce all possible interactions

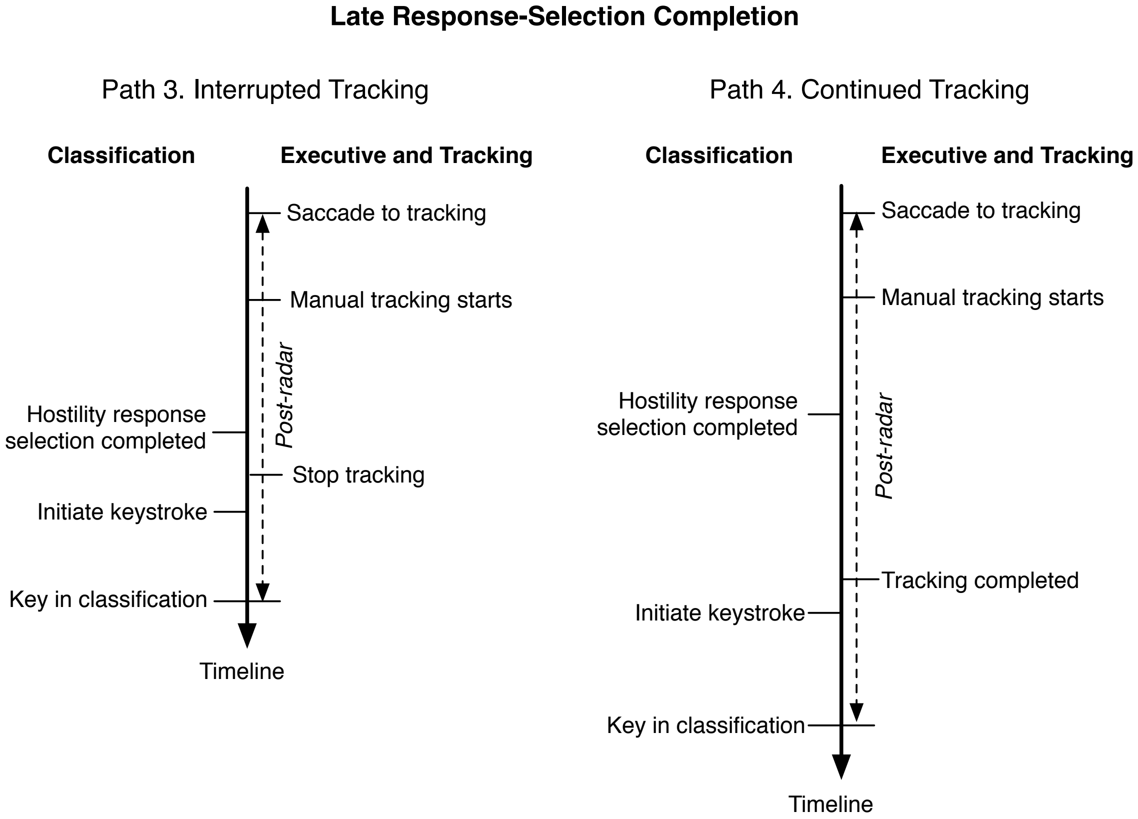


FIGURE 28. Two possible sequence of cognitive and motor events for the circumstances in which the eyes move back to tracking and in which the model selects a hostility response after resuming manual tracking. Path 1 interrupts manual tracking to classify, whereas Path 2 continues tracking until the cursor becomes green and then classifies.

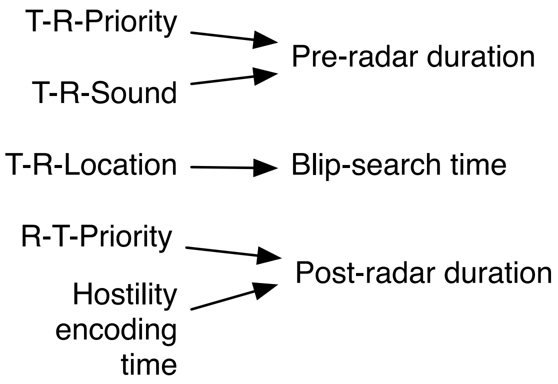


FIGURE 29. The influence of the four strategy dimensions and the free parameter on the three classification stages.

of the different strategy dimensions. For example, the sequence of events might not unfold as imagined above if multiple blips appear at the same time. For example, the Eyes-to-All-Sounds strategy may interact with the Look-prior-Blip-Location strategy because Eyes-to-All-Sounds leads to more glances to the radar display, which could help the model discover new blips as soon as they arrive, and permit the model to better remember their locations. These memorized blip locations can then be used by the Look-prior-Blip-Location strategy to shorten the blip search time. Such interactions across strategy dimensions depend on the moment-to-moment task situations, and they cannot be analyzed without running the model through all of the trials. To efficiently run the model with all the different strategy and parameter settings, however, enormous computational power is needed. This dissertation research addressed this computational challenge by developing a parallelized cognitive modeling system, discussed next.

The Parallelized Cognitive Modeling System

Fully exploring strategic dimensions and parameter settings entails a large number of models. In this dissertation research, there are 24 different strategies (the combinatorial product of the four strategy dimensions) and 16 settings for the hostility recoding time parameter, resulting in 384 models for each participant. Running one dual task model takes about 3 minutes on a contemporary desktop machine, and thus running 384 models for a single participant would require about 19 hours. Since in this research, models are built for each participant, this number would be further multiplied by the number of participants, ten. It thus would take about 190 hours, or 8 days, to run this model on a desktop machine. Given that many runs of the model are needed, such as to debug the production rules, this long model running time makes it impractical to explore a large strategy and parameter space using a traditional modeling system on a contemporary desktop computer.

To address this computational challenge, I developed a parallelized cognitive modeling (PCM) system that utilizes a computer cluster to speed up large-scale model explorations. The specific cluster used here is a part of the University of Oregon's Applied Computational Instrument for Scientific Synthesis project (ACISS, NSF Award #0960354, Principle Investigator Allen Malony). Though previous research (Gluck et al., 2010) used a computer cluster to explore model parameters, the system presented here pushes the boundary further by using a cluster to explore task strategies.

Figure 30 shows the components of the PCM system and illustrates how it generates and parallelizes model executions. The system consists of two main programs: a model spawner and a job scheduler. The model spawner takes three files as inputs: the basic model, the parameter space, and the strategy space. The basic model implements a partially instantiated task strategy using production rules. For this task, the basic model implements the structure described by Figure 25. The basic model also includes slots that will be filled in later with specific parameter and strategy settings to generate complete models. The parameter space defines the ranges and sampling intervals for the model's free parameter (one in this run of the system). The strategy space defines instructions about how to modify the basic model to implement each strategy. These instructions include which production rules to modify, and what conditions (lefthand side) and actions (righthand side) to add and to delete from the rules. The model spawner takes the three input files and generates all possible versions of the model across the space of strategy and parameter settings.

The job scheduler takes the various models generated by the spawner and parallelizes them on a computer cluster. The job scheduler grabs as many computer cores

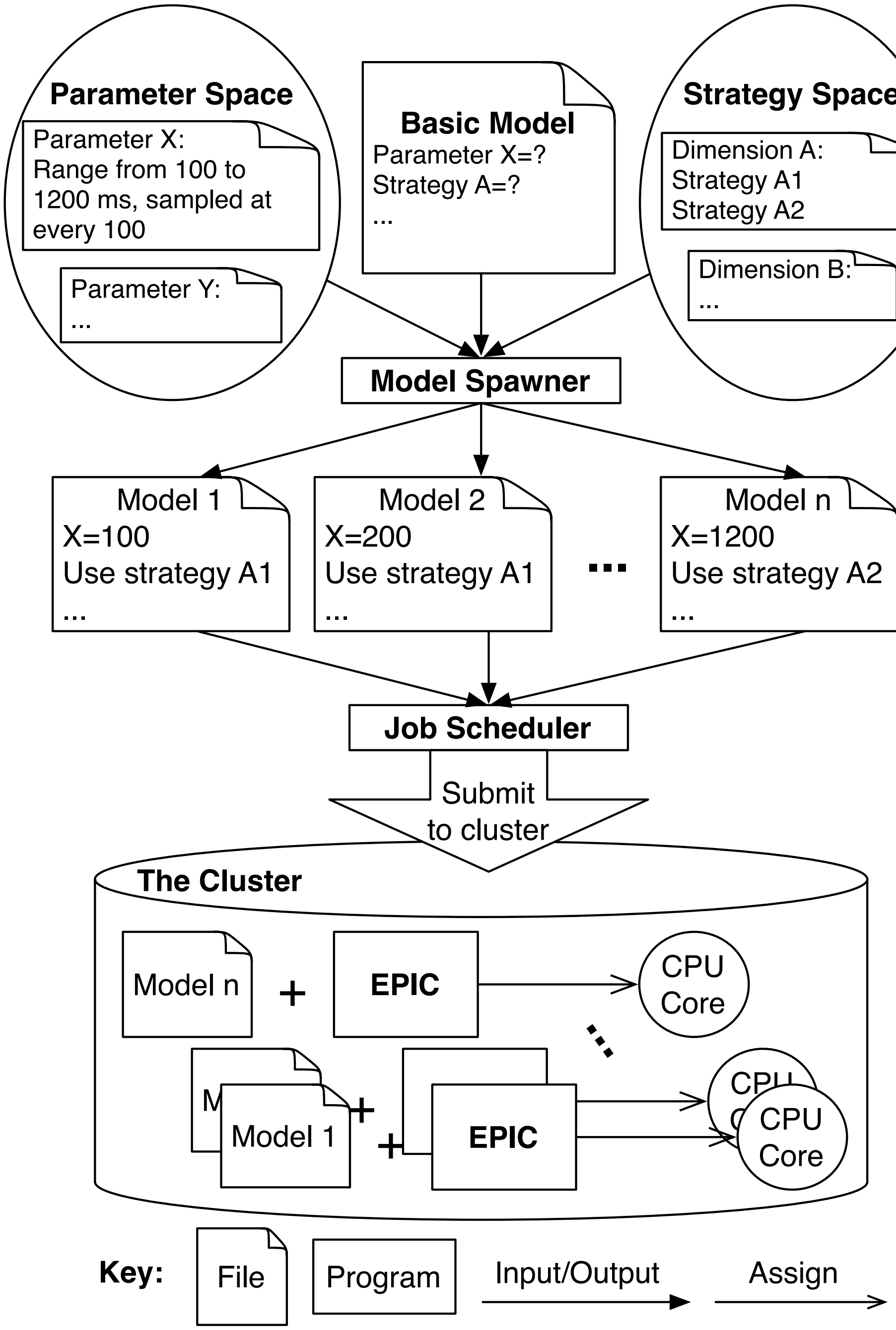


FIGURE 30. The components of the Parallelized Cognitive Modeling system.

as possible from the cluster (limited by a per-user quota), and assigns each model to run on a CPU core. There are typically fewer cores than the number of models, and so the models cannot be all run at once. Rather, the scheduler uses a queue to keep the models that are yet to be run, and once a model finishes its execution and a CPU core is freed, the scheduler dequeues a new model and assigns it to the freed core. By using 240 CPU cores on the cluster, the original 190-hour running time on a desktop machine is reduced to less than 50 minutes. This enabled me to conduct truly large-scale modeling and to collect more comprehensive results than previous cognitive modeling studies.

Running the Models

As discussed earlier, 384 models were run for each participant across all strategy and parameter configurations. Each model was run through the participant's original stimulus conditions. That is, a model experienced exactly the same tracking target movements and blip stimuli that a participant experienced. EPIC's various noise factors were turned on to simulate noisy human behaviors. To smooth out random variations in model predictions, each model was run ten times and the predictions of all ten runs were averaged.

Many data were collected from the models, including classification time for each blip, tracking errors sampled at 12 Hz (just like in the human experiment), eye movements, and payoff. The model payoff is calculated using the same scheme as that used for the participants.

Summary

This chapter introduced the EPIC cognitive architecture, discussed in details all the components of the dual task model, and illustrated the parallelized cognitive modeling system that is developed for extensive strategy and parameter exploration. The EPIC cognitive architecture provides many necessary components for modeling general multitasking scenarios, which served as an excellent foundation for modeling this complex dual task experiment. Great care was taken in building each component of the dual task model to ensure that each step of the task is characterized by the model as accurately as possible. In particular, the tracking movement implementation in the EPIC architecture was improved in this dissertation based on a thorough analysis of the moment-to-moment tracking movement data that we have. Despite the many components of the model, the number of free parameters was kept minimum by calibrating as many parameters as possible to parts of the human data that are not used for model evaluation. These efforts made sure that the two single-task components, tracking and classification, are comprehensive and are accurately tuned to each participant's own cognitive, motor, and perceptual characteristics. On top of these two single-task components, this chapter explored the possible variations of the executive processes that manage the task-switching processes. A comprehensive strategy space comprised of four strategic dimensions was proposed, and the potential effects of the different substrategies on model predictions were analyzed qualitatively. The results from such a qualitative analysis would later help interpret the myriad of model predictions. Finally, to meet the substantially increased computational demand of the strategy and parameter space exploration, a parallel cognitive modeling system was developed, making the exploration feasible.

The next chapter discusses the decisions and efforts that were made in developing rigorous approaches for finding the best-fitting models, and presents the results of this large-scale strategy and parameter exploration. The results point to new approaches for conducting principled model evaluations, and reveal new insights into human multitasking.

CHAPTER V

MODELING RESULTS

This chapter is divided into two sections. The first section shows the methods that I developed for finding best-fitting models, and discusses how these methods provide a rigorous approach for evaluating and comparing cognitive models. The second section presents the best-fitting models for every participant in the dual task experiment and discusses the implications of the strategic differences found between the top and bottom performers.

Approaches to Finding the Best-Fitting Models

As in other cognitive modeling studies, one goal of this dissertation is to determine, among all the alternative models, which model best explains the human data. The bigger goal, and challenge, is to figure out new ways to identify the strategies, substrategies, and strategic overlapping that people employ when multitasking, and to figure out new ways to probe human data to validate hypotheses pertaining to strategic decisions.

In the search for the best-fitting model, several decisions will impact the validity of the modeling results. I must:

- Decide whether to explain the aggregated average performance or the individual participants' performance.
- Decide on the data and measures for evaluating the models' goodness-of-fit.
- Decide which models best fit the data, and which can be ruled out.

This dissertation pursues a rigorous approach to each step of the analysis. This section presents the exploration that has been taken to arrive at the final model-evaluation approach, and shows an example of how this modeling approach is applied for one participant, participant P10.

Decide Whether to Explain the Average Performance or the Individual Performance

Cognitive scientists and cognitive modeling research often only examines aggregated human data (e.g., Byrne & Anderson, 2001; Hornof, 2001; Pashler, 1989; Salvucci & Taatgen, 2008), but for multitasking research, recent studies (e.g., Howes et al., 2009; Schumacher et al., 2001) show that there are often differences across individual participant behavior. Explaining such differences of course requires analyzing individual data.

To determine whether modeling average performance is sufficient for explaining individual behavior in this research, models were first developed with parameters calibrated using the average of all participants' data instead of a single participant's data. The exploration of the strategy and parameter space was conducted as discussed in Chapter IV (with semi-automatic strategy generation and parallel model execution on a computer cluster), which resulted in 384 models that span across 24 different strategy configurations and 16 different hostility-encoding time settings. The resulting model output was compared against all of the individual performance data.

Figure 31 helps with the decision of whether to explain average or individual performance. The figure shows that the average-performance model does not account for the variation in individual performance. Specifically, Figure 31 compares the average-performance model's predictions with the individual participants' data. The graph plots the classification time against the RMS tracking error. Each panel plots the results for one of the four experimental conditions. Each diamond symbol and error bar shows one of the ten participants' mean performance and, for the classification time, the 95%

confidence intervals. The plot symbols for the top (P06) and bottom (P04) performer are annotated. (There are no horizontal error bars because each session only resulted in one RMS tracking error.) The clouds show the predictions of the average-performance model, with each point representing one of the 384 models. The more the clouds fall within the human data brackets, the better the models explain the observed data. To most vividly illustrate the situation, the graph only shows the observed data and model predictions for the yellow blips.

As can be seen in the graph, the individual participants' data span across large areas that sometimes, particularly in the bottom panels, fall outside the clouds. This shows that the models built for fitting the average-performance data indeed cannot account for the variations in individual performance.

The above results suggest that individualized models are needed to explain how people completed this dual task. Participants have different cognitive, perceptual, and motor capabilities and, to generate accurate predictions for each participant, these differences need to be reflected in the model's parameter settings. Such individualized parameter settings were obtained by calibrating the parameters to each individual's data rather than the aggregated data. Then, the strategy and parameter space can be explored to find the best-fitting model for each individual participant.

Figure 32 shows the results of this individualized modeling for the top performer P06 and the bottom performer P04. (Again, only the results for yellow blips are shown.) Note how the prediction clouds for each condition shift across the two participants. When compared to Figure 31, it can be seen that the individualized-parameter models explain the individual data better than the model that was parameterized based on the aggregated human data. The plots for P06 in Figure 32 show how the clouds now approach P06's data more closely than the clouds in Figure 31. The plots for P04 in Figure 32 show the

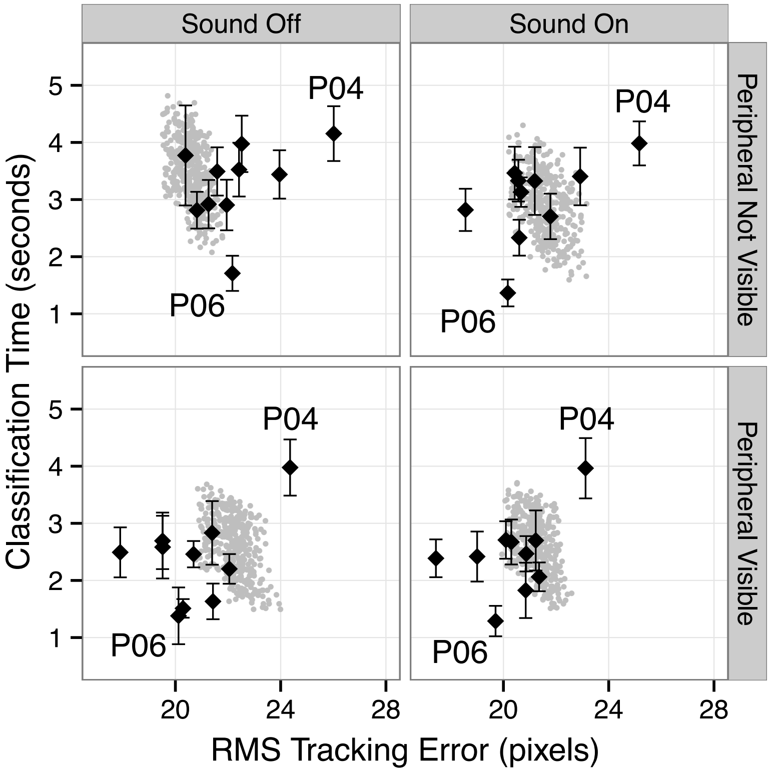


FIGURE 31. The predictions (gray clouds) of the models that were built for the aggregated human data, compared to individual data from each of the ten participants (the diamond-shaped plot symbols). Each of the four panels shows one experimental condition. The columns differentiate the two sound conditions, and the rows differentiates the two peripheral-visibility conditions. The error bars show the 95% confidence intervals of each each of the participant's average classification time, calculated using each participant's trial data. There are no horizontal error bars because every condition (session) had only one RMS tracking error. Each point in the clouds represents one of the 384 models that resulted from the large-scaled strategy and parameter exploration. For clarity, only the data of yellow blips are shown. P06 (the top performer) and P04's (the bottom performer) data are indicated.

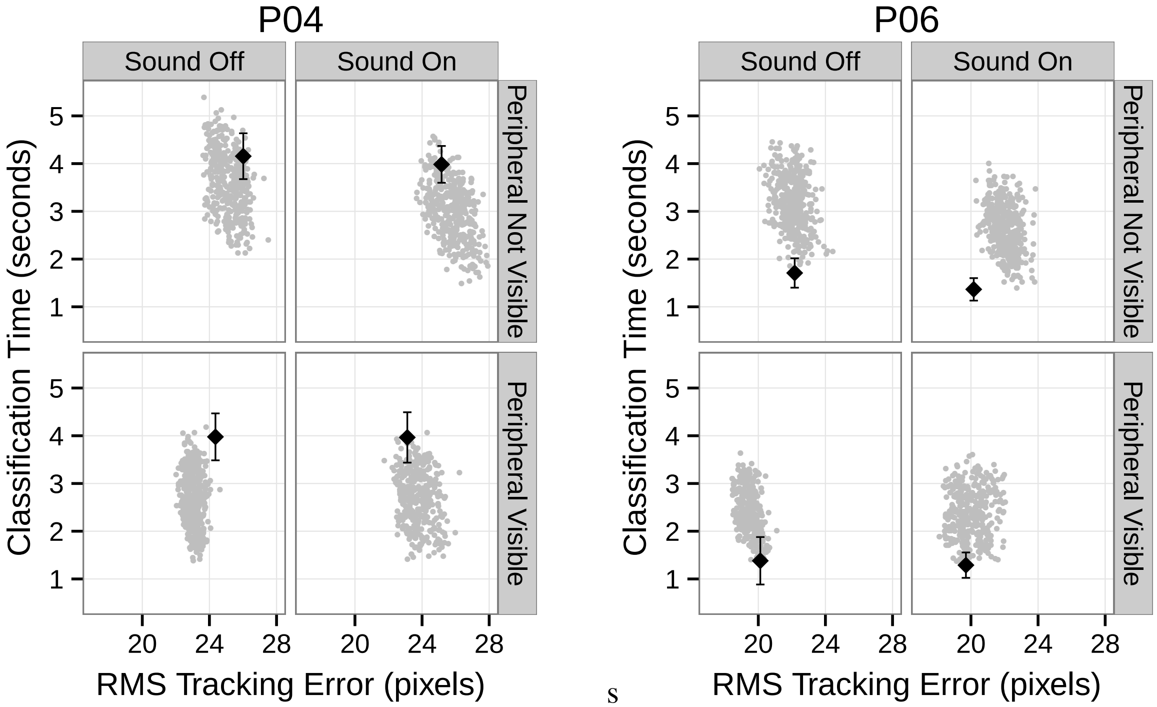


FIGURE 32. The observed data, and the predictions of the individualized models, for the bottom performer, P04, and the top performer, P06. Only data for yellow blips are shown.

prediction clouds readily encompassing P04's data in all conditions whereas, in Figure 31, P04 fell outside the clouds.

The above results illustrate a situation in which modeling individual performance appears to be needed to understand multitasking performance. It perhaps raises an alarm for multitasking research that does not examine individual differences in that it may be over-generalizing the data and drawing incorrect conclusions (perhaps such as Strayer & Johnston, 2001). Though modeling individual performance increases the computational demand of modeling, and the difficulty of the analysis, it improves the rigor and the reliability of the conclusions. This dissertation applies individualized modeling to the remaining analyses.

Decide on the Measures for Model Evaluation

Another important step in finding the best-fitting model is to decide on the measures for evaluating the models. Many cognitive modeling studies aim at fitting high-level data such as reaction time (e.g., Byrne & Anderson, 2001; Hornof & Kieras, 1997; Kieras et al., 2000; Salvucci, 2009). Such high-level data are typically influenced by many factors, and may not provide sufficient constraints for inferring the low-level strategies that participants adopted. As discussed in Chapter II, previous research (Hornof & Zhang, 2010) showed that when only modeling the classification time of the dual task experiment, it is easy to arrive at a set of multitasking strategies that predicts the classification time correctly, but entirely mispredicts the eye movement patterns. Such mispredictions indicate that reaction time alone does not provide tight constraints for identifying strategies. Fitting such high-level measures with many free parameters and strategies will sometimes surely lead to incorrect conclusions about human behavior.

This dissertation avoids this problem by using detailed eye movement data to evaluate the models. In Chapter IV, the qualitative analysis of the effects of the strategies showed that some eye-movement related measures, such as the pre-radar duration (the time between when a blip changes color and when the eyes land on the radar display), are directly influenced by one or two strategies that manage visual task switching. This direct influence, along with the short time span of these measures, suggest that they could be used to evaluate the models more rigorously.

One challenge when using the detailed eye movement data, however, is that the eye movement data can produce multiple measures to fit. In this analysis, for example, there are the pre-radar duration, the blip search time, and the post-radar duration (the time between eyes-to-tracking and keying-in classification). Fitting a model to multiple measures can be problematic because if the measures are used separately, each measure may lead to a different best-fitting strategy and parameter settings. The challenge is to use multiple measures in parallel to triangulate on how people were really doing the task.

One way to avoid conflicting results in modeling multiple measures is to combine the measures into one, such as by converting each into a unitless measure and then averaging the converted measures. Zhang and Hornof (2014) used this method to evaluate the dual task models. In that study, the models were evaluated based on their average absolute percentage error on two measures: the classification time and the RMS tracking error. The two error percentages were averaged to produce a single goodness-of-fit measure. Although this approach of combining goodness-of-fit arguably permitted good, comprehensive evaluation of the models, it relied on the assumption that a 1% error in the prediction of one measure is equivalent to 1% error in another. This is not always true. Dealing with multiple measures by combining them, at least in terms of average absolute percentage errors, is not always a reliable solution. Hence, this dissertation takes a different approach.

Another way to avoid conflicting results in modeling multiple measures is to carefully pick the measures such that each measure is used to fit a different set of strategies or parameters; this way, no two measures are used to infer the setting for the same strategy or parameter dimension. This is the approach taken in this dissertation. As shown previously in Figure 29, each strategy dimension only affects the duration of one classification stage: T-R-Priority and T-R-Sound affect pre-radar duration, T-R-Location affects blip search time, and R-T-Priority and hostility-encoding time affect post-radar duration. In turns, each measure can be used to fit a different set of strategies: (a) The pre-radar duration is used to determine the best-fitting strategies in the T-R-Priority and T-R-Sound dimensions; (b) The blip search time is used to determine the strategies in the T-R-Location dimension; and (c) the post-radar duration is used to determine the strategies in the R-T-Priority dimension and the hostility-encoding time. Therefore, there is no overlap between the strategies that each measure is used to fit, and the strategies in all four dimensions (as well as the single free parameter) can be determined with the three eye-movement measures without interference.

The above discussion outlines a useful and general way to use detailed data and multiple measures to evaluate complex models. Instead of trying to find a unifying measure that incorporates all measures into one, this approach uses measures that are influenced by separate sets of strategies to find the best-fitting model. This method and its efficacy will be further illustrated by a concrete analysis of Participant P10's models.

Decide Which Models Best Fit the Data

Having decided the appropriate measures for model evaluation, the final step—deciding which one is the best-fitting model—may appear straightforward, but it is not. And many cognitive modeling studies do not approach this step in in a clearly principled manner. For example, as discussed in Chapter II, some modeling studies do not report alternative models that were considered, how models besides the best-fitting model were considered and rejected, and whether there was any statistical basis to reject competing models (e.g., Byrne & Anderson, 2001; Fleetwood & Byrne, 2006; Salvucci, 2006; Taatgen et al., 2007). There is a need to establish principled approaches for evaluating and comparing models, and developing such approaches would improve the rigor of modeling studies.

The large-scaled exploration of the strategy and parameter space conducted in this research, which is likely to be pursued by other researchers in the future, further points to the need for principled methods for finding best-fitting models. Enormous prediction spaces are hard to analyze and quite often, different models will likely predict similar results, making it difficult to promote one over the other. Previous research such as Brumby et al. (2009) used visualizations similar to Figure 32 to address the problem. Such visualizations can indicate roughly where the human data sits in the prediction space, and suggest strategies that can explain the human data. However, such visualizations cannot be used to accurately decide which model best fits the human data, nor can they provide a statistical basis for accepting or rejecting models. Previous research (Zhang & Hornof, 2014) on modeling the dual task data presented here also failed to properly address this challenge in that the analysis selected the best-fitting models without showing that other alternative models provided, statistically speaking, significantly worse fits.

This dissertation addressed this challenge head-on and develops a set of principled approaches for (a) visualizing model predictions, (b) finding the best fitting strategy and parameter settings, and (c) ruling out competing models with supporting statistics. The detailed procedure will be illustrated below through a concrete analysis of one participant's models.

Example: Finding the Best-Fitting Models for P10

This section presents the analysis of the models for P10 to illustrate the model-evaluation processes developed in this dissertation work, and the efficacy of this approach. A similar analysis was applied to each of the three classification stages—pre-radar, blip search, and post-radar—to determine the best-fitting strategies and parameter. The analysis takes the following three steps:

1. Determine which strategy dimensions or parameters, and which experimental manipulations, have the greatest effect on the model predictions. (This reduces the number of relevant factors that need to be analyzed in the subsequent two steps.)

2. Create visualizations to compare the model predictions with the human data. (This helps the analyst to form an intuitive understanding of how interaction between (a) the model's strategies and parameters and (b) the experimental manipulations will change the model's predictions.)
3. Conduct statistical analyses to determine the goodness-of-fit of all models, and to find the best-fitting model or models.

These steps are illustrated below for Participant P10. P10 was chosen because this participant ranks sixth in terms of the payoff earned, which should represent the midrange of the ten participants' behaviors. P10 is also chosen because this analysis identified two best-fitting models for this participant, and showing how these two best-fitting models are determined and why they cannot be further differentiated helps to illustrate the analysis process, and how it does not always lead to a single irrefutable answer.

Note that the graphs presented in this section show a single participant's data, and thus the error bars in these graphs represent the 95% confidence intervals calculated for that single participant. That is, the error bars represent the variability of the individual's performance, given that a participant is tested in each stimulus condition multiple times (e.g., for each participant, an yellow blip is tested 24 times in each condition). This is different from the error bars presented in Chapter IV, as those error bars represent the variabilities across participants.

The following subsections illustrate the three steps of the analysis procedure, specifically for the pre-radar stage.

Step 1: Determine what factors affect the model's predictions regarding pre-radar duration

For the first step of the analysis—determining which factors affect the model's predictions—the previous qualitative analysis of the effects of strategies provides some useful directions. For the pre-radar stage (from when a blip changes color to when the eyes land on the radar display), as previously discussed, among the four strategic dimensions and the free parameter, the T-R-Priority and T-R-Sound strategy dimensions affect its duration. However, as discussed in Chapter IV, the relationships between strategies and the durations of three classification stages derived from the qualitative analysis might not stand in the quantitative simulation of the whole experiment, as the simulation takes into account nuanced situations such as classifying multiple blips at the same time. To make sure that the relationships hold in the simulation, a sensitivity analysis is needed.

The goal of a sensitivity analysis is to find out how strongly various strategy and parameter settings affect the the model predictions. To conduct a sensitivity analysis, first an ANOVA model is built using the predicted pre-radar duration as the response variable, and the experimental conditions, strategies, and the free parameter—hostility encoding time (HET) for yellow blips—as the predictors. Note that in this step of the analysis, the human data are not used because this step does not calculate the models' goodness of fit, but instead examine what factors affect the model's output. The experimental conditions are included as predictors to isolate their effects and to get better estimates for the effect size of strategies and the HET parameter. After constructing the ANOVA model, an analysis of effect size is applied to all the predictors to find out how much each predictor affects the model predictions.

Table 3 shows the effect size of all the predictors on the pre-radar duration. The first column lists the predictors included in the ANOVA model. The second column shows the effect size eta-squared η^2, which measures the ratio of variance explained in the response variable by each predictor variable. In other words, η^2 represents how much a response variable (such as test score) fluctuates in response to the changes in a predictor

TABLE 3. Effect size of the experimental factors and the model strategies and parameter on pre-radar duration. The × symbol represents the interaction term between two predictors.

Source of Effect	Predictor	$\eta^2(\%)$
Experimental Factors	Sound	19.4
	Peripheral Visibility	46.0
	Sound × Peripheral Visibility	17.0
	Blip Color	00.7
Model Strategies and Parameter	T-R-Priority	05.2
	T-R-Sound	01.4
	T-R-Location	00.0
	R-T-Priority	00.1
	Hostility Encoding Time	00.3

variable (such as age); predictors with larger η^2 have a larger influence on the response variable. Using η^2 is just one way to determine the effect size. Other effect size measures such as partial eta squared η_p^2 can also be used (though see Levine and Hullett, 2002, for why η^2 is generally preferred to η_p^2). The goal of this effect-size analysis, regardless of which measure is used, is to determine the relevant factors so that irrelevant factors can be removed from later analyses.

Table 3 shows that as the qualitative analysis predicted, among the strategy dimensions and the free parameter, only the T-R-Priority and the T-R-Sound strategy dimensions influence the pre-radar duration. As expected, based on how the model uses different sensory information in different experimental conditions as cues for the classification task, the sound and peripheral visibility conditions substantially affect the predicted pre-radar duration: Peripheral visibility accounts for 46% of the variance in pre-radar duration, while sound and the interaction between sound and peripheral visibility each account for about 20% of the variance. Strategies and the HET parameter have much less influence. Among all the strategies, only the T-R-Priority and T-R-Sound dimensions account for more than 1% of variance.

The above results suggest that although the model predictions are numerous (3072 data points resulted from 24 strategy configurations, 16 parameter settings, four experimental conditions, and two blip colors), these predictions can be aggregated across several factors that had little impact on the pre-radar durations without affecting the accuracy of subsequent analyses. Specifically, for the pre-radar stage, the model predictions are aggregated across the blip color, T-R-Location, R-T-Priority, and hostility encoding time, leaving only the sound, peripheral visibility, T-R-Priority, and T-R-Sound factors to analyze. The analysis of effect size can dramatically reduce the complexity of the data and subsequent analyses.

Step 2: Visualize and compare the predicted pre-radar duration with the human data

After aggregating the model predictions across the factors with little effect on the pre-radar duration, the resulting model predictions can then be compared against the human data using visualizations that illustrate the specific effects of the remaining factors. Figure 33 compares P10's pre-radar duration with the model's predictions across three experimental conditions. All of the factors that impacted the pre-radar duration are shown in the graph, including the experimental conditions, and the T-R-Sound and T-R-Priority dimensions. The human data are represented by the error bars, which show the average human performance and the associated 95% confidence intervals. The model predictions, after aggregated across many factors, are represented by the filled plotting symbols. The closer the model predictions are to the human data, the better the fit.

The sound-off peripheral-not-visible (SOff PNV) condition is excluded from Figure 33 because the duration of the pre-radar stage in that condition is not determined by the T-R-Sound and T-R-Priority strategy dimensions. As discussed previously, in the SOff PNV condition, there are no sound or visual cues for the classification task, and

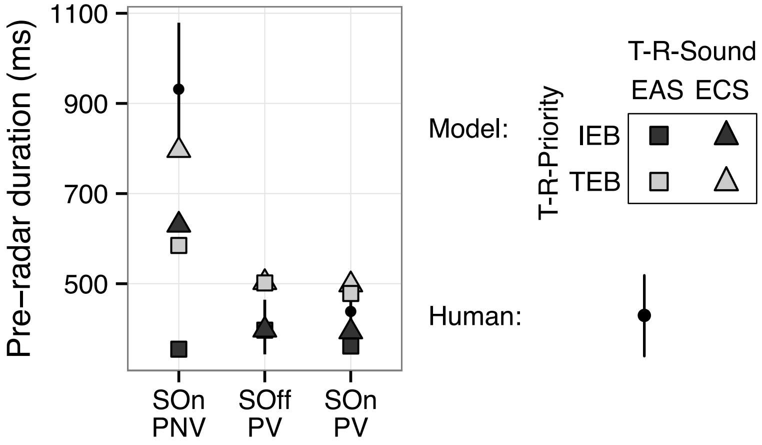


FIGURE 33. P10's pre-radar duration compared against the models' predictions across three experimental conditions. Triangles and squares represent different combinations of T-R-Sound and T-R-Priority dimensions. Model predictions were aggregated across the HET settings and other strategy dimensions not shown here. The error bars represent 95% confidence intervals of P10's average performance. Note the non-zero axis.

the model has to periodically interrupt tracking and deliberately move the eyes to the radar display to check if there are any active blips. The frequency with which the model checks the radar display is determined by a time-interval parameter, which is set to 1.5 s to capture how long participants typically wait between two visits to the radar display. Thus, it is this time-interval parameter, not the two strategies, that somewhat directly determines when the model will move the eyes to the classification display, which in turn determines the pre-radar duration. The pre-radar duration of the SOff PNV condition thus cannot reliably contribute to finding the best-fitting strategies for the T-R-Sound and T-R-Priority dimensions.

The goal for examining Figure 33 is to see how the model predictions compare to the human data, and to possibly determine the best-fitting strategies from the visualization. Note that this research assumes that the participant uses the same strategy across all conditions. (Strategies that directly depend on sound or peripheral-visibility, such as the Eyes-to-All-Sounds strategy, can still be assumed to be applied across all conditions, though they would have no effect on the conditions that lack the necessary sensory information.) This assumption is due to: (a) There is not a strong reason for the participant to change the strategies across the sound and peripheral-visibility conditions; and (b) in the absence of any compelling reason to believe that people change strategies across conditions, the parsimonious explanation is that they do not. Thus, the different strategies in Figure 33 should be evaluated based on how well they match the data across all three conditions. Based on this criterion, for the T-R-Sound dimension, Eyes-to-Color-change-Sounds (ECS, triangles) should be the best-fitting strategy. This is because that in the sound-on peripheral-not-visible (SOn PNV) condition, the predictions of the ECS strategy are closer to the human data than those of the Eyes-to-All-Sounds (EAS, squares) strategy; whereas in the other two conditions, in which the peripheral display is visible, the predictions of these two strategies are nearly identical, as can be seen in how the squares overlap with the triangles. Taking all three conditions into consideration, the ECS strategy fits the human data better than the EAS strategy.

Figure 33 also provides an opportunity for examining whether some of EPIC's default parameter settings were correct. Two parameters that were kept at the default settings affect the model's pre-radar duration: the luminance-change detection time and the sound-onset detection time. Both parameters were set to 100 ms, following EPIC's default settings (see Table 1). The luminance-change detection time affects the model's pre-radar duration in the two peripheral-visible conditions and, as can be seen in Figure 33, in these two conditions, the predictions are all very close to the data. This suggests that the default setting for this parameter is likely correct. The sound-onset detection time affects the predictions of the EAS strategy in the SOn PNV condition. It is difficult to judge, based on Figure 33, whether the sound-onset parameter was set correctly because the graph shows that P10 likely used the ECS strategy rather than EAS, and thus how well the EAS strategy fits P10's data does not offer any information about the correct sound-onset detection time. However, there is one participant, P06, whose best-fitting strategy in the T-R-Sound dimension seems to be EAS. From P06's data, it could be determined that the sound-onset detection time is roughly correct because the EAS strategy's predicted post-radar duration is only about 25 ms less than P06's average post-radar duration. If the sound-onset detection time were set incorrectly, then it would affect the EAS strategy's prediction, and the prediction likely would not match the observed data for P06. Thus, the default settings for both parameters seem correct and should not affect the reliability of the model fitting process.

Though visualizations such as Figure 33 are useful in providing an initial diagnosis of the model, they cannot always help determine best-fitting strategies. For example, Figure 33 cannot be used to determine the best-fitting strategy in the T-R-Priority dimension. This is because neither one of the two strategies in the T-R-Priority dimension fit the data better than the other one across all three conditions: The Track-then-Eyes-to-Blip (TEB, light gray symbols) strategy has better fit in the SON PNV condition, but the Immediate-Eyes-to-Blip (IEB, dark gray symbols) strategy has better fit in the two PV conditions. Thus, to determine the best-fitting strategy in the T-R-Priority dimension, more analysis is needed.

Step 3: Determine the best-fitting strategies for the pre-radar stage

This step of the analysis aims to decisively determine which strategy in the T-R-Priority and T-R-Sound dimensions fits the data the best. In this step, the models are compared based on the root-mean-squared deviations (RMSDs) and r^2s of their predicted pre-radar durations. The RMSD and r^2 of each model are calculated by first aggregating the model's predictions for pre-radar duration (across all trials of an experimental run) into eight data points, one for each experimental manipulation (consists of the sound condition, the peripheral-visibility condition, and the blip-color class condition). These eight data points are then compared with the human data that are aggregated in the same manner.

The best-fitting strategy setting for the pre-radar duration can be determined by finding the strategies that led to the lowest RMSD and highest r^2. Figure 34 shows the RMSD (left graph) and r^2 (right graph) across all four combinations of strategies in the T-R-Priority and T-R-Sound dimensions. As can be seen, the two strategy configurations with the ECS strategy, namely IEB-ECS and TEB-ECS, produced the lowest RMSDs and highest r^2s. Between the two configurations, however, there is no clear winner because TEB-ECS has a lower RMSD and IEB-ECS has a higher r^2. In this dissertation, for situations like this where competing models have similar RMSDs and similar r^2s, the one with the lowest RMSD is chosen as the best-fitting model because this research focuses more on predicting the locations of the data points rather than the trends. Thus, the best-fitting strategy for P10 is TEB-ECS because TEB-ECS has the lowest RMSD (a paired t-test between TEB-ECS and IEB-ECS shows $t(189) = -13.6$, $p < .0001$).

That the TEB-ECS strategy is the best-fitting strategy for P10's pre-radar stage suggests that P10 tried to optimize tracking performance, at least when switching visual processing from tracking to classification. Recall that the TEB strategy, after knowing that a blip changed color, continues tracking until the tracking cursor turns green, and the ECS strategy switches visual to the classification task after an auditory alert indicates a color change. Both strategies aim to optimize tracking: TEB optimizes tracking by keeping it in a money-making state before switching to classification, and ECS optimizes tracking by minimizing interruptions to tracking. Thus, that these strategies are the best-fitting

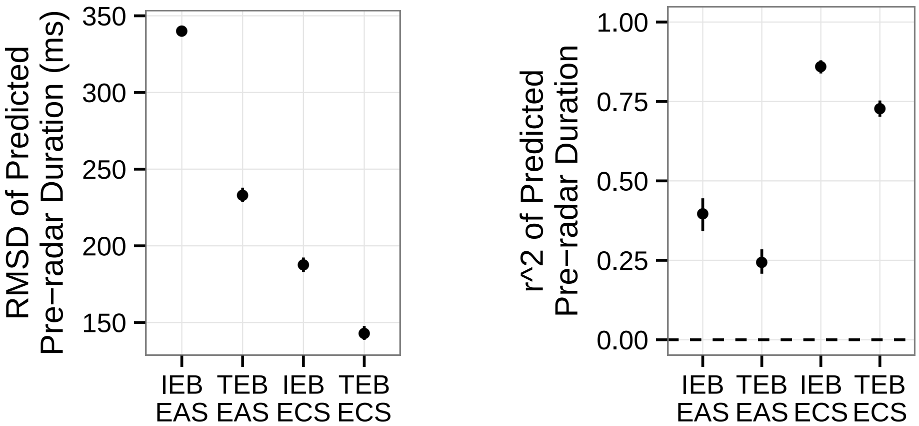


FIGURE 34. RMSD and r^2 of the models' pre-radar predictions across combinations of strategies in the T-R-Priority and T-R-Sound dimensions. For RMSD, a lower value indicates a better fit. For r^2, a higher value indicates a better fit. 95% CI are drawn but because models with the same combination of T-R-Priority and T-R-Sound strategies have similar predictions, they are too short to be visible. Note the non-zero *y* axes on the graph that shows RMSDs.

strategies suggest that P10 prioritized tracking over classification. This reveals how this detailed analysis of strategies can identify trends or perhaps biases in human behavior.

Summary of the Model Evaluation and Comparison Procedure

The above subsections show the new model evaluation and comparison procedure developed in this dissertation. The procedure is specifically designed to handle the results of large-scale model exploration, but it can also be applied to more traditional modeling studies, in which only a few models need to be compared. The procedure addresses a general concern that researchers have expressed about cognitive modeling research, that there is a serious need for more principled approaches to model evaluations (e.g., Howes et al., 2009; Schunn & Wallach, 2005).

Each step of the procedure proposed here is necessary for reaching a decisive conclusion about which strategy or parameter settings best explain the human data. The first step—using an effect-size analysis to determine what factors affect the model predictions—reduces the complexity of the data so that later analyses will not be clouded by irrelevant factors. The second step—visualizing the model predictions in comparison to the human data—gives the analyst an opportunity to form an intuitive judgment about how well the model is capturing the trends in the data. The third step—using statistics and visualizations to determine the best-fitting strategies—decisively determines the best-fitting strategies and parameters. Note that both the second and third steps use visualizations, but only the second step incorporates the human data in the visualizations, because only in the second step, such information is needed for judging how well the model explains the data. This comparison to the human data across all experimental conditions is important because it can help the analyst to discover some potential problems in the model, such as the model matching the data in some but not all of the experimental conditions.

The same analysis procedure is applied to the other two classification stages to determine the best-fitting settings for the remaining two strategy dimensions and the hostility encoding time parameter.

Determining the Best-Fitting Strategy for the Blip-Search Stage

The same three steps just applied to the pre-radar stage are next applied to the blip-search stage. As in the pre-radar stage, a sensitivity analysis is first conducted to determine the factors that affect the blip search time, which is the time from when the eyes arrive on the radar display to when the eyes land on the target blip. Table 4 shows the results of the sensitivity analysis. The peripheral-visibility factor has the greatest effect on blip search time, accounting for 70% of the variance in blip search time. Among the strategy and parameter settings, the T-R-Location dimension accounts for 13% of variance, and other predictors each account for less than 1% of variance. Thus, as previous qualitative analysis suggested, among the strategic dimensions and the free parameter, only the T-R-Location dimension affects how long the model takes to find a target blip, and Steps 2 and 3 of the analysis should only need to examine the effects of the peripheral visibility and the T-R-Location dimension on the models' goodness of fit.

TABLE 4. Effect size of various predictors on blip-search time.

Source of Effect	Predictor	$\eta^2(\%)$
Experimental Factors	Sound	00.8
	Peripheral Visibility	70.7
	Sound × Peripheral Visibility	00.7
	Blip Color	00.0
Model Strategies and Parameter	T-R-Priority	00.0
	T-R-Sound	00.8
	T-R-Location	13.1
	R-T-Priority	00.0
	Hostility Encoding Time	00.0

Figure 35 compares P10's blip search time with the models' predictions across the two peripheral-visibility conditions and the two T-R-Location strategies. Other strategy dimensions and experimental conditions were aggregated because they do not influence blip search time. As can be seen, in the peripheral-not-visible (PNV) condition, the Look-Window-Center (LWC) strategy predicts a longer blip search time than the Look-prior-Blip-Location (LBL) strategy whereas, in the peripheral visible condition, because the model can use peripheral vision to see the active blip and fixate it directly, the blip search time is not affected by the two strategies and is thus zero. It is apparent from this graph that the LBL strategy fits the participant's blip-search time much better than the LWC strategy, particularly in the peripheral-not-visible condition.

Step 3 of the model evaluation analysis—identifying the best-fitting model—is next applied. In this step, to make sure that the LBL strategy does indeed best explain the blip search time, the models' goodness of fit with the human data, measured in terms of the

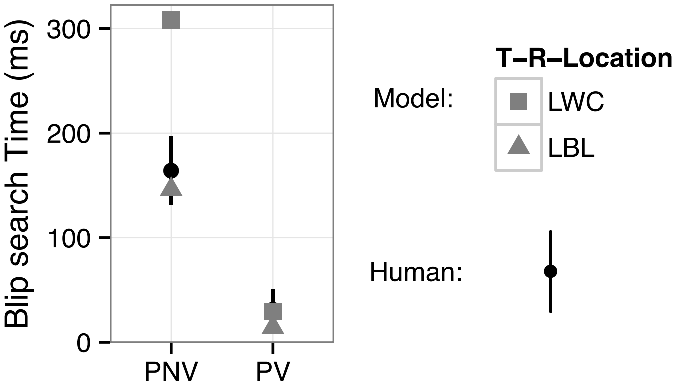


FIGURE 35. P10's blip search time compared with models' predictions across two peripheral-visibility conditions. The triangles and squares represent the model predictions. The error bars show the 95% CI of P10's average blip search time.

RMSD and r^2, need to be analyzed and compared. Figure 36 shows the RMSD and r^2 of the blip-search time predictions across the two T-R-Location strategies. Again, because all models with the same T-R-Location strategy predict similar average blip search times, the error bars are too small to see on the graph. As the two graphs show, when compared to the data observed for Participant P10, the LBL strategy resulted in a smaller RMSD ($t(381) = 119$, $p < .0001$) than the LWC strategy, and the two strategies have about the same r^2s. These results indicate that, for P10, the LBL strategy is the best-fitting strategy in the T-R-Location dimension. Recall that in the peripheral-not-visible conditions, the LBL strategy uses visual memory to help quickly locate active blips, whereas the LWC strategy almost always requires additional visual search to find the active blips. That the LBL strategy provides the best fit to P10's data suggests that P10 attempted to optimize classification performance by shortening the visual search time. This concludes the examination of blip search time. The detailed analysis of eye movement data and large-scale exploration of task strategies now proceeds to the next stage in the task, the post-radar stage.

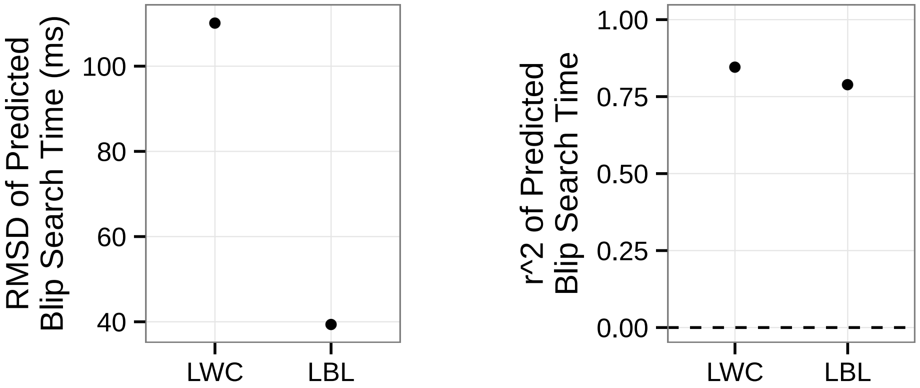


FIGURE 36. The RMSD and the r^2 of the models' blip-search time predictions, when compared to Participant P10's observed performance, across the Look-Window-Center (LWC) and Look-prior-Blip-Location (LBL) strategies. Note the non-zero *y* axes on the graph that shows RMSDs.

Determining the Best-Fitting Strategy and Parameter for the Post-radar Stage

The post-radar duration is the time from when the eyes switch back to the tracking display to when the blip number is keyed in. Table 5 shows the effect size of all predictors on the post-radar duration. As can be seen, only three factors have more than 1% influence on the variance of the post-radar duration. These factors are: blip color (an experimental condition), the R-T-Priority strategy dimension, and the hostility-encoding time (HET) parameter. The following analyses thus focus on the effects of these three factors on post-radar duration.

TABLE 5. Effect size of various predictors on post-radar duration.

Source of Effect	Predictor	$\eta^2(\%)$
Experimental Factors	Sound	00.0
	Peripheral Visibility	00.5
	Sound × Peripheral Visibility	00.0
	Blip Color	34.9
Model Strategies and Parameter	T-R-Priority	00.1
	T-R-Sound	00.0
	T-R-Location	00.1
	R-T-Priority	39.3
	Hostility Encoding Time	08.8

Figure 37 shows how blip color, the R-T-Priority strategy dimension, and the HET parameter jointly affect the post-radar duration. The graph plots the post-radar duration against the HET, which ranged from 100 to 1600 at 100 ms intervals. The solid, horizontal lines indicate P10's average post-radar duration, and the dashed lines indicate the 95% CIs. The human data are horizontal lines because the data are not affected by the model's HET setting. The model predictions for red/green blips also largely fall on three horizontal lines because HET only affects the predictions for yellow blips. The three plot symbols show the three different R-T-Priority strategies. The closer the model predictions are to the solid lines, the better the fit.

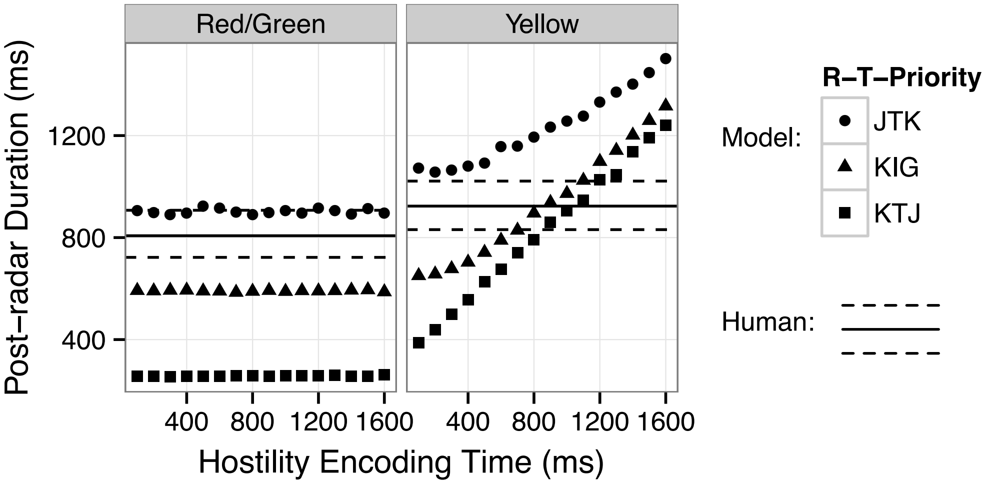


FIGURE 37. The predicted post-radar duration as a function of hostility encoding time, for red/green blips in the left panel, and yellow blips in the right panel. The three plot symbols show the three different R-T-Priority strategies. P10's average post-radar duration is shown as solid horizontal lines, with the 95% CI shown as dashed lines.

Figure 37 shows that the best-fitting strategy varies depending on the HET setting. For example, if HET is set to 200 ms, then the Joystick-Then-Keypad strategy (JTK, the circular plot symbols) fits the data best because its predictions of the post-radar duration are closest to the observed data for blips of all color. If HET is set to 900 ms, however, then the Keypad-If-Green strategy (KIG, triangles) might fit the data best because its predictions almost perfectly match the data for yellow blips, and are close to the data for red/green blips.

The best-fitting R-T-Priority strategy also depends on four other parameters that are not shown in Figure 37: the hostility encoding time for the red/green blips, and the preparation and execution time for manual keystrokes. The first two parameters affect the post-radar duration for the red/green blips, similar to how the HET parameter affects the post-radar duration for the yellow blips. The other two parameters determine how long the classification keystrokes take, and affect blips of all color.

This dependency between strategy and parameter settings poses a problem for fitting models, both in general and in this particular case, because if the parameters are set incorrectly, then the resulting best-fitting strategy might also be incorrect, in effect compensating for the incorrect parameter settings. Among the parameters that affect the post-radar duration, the keystroke execution time was calibrated and thus likely set reasonably correctly for each participant's model. The hostility encoding time for red/green blips and the keystroke preparation time were maintained at their default settings. These default settings seem to be reasonably accurate because, for each participant, one of the three strategy's predictions for red/green blips always falls in the 95% CI of the observed post-radar duration. If these parameters were set incorrectly, then it is likely that none of the three strategies would accurately predict each participant's post-radar duration for red/green blips. The HET is a free parameter, and in this dissertation, I chose to simultaneously determine the best-fitting setting for the HET parameter and the R-T-Priority strategy by finding the combination of parameter and strategy settings that generates the best fit. Given the lack of information to independently nail down either the strategy or the parameter, this approach seems to be the most straightforward way to fit the model. Note, however, that by using the detailed eye movement measures, this intertwinement between strategies and parameters is at least confined to just one strategy dimension. If the model were instead fit to the classification time, then all four strategy dimensions would be convolved with the HET parameter, which would further increase the chance of obtaining wrong conclusions.

Having decided to find the combination of HET setting and R-T-Priority strategy that fits the data best, the next step is to compare the models' goodness-of-fit. Given that the HET parameter is numeric, a line graph can be drawn to show how the goodness-of-fit changes continuously across different HET settings. Figure 38 plots the RMSD and r^2 of the models' predictions of the post-radar durations as a function of the HET setting across the three strategies in the R-T-Priority dimension. The plot permits us to see at what HET setting the different strategies reach the best fits. Again, lower RMSD and higher r^2 indicate better fits. As can be seen, the JTK curve and the KIG curve seem to reach comparably low RMSD values, JTK with an HET of 200 ms, and KIG with an HET of 900 ms. Specifically, the JTK strategy with an HET of 200 ms produces an RMSD of 172 ms ($SD = 36.1$), and the KIG strategy with an HET of 900 ms produces an RMSD of 188 ms ($SD = 15.3$). A paired t-test performed on these two strategy and parameter settings shows that their RMSDs are not significantly different, $t(9.44) = 1.15$, $p = 0.277$. Thus, there is no statistical basis to prefer one over the other. Both should be considered as the best-fitting strategy and parameter configurations for the post-radar duration.

The reader may be alarmed by the relatively low r^2s shown in Figure 38; these low r^2s are likely due to the model's failure to explain one of the two trends in the observed post-radar duration. The two trends can be seen in Figure 15: The post-radar duration increases (a) from red/green blips to yellow blips, and (b) from the peripheral-not-visible condition to the peripheral-visible condition. The model accounts for the first trend, the effect of blip color, as shown in Figure 37. The model does not account for the second

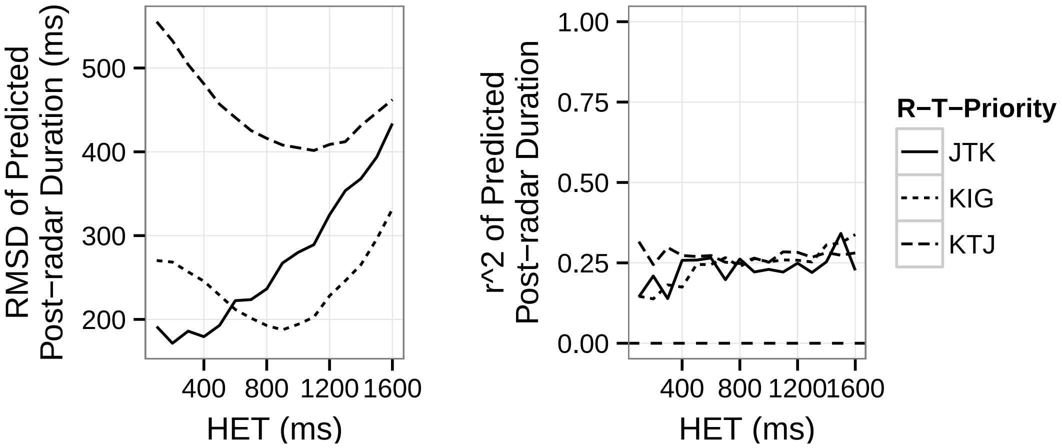


FIGURE 38. RMSD and r^2 of the models' post-radar duration as a function of the HET parameter setting, across the three R-T-Priority strategies. Note the non-zero y axis in the left panel.

trend, the effect of peripheral visibility, as the model predictions do not change across the two peripheral visibility conditions (the effect size of peripheral visibility is only 0.5%). This failure to explain the second trend, however, should not affect the validity of the conclusions about R-T-Priority strategy and the best-fitting HET parameter because the two trends do not interact, $F(1,9) = 0.06, p = 0.81$. In other words, if we add a new component to the model that accounts for the effect of peripheral-visibility, the resulting RMSDs and r^2s of all models would likely all increase by similar values, and the best-fitting R-T-Priority strategy and HET parameter setting would likely remain the same.

The two best-fitting strategy and parameter configurations suggest that P10 was either prioritizing the tracking task at all times (the JTK strategy) or prioritizing the tracking task only when the tracking cursor was not green (the KIG strategy). Either way, it seems that P10 was not prioritizing the classification task (the KTJ strategy) for the transition from classification to tracking. This suggests that in multitasking, some participants may focus more on the immediately perceived task demands, which in this case might be to reduce the tracking error to get the tracking cursor back to green.

It may appear unconventional to have two best-fitting models in a cognitive modeling study, but this is a consequence of adopting a thorough and rigorous approach to evaluating models: When there is not enough evidence to reject an alternative model, each model should be accepted as a plausible explanation for the data. The uncertainty associated with multiple best-fitting models may appear unsettling at first, but this uncertainty should be exposed and discussed, rather than swept under the rug. Only by exposing what is still unknown can we effectively advance cognitive modeling research and cognitive science.

The Best-Fitting Model(s) for P10

The above analyses of the three classification stages revealed the best-fitting strategy for each strategic dimension and the best-fitting hostility encoding time. The best-fitting models for P10 are (a) a strategy combination of TEB-ECS-LBL-JTK with a HET of 200 ms, and (b) a strategy combination of TEB-ECS-LBL-KIG with a HET of 900 ms. Note that based on Figure 38, HET can vary slightly without dramatically changing the model's goodness-of-fit. However, these variant models are not explored because their predictions and the implications with regards to human performance should be very similar to the two best-fitting models in all regards.

If the components within the model are veridical, the two best-fitting models derived from the detailed eye-movement data analyses should also explain the two high-level measures, which are the classification time and the RMS tracking error. Figure 39 compares the two best-fitting models' classification time and RMS tracking error with the human data. The circular plot symbols and the error bars represent the human data, and the clouds represent all models with different strategy and parameter settings. The best-fitting model TEB-ECS-LBL-JTK with HET=200 is indicated by the diamond-shaped plot symbols, and the best-fitting model TEB-ECS-LBL-KIG with HET=900 is indicated by the triangular plot symbols. The graph shows that the two best-fitting models' predictions are very close to the human data in almost all conditions. In particular, in the two peripheral-visible conditions, the predicted classification time match P10's classification time very well, and almost all predictions about the classification time fall within the 95% CI. The only deviation between the predictions and the human data appear in the sound-on peripheral-not-visible condition, in which the model with the KIG strategy (triangles) overestimated the RMS tracking error by about two pixels. In terms of absolute percentage error, however, this deviation is still within 10% range of the observed RMS tracking error, which is within an acceptable range for engineering models. These results show that fitting the detailed measures does indeed lead to models that also reliably capture the aspects of larger-scale human performance.

The two best-fitting models' goodness-of-fit for the two high-level measures and the three detailed measures are listed in Table 6. The goodness-of-fit are calculated in terms of RMSD and r^2. It can be seen that the r^2 is moderately high (above 0.5) for most measures, suggesting that the two models predicted how P10's performance varied across experimental conditions. The r^2 for the post-radar stage is low in both models (though particularly low in the model with the KIG strategy). But as discussed previously, this should not affect the validity of the conclusions about what strategy and parameter settings best explain the data. In addition, the RMSD for the post-radar stage is still within 10% (for the first model) or 20% (for the second model) of observed post-radar duration (on average, 1 second). Because the RMSD can still be used to adjudicate the alternative strategies, and because the misprediction is confined to that single data point, the results of this analysis should still be reliable. Overall, the two best-fitting models

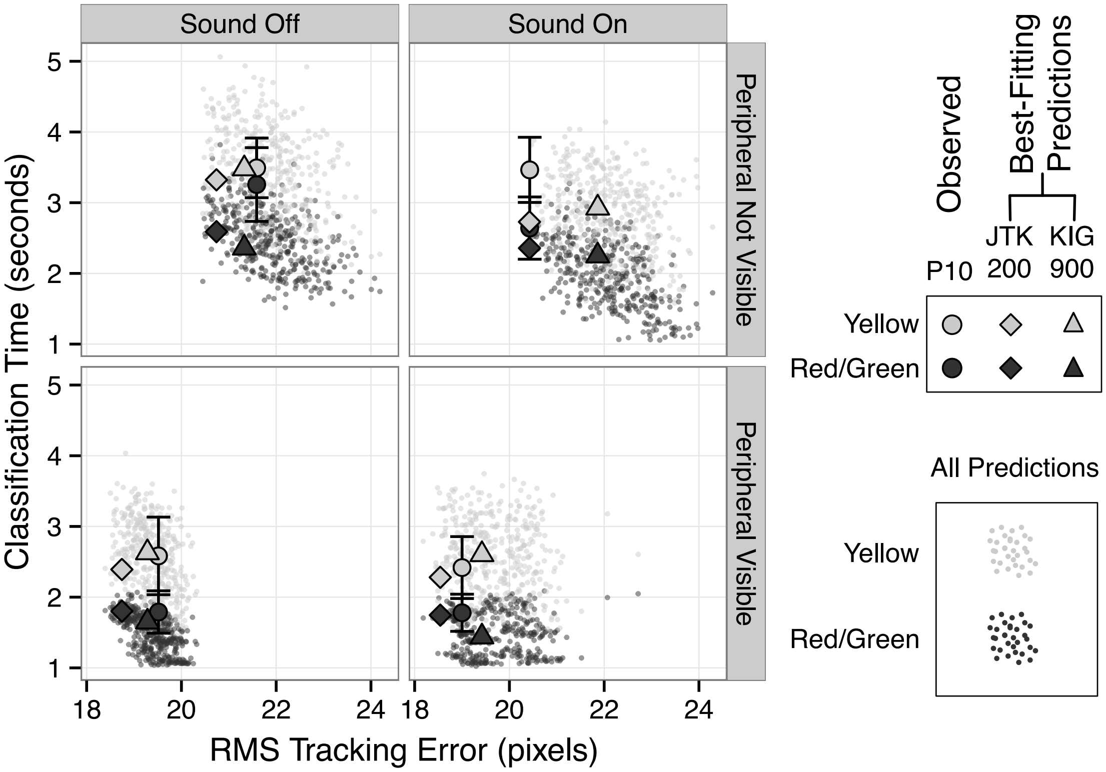


FIGURE 39. Participant P10's average classification time and root-mean-squared (RMS) tracking error across all four experimental conditions, compared against the predictions of all 384 models, including the two best-fitting models. The error bars show the 95% CIs.

seem to account for the participants' data in most measures. These models provide support for the validity of the modeling approaches developed in this dissertation.

TABLE 6. The goodness-of-fit of the two best-fitting models, calculated in RMSD and r^2 for five measures: the three classification stage durations, the total classification time, and the RMS tracking error. Tracking error is measured in pixels, and all others are measured in milliseconds.

Best-Fitting Model	**Measure**	**RMSD (ms)**	r^2
TEB-ECS-LBL-JTK HET=200	Pre-radar	157	0.73
	Blip search	34	0.93
	Post-radar	122	0.43
	Classification Time	380	0.87
	RMS TE (pixels)	0.62	0.89
TEB-ECS-LBL-KIG HET=900	Pre-radar	171	0.57
	Blip search	35	0.90
	Post-radar	209	0.21
	Classification Time	422	0.74
	RMS TE (pixels)	0.77	0.64

Summary of the Approaches to Finding the Best-Fitting Models

This section described the challenges associated with conducting rigorous model evaluations, and presented new methods for overcoming these challenges. This section established that to understand the range of behaviors that exhibited in multitasking scenarios, it may be necessary to model individual performance. This section argued that using detailed measures derived from eye movement data can determine the participants' strategies more reliably than high level measures because the detailed measures provide more stringent constraints on possible explanations of how participants completed each the task. Fitting multiple detailed measures is a challenge in that the different measures may lead to conflicting results about what strategies and parameters best fit the observed data. The section showed how this challenge can be addressed by carefully choosing the measures such that each measure is used to fit a disjoint set of strategies and parameters.

This section presented new methods for addressing the challenges in determining the best-fitting strategies and parameters. The first challenge is to handle the multitude of model predictions that result from a large-scale strategy and parameter space exploration. I showed that by conducting effect-size analyses, the numerous predictions can be effectively compressed and aggregated, leaving only a few data points to analyze; and that by visualizing the model predictions in comparison to the context of the data, the patterns of the predictions can be understood and potential problems with the models may be discovered. The second challenge in determining the best-fitting model is to compare different models to each other and provide statistical evidence for accepting or rejecting competing models. This dissertation shows that such statistical evidence can be provided and that in some cases, multiple best-fitting models may explain the data equally well.

The above model evaluation processes—for deciding what data to fit and for determining the best-fitting strategies and parameters—constitute a contribution to the field of cognitive modeling. The essential requirement of these rigorous processes is the large-scale strategy and parameter exploration, which is another contribution of this research. Without the thorough explorations of the strategies and parameter settings, the model evaluation would not be as complete.

The next section compares the best-fitting strategies and parameters of the top performers to those of the bottom performers. The similarities and differences between the two groups of participants' strategies reveal aspects of how people select strategies for multitasking, and further illustrate the importance of strategy exploration and individualized modeling.

Best-Fitting Models for Each Participant, and Individual Differences

The sequence of analysis discussed in the previous section was applied to all ten participants to find the best-fitting model or models for each individual. Table 7 shows the resulting best-fitting models and their RMSDs across three measures: a sum of the RMSDs of the predictions for the three eye-movement stage durations (EM), the RMSD of the predicted classification time (CT), and the RMSD of the predicted RMS tracking error (TE). Note that P10 and P17 each have two equally good fitting models, with P10 having two plausible best-fitting strategies in the R-T-Priority dimension, and P17 having two plausible best-fitting strategies in the T-R-Priority dimension. The participants in Table 7 are sorted from the best to the worst performer, as determined by the bonus that each participant earned. The ten participants are then divided into two groups, the top performers and the bottom performers. By comparing the strategies of the top performers with those of the bottom performers, it is possible to explore how the choice of strategies may have affected participants' performance.

This section examines these best-fitting models, and discusses the parameter and strategy differences between the top and bottom performers, and the implications of the strategic choices made by the participants.

The Influence of Parameters vs. the Influence of Strategies on Task Performance

Participants' fundamental perceptual, cognitive, and motor characteristics clearly had an impact on task performance. This can be seen, for example, in Table 7 in how the top performers' best-fitting HETs tend to be short, while the bottom performers' HETs tend to be long. To further gauge the influence of the parameters and strategies on performance, an effect-size analysis was conducted with the model's predicted payoff as the dependent variable, and the strategy dimensions and parameters (including the

TABLE 7. The best-fitting models for each participant, and the models' goodness-of-fit (measured in RMSD) across the eye movement data (EM), the classification time (CT), and the RMS tracking error (TE). Note that P17 and P10 have two equally good-fitting model. The definition of the four strategic dimensions are: (a) The T(racking)-R(adar)-Priority dimension controls whether to visually switch to radar immediately after knowing a blip changed color. (b) The T-R-Sound dimension controls whether to visually switch to radar for all sound alerts or only for color-change alerts. (c) The T-R-Location dimension controls where to look at when visually switching from tracking to radar. (d) The R-T-Priority dimension controls what manual task to do after visually switching from radar back to tracking.

Participants		Strategy Dimensions				HET (ms)	RMSD		
		T-R- Priority	T-R- Sound	T-R- Location	R-T- Priority		EM (ms)	CT (ms)	TE (pixels)
Top Performers	P06	IEB	EAS	LBL	KTJ	100	304	189	1.61
	P20	TEB	ECS	LBL	KIG	600	503	601	0.72
	P07	IEB	ECS	LBL	KIG	200	279	469	1.96
	P11	TEB	ECS	LBL	JTK	600	427	528	2.34
	P17	TEB	ECS	LBL	KIG	800	285	425	1.57
		IEB	ECS	LBL	KIG	800	349	441	1.76
Bottom Performers	P10	TEB	ECS	LBL	JTK	200	313	380	0.62
		TEB	ECS	LBL	KIG	900	415	422	0.77
	P09	TEB	ECS	LBL	JTK	1000	474	622	1.57
	P12	IEB	ECS	LBL	KIG	600	212	318	1.26
	P16	TEB	ECS	LBL	JTK	1600	437	679	0.89
	P04	TEB	ECS	LBL	JTK	1100	335	506	1.37

parameters calibrated to each individual participant and the free parameter HET) as the independent variables. The results show that the HET parameter accounted for 8.7% of the variance in payoff, while all other parameters together accounted for 39.7%, and the four strategy dimensions together accounted for 4%. The seemingly small influence of the strategy dimensions is likely due to the fact that the strategy dimensions control only small parts of the classification task, whereas some parameters such as the manual tracking parameters affect almost the entire duration of the experiment. In other words, slight changes in the tracking parameters would cause the payoff to change substantially, whereas the changes in strategies might have big influence on classification, but little on the whole experiment.

That the hostility encoding time accounts for 8.7% of the variance in payoff suggests that the participants' performance could be much improved by changing the dual task interface to better facilitate the hostility encoding of yellow blips. For example, the hostility encoding could potentially be sped up by adding a visual cue around each yellow blip that marks the blip's trajectory, with a longer trajectory line indicating a faster moving speed. This interface change should reduce the time required to encode each yellow blip's direction and speed, and hence improve the dual task performance.

Though the influence of strategies may seem small, it is still important to examine the strategic differences between the top performers and the bottom performers for at least two reasons: First, for some bottom performers, strategies have a large influence, sometimes affecting the payoff by as much as 10%. Second, understanding the influence of strategies may have more practical value than understanding the influence of parameters because it will at least sometimes be easier for people to change their strategies than to change their fundamental information processing capabilities. This point was elegantly demonstrated by Gray et al. (2006), which showed how people can quickly and appropriately adapt their strategies to different task conditions. For these two reasons, the best-fitting strategies of individual participants were carefully examined to see whether the strategies would have likely contributed to the performance difference between the top and bottom performers, and whether bottom performers may have failed to adopt optimal strategies. The remainder of this section focuses on this analysis of strategies.

Participants Chose Optimal Strategies When It Was Clear What Was Optimal.

Participants seemed to adopt optimal strategies when it was clear which strategy would lead to optimal performance. This is evidenced by how all participants appeared to have adopted an Look-priori-Blip-Location (LBL) strategy for the Tracking-Radar-Location (T-R-Location) dimension, and how almost all participants seemed to have adopted an Eyes-to-Color-change-Sounds (ECS) strategy for the T-R-Sound dimension. For the T-R-Location dimension, the LBL strategy is clearly better than the Look-Window-Center strategy, because the LBL strategy uses visual memory to reduce the need for searching for the active blips in the peripheral-not-visible conditions. Thus the LBL strategy shortens the classification time while likely having no detrimental effect on the tracking performance.

For the T-R-Sound dimension, the Eyes-to-Color-change-Sounds (ECS) strategy is clearly more optimal than the Eyes-to-All-Sounds (EAS) strategy, because the ECS strategy can reduce the interruptions to tracking while only slightly delaying the response to the classification task (as needed to listen a little longer to the sound before moving the eyes). An odd exception is that P06, the top performer, seemed to have adopted an EAS strategy. Detailed analysis of the human data suggests that P06 was the only participant who actively monitored black blips and tried to classify blips immediately after they become active without even looking at them again. Thus, P06 was able to take special advantage of the Eyes-to-All-Sounds (EAS) strategy by starting to classify blips immediately after they appeared on the screen. For other participants, however, EAS interrupted tracking and provided little benefit.

These results lend some support to the hypothesis of bounded rationality, which states that humans tend to make decisions that optimize performance or utility, but that peoples' ability to optimize is limited by (a) the information available and (b) the cognitive, perceptual, and motor constraints (Gray et al., 2006; Howes et al., 2009; Simon, 1955). The results presented here provide additional evidence that in developing *a priori* cognitive models, if a certain plausible strategy is clearly optimal, then there is a good chance that it is correct, and it should probably be directly incorporated into the model to predict performance.

Participants Made a Variety of Choices When an Optimal Strategy Was Not Clear.

Participants seemed to adopt different strategies when it was not obvious which of the alternative strategies would lead to overall optimal performance. This is evidenced by the strategies selected for the T-R-Priority and R-T-Priority dimensions, as shown in Table 7. For the T-R-Priority dimension, the Immediate-Eyes-to-Blip (IEB) strategy prioritizes classification, and though this strategy shortens the classification time by responding to blips faster, it can also sometimes leave the tracking task in a money-losing state (when the tracking error is larger than 50 pixels). Thus, the IEB strategy is good for classification, but bad for tracking. The converse is true for the Track-then-Eyes-to-Blip (TEB) strategy. Thus, both strategies trade the performance of one task for the performance of the other, and it is difficult to determine which strategy would lead to a higher overall payoff. Similarly, for the R-T-Priority dimension (which controls what manual task to do after visually switched from radar back to tracking), the three strategies—Keypad-Then-Joystick (KTJ), Keypad-If-Green (KIG), and Joystick-Then-Keypad (JTK)—also trade one task for the other. KTJ prioritizes classification over tracking; JTK prioritizes tracking over classification; and KIG prioritizes classification if, when the model switches back to the tracking task, the tracking error is small, and otherwise it prioritizes tracking. In both strategic dimensions, participants seemed to have adopted different strategies, presumably because in each case there is no clear benefit to any one strategy.

Despite the lack of a strong indication of an overall optimal strategy, the top performers and the bottom performers seemed to converge on different sets of strategies for the T-R-Priority and R-T-Priority dimensions. For the T-R-Priority dimension, three of the five top performers may have adopted the IEB strategy, whereas only one of the bottom performers seemed to use this strategy. For the R-T-Priority dimension, four of the bottom performers seemed to have adopted the JTK strategy, whereas only one of the top performers seemed to use this strategy. More generally speaking, it seems that the top performers were more likely to prioritize the classification task, because the IEB and KTJ strategies, and to some extent the KIG strategy, are more beneficial for classification; whereas the bottom performers were more likely to prioritize the tracking task, because the TEB and JTK strategies are beneficial for tracking. This suggests that perhaps participants made a high-level decision to emphasize one of the two subtask for the entire experiment. It also suggests that prioritizing classification led to better overall performance for this experiment.

As in previous research on multitasking (Howes et al., 2009; Schumacher et al., 2001), this dissertation shows how analysis of individual performance and modeling of individual data can reveal rich insights about multitasking behaviors. In the work presented here, the individualized data analysis and individualized modeling helped to identify the strategic differences between the top and bottom performers. Similar practices should perhaps be carried on in future studies on multitasking.

Prioritizing Classification Would Lead to Optimal Payoff for All Participants.

Individualized modeling reveals the different strategic choices among participants, but figuring out which strategy is globally optimal sometimes requires additional analysis of the model output. The fact that most top performers seemed to prioritize classification suggests that this might be the optimal strategy for all participants to pursue. However, it is possible that, given the specific perceptual, motor, and cognitive characteristics of the bottom performers, prioritizing classification may lead to sub-optimal global performance for these participants. In other words, there may not be a "one-size-fits-all" globally optimal strategy. Figuring out which of the two hypotheses—(H1) there is a single optimal strategy for all participants, or (H2) different participants will have different optimal strategies—is correct would further an understanding of multitasking behaviors and suggest new approaches to developing *a priori* cognitive models for predicting multitasking performance.

One way to adjudicate on the above two competing hypotheses is to examine the payoffs earned by the models. Because the models are already individualized with each participant's parameters, the models should reflect how the participants may perform under different strategic choices. If it can be shown for all participants that one strategy leads to a better payoff than all other alternative strategies, then perhaps the bottom performers could have improved their performance by adopting that optimal strategy. For this payoff-based analysis, only the R-T-Priority strategies need to be explored because (a) for the T-R-Sound and T-R-Location dimensions, almost all participants used the same strategies, and (b) for the T-R-Priority dimension, the two alternative strategies predict very similar payoffs.

As part of exploring the payoffs of the different models, Figure 40 was created to show how the predicted payoffs varied across different R-T-Priority strategies.

Again, for this payoff-based analysis, only the R-T-Priority strategy is varied; the other strategy dimensions and the HET parameter were set to the best-fitting configurations. In Figure 40, the filled plot symbols indicate the best-fitting strategy in the R-T-Priority dimension. As can be seen, the Keypad-Then-Joystick (KTJ) strategy, which prioritizes classification, earns the highest payoff consistently across all individual participants, and the Joystick-Then-Keypad (JTK) strategy, which prioritizes tracking, earns the lowest payoff across all individuals. These results suggest that prioritizing classification would indeed be a good strategy for all participants.

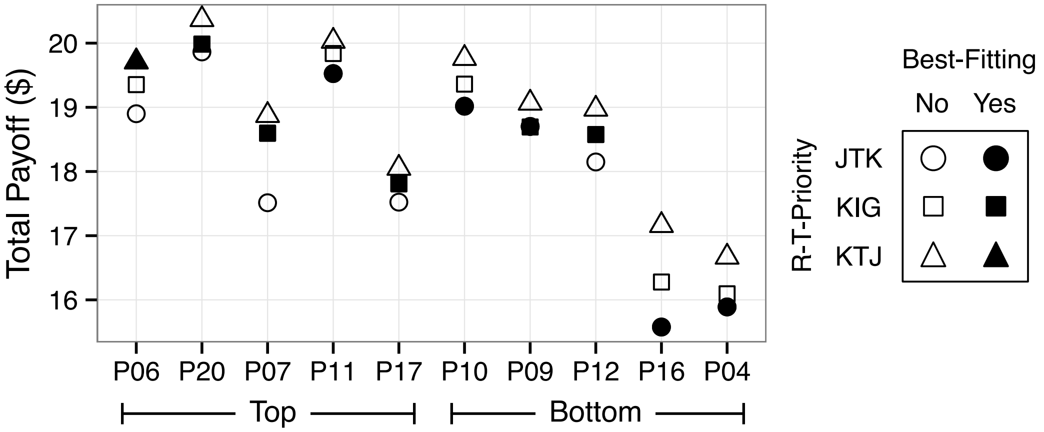


FIGURE 40. The total payoff predicted by models with the three different R-T-Priority strategies across all individual models. Filled plotting symbols represent the best-fitting strategies.

Figure 40 also shows that perhaps the bottom performers could improve their payoff by changing their strategies. For the bottom performers, from among the three R-T-Priority strategies, the best-fitting strategy tends to generate the lowest payoff. It seems that if the bottom performers choose the KTJ strategy, they could increase their payoff by as much as $1.5, which would amount to about 10% in pay for some of the bottom performers.

That the task performance is substantially influenced by strategies has strong implications for personnel training and user interface design for multitasking scenarios. One implication is that perhaps people can be better taught about efficient strategies. Too often, poor performance in some professions (such as some military positions) is attributed to a person's intrinsic abilities (or lack thereof) that cannot be easily improved. The results presented here suggest that strategy may also play an important role and perhaps teaching people efficient strategies can improve their performance. Another implication is that perhaps user interfaces intended for multitasking could be designed to motivate optimal strategies. The results presented here suggest that a possible reason that people adopt inferior strategies might be that there is no clear indication of what strategy could be optimal. Perhaps adding some kind of visual or auditory feedback to the user interface that enables people to monitor how strategic choices influence the overall performance could help people form optimal integrated task strategies.

Computational cognitive modeling played an indispensable role in this payoff-based analysis. Only by developing computational models that accurately represent each individual's cognitive, perceptual, and motor characteristics could people's performance under different strategic choices or task environments be inferred. This is one of many powerful abilities of computational cognitive modeling that permits it to contribute substantially to theory development.

Participants May Have Optimized Performance Locally.

One important implication of the above results is that applying the cognitive bounded rational (CBR) analysis would not lead to correct models for this dual task. As discussed earlier, the CBR analysis assumes that, given enough practice, participants will find optimal strategies to achieve maximum payoff. But as the payoff-based analysis shows, the bottom performers clearly did not choose optimal strategies. Following the CBR analysis would not lead to this finding. Rather, because the optimal strategy would not fit the observed data, the CBR analysis would suggest that participants had different interpretations of the payoff functions or that the cognitive architecture is incorrect, and therefore either the payoff function needs to be tuned or the architectural assumptions reexamined. However, continuing down this path would increase the model's degrees of freedom (due to the tuning of the payoff function), contrary to the original goal of the CBR analysis.

One possible reason why the bottom performers did not use the optimal strategy is that the participants focused on the moment-to-moment performance and chose strategies that optimized locally. For example, there is an immediate, salient visual feedback associated with the tracking task: the tracking cursor turns green when the tracking error is small. This visual feedback may have propelled some participants to first focus on the tracking task after moving the eyes back to the tracking display.

If the above hypothesis is true, then a viable approach to building *a priori* models would include an exploration of locally optimal strategies. Such an approach, to some degree, extends the CBR analysis to encompass more plausible behaviors, and may be a promising future research direction to explore.

Summary

This section showed that through individualized modeling and an extensive strategy and parameter exploration, the diverse multitasking behaviors across individual participants can be revealed. By examining the similarities and differences between the strategies of the top performers and those of the bottom performers, we now have a better understanding about when people would or would not adopt an optimal strategy.

The results also suggest new approaches to developing predictive models, such as by exploring locally-optimizing strategies. This section also showed that cognitive modeling, particularly individualized modeling, is indispensable for gaining insights on human behaviors and for theory development. The next section concludes the thesis, summarizes the findings of this modeling research, and suggests future research directions for multitasking and cognitive modeling.

CHAPTER VI

CONCLUSION

Through an eye tracking experiment and large-scale cognitive modeling, this research investigated human-computer interaction phenomena in a time-pressured multimodal multitasking environment. The computational model developed in this dissertation explains many detailed aspects of human behavior observed in the dual task experiment, and provides new insights into general human multitasking performance. In developing and evaluating the cognitive models, this research also proposed new methods to address several challenges involved in conducting a comprehensive exploration of the strategy and parameter space. These new methods potentially pave the road for building *a priori* models of human performance. This chapter summarizes these contributions and discusses the implications for designing user interfaces that better support multitasking, as well as the implications for future research on multitasking and cognitive modeling.

A Computational Theory of Multitasking

The computational theory of multitasking outlined in this research consists of both the invariable psychological factors and the multitasking strategies. For invariable factors, the models presented in this dissertation largely inherited the implementations from the EPIC architecture, including the processing capabilities of the extra-foveal vision, the parallel execution of multiple production rules, and the motor programming framework. These computational implementations are the important components of the model presented here, and the model's success in explaining the data helps to promote the EPIC architecture as a viable theoretical framework for exploring and explaining somewhat complex interleaved dual task performance.

Besides exploring the invariable factors underlying multitasking performance, a bigger focus of this research was to determine general patterns of multitasking task strategies and possible variations of strategies. Following Meyer and Kieras (1997a), in this research, multitasking strategies are divided into task processes and executive processes, with the executive processes assuming the responsibility of coordinating the task processes. The executive processes are then further divided into independent groups, each of which either manages access to a motor processor (in this case, the motor processor is either ocular or manual) or manages the transition of the processor from one task to the other. This division of labor within the executive processes is consistent with Kieras et al.'s (2000) exploration of general executives. A unique contribution of this dissertation is to provide, within this general approach for organizing cognitive strategies, a systematic approach for exploring variations in the executive processes and task strategies. This exploration is achieved by first identifying the parts of the executive processes that can be accomplished by several alternative strategies, and then semi-automatically generating all of the different configurations of the strategies. These variations of strategies typically alter how the executive processes prioritize tasks or respond to immediate task demands. This large-scale exploration of the strategy space is enabled by a computer-cluster-based parallelized cognitive modeling framework developed as part of this dissertation work. This new approach to exploring multitasking strategies can now be applied to other multitasking scenarios for understanding and predicting performance.

Insights into Multitasking Performance

This dissertation provides theoretical and empirical insights into the human information processing involved in multitasking.

Both the empirical data and the modeling results show that the choice of perceptual modality for delivering task information affects performance. The empirical data show that delivering information via the visual channel has clear advantages over delivering information via the auditory channel because (a) people seem to respond more promptly to visual changes than to auditory alerts and (b) a visual presentation can convey the locality of an event more precisely than an auditory presentation. The first observation may seem incorrect at first glance, because previous psychological studies found responses to sound to be faster than responses to light (see Welford & Brebner, 1980). A closer examination based on the modeling work presented here suggests that this is not a contradiction to previous research, but rather a result of participants treating the information from the two modalities differently. Specifically, the modeling results suggest that participants responded to visual signals as performing a simple reaction task, in which little or no cognitive decision was involved, whereas participants responded to sound as performing a choice reaction task, in which they spent time deciding the alert (color change vs. appearance). Perhaps because of this difference, in our experiment, responses to sound were slower than responses to visual changes.

The modeling results suggest that, in multitasking scenarios, identifying the circumstances in which tasks interfere with each other is more challenging than what the multiple resource theory suggests. The multiple resource theory predicts that there will be strong interference between tasks that share the same resources. This research, however, shows how even when two tasks employ the same visual and manual processes, people can find opportunities to overlap the processes across two tasks. The main empirical evidence of such overlapping in this research is that the participants almost always keyed-in the classification responses while looking at the tracking screen. In addition, the modeling results suggest that the participants selected classification responses while doing manual tracking. This evidence suggests that when the perceptual and response stages of a task are decoupled, as is done in the classification task, multitasking may occur. In other words, one condition for efficient multitasking is to not require people to constantly monitor their actions while carrying them out.

This dissertation provides new evidence for a claim that has been demonstrated previously by a few studies (e.g. Gray & Boehm-Davis, 2000; Kieras et al., 2000), which is that strategies can substantially influence task performance. New evidence presented here relates to (a) how the strategic difference between the top and bottom performers seemed to contribute to the difference in task performance and (b) how the predicted payoff of one strategy, Keypad-Then-Joystick, is sometimes 10% larger than that of the Joystick-Then-Keypad strategy (see Figure 40). In other words, it appears that by merely changing the task-switching strategy for the brief post-radar period, the participant's performance can change by 10%. In some multitasking scenarios, such effects could be critical and even life-saving.

Implications for Designing User Interfaces to Support Multitasking

Based on the above insights on multitasking performance, the following four guidelines can be offered for designing user interfaces to support multitasking.

1. Incorporate into each subtask visual cues about the critical events of other subtasks. Based on the model predictions, auditory cues alone may not help a user to quickly differentiate peripheral events, but visual cues for these events might convey critical information more efficiently. For example, for the dual task interface used in this experiment, a visual cue could perhaps be added to the tracking display to notify the participant of any blip-color-change events. This visual cue could be as simple as an icon that appears on the left side of the tracking display when a blip changes color, or as sophisticated as some sort of miniature classification display that is always visible in a corner of the tracking display. The first design would likely help the participant, at least when the classification display is not peripherally visible, to more quickly respond to color changes; the second design might also help the participant to more quickly locate the active blip.

The design guideline could be applied to many real-world multitasking scenarios. For example, many contemporary automobiles alert the driver to dangerous situations such as the vehicle drifting out of the lane or coming too close to a vehicle in front. Many empirical studies (e.g., Campbell, Richard, Brown, & McCallum, 2007; Ho, Reed, & Spence, 2007; Lee, Hoffman, & Hayes, 2004) were conducted to determine the effectiveness of the warning cues delivered via different modalities including the visual, auditory, and tactile modalities, and at least one study (Scott & Gray, 2008) found that drivers respond faster to auditory collision warnings than visual warnings. The modeling results presented in this dissertation, however, indicate that when there are many auditory cues, participants might respond more slowly to auditory cues than to visual cues because of the need to comprehend the auditory alerts. Thus, this dissertation suggests that a good design for such warning systems might be to use salient visual changes on a head-up display, in addition to auditory cues, to help the driver to detect and respond to the emergency more quickly.

2. Enable the user to make responses without constantly monitoring their actions. As discussed earlier, both the empirical data and the modeling results showed that the participants in the dual task experiment overlapped parts of the tracking task with parts of the classification task, and that this overlapping was possible because the classification responses could be made without visual monitoring. One implication of this is that touchscreen interfaces are not likely to support optimal multitasking performance because they require constant visual monitoring. Systems that will be used in multitasking scenarios, such as in-car entertainment systems, might be better off using physical controls rather than touchscreens.

3. To maximize subtask overlapping, use different perceptual and response modalities for each subtask. As suggested by the modeling work presented here and by Schumacher et al.'s (2001) study, dual-task interference can be reduced or removed when two tasks do not share the same modality. Thus, for the dual task presented here, the tracking task might be less interrupted if the classification responses were made vocally as opposed to manually, or if blip information could be fully acquired via sound. Such designs will not always lead to more efficient performance in competing subtasks because some information such as route information is better presented visually than auditorily, and today's error-prone speech recognition systems might cause distractions (Strayer, Turrill, Coleman, Ortiz, & Cooper, 2014). Cognitive modeling could be applied to evaluate alternative designs to determine whether the benefits of overlapping the tasks outweighs the loss of single-task efficiency.

4. Design user interfaces to motivate optimal strategies. That strategies can have substantial impact on multitasking performance suggests that multitasking user interfaces should be designed with a consideration of a variety of possible task strategies, and to motivate optimal strategies. For example, the dual task interface could incorporate visual cues, such as circles around active blips, to motivate participants to prioritize the classification task. Similarly, for driving, salient visual cues could be displayed when the car is drifting away from the center of the lane, in order to motivate the optimal strategy which, in this case, is likely to focus on driving.

Applying the above four guidelines should help improve user interfaces meant to support multitasking.

Future Research Directions for Multitasking

The results of this dissertation suggest several research directions that could be pursued to further deepen an understanding of human multitasking. One direction would be to examine the effects of learning, and there would be at least two ways to do this. One would be to teach the participants the optimal strategy (in this task to prioritize classification, and in driving tasks to prioritize driving) and see if the bottom performers can adopt the optimal strategy and improve their payoff. If they do adopt the optimal strategy and their performance indeed improves, then it would provide further evidence of the influence of task strategies, and further support the hypothesis that strategies can be taught to improve performance.

Another way to study the effects of learning would be to give participants more time on task and see if the bottom performers change their strategies as they gain more practice. If they do, it might suggest that, with sufficient practice, people can discover that they are only optimizing locally, and learn on their own how to optimize globally. If they do not change strategies, however, this might suggest that it is at least sometimes difficult for a person to notice their own inefficiencies and improve their strategies. It may be that, at least for some perceptual-motor tasks, people have a natural tendency to optimize locally.

Another future research direction would be to adapt the model developed here to more complex, real-world tasks such as driving. Though the model developed in this dissertation cannot be directly applied to driving, the way the strategies are explored could be reused to model multitasking scenarios that involve driving. Recently, some studies (e.g., Salvucci, 2006, 2009) modeled multitasking in a driving environment, but they did not explored many alternative strategies. Moreover, at least some of the empirical experiments do not gather eye movement data and hence may not have provided enough data for highly-constrained model validation. More eye tracking studies are needed in the field of driving research to understand the detailed visual interactions involved in driving, and to build driving models that make *a priori* predictions of multitasking performance.

Contributions to Modeling Methodologies

Besides providing insights into multitasking, this dissertation also addressed several challenges involved in the process of developing and validating cognitive models. This section discusses these contributions.

Use fine-frained data to more accurately find the best-fitting models. One of the challenges in developing cognitive models is to find the best-fitting models, and this research presents a new way to use fine-grained data to achieve this goal. The approach presented here includes five steps: (1) Identify the critical events that capture participants' typical behavior in the task. (2) Derive detailed measures of human performance from these critical events. (3) Develop and run the models. (4) Determine which strategy dimensions and parameters likely contribute to the model's prediction for each measure. This can be done either through a qualitative analysis of the model's behavior or a sensitivity analysis of the model's predictions. (5) Based on the results of Step 4, for each measure, aggregate the model predictions across the strategy dimensions and parameters that do not affect the predictions, and then fit the model to the observed data. This five-step approach should help researchers to use fine-grained data to arrive at reliable estimates of strategy and parameter settings.

Develop individualized models for detailed insights into strategy variations across individuals. The modeling work presented in this dissertation illustrates the benefits of building models for individual participants. Specifically, individualized modeling revealed how strategies influenced performance, and how strategies contributed to the differences between the top and the bottom performers. For other modeling studies, individualized modeling might have similar benefits. Though individualized modeling requires more effort than the traditional practice of modeling the average performance, it can offer unique insights into human behavior and sometimes the additional effort is warranted.

Conduct comprehensive parameter and strategy explorations with large-scale modeling and sensitivity analyses. An important contribution of this research is the framework that was developed for the large-scale exploration of cognitive task strategies and for the analysis of the massive data that results from such exploration. As discussed previously, to account for a variety of plausible strategies and to find the most accurate model of the observed behavior, it is important to thoroughly explore strategies and parameter settings. But, until now, two difficulties hindered such exploration: (a) It was tedious to program many different strategies and (b) it was computationally expensive to run many different models. The parallelized cognitive modeling (PCM) system developed in this research addresses both challenges. The first challenge was addressed by building a model spawner that automatically applies predefined changes to a template model to implement different strategies. The spawner eliminates the need to manually program the myriad possible combinations of strategies, and makes it relatively easy to systemically explore strategies. The second challenge was addressed by developing a cluster-based computing architecture that automatically distributes the thousands of different models to hundreds of CPUs, reducing the running time by more than a hundredfold. This dissertation presents one of the first modeling studies to adapt a cognitive architecture to a computer cluster, and perhaps the very first to use a cluster to explore strategies. As cognitive modeling research moves to more complex tasks and adopts more computationally expensive methods, parallelized cognitive modeling will become more important and more prevalent.

A PCM system enables large-scale modeling, but also produces an enormous amount of data. The sensitivity analysis techniques developed in this research address this problem and make it easier to deal with large sets of model predictions. This approach first uses effect-size analysis to extract from the various factors the few that affect a target measure, and then uses visualizations and statistical tests to find out which strategy and parameter configurations produce the best fit. Without such a process, it may be impossible to examine the influence of the strategies as the influence is buried within thousands of different models. Thus, as modeling research moves to large-scale exploration, better analysis techniques are needed to parse the myriad model predictions, and the approach shown in this dissertation provides an example of such a technique.

Select strategies that optimize locally. This research sheds light on how to constrain the variability of strategies and sharpen the predictive power of cognitive modeling. As discussed previously, cognitive task strategies are very flexible and are sometimes explored as free parameters to fit behavioral data. This flexibility makes it difficult to validate a model's architectural and strategy assumptions. Addressing this challenge requires constraining task strategies only to those that are likely to actually be used by people. To this end, Howes et al. (2009) proposed the cognitive bounded rational (CBR) analysis, which reduces the strategy space by assuming that people only choose strategies that maximize utility. This dissertation shows, however, that this assumption may not always hold because it seems the participants in the dual task experiment did not always select the strategies with the highest utility, or at least with the highest monetary payoff.

The results of this dissertation seem to be more in line with Gray et al.'s (2006) soft constraints hypothesis, which asserts that people optimize interactive behavior at a time scale of 300 ms to 3 seconds, and that such local optimization does not necessarily lead to global optimization. In our experiment, for the parts of the task in which there is a clear optimal strategy, participants appear to use it. This suggests that humans tend to be rational. However, for other parts of the task in which different strategies seem to optimize the performance of different subtasks, participants did not seem to appreciate how their local strategies might impact their global performance, and often chose what could be characterized as a greedy, or locally-optimizing, strategy. That is, they appeared to optimize based on immediate task feedback. For example, a possible reason that many participants chose to prioritize tracking after moving their eyes back to the tracking display might have been related to the simple control-feedback satisfaction of moving the joystick to see the tracking cursor turn green. This may have given them a bigger mental payoff than the financial rewards associated with the experiment. This suggests that, to build accurate predictive cognitive models, researchers need to explore many strategies and give special consideration to those that achieve local maxima. This approach of selecting locally optimal strategies could reduce the strategy space and still permit accurate predictions of suboptimal performance, as was done here.

A Methodological Framework for Model Fitting and Predictive Modeling

Overall, the above approaches constitute a methodological framework for cognitive modeling that can be carried forward for future explanatory and predictive cognitive models. The previous subsections illustrated what this methodological framework offers for model fitting. For *a priori* predictive cognitive modeling, such as for engineering purposes, this framework could also be useful. In particular, the parallelized cognitive modeling (PCM) system could facilitate the predictions of the range of performance seen in experts versus novices. Predicting a range of the possible performance, also referred to as bracketing (Kieras et al., 2001), is needed for *a priori* modeling because it is often difficult to anticipate exactly which strategies people will use (see Kieras et al., 2001, 1997). The PCM system could make it easier to explore strategies that may be adopted. The resulting comprehensive prediction space might be more valuable for evaluating a user interface design than the smaller prediction space that would result from exploring only a few strategies.

The PCM system could also be incorporated into existing design-oriented, predictive cognitive modeling tools such as CogTool (John, Prevas, Salvucci, & Koedinger, 2004) or GLEAN (Kieras, Wood, Abotel, & Hornof, 1995) to enhance the comprehensiveness of the predictions made by these tools. These tools provide a graphical user interface or a simplified programming language that allows a designer to easily specify a sequence of interactions (such as button pressing and mouse clicking) needed to complete a task and, as such, are easier to use than cognitive architectures for evaluating user interface designs for conventional desktop or mobile platforms. If the PCM system were incorporated into these tools such that alternative strategies could be specified for each step of a task or for each interaction procedure, these modeling tools could potentially provide predictions for all plausible strategies, and inform the designer of the entire range of possible performance, potentially making these tools more valuable for evaluating user interface designs.

Future Research Directions for Cognitive Modeling Methodologies

Though this dissertation addresses many issues that arise in developing and evaluating cognitive models, there are still many outstanding research questions.

One future research direction would be to develop goodness-of-fit measures that are independent of scale. Many goodness-of-fit measures, such as the root mean squared deviation (RMSD) and the mean absolute deviation, are dependent on the scale of the observed data. For example, if the data are in milliseconds, then the RMSD is also in milliseconds. This poses a problem because RMSDs measured for different types of data, such as classification time and tracking error, will have different scales and thus cannot be easily compared or combined. Scale-independent measures could enable comparisons across models that are fitted to different types of data and different measures (such as reaction time and eye movements), or even models that are developed for different tasks. Scale-independent goodness-of-fit measures could also help combine multiple dependent variables (DVs) that measure task performance into a single DV (such as by summing the goodness-of-fit values across DVs). The resulting single DV could then serve as an overall measure of how well the model explains the observed task performance. One commonly used scale-independent measure is r^2, but other scale-independent measures are needed to characterize how well a model explains absolute positions rather than just the trends of the observed data. Schunn and Wallach (2005) offered some recommendations for scale-independent measures. One such measure, for example, scales down RMSD by the standard error of the participant mean. However, more evidence and evaluation is needed to show how new goodness-of-fit measures can integrate across different measures.

Another future research direction is to develop methods to address the situation in which the best-fitting strategy varies depending on the parameter setting. In this dissertation, this situation was seen in how the best-fitting Radar-to-Tracking-Priority strategy changed depending on the setting of the hostility encoding time. When such situations occur, it might be difficult to determine which strategy and parameter configuration is correct. In this dissertation, this problem was addressed by, in effect, giving equal weight to the free parameter and to the strategy dimensions. Other approaches might put more weight on the parameters and report the best-fitting strategy for different ranges of parameter settings. Such approaches, however, would likely complicate analyses because they could produce many different best-fitting models. Future research could explore methods to narrow the set of the best-fitting configurations, such as by setting a threshold on the goodness-of-fit and only considering models above the threshold.

In addition to the above research directions regarding model evaluation, this dissertation also suggests new directions for developing *a priori* predictive models. One direction is to further automate and expand the exploration of task strategies. The model spawner included in the parallel cognitive modeling system partially automates the generation of task strategies. The analyst still needs to provide a basic model template and instructions for how to modify the template to implement each strategy. A fully automated system would automatically identify parts of the task that can be completed with different strategies and automatically implement the alternative strategies. Such a modeling system should be possible to implement for common human-computer interaction paradigms such as desktop computing because the alternative strategies for completing common interaction routines such as filling out a form and opening a file can be pre-defined and reused. Such fully automated exploration could be incorporated into existing modeling tools to enable more comprehensive predictions of task performance.

While automating strategy exploration is important, finding a way to guide the exploration towards strategies that would likely be adopted by users would make cognitive modeling more useful for evaluating user interface designs. This dissertation suggests that one useful heuristic for strategy exploration is to only examine the strategies that optimize performance locally. This hypothesis, however, still requires further examination, and one way to examine it would be to conduct variants of the dual task experiment with different payoffs and feedback. One variant could be to change the payoff to put more weight on one task or the other, and to see if participants change their priorities accordingly. Other experiments could include “payoff traps” that lead participants to local maxima that prevent maximum overall payoff. These experimental designs could help reveal the extent to which people are sensitive to payoff incentives and feedback, and when and how people optimize locally. This could contribute to *a priori* predictive models that simulate the strategic optimization that is most appropriate for a given task.

This dissertation work suggests another way to use cognitive modeling to guide user interface design: Critical parts of task processes that have a large influence on the overall task performance could be identified by exploring parameter settings. Understanding the influence of different parameters could help a designer to decide which parts of the user interface to optimize. For example, if it can be determined that the text encoding time is critical for the overall task performance, then the user interface might use large fonts to reduce text encoding time (Beymer, Russell, & Orton, 2008). This use of parameter exploration was demonstrated in this dissertation with the exploration of the hostility encoding time (HET) parameter. Because it was determined that the HET parameter impacted task payoff, a few user interface improvements were proposed for reducing the hostility encoding time. Future research could expand such parameter exploration to many parameters, including parameters that have default architectural settings, or parameters that could be calibrated from the data. These parameter explorations could reveal which parameters most affect the performance, and the designer could spend more effort to improve the corresponding parts of the user interface. Though such large-scaled parameter exploration would likely demand even more computational power than what was needed for this dissertation, for life-critical real-world tasks such as driving, such exploration might be warranted.

All of the above research directions can be explored in future studies to further advance cognitive modeling methodologies.

Concluding Remarks

Designing user interfaces to support efficient multitasking requires a rich understanding of the capabilities and limitations of human multitasking, as well as a consideration of the detailed human-machine interactions that might unfold in a multitasking environment. This dissertation demonstrates that such detailed interactions might be best understood with highly parallelized strategy exploration in the context of individualized computational cognitive modeling. By running and analyzing 3,840 models on a cluster-based parallelized cognitive modeling system, it is shown that the difference between the top and bottom performers appears to partly result from their skill at orchestrating a symphony of strategy selection, coordination, and execution. The effect of strategy selection on overall task performance suggests that multitasking is not accomplished by simply combining subtask processes but rather with a controlled process that involves many micro-decisions which together determine overall task performance. These new understandings about multitasking, as well as the tools and models developed in this research, ultimately advance and benefit human-computer interaction research.

REFERENCES CITED

Altmann, E. M. (2007). Control signals and goal-directed behavior. In W. D. Gray (Ed.), *Integrated models of cognitive systems* (pp. 380–387). Oxford University Press.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Lawrence Erlbaum Associates.

Anderson, J. R., Taatgen, N. A., & Byrne, M. D. (2005). Learning to achieve perfect time sharing: Architectural implications of hazeltine, teague, and ivry (2002). *Journal of Experimental Psychology*, *31*(4), 749–761.

Ballas, J. A., Heitmeyer, C. L., & Pérez-Quiñones, M. A. (1992). Evaluating two aspects of direct manipulation in advanced cockpits. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 127–134). New York, NY: ACM.

Beymer, D., Russell, D., & Orton, P. (2008). An eye tracking study of how font size and type influence online reading. In *Proceedings of the 22nd british hci group annual conference on people and computers: Culture, creativity, interaction - volume 2* (pp. 15–18). Swinton, UK, UK: British Computer Society.

Broadbent, D. E. (1958). *Perception and communication*. New York, NY: Oxford University Press.

Brooks, L. R. (1968). Spatial and verbal components of the act of recall. *Canadian Journal of Psychology*, *22*(5), 349–368.

Brumby, D. P., Salvucci, D. D., & Howes, A. (2009). Focus on driving: How cognitive constraints shape the adaptation of strategy when dialing while driving. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 1629–1638). doi: 10.1145/1518701.1518950

Byrne, M. D., & Anderson, J. R. (2001). Serial modules in parallel: The psychological refractory period and perfect time-sharing. *Psychological Review*, *108*(4), 847–869.

Campbell, J. L., Richard, C. M., Brown, J. L., & McCallum, M. (2007). *Crash warning system interfaces: human factors insights and lessons learned* (Tech. Rep. No. DOT HS 810 697). US Department of Transportation, National Highway Traffic Safety Administration.

Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a crt. *Ergonomics*, *21*(8), 601–613.

Cherry, E. C. (1953). Some experiments on the recognition of speech, with one and with two ears. *Journal of the Acoustical Society of America*, *25*(5), 975–979.

De Jong, R. (1993). Multiple bottlenecks in overlapping task performance. *Journal of Experimental Psychology: Human Perception and Performance*, *19*(5), 965–980.

Deutsch, J. A., & Deutsch, D. (1963). Attention: Some theoretical considerations. *Psychological review*, *70*(1), 80–90.

Duchowski, A. T. (2007). *Eye tracking methodology: Theory and practice* (2nd ed.). New York: Springer.

Findlay, J. M., & Gilchrist, I. D. (2003). *Active vision: The psychology of looking and seeing*. New York, NY: Oxford University Press.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, *47*(6), 381–391.

Fleetwood, M., & Byrne, M. (2006, May). Modeling the visual search of displays: A revised ACT-R model of icon search based on eye-tracking data. *Human–Computer Interaction*, *21*(2), 153–197.

Foyle, D. C., & Hooey, B. L. (2008). *Human performance modeling in aviation*. CRC Press.

Fu, W.-T., & Gray, W. D. (2006). Suboptimal tradeoffs in information seeking. *Cognitive Psychology*, *52*(3), 195–242.

Gluck, K. A., Stanley, C. T., Moore, L. R., Reitter, D., & Halbrügge, M. (2010). Exploration for understanding in cognitive modeling. *Journal of Artificial General Intelligence*, *2*(2), 88–107.

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

Gray, W. D., Sims, C. R., Fu, W.-T., & Schoelles, M. J. (2006). The soft constraints hypothesis: A rational analysis approach to resource allocation for interactive behavior. *Psychological Review*, *113*(3), 461–482. doi: 10.1037/0033-295X.113.3.461

Hamming, R. R. (2003). *Art of doing science and engineering: Learning to learn*. Taylor & Francis.

Ho, C., Reed, N., & Spence, C. (2007, Dec). Multisensory in-car warning signals for collision avoidance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *49*(6), 1107–1114. Retrieved from `http://dx.doi.org/10.1518/001872007X249965` doi: 10.1518/001872007x249965

Hornof, A. J. (2001, September). Visual search and mouse-pointing in labeled versus unlabeled two-dimensional visual hierarchies. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *8*(3), 171–197. doi: 10.1145/502907.502908

Hornof, A. J., & Halverson, T. (2002). Cleaning up systematic error in eye-tracking data by using required fixation locations. *Behavior Research Methods*, *34*, 592–604. Retrieved from `http://dx.doi.org/10.3758/BF03195487` (10.3758/BF03195487)

Hornof, A. J., & Halverson, T. (2003). Cognitive strategies and eye movements for searching hierarchical computer displays. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 249–256).

Hornof, A. J., Halverson, T., Isaacson, A., & Brown, E. (2008). Transforming object locations on a 2d visual display into cued locations in 3d auditory space. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *52*(18), 1170–1174. doi: 10.1177/154193120805201804

Hornof, A. J., & Kieras, D. E. (1997). Cognitive modeling reveals menu search in both random and systematic. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 107–114). New York, NY, USA: ACM. doi: 10.1145/258549.258621

Hornof, A. J., & Zhang, Y. (2010). Task-constrained interleaving of perceptual and motor processes in a time-critical dual task as revealed through eye tracking. In D. D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the 10th international conference on cognitive modeling* (pp. 97–102). Philadelphia, PA: Drexel University.

Hornof, A. J., Zhang, Y., & Halverson, T. (2010). Knowing where and when to look in a time-critical multimodal dual task. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 2103–2112). doi: 10.1145/1753326.1753647

Howes, A., Lewis, R. L., & Vera, A. (2009). Rational adaptation under task and processing constraints: Implications for testing theories of cognition and action. *Psychological Review*, *116*(4), 717–751. doi: 10.1037/a0017187

John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 455–462). New York, NY, USA: ACM. doi: 10.1145/985692.985750

Kaber, D. B., & Kim, S.-H. (2011). Understanding cognitive strategy with adaptive automation in dual-task performance using computational cognitive models. *Journal of Cognitive Engineering and Decision Making*, *5*(3), 309–331.

Kandel, E., Schwartz, J., & Jessell, T. (2000). *Principles of neural science* (4th ed.). McGraw-Hill Medical.

Karsh, R., & Breitenbach, F. W. (1983). Looking at looking: The amorphous fixation measure. In R. G. Groner, C. Menz, D. F. Fisher, & R. A. Monty (Eds.), *Eye movements and psychological functions: International views* (pp. 53–64). L. Erlbaum Associates.

Keele, S. W. (1973). *Attention and human performance*. Pacific Palisades, CA: Goodyear.

Kieras, D. E. (2009). Why EPIC was wrong about motor feature programming. In A. Howes, D. Peebles, & R. Cooper (Eds.), *Proceedings of the 9th international conference on cognitive modeling.* Manchester, UK.

Kieras, D. E., Ballas, J., & Meyer, D. E. (2001). *Computational models for the effects of localized sound cuing in a complex dual task* (Tech. Rep. No. TR-01/ONR-EPIC-13). University of Michigan.

Kieras, D. E., & Hornof, A. J. (2014). Towards accurate and practical predictive models of active-vision-based visual search. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 3875–3884). New York, NY, USA: ACM. doi: 10.1145/2556288.2557324

Kieras, D. E., Hornof, A. J., & Zhang, Y. (2015). Visual search of displays of many objects: Modeling detailed eye movement effects with improved epic. In *Proceedings of the 13th international conference on cognitive modeling.*

Kieras, D. E., & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human–Computer Interaction*, *12*(4), 391–438.

Kieras, D. E., Meyer, D. E., Ballas, J. A., & Lauber, E. J. (2000). Modern computational perspectives on executive mental processes and cognitive control: Where to from here. In S. Monsell & J. Driver (Eds.), *Control of cognitive processes: Attention and performance xviii* (pp. 681–712). MIT Press.

Kieras, D. E., Wood, S. D., Abotel, K., & Hornof, A. J. (1995). Glean: a computer-based tool for rapid goms model usability evaluation of user interface designs. In *Proceedings of the 8th annual acm symposium on user interface and software technology* (pp. 91–100). New York, NY, USA: ACM. doi: 10.1145/215585.215700

Kieras, D. E., Wood, S. D., & Meyer, D. E. (1997). Predictive engineering models based on the EPIC architecture for a multimodal high-performance human-computer interaction task. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *4*(3), 230–275.

Langolf, G. D., Chaffin, D. B., & Foulke, J. A. (1976). An investigation of fitts' law using a wide range of movement amplitudes. *Journal of Motor Behavior*, *8*, 113–128.

Lee, J. D., Caven, B., Haake, S., & Brown, T. L. (2001). Speech-based interaction with in-vehicle computers: The effect of speech-based e-mail on drivers' attention to the roadway. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *43*(4), 631.

Lee, J. D., Hoffman, J. D., & Hayes, E. (2004). Collision warning design to mitigate driver distraction. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 65–72). New York, NY, USA: ACM. doi: 10.1145/985692.985701

Levine, T. R., & Hullett, C. R. (2002, Oct). Eta squared, partial eta squared, and misreporting of effect size in communication research. *Human Communication Research*, *28*(4), 612–625. doi: 10.1111/j.1468-2958.2002.tb00828.x

Maloney, L. T., & Zhang, H. (2010, Nov). Decision-theoretic models of visual perception and action. *Vision Research*, *50*(23), 2362–2374. doi: 10.1016/j.visres.2010.09.031

Martin-Emerson, R., & Wickens, C. D. (1997). Superimposition, symbology, visual attention, and the head-up display. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *39*(4), 581–601.

Meyer, D. E., & Kieras, D. E. (1997a). A computational theory of executive cognitive processes and multiple-task performance: Part 1. Basic mechanisms. *Psychological Review*, *104*(1), 3–65.

Meyer, D. E., & Kieras, D. E. (1997b). A computational theory of executive cognitive processes and multiple-task performance: Part 2. Accounts of psychological refractory-period phenomena. *Psychological Review*, *104*(4), 749–791.

Monsell, S. (2003). Task switching. *Trends in Cognitive Sciences*, *7*(3), 134–140. doi: 10.1016/S1364-6613(03)00028-7

Moore, B. C. J. (1986). *Frequency selectivity in hearing*. London: Academic Press.

Moore, T., & Armstrong, K. M. (2003). Selective gating of visual signals by microstimulation of frontal cortex. *Nature*, *421*, 370–373.

Navon, D., & Gopher, D. (1979). On the economy of the human-processing system. *Psychological review*, *86*(3), 214–255.

Newell, A. (1973). You can't play 20 questions with nature and win. In W. G. Chase (Ed.), *Visual information processing* (pp. 238–308). New York: Academic Press.

Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice Hall.

Olson, R. L., Hanowski, R. J., Hickman, J. S., & Bocanegra, J. (2009). *Driver distraction in commercial vehicle operations* (Tech. Rep. No. FMCSA-RRR-09-045). U.S. Department of Transportation.

Pashler, H. (1989). Dissociations and dependencies between speed and accuracy: Evidence for a two-component theory of divided attention in simple tasks. *Cognitive Psychology*, *21*(4), 469–514.

R Core Team. (2015). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `http://www.R-project.org/`

Recarte, M. A., & Nunes, L. M. (2000). Effects of verbal and spatial-imagery tasks on eye fixations while driving. *Journal of Experimental Psychology: Applied*, *6*(1), 31–43.

Richtel, M. (2009, December 6). Promoting the car phone, despite risks. *New York Times*, A1. Retrieved from `http://www.nytimes.com/2009/12/07/technology/07distracted.html`

Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? a comment on theory testing. *Psychological Review*, *107*(2), 358–367.

Rollins, H. A., & Hendricks, R. (1980). Processing of words presented simultaneously to eye and ear. *Journal of Experimental Psychology: Human Perception and Performance*, *6*(1), 99–109.

Rosenbaum, D. A. (1980). Human movement initiation: Specification of arm, direction, and extent. *Journal of Experimental Psychology: General*, *109*(4), 444–474.

Rosenbaum, D. A. (2009). *Human motor control* (2nd ed.). Academic Press.

Salvucci, D. D. (2006). Modeling driver behavior in a cognitive architecture. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *48*(2), 362–380.

Salvucci, D. D. (2009, June). Rapid prototyping and evaluation of in-vehicle interfaces. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *16*(2), 9:1–9:33. doi: 10.1145/1534903.1534906

Salvucci, D. D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*, *115*(1), 101–130.

Schumacher, E. H., Lauber, E. J., Glass, J. M., Zurbriggen, E. L., Gmeindl, L., Kieras, D. E., & Meyer, D. E. (1999). Concurrent response-selection processes in dual-task performance: Evidence for adaptive executive control of task scheduling. *Journal of Experimental Psychology: Human Perception and Performance*, *25*(3), 791–814. doi: 10.1037/0096-1523.25.3.791

Schumacher, E. H., Seymour, T. L., Glass, J. M., Fencsik, D. E., Lauber, E. J., Kieras, D. E., & Meyer, D. E. (2001). Virtually perfect time sharing in dual-task performance: Uncorking the central cognitive bottleneck. *Psychological Science*, *12*(2), 101–108.

Schunn, C. D., & Wallach, D. (2005). Evaluating goodness-of-fit in comparison of models to data. In W. Tack (Ed.), *Psychologie der kognition: Reden and vorträge anlässlich der emeritierung von werner tack* (pp. 115–154). Saarbrueken, Germany: University of Saarland Press.

Scott, J. J., & Gray, R. (2008, Apr). A comparison of tactile, visual, and auditory warnings for rear-end collision prevention in simulated driving. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *50*(2), 264–275. Retrieved from `http://dx.doi.org/10.1518/001872008X250674` doi: 10.1518/001872008x250674

Simon, H. A. (1955). A behavioral model of rational choice. *The quarterly journal of economics*, 99–118.

Sperling, G., & Melchner, M. J. (1978). The attention operating characteristic: Examples from visual search. *Science*, *202*(4365), 315–318.

Strayer, D. L., & Johnston, W. A. (2001). Driven to distraction: Dual-task studies of simulated driving and conversing on a cellular telephone. *Psychological Science*, *12*(6), 462–466.

Strayer, D. L., Turrill, J., Coleman, J. R., Ortiz, E. V., & Cooper, J. M. (2014, October). *Measuring cognitive distraction in the automobile II: Assessing in-vehicle voice-based interactive technologies* (Tech. Rep.). AAA Foundation for Traffic Safety.

Taatgen, N. A., Van Rijn, H., & Anderson, J. (2007). An integrated theory of prospective time interval estimation: the role of cognition, attention, and learning. *Psychological Review*, *114*(3), 577–598.

Welford, A., & Brebner, J. M. T. (1980). *Reaction times*. Academic Press.

Wickens, C. D. (2002). Multiple resources and performance prediction. *Theoretical Issues in Ergonomics Science*, *3*(2), 159–177. doi: 10.1080/14639220210123806

Wierwille, W. W. (1993). Visual and manual demands of in-car controls and displays. In B. Peacock & W. Karwowski (Eds.), *Automotive ergonomics* (pp. 299–320). Philadelphia, PA: Taylor and Francis.

Zhang, Y., & Hornof, A. J. (2012). A discrete movement model for cursor tracking derived from moment-to-moment tracking data and the modeling of a dual-task experiment. In *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 56, pp. 1000–1004). doi: 10.1177/1071181312561209

Zhang, Y., & Hornof, A. J. (2013). Using model tracing and evolutionary algorithms to determine parameter settings for cognitive models from time series data such as visual scanpaths. In R. West & T. Stewart (Eds.), *Proceedings of the 12th international conference on cognitive modeling* (pp. 433–438). Ottawa, Canada: Carleton University.

Zhang, Y., & Hornof, A. J. (2014). Understanding multitasking through parallelized strategy exploration and individualized cognitive modeling. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 3885–3894). doi: 10.1145/2556288.2557351