Queueing Network Modeling of Human Performance

in Complex Cognitive Multi-task Scenarios

by

Shi Cao

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Industrial and Operations Engineering)
in the University of Michigan
2013

Doctoral Committee:

Professor Yili Liu, Chair
Professor Benjamin J. Kuipers
Associate Professor Bernard J. Martin
Professor Nadine B. Sarter

To

Zhongyi Cao

Linyan Shi

Xi Chen

&

Bruce Yi Cao

Acknowledgements

First of all, I must gratefully appreciate Dr. Yili Liu for his irreplaceable mentorship, altruistic support, intelligent discussion, and inspiring role model throughout this five-year-and-four-month endeavor, which has fulfilled my childhood dream, completely transformed me, and made me fully confident in my future independent research, work, and life. Equally important appreciations must go to Dr. Benjamin Kuipers, Dr. Bernard Martin, and Dr. Nadine Sarter who kindly served as my dissertation committee members and provided invaluable comments and suggestions. I have learned from all my committee members a tremendous amount of knowledge and professional skills that have greatly improved the quality of this dissertation work and broadened my horizons.

I would also like to thank many professors at University of Michigan, especially Dr. Paul Green, Dr. Thomas Armstrong, Dr. Don Chaffin, Dr. Xiuli Chao, Dr. Mark Daskin, and Dr. Monroe Keyserling, as well as many professors from other universities, especially Dr. Wayne Gray, Dr. Changxu Wu, and Dr. Jibo He, for their help and support. A lot of thanks also must go to my labmates Dr. Dan Nathan-Roberts, Dr. Christopher Best, and Soon-To-Be Doctor Fred Ruijia Feng as well as many other fellow students, because we help and support each other in this sometimes stressful process. I also sincerely appreciate many staff members who made my life in the department much easier and enjoyable, especially Tina Blay, Gwen Brown, Rod Capps, Eyvind Claxton, Wanda Dobberstein, Candy Ellis, Elizabeth Fisher, Matt Irelan, Christopher Konrad, "Mint" Minto, Sheryl Ulin, Amy Warhaft, and Charles Woolley. They are always there ready and willing to help with any kind of job otherwise difficult or impossible to complete.

This dissertation cannot be completed without the help, support, understanding, and love from my wife Soon-To-Be Doctor Xi Chen, my parents Zhongyi Cao and Linyan Shi, and my parents-in-law Zhong Chen and Guilan Han. And also many thanks to Bruce Yi Cao, who was very cooperative during the first few days of his life in the world that happened to be the last few days before my dissertation defense.

This dissertation work is partially supported by a Rackham International Research Award from the Horace H. Rackham School of Graduate Studies and the University of Michigan International Institute and a Rackham Graduate Student Research Grant from the Horace H. Rackham School of Graduate Studies at the University of Michigan. I would also like to thank the Department of Industrial and Operations Engineering at the University of Michigan for providing me fellowship and graduate student instructor positions that supported my study and dissertation work.

Finally, thank you (Yes, You!) for reading this (at last, there is someone interested in this thing).

Table of Contents

List of Figures

List of Tables

Abstract

As the complexity of human-machine systems grows rapidly, there is an increasing need for human factors theories and computational methods that can quantitatively model and simulate human performance and mental workload in complex multi-task scenarios. In response to this need, I have developed and evaluated an integrated cognitive architecture named QN-ACTR, which integrates two previously isolated but complementary cognitive architectures – Queueing Network (QN) and Adaptive Control of Thought-Rational (ACT-R). Combining their advantages and overcoming the limitations of each method, QN-ACTR possesses the benefits of modeling a wider range of tasks including multi-tasks with complex cognitive activities that existing methods have difficulty to model. These benefits have been evaluated and demonstrated by comparing model results with human results in the simulation of multi-task scenarios including skilled transcription typing and reading comprehension (human-computer interaction), medical decision making with concurrent tasks (healthcare), and driving with a secondary speech comprehension task (transportation), all of which contain important and practical human factors issues. QN-ACTR models produced performance and mental workload results similar to the human results. To support industrial applications of QN-ACTR, I have also developed the usability features of QN-ACTR to facilitate the use of this cognitive engineering tool by industrial and human factors engineers. Future research can apply QN-ACTR – which is a generic computational modeling theory and method – to other domains with important human factors issues.

Chapter 1. Introduction

Chapter Summary

How to computationally model human performance in complex cognitive and multi-task scenarios has become an important yet challenging question for human performance modelling and simulation. This chapter introduces existing cognitive architectures and the benefits of integrating Queueing Network and Adaptive Control of Thought Rational architectures. This integration can overcome the limitations in each method and model a wider range of tasks including cognitive multi-task scenarios that contain important and practical human factors issues.

1. Human Performance Modeling (HPM)

The analysis and evaluation of human performance have become increasingly crucial in the design and control of complex systems in a wide range of industrial and system engineering domains including healthcare, human-computer interaction, transportation, manufacturing, and aviation. System designers and policy makers have an increasing need for cognitive engineering methods that can quantitatively model and simulate human performance. Addressing this need, Human Performance Modeling (HPM) research focuses on the use of mathematical algorithms and computational simulation to model and simulate human performance. In this dissertation, the scope of human performance (i.e., behavioral results) focuses on the cognitive aspects rather than the physical aspects of performance. For example, the models in this dissertation work produce decision choice, reaction time, and correct rate results but do not simulate human motion trajectory, postures, the senses of touch and proprioception, the force of motor activities, or biomechanics. This cognitive focus is selected to address the rapidly increasing need for cognitive models in both the research field and the industry. In recent years, major challenges in the HPM research field were created by the increasingly complex human-machine interactions that are often required by complex human-machine systems being used or developed. Common tasks in the workplace today often require multi-task performance and complex cognitive activities in addition to physical activities. For example, a typist often needs to type quickly on the keyboard and comprehend the typing materials, while responding to incoming short text messages. An automobile driver often needs to control the vehicle's lane position and speed while communicating on a mobile phone. A physician in an emergency department needs to monitor multiple vital signs of multiple patients and make critical decisions under time pressure. The increased task complexity raises an important yet challenging human factors question: How to model and simulate human performance in complex cognitive and multi-task scenarios?

One answer to this question comes from developing integrated computational cognitive architectures. A cognitive architecture is a comprehensive representation of the human mind and possesses the following properties.

First, a cognitive architecture integrates isolated cognitive theories under the same framework – a unified theory of cognition – where different mental modules execute different aspects of mental functions. Since human performance involves all aspects of cognition including perception, memory, and response selection, a general human performance model needs to unify all the underlying cognitive mechanisms (Newell, 1990, 1973). For instance, in a study that reviewed existing models related to helicopter pilot performance, researchers found hundreds of isolated models and concluded that the central problem is to integrate them into a coherent unity that works together (Elkind, Card, Hochberg, & Huey, 1989).

Second, the modules of a cognitive architecture and their computational mechanisms are biologically inspired, based on psychological and neurological evidence of how the human brain works. Previous studies have identified the correspondence between the functional areas in the human brain and the information processing modules of cognitive architectures (J. R. Anderson, Fincham, Qin, & Stocco, 2008; Liu, Feyen, & Tsimhoni, 2006).

Third, a cognitive architecture can be implemented as a computerized simulation program to generate simulated behaviors and quantitatively model human performance. In a cognitive architecture, the human mind is represented as an information processing system. Mental representations and knowledge are programmed as symbols with their own data structures, and mental processing is programmed as computational functions applied to the symbols. The parameters of the architecture can represent human factors such as visual processing speed and working memory capacity. System inputs include task description and task-specific knowledge description, and outputs include simulated performance such as the contents of responses, processing time, correct rates, mental workload, and strategies. In this dissertation, as well as many cognitive-architecture-based modeling studies, the term “model” often refers to a set of task-specific information, including the tasks to be performed, the knowledge required to perform the tasks, and the parameters. On the other hand, the term “architecture” often refers to the generic cognitive framework representing non-task-specific information processing capabilities and mechanisms of the human mind in general.

Over the past several decades, numerous efforts have been made in developing unified theories of cognition and led to the creation of several important cognitive architectures including Adaptive Control of Thought-Rational (ACT-R; J. R. Anderson et al., 2004), Executive-Process Interactive Control (EPIC; D. E. Meyer & Kieras, 1997), Soar (Laird, Newell, & Rosenbloom, 1987), and Queueing Network-Model Human Processor (QN-MHP; Liu et al., 2006). The evolution of these architectures is a testimony to model integration. ACT-R originally focused on modeling pure cognitive activities and later acquired perceptual-motor modules conceptually based on EPIC’s perceptual-motor processors (Byrne & Anderson, 1998). Soar’s declarative memory modeling mechanisms are inspired by ACT-R’s declarative knowledge representation and computations (Laird, 2008). QN-MHP integrates mathematical Queueing Network (QN) modeling with Model Human Processor and GOMS (Goals, Operators, Methods, and Selection rules) models (Card, Moran, & Newell, 1983; John & Kieras, 1996); QN modeling itself has integrated a wide range of mathematical models of mental processes (Liu, 1996, 1997). Although previous work of model integration has significantly expanded the scope of tasks that can be modeled by a cognitive architecture, existing architectures still have difficulties in modeling human performance in multi-task scenarios involving complex cognitive activities. ACT-R and Soar have the capability to model complex cognition such as reading comprehension, learning, and decision making, but they lack a mechanism that can schedule multi-task processing at the local module level. EPIC focuses on modeling perceptual-motor tasks but lacks the mechanisms to model complex cognitive activities that involve declarative memory, for example, language learning and sentence comprehension. QN-MHP uses queueing as a multi-task coordination mechanism but is limited in modeling complex cognition. These limitations indicate the need for further model integration.

This dissertation research focuses on the integration of QN and ACT-R, because the two previously isolated cognitive architectures have many complementary properties (Liu, 2009). An integration of the two architectures could take the advantages and overcome the limitations of each in modeling cognitive multi-task scenarios. The following sections will introduce QN and ACT-R in more details and then introduce the theoretical values of the integration.

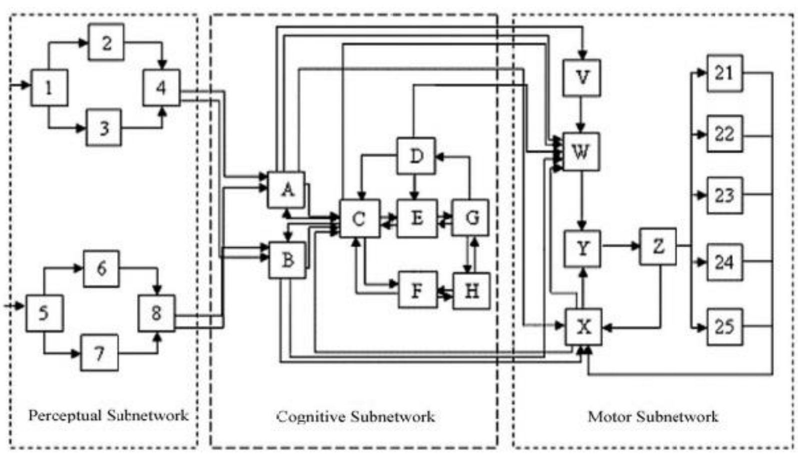
2. Queueing Network (QN) Architecture of Human Performance

Queueing theory is the mathematical study of waiting before processing. In a queueing system, customers/entities arrive at the system, get processed in servers, and leave the system. If a server's capacity is reached, incoming entities wait in a queue until the server is free. Queueing network (QN) is a network of servers connected by paths. Since the early 20^{th} century, QN has been widely used in system engineering, such as telecommunications, traffic engineering, and product line design, to evaluate system performance including processing time, waiting time, and server utilization. Although QN and queueing theory are well-established mathematical formulations that have been widely used in the modeling and simulation of complex engineering systems, they were not explicitly used in the modeling and simulation of human mental performance until recently. Nevertheless, there have been evidences that suggest the existence of queues in the cognitive system. At the neuron and synapse level, it has been found that the mobilization of synaptic vesicles (which store and release neurotransmitters to transfer information between neurons) follows a queueing mechanism that gives some vesicles higher priority than others (Holt & Jahn, 2004). At the cortex level, motor commands are also believed to form queues in the human brain before their execution.

> "A sequence of different isolated finger movements requires programming in the supplementary motor areas. I suggest that the supplementary motor areas are programming areas for motor subroutines and that these areas form a queue of time-ordered motor commands before voluntary movements are executed by way of the primary motor area" (p. 118, P. E. Roland, Larsen, Lassen, & Skinhøj, 1980).

QN can be used to model human performance in two ways, as a mathematical tool and as a computerized simulation framework. As a mathematical tool, QN can be used to derive a closed-form expression of reaction time, given inter-arrival time distributions and service time distributions. Despite of being widely used in system engineering, QN is not among the first models selected by mathematical psychologists to represent and analyze the human mind. Instead, isolated mathematical methods and models for reaction time were developed (for a review, see Liu, 1996), including the subtractive method (Donders, 1868), the additive factors method (Sternberg, 1969), the cascade model (McClelland, 1978), the queue-series model (Miller, 1993), and the Critical Path Network model (Schweickert, 1978). These methods and models were later proved to be special cases of QN (Liu, 1996), and QN was further shown to be able to represent cognitive structures (such as feedback and bypass structures) that cannot be modeled by the previous methods. Therefore, QN is a more general mathematical tool for reaction time modeling and has more explanatory power than the previous mathematical models.

As a framework in computerized simulation of HPM, QN has been used in the recent work of QN-MHP (Liu et al., 2006). QN serves as the structural framework. Natural Goals, Operators, Methods, and Selection rules Language (NGOMSL) is the task analysis method. Model Human Processor (MHP) provides the empirical data for processing logics and model parameters. The task representations of stimuli and task procedures are stored in spreadsheets as well as the entity attributes and array variables used in the simulation platform (ProModel™). Compared with MHP's three processors of perception, cognition, and motor, QN-MHP has a finer granularity. The granularity of QN-MHP was set at a level where each server represents millions of neurons from a certain functional field of the brain. This server selection was strongly biologically-inspired and followed the cortical field and neuronal population activation theory (P. Roland, 1993). Figure 1 illustrates QN-MHP's server network and the functional mapping onto the corresponding brain areas. QN-MHP has successfully modeled laboratory tasks like the psychological refractory period (Wu & Liu, 2008a) and real-world tasks like driving and in-vehicle map reading (Liu et al., 2006) and transcript typing (Wu & Liu, 2008b).



Perceptual Subnetwork

1. Common visual processing (eyes, lateral geniculate nucleus, superior colliculus, primary and secondary visual cortex)
2. Visual recognition (dorsal system)
3. Visual location (ventral system)
4. Visual recognition and location integration (distributed parallel area including the connections among V3 , V4 and V5, superior frontal sulcus, and inferior frontal gyrus)
5. Common auditory processing (middle and inner ear)
6. Auditory recognition (area from dorsal and ventral cochlear nuclei to the inferior colliculus)
7. Auditory location (area from ventral cochlear nucleus to the superior olivary complex)
8. Auditory recognition and location integration (primary auditory cortex and planum temporale)

Cognitive Subnetwork

A. Visuospatial sketchpad (right-hemisphere posterior parietal cortex)
B. Phonological loop (left-hemisphere posterior parietal cortex)
C. Central executive (dorsolateral prefrontal cortex (DLPFC), anterior-dorsal prefrontal cortex (ADPFC) and middle frontal gyrus (GFm))
D. Long-term procedural memory (striatal and cerebellar systems)
E. Performance monitor (anterior cingulate cortex)
F. Complex cognitive function: decision, calculation, anticipation of stimulus in simple reaction etc. (intraparietal sulcus (IPS), the superior frontal gyrus (SFS), the inferior frontal gyrus (GFi), the inferior parietal cortex and the ventrolateral frontal cortex, the intraparietal sulcus and the superior parietal gyrus)
G. Goal initiation (orbitofrontal region and amygdala complex)
H. Long-term declarative & spatial memory (hippocampus and diencephalons)

Motor Subnetwork

V. Sensorimotor integration (premotor cortex)
W. Motor program retrieval (basal ganglia)
X. Feedback information collection (somosensoy cortex)
Y. Motor program assembling and error detecting (supplementary motor area (SMA) and the pre-SMA)
Z. Sending information to body parts (primary motor cortex)
21-25: Body parts: eye, mouth, left hand, right hand, foot

Figure 1. The server structure of QN-MHP (from Wu & Liu, 2007).

An advantage of the QN architecture is modeling human multitasking performance, as explained in details as follows. First, QN does not need executive control to model multitasking performance, for example, as demonstrated in the modeling of the sub-additive effect of psychological refractory period (Wu & Liu, 2008a). In contrast, EPIC and ACT-R both need executive control (strategic response deferment) that strategically locks and unlocks a task to explain the effect. Executive control is not a preferred method for HPM, because it strongly depends on the task characteristics and an individual's strategy, and needs to be customized for different tasks and different individuals. QN's method to model multitasking is a generic one independent from task characteristics and strategies, because it resolves resource confliction at the local server level rather than high-level executive control.

Second, QN has a hybrid central cognitive processing system containing both serial and parallel servers. In contrast, other architectures assume a single central cognitive processor that is either serial (ACT-R and Soar) or parallel (EPIC). Dual-task results from experiments involving simple cognition are more in favor of a parallel processor, but results from experiments involving complex cognition (e.g., arithmetic computation, Byrne and Anderson, 2001) are more in favor of a serial one. Results from both sides can be easily explained by the QN architecture, which has a multiple-capacity Server C and a single-capacity (serial) Server F (see Figure 1). Server C can process several low-level cognitive operations in parallel, but Server F has to serially process multiple high-level cognitive operations. Predicted by this hybrid system, a simple task and a complex task are likely to be processed in parallel, given no bottlenecks in the perception and the motor sub-networks; multitasking involves two or more complex cognitive tasks are likely to be processed in serial. It is important to note that this two-server formalization was not specifically designed to explain the mixed empirical results. Instead, it was inspired by the cortical field theory to match human brain cortical functions.

Third, since QN has a finer granularity in the cognitive sub-network, the same task may be processed through different paths, which may be used to model the effect of secondary task demands and the effect of learning. For example, the processing path at the novice level may go through servers A-C-F-C-W, and the processing path at the expert level may go through A-C-W. The expert process may not need Server F and is therefore faster.

Another advantage of QN is its ability to model mental workload using the concept of utilization. In queueing theory, utilization is the proportion of the system's resources that is used by the entities in the system. Considering server capacities in QN as mental resources in the brain, one can intuitively link utilization with mental workload. Previous QN-MHP studies have successfully modeled workload measurements including both an objective measurement of P300 in Event-Related Potential (Wu, Liu, & Quinn-Walsh, 2008) and a subjective measurement of NASA-task load index (Wu & Liu, 2007).

3. Adaptive Control of Thought-Rational (ACT-R)

ACT-R (J. R. Anderson et al., 2004) is a cognitive architecture implemented as a production rule system. It assumes two types of knowledge representations: chunks and production rules (rules, for short). Chunks represent declarative knowledge such as the fact that the sum of 3 and 4 is 7. In contrast, rules represent procedural knowledge of how to do things and are executed to produce actions (Squire, 2004). Rules are coded in ACT-R as condition-action (if-then) pairs. For example, a simple rule could be, if seeing a light is illuminated, then press a button. A rule can only be executed or fired when it is matched, which means its condition part matches the agent's current "mental state". A mental state consists of the state of each mental module. ACT-R's module structure is illustrated in Figure 2. Each ACT-R module has a buffer that can hold only one chunk at a time. As a production rule system, ACT-R "thinks" and "acts" by firing production rules until a pre-defined goal state is reached.

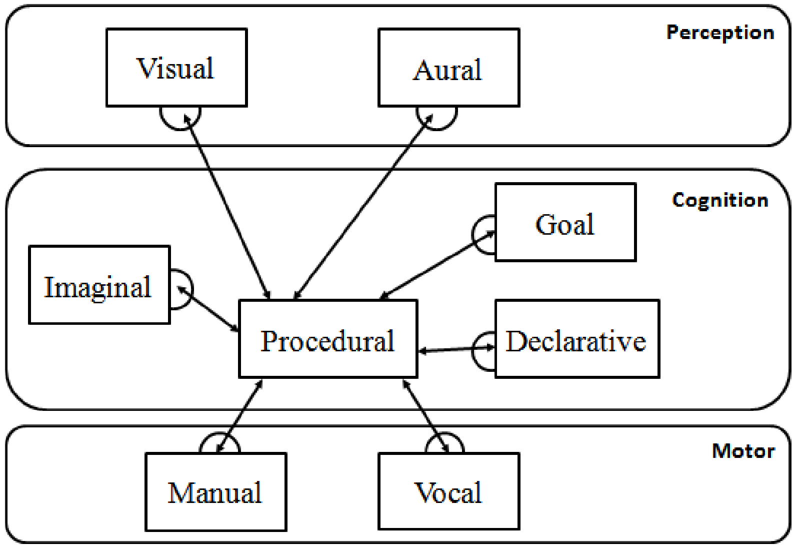


Figure 2. Module structure of ACT-R version 6.0. Circles represent buffers for each module (adapted from J. R. Anderson et al., 2004).

For chunks, ACT-R has algorithms for calculating chunk retrieval time and retrieval probability based on chunk activation and association in its declarative module, which stores and retrieves declarative knowledge. Larger values of chunk activation lead to faster and more successful retrieval. For rules, ACT-R has algorithms for rule selection and learning based on rule utility in its production module, which matches, selects, and executes rules. Every production rule in ACT-R has a firing cycle of 50 ms (by default) and a utility value that represents the relative desirability to fire the rule. Since only one rule is allowed to fire within a firing cycle, multiple rules matched in the same cycle will compete. Rules with greater utilities have greater chances of being fired, determined by a soft-max function (for details, see J. R. Anderson et al., 2004; J. R. Anderson & Lebiere, 1998). These algorithms are part of the general architecture that is the same for modeling different tasks, whereas task-specific knowledge of chunks and production rules need to be specified individually for each task in each model.

Previous studies using ACT-R have modeled complex cognitive tasks including reading comprehension (J. R. Anderson, Budiu, & Reder, 2001), decision making (Fu & Anderson, 2006), and language learning (Taatgen & Anderson, 2002). However, these tasks are mostly single-tasks. Recently, a multi-task scheduling mechanism called threaded cognition has been reported and tested (Salvucci & Taatgen, 2008). Threaded cognition is a general theory about the mechanisms of human multitasking and can be implemented in ACT-R. The term thread is borrowed from computer technology where a single central processing unit (CPU) fast switches between different task threads. Each thread represents the task demands from each task. On the basis of ACT-R's assumption of a serial central cognitive processor, threaded cognition has two additional assumptions for multitasking mechanisms. First, it assumes that the goal buffer can hold more than one goal simultaneously. Second, "Conflict Resolution Assumption: When multiple threads contend for the procedural resource, the least recently processed thread is allowed to proceed" (Salvucci & Taatgen, 2008, p. 111). In essence, the conflict resolution assumption is a type of queueing scheme (Liu, 1997). As described in a previous QN-MHP study, "If more than one goal may be executed (i.e., a multiple task scenario), …, three schemes have been included: one, randomly select between the various goals; two, choose the goal which has waited the longest since its last execution; or three, select the goal with the highest priority value in the goal list array" (Feyen, 2002, p. 138).

4. An Integrated Cognitive Architecture to Model Cognitive Multitasking Performance

Existing HPM architectures have difficulties and limitations in modeling complex cognitive multi-task scenarios. QN-MHP lacks a sophisticated declarative memory reflecting activation and association; therefore, it lacks the ability to model complex cognitive tasks involving long-term memory retrieval, such as reading comprehension, problem solving, and decision making. On the other hand, symbolic production rule architectures like ACT-R can model (i.e., can be used to model, the same below) complex cognitive tasks but have limitations in modeling multi-task scenarios. The focus of this dissertation research is to develop and examine an integrated cognitive architecture that combines the advantages of QN-MHP's server network and queueing mechanisms and ACT-R's algorithms for complex cognition. Among several symbolic production rule architectures, ACT-R is chosen because of its well-developed declarative memory retrieval algorithms based on chunk activation and association and also its production rule selection algorithms based on utility. Previous single-task models using ACT-R are great resources for developing multi-task models in the integrated architecture.

As discussed by Liu (2009), the further integration of the QN mathematical architecture and symbolic architectures continues along the line of work using QN to model and simulation human performance (Liu, 1996, 2009; Liu et al., 2006; Wu & Liu, 2008a) and can examine and potentially resolve several theoretical and methodological issues in human performance modeling. In particular, the integration of QN and ACT-R architectures has the potential to model performance and mental workload in multi-tasks with complex cognitive components (Cao & Liu, 2011a, 2011b, 2012c). Continuing model integration and the development of a unified theory of cognition, the goals of the present study are to integrate QN and ACT-R by merging their structures and modeling mechanisms and build a cognitive architecture that can model a wider range of cognitive and concurrent tasks. This integration also allows the examination of several important theoretical issues in ACT-R – including concurrent goal scheduling and module jamming – from the QN perspective.

Modeling complex cognitive and multi-task scenarios. The integration of QN and ACT-R combines QN's strength in modeling multi-task performance and ACT-R's strength in modeling complex cognition. Two unique theoretical positions in QN are the queueing mechanism and the hybrid server network (Liu et al., 2006; Wu & Liu, 2008a). The queueing mechanism can serve as a natural multi-task coordination mechanism at the local server level, and the hybrid network with both serial and parallel processing servers can model multi-task processing with a finer granularity. Previous studies using the QN architecture (i.e., QN-MHP) have modeled multi-task scenarios including the psychological refractory period dual-tasks (Wu & Liu, 2008a) and a driving and map viewing dual-task (Liu et al., 2006). On the other hand, ACT-R's strength lies in its symbolic knowledge representations and sub-symbolic computations that can model memory retrieval and learning. Since QN and ACT-R possess unique and complementary modeling capabilities, the integration of the two architectures can benefit from their strengths and overcome the limitations of each one.

Concurrent goal scheduling. In ACT-R, each module has a buffer as an interface to connect with other modules. One of ACT-R's theoretical positions is that each buffer (e.g., the goal buffer storing the information about the current task) can hold only a single chunk at a time (J. R. Anderson et al., 2004). This theoretical position, although without clear neurological evidence, has been shown to be suitable in the modeling of single-tasks. However, in dual-task models, this position often requires dual-task models in ACT-R to combine the two goals from the two task components into one task-specific goal for the dual-task in question (e.g., Byrne & Anderson, 2001). This task-specific modeling method makes it difficult to model a wide range of dual-task scenarios, because modelers must define the task-specific knowledge for each scenario. From the QN perspective, multiple goals can co-exist in the goal buffer, and multi-task performance emerges as the behavior of multiple streams of information flowing through a network, with no need for multitask-specific goals to interleave production rules into a serial program or for an executive process to interactively control task processes (Liu, 1997; Liu et al., 2006). Encouragingly, this QN position has been adopted in recent ACT-R based threaded cognition work for multi-task scheduling (Salvucci & Taatgen, 2008). In fact, the core mechanism of threaded cognition can be represented as a special type of queueing with a serial server (i.e., the production module in ACT-R) that gives priority to the longest waited task entity. The full integration of QN and ACT-R can further test and examine different queueing scheduling mechanisms to model a wider range of multi-task scenarios, especially those involving dynamic and complex cognitive tasks and producing high mental workload.

Module jamming. Another theoretical assumption in ACT-R is that a module processes one request at a time, and its buffer content is limited to a single declarative chunk (ACT-R Tutorial Unit 2; J. R. Anderson et al., 2004). As a result of this serial assumption, if a module is busy processing a request and receives another request at the same time, the module will be jammed. Technically, the handling of module jamming is not included in the theory of ACT-R, and the programming consequences of jamming may vary (e.g., removing the first request, ignoring the second request, or stopping the whole simulation), so the modeling guideline is to avoid it. To avoid this jamming issue, ACT-R modeling requires modelers to "query the state in every production that makes a request that could potentially jam a module" (ACT-R Tutorial Unit 2, p. 9). From the QN perspective, this module jamming issue can be naturally resolved by adding queues to the modules. When a request arrives at a busy module, the request can wait in a queue until the module is free. This QN method does not require a production rule to query the state of the module or wait for the process in the module to finish, and therefore it is more suitable to model fast motor performance such as skilled transcription typing. The existence of queues in the motor processors has also received neurophysiological support from previous research (P. E. Roland et al., 1980).

5. Thesis Structure

The value of integrating QN and ACT-R in multitask modeling is demonstrated in this dissertation, and the resulting architecture is named QN-ACTR.

Chapter 2 introduces the theoretical assumptions and implementation of QN-ACTR and describes the verification of QN-ACTR. Then the benefits of the integration are demonstrated in the simulation of transcription typing tasks involving multi-task performance and reading comprehension, showing QN-ACTR's improved modeling capabilities in complex cognitive and multi-task scenarios.

Chapter 3 reports a laboratory experiment that examined the effects of concurrent tasks on diagnostic decision making, which is an important human factors issue in the healthcare work environment. The human data collected from this experiment will be compared with model simulation results in Chapter 4.

Chapter 4 describes the evaluation of cognitive models built in QN-ACTR to model cognitive multitasking performance in diagnostic decision making tasks, demonstrating that the models can produce performance and mental workload results very similar to the human results reported in Chapter 3.

Chapter 5 and 6 describe a simulated driving experiment and the corresponding model simulation conducted to further examine QN-ACTR's modeling capability in the transportation domain. The experiment collected detailed human performance and mental workload results in a driving and speech comprehension dual-task scenario. Then QN-ACTR model was built to simulate these human results.

Chapter 7 describes the usability development of QN-ACTR for cognitive engineering applications. These usability features improve the accessibility for industrial engineers who are not experts of human performance modeling to use QN-ACTR as a cognitive engineering tool.

Chapter 8 summarizes the results and conclusions from this dissertation research and discusses future research directions.

Chapter 2. Framework and verification of Queueing Network – Adaptive Control of Thought Rational (QN-ACTR)

Chapter Summary

This chapter introduces the theoretical assumptions and implementation of the integrated cognitive architecture Queueing Network–Adaptive Control of Thought Rational (QN-ACTR) and describes the verification of QN-ACTR by testing and examining it in the simulation of 20 typical tasks from the ACT-R literature and comparing the results. After the verification, the benefits of the integration are demonstrated in the simulation of transcription typing tasks involving multi-task performance and reading comprehension, showing QN-ACTR's improved modeling capabilities in complex cognitive and multi-task scenarios that are difficult to model by either QN or ACT-R alone.

1. Queueing Network-Adaptive Control of Thought Rational (QN-ACTR)

Theoretically, the structure of QN cognitive architecture and ACT-R cognitive architecture can be merged to form an integrated architecture. Figure 3 shows the current mental structure of QN-ACTR resulting from merging QN and ACT-R architectures. The servers in QN-ACTR correspond to ACT-R's modules and buffers, some of which are grouped to match the corresponding servers of the QN structure as in QN-MHP. Entities travelling between these servers correspond to ACT-R's information units including buffer requests, chunks, and production rules. Table 1 shows the functional correspondence between ACT-R modules and QN servers that supports the integration of their structures. The processing logics in each QN-ACTR server are identical to the algorithms in the corresponding ACT-R module, including the sub-symbolic computations in the production and the declarative modules. As previously described in Chapter 1, ACT-R modules do not have queueing mechanisms, but queues can be added from the QN perspective to support the scheduling of multi-tasks at the local server level. In QN-ACTR, queues are added to the modules that have non-zero processing time and a limited capacity. Currently, I do not apply any constraint to the capacity of queues and assume no time is needed for an information entity to enter or leave a queue. These assumptions are the same as the ones in previous QN modeling work (Liu et al., 2006; Wu & Liu, 2008a).

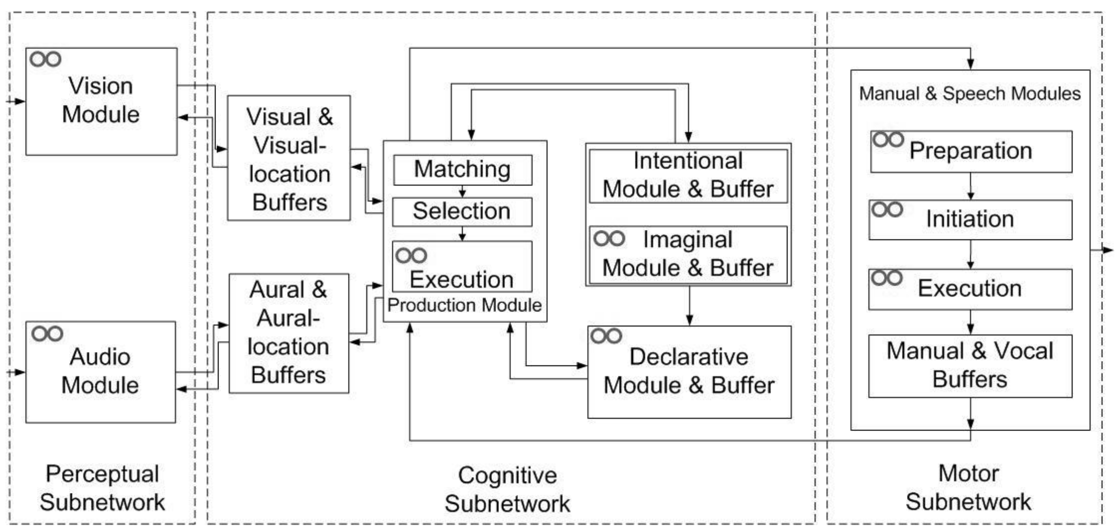


Figure 3. Server structure of QN-ACTR. Queue symbols (shown as two circles) mark the servers where queues are added from the QN's perspective. All the server processing logics in the QN-ACTR are identical to the corresponding algorithms in ACT-R (adapted from Cao & Liu, 2012c).

As shown in Figure 3, the servers are mainly based on ACT-R's modules and buffers. In ACT-R, there is no special module of working memory. Instead, "working memory can be equated with the portion of declarative memory above a threshold of activation" (p. 221, J. R. Anderson, Reder, & Lebiere, 1996), and the chunks that are temporarily held in the cognitive subnetwork (e.g., in the goal buffer) may provide a context and spread activation to the chunks in declarative memory, facilitating the retrieval of context-relevant declarative chunks. In this regard, entities (e.g., chunks) in the queues of the cognitive subnetwork may provide sources of activation for the working memory. In the perceptual subnetwork, there may be a connection between the queues and short term sensory storages, including the visual sensory memory (Dick, 1974) and the auditory sensory memory (Darwin, Turvey, & Crowder, 1972), because they both have the functionality of temporarily storing information. In the motor subnetwork, queues are believed to temporarily store time-ordered motor commands (P. E. Roland et al., 1980). The correspondence between queues and cognitive or neurological constructs, however, is not the focus of the current study. Instead, I focus on the verification of QN-ACTR in terms of human performance modeling capabilities and demonstrate the importance of the theoretical concepts of "queueing" in understanding cognitive architecture and multi-task performance.

Table 1. Correspondence between ACT-R modules and QN servers (from Cao & Liu, 2012c).

ACT-R modules and buffers	**Corresponding QN servers. See (Wu & Liu, 2008a) for server details.**
Vision module	Server 1 – 4
Audio module	Server 5 – 8
Visual and visual-location buffers	Server A
Aural and aural-location buffers	Server B
Production module	Server C, D, F
Intentional and imaginal modules and buffers	Server E, G
Declarative module and buffer	Server H
Manual and speech modules and buffers	Server V, W, X, Y, Z, 21 – 25

At the implementation level, QN-ACTR is a computerized program built on a discrete event simulation platform Micro Saint® Sharp (http://www.maad.com). A full integration was implemented, which means that all the data structures and the functions in ACT-R have been ported from the original Lisp implementation to Micro Saint® Sharp. This programming platform is selected because of three major reasons. First, this platform provides graphical interfaces for easy QN construction and simulation and also supports the visualization of server network and task interaction. Second, this C# based platform supports C# programming plug-in functions and the connection with other C# applications, which help the development of QN-ACTR as an easy-to-use cognitive engineering tool with features such as model building assistants and human experiment platforms. Third, it is the same platform on which IMPRINT (Allender, Kelley, Archer, & Adkins, 1997) is implemented and therefore supports future seamless integration between QN-ACTR and IMPRINT – a widely used system and task analysis tool that has its unique features in mental workload modeling.

For modelers using QN-ACTR, the model development process includes the steps of task setup, task-specific knowledge setup, and parameter setup (Figure 4). The task setup refers to the specification of display and control mechanisms in the task or experiment. The task-specific knowledge setup refers to the specification of knowledge, i.e., declarative chunks and production rules, required to perform the task. The parameter setup adjusts parameters that control the model performance. QN-ACTR provides two methods to build a model – a text-based syntax method and a click-and-select interface (Cao & Liu, 2012b). The syntax method supports fast and direct model editing (i.e., copy and paste), which is designed for advanced users. Since the syntaxes for the task-specific knowledge and the parameter setups are identical to ACT-R syntaxes, available ACT-R codes can be directly used for QN-ACTR models. The click-and-select interface assists novice users and allows them to describe the model following experiment logic and knowledge by selecting from menu items and filling in blanks using a natural language (English), without the need to learn any special programming or cognitive engineering language.

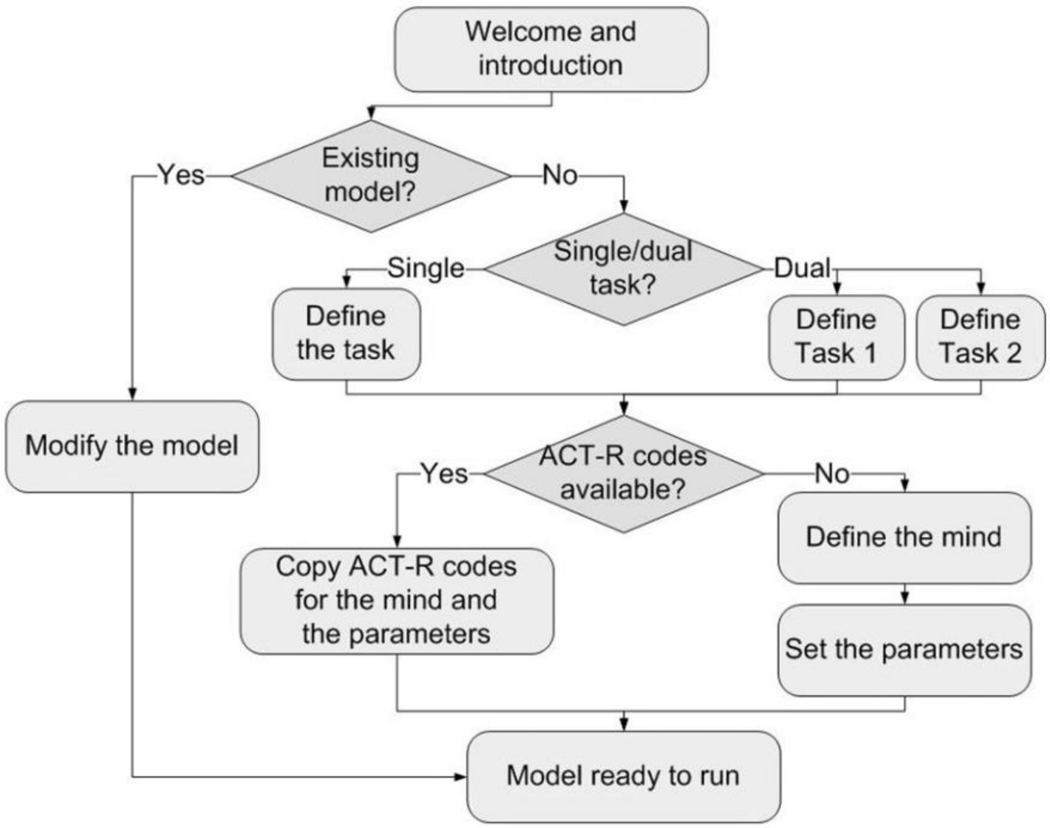


Figure 4. Flow chart showing the model development process in QN-ACTR (from Cao & Liu, 2012b).

QN-ACTR's modeling results include the trace of simulated mental activities, behavioral results, reaction times, correct rates, and mental workload. These results can be visualized while the model is performing the task and recorded for future analyses. QN-ACTR also has an integrated human experiment platform for human data collection. For example, a modeler can use QN-ACTR to conduct simulated driving experiments with steering wheels and pedals. This feature allows models and humans to perform and be compared in the same tasks with identical interfaces, with no need to replicate the real world experiment system in the modeling platform for models to interact with. Using the same experiment platform saves programming work and, more importantly, avoids any discrepancy between human and model tests caused by different experiment setups.

2. Model Verification

2.1 Method

After merging the structures and functions of QN and ACT-R architectures, I conducted a verification study with the purpose of examining whether the ACT-R functions built in QN-ACTR are accurate and complete. I tested QN-ACTR in the simulation of 20 tasks (summarized in Table 2) that have been modeled by ACT-R and threaded cognition. Among these tasks, 17 tasks were selected from ACT-R 6.0 (v1.3) tutorial to cover all ACT-R modeling mechanisms/algorithms and a wide range of typical single-tasks modeled in ACT-R, including visual-motor, auditory-vocal, declarative learning, procedural learning, and decision making tasks. The other three tasks were selected from dual-task experiments (Schumacher et al., 2001) using the psychological refractory period paradigm and have been modeled by threaded cognition implemented in ACT-R. As previously described, threaded cognition can be considered as a special type of queueing mechanism applied to ACT-R's production module, giving priority to the longest waited task. All together, these 20 tasks provide a thorough test bed to verify the implementation and programming of QN-ACTR, and this work of verification has been presented to and passed the review of the research community (Cao & Liu, 2011a, 2012a).

Models for the 20 tasks were developed strictly following their corresponding ACT-R models. The task displays and controls were modeled using QN-ACTR's syntaxes and templates. The codes specifying task-specific knowledge and parameters were directly copied from the codes used in the original ACT-R models. In addition, I also used the Common Random Numbers technique (McGeoch, 1992) to assign ACT-R and QN-ACTR models the same randomization method and the same seeds in order to further control the sources of variance in the verification. Table 3 shows excerpts from the QN-ACTR model codes for the Demo task. This task presents a random letter on the screen and requires a key press response corresponding to the displayed letter.

Table 2. Descriptions of the 20 tasks modeled to verify QN-ACTR (adapted from Cao & Liu, 2011a).

Model	**Description**
	ACT-R 6.0 (v1.3) tutorial models
Addition	Compute 5 + 2 by counting 5, 6, and 7.
Count	Count from 2 to 4.
Semantic	Judge if an animal belongs to a category.
Tutor-model	Compute 36 + 47 by first adding the single digits and then the ten digits.
Demo	See a letter. Press the key for the letter.
Unit-2-assignment	See three letters. Two are the same. Press the key of the single letter.
Sperling	Briefly see 12 letters in three rows. Press keys for letters in the target row, which is indicated by the pitch of a tone. Higher pitch = higher row.
Subitize	See several “x”s. Say how many “x”s there are.
Paired	Memorize and recall 20 word-number pairs, one pair a trial.
Zbrodoff	Judge alphabetic arithmetic problems by pressing keys. For example, A + 2 = C is correct, but B + 3 = F is not.
Fan	Remember a set of people-location facts. Answer queries such as “is the captain in the park?”
Group	Imperfect recall of 9 numbers in three groups, 123-456-789.
Siegle	Imperfect single-digit addition due to number similarity.
Bst-learn	Create a stick of a particular target length by selecting building sticks with three different lengths using the mouse.
Choice	Repetitively guess a biased coin.
Paired-learning	Same as the paired model except starting the task-specific knowledge from descriptive instructions instead of procedural rules.
Past-tense	Learn English past tenses from examples, demonstrating the overregularization of irregular verbs.
	Schumacher et al. (2001) experiments modeled by threaded cognition
Exp. 1	Psychological refractory period, visual-motor and auditory-vocal dual-task. No specific response order.
Exp. 2	Same as Exp. 1 except prioritizing on the auditory-vocal task.
Exp. 3	Same as Exp. 1 except using incompatible stimulus-response associations instead of compatible ones.

2.2 Results

Both the mental activity results recorded in the model output traces and the behavioral results such as reaction times and correction rates were compared between QN-ACTR and ACT-R models. Output traces were compared line by line to examine both the time and event contents. For example, the following line of trace,

222.423 DECLARATIVE RETRIEVED-CHUNK pair18-0

from the Paired model results in QN-ACTR showed that at clock time 222.423 second, the model’s declarative module retrieved the chunk *pair18-0*. This trace was identical to the corresponding ACT-R trace. For the tasks with quantitative behavioral results, mean absolute percentage error (*MAPE*), root mean square error (*RMSE*), and coefficient of determination (R^2) were computed between QN-ACTR and ACT-R results.

Table 3. Model code excerpts from a QN-ACTR model for the Demo task in ACT-R's tutorial. The syntaxes specifying task-specific knowledge and parameters are identical to the ones used in ACT-R (from Cao & Liu, 2013c).

Model setup step	Model codes
Task description	(use_task_dbt_template :method discrete_display_feedback_two_stages_method) (add_trials_from_discrete_display_feedback_two_stages_method :add_number_of_trials_per_block 1 :number_of_responses_per_trial 1 (:item_type display_item_visual_text :visual_text ("B" "C" "D" "F" "G" "H" "J" "K" "L" "M" "N" "P" "Q" "R" "S" "T" "V" "W" "X" "Y" "Z") :correct_response_to_each_visual_text (b c d f g h j k l m n p q r s t v w x y z) :text_randomization without_replacement :display_item_screen_location_x (125) :display_item_screen_location_y (150)))
Declarative knowledge description	(chunk-type read-letters state) (chunk-type array letter) (add-dm (start isa chunk) (attend isa chunk) (respond isa chunk) (done isa chunk) (goal isa read-letters state start))
Procedural knowledge description	(P find-unattended-letter =goal> ISA read-letters state start ==> +visual-location> ISA visual-location :attended nil =goal> state find-location)
Parameters	(sgp :v t :needs-mouse nil :show-focus t :trace-detail high)

As summarized in Table 4, the verification results showed that QN-ACTR models generated the same results as the original ACT-R models. Results from 15 of the 20 tasks were identical between QN-ACTR and ACT-R. For the other five models, results were very similar (*MAPE* < 5.0% and R^2 > 0.9). The sources of the remaining variances include the difference of built-in random number functions between Lisp and C#, which were used in randomly focusing visual attention on a visual item, and the difference in rounding digits between Lisp and C#. These results support the conclusion that the ACT-R functions built in QN-ACTR are accurate and complete.

Table 4. QN-ACTR verification results (adapted from Cao & Liu, 2011a).

Model	Results
	ACT-R 6.0 (v1.3) tutorial models
Addition	Same output traces ended at 0.5 s.
Count	Same output traces ended at 0.3 s.
Semantic	Same output traces for test 1 (ended at 0.15 s), test 2 (0.25 s), and test 3 (0.35 s).
Tutor-model	Same output traces ended at 0.45 s.
Demo2	Same output traces for both home-location key condition (ended at 0.785 s) and other key condition (1.035 s).
Unit-2-assignment	Same output traces for both home-location key condition (ended at 1.055 s) and other key condition (1.305 s).
Sperling	1000 run results (numbers of correct responses): *MAPE* = 1.4%, *RMSE* = 0.032, R^2 = 0.997.
Subitize	Same output traces. 1 run results (both reaction time and correction rate): *MAPE* = 0%, *RMSE* = 0, R^2 = 1.
Paired	
Zbrodoff	
Fan	
Group	Same output traces ended at 18.535 s.
Siegle	Same output traces. 500 run results (response distribution): *MAPE* = 0%, *RMSE* = 0, R^2 = 1.
Bst-learn	Same output traces. 5 run results (overshoot chances and rule utilities): *MAPE* = 0%, *RMSE* = 0, R^2 = 1.
Choice	Same output traces. 1 run results (rate of guessing heads): *MAPE* = 0%, *RMSE* = 0, R^2 = 1.
Paired-learning	Same output traces. 1 run results (both reaction time and correction rate): *MAPE* = 0%, *RMSE* = 0, R^2 = 1.
Past-tense	Same production rules composed and same output traces over 10 trials, except for a 0.001% time difference.
	Schumacher et al. (2001) experiments modeled by threaded cognition
Exp. 1	1 run results (reaction time): *MAPE* = 1.3%, *RMSE* = 7.554, R^2 = 0.998.
Exp. 2	30 run results (reaction time): *MAPE* = 2.3%, *RMSE* = 10.851, R^2 = 0.983.
Exp. 3	3 run results (reaction time): *MAPE* = 4.0%, *RMSE* = 37.742, R^2 = 0.939.

3. QN-ACTR Simulation of Transcription Typing and Reading Comprehension Tasks

The verification of QN-ACTR described in the previous section established the basis for further incorporating unique QN mechanisms into QN-ACTR. In this section, I demonstrate the improved modeling capability of QN-ACTR after adding QN mechanisms. A model was built using QN-ACTR to simulate transcription typing tasks involving dual-task performance and reading comprehension, illustrating the benefits of the integrated cognitive architecture in modeling complex cognition and multi-task scenarios and resolving the concurrent goal scheduling and the module jamming issues in ACT-R. The transcription typing tasks were selected, because (1) previous studies about transcription typing have accumulated numerous detailed empirical results that are very useful for comparing models, (2) the previous QN architecture (i.e., QN-MHP) has modeled many transcription typing phenomena but has difficulty in modeling the phenomena that involve reading comprehension, and (3) skilled transcription typing and related dual-task scenarios may cause the concurrent goal scheduling issue and the module jamming issue in ACT-R and have not been modeled in ACT-R.

3.1 Method

Transcription typing is one of the most common activities in human-computer interaction. It involves complex interaction of a series of perception, cognition, and motor processes each assumed to be in the scale of tens to hundreds milliseconds. A considerable amount of phenomena in transcription typing has been studied and documented, including basic behavioral performance and the effects of skills, typing contents, and concurrent tasks (Gentner, 1983; Inhoff & Wang, 1992; Rayner, 1998; Salthouse, 1984, 1986; Salthouse & Saults, 1987). It has been regarded as one of the major tasks to test cognitive architectures (Newell, 1990). Several qualitative and quantitative models have been developed to model transcription typing, and a recent study using QN-MHP has modeled most of the phenomena in the literature but cannot model the two phenomena that involve reading comprehension (Wu & Liu, 2008b).

The integrated QN-ACTR cognitive architecture can model reading comprehension using ACT-R's declarative memory mechanisms. In ACT-R, sentence memory can be modeled as declarative chunks of syntactic and semantic representations, and reading comprehension can be modeled as memory retrieval and inference/interpretation (J. R. Anderson et al., 2001; Budiu & Anderson, 2004; Lewis & Vasishth, 2005). Also, QN-ACTR can model transcription typing and concurrent task scheduling using QN mechanisms that have been demonstrated in QN-MHP. These mechanisms are not included in ACT-R but are necessary for modeling skilled transcription typing phenomena and dual-task coordination.

First, queueing mechanisms coordinate multi-task performance at the local server level without the need to define any multitask-specific knowledge or any executive process. In QN-ACTR, the goal buffer of the intentional module can hold multiple goal chunks simultaneously representing the concurrence of multiple task components, and a sorted queue is implemented in the production module to coordinate concurrent tasks.

The queue consists of entities each representing a task component and sorts entities based on their waiting time. The waiting time of a task component is initialized as zero at the beginning of the simulation and is reset to zero each time when the task component is processed by the production module executing a production rule. If multiple task components are competing for the limited production module resource, the task component closest to the front of the queue will receive priority. In the motor subnetwork, first-in-first-out queues are implemented in the servers of preparation, initiation, and execution. These servers explicitly represent the three processing stages in ACT-R's motor module, but in ACT-R they have no queue. Without any queue, an ACT-R model would execute a production rule and query the state of the manual module before typing each key to avoid jamming the module (e.g., the Sperling model in ACT-R Tutorial Unit 3). This modeling method has difficulty in modeling fast and skilled typing performance. In QN-ACTR, a production rule can execute a manual action to type in the unit of a word. Motor servers still process one letter at a time, but extra letters in a word can wait in a queue, as a way to model skilled typing performance and avoid the module jamming issue.

Second, parallel processing is created in the motor execution server, allowing parallelism between individual motor effectors. For example, a hand can move simultaneously with another hand or a foot. This parallel processing has not been included in ACT-R but is necessary for modeling the parallelism of motor movements evidenced in transcription typing. For instance, a typist's concurrent task of pressing a foot pedal as soon as they heard a tone did not affect typing performance (Salthouse & Saults, 1987). In addition, successive keystrokes from fingers on alternate hands are faster than successive keystrokes from fingers on the same hand (Wu & Liu, 2008b). The motor servers of QN-ACTR follow the corresponding parallel processing implementation in QN-MHP.

Third, the learning effect on server processing time is mathematically modeled in two motor servers – preparation and initiation, modeling motor learning. For example, one of the typing phenomena shows that repetitive one-finger tapping time decreases with typing skills (Salthouse, 1984). In the QN architecture, the effect of motor skill learning on reaction time is modeled mathematically using an exponential function (Feyen, 2002).

Although the power function was also used to model practice and learning effects (A. M. Anderson, Mirka, Joines, & Kaber, 2009; Newell & Rosenbloom, 1981), research has shown that the exponential function is a better candidate for the law of practice than the power function (Heathcote, Brown, & Mewhort, 2000). As a result, in QN-ACTR, the server processing time (T_i) is modeled as an exponential function of the number of entities (N_i) processed by the server (i),

$$T_i = A_i + B_i \, Exp\,(-\alpha_i N_i)\,. \tag{1}$$

A_i represents the expected minimal processing time after intensive practice; B_i is the change in the expected processing time from the beginning to the end of practice; α_i represents the learning rate. This equation has been used in QN-MHP and successfully modeled the effects of motor skill learning in psychological refractory period experiments (Wu & Liu, 2008a) and transcription typing experiments (Wu & Liu, 2008b). QN-ACTR has implemented this equation in two motor servers to model motor learning. Currently, the four parameters in this equation are the only parameters that have been integrated into QN-ACTR from the QN architecture. The other parameters in QN-ACTR are from the ACT-R architecture. Detailed descriptions are not included here but can be found in the ACT-R reference manual (ACT-R Group, 2011).

The integrated QN-ACTR architecture was tested in the simulation of 29 transcription typing phenomena, particularly the two phenomena involving reading comprehension and a phenomenon involving concurrent tasks. The descriptions and empirical results of these phenomena are summarized in Table 5. Salthouse (1984) tested skilled typists using the Nelson-Denny Reading Test and found that the typing speed in the typing-and-reading condition (58 words-per-minute) was much slower than the reading speed in the reading-only condition (253 words-per-minute). The accuracy of reading comprehension was lower in the typing-and-reading condition (44.7%) compared with the reading-only condition (58.1%). The typing interkey time (177 ms) in the typing-and-reading condition was similar to the time in the typing-only condition (181 ms; Salthouse & Saults, 1987). In another study, the author found that the correlation between typing speed and comprehension scores obtained when typing was not significant

(Salthouse, 1986) and concluded that the typing skill and the comprehension skill are independent. Another important phenomenon is that a concurrent task does not affect typing performance. Salthouse and Saults (1987) added a secondary task in parallel with the primary task of transcription typing. Instructions asked typists to press a foot pedal as soon as they heard a tone signal but prioritize the typing task as the primary task. The results showed that this concurrent auditory-pedal task did not affect typing performance, as the typing interkey time was 185 ms in the concurrent-task condition versus 181 ms in the typing-only condition.

In the model, the task displays and controls were coded using QN-ACTR's task template for discrete and trial-based experiments. The declarative and procedural knowledge for typing was modeled following the Demo model in ACT-R Tutorial Unit 2. Four production rules were defined to model the procedure of typing, including *find-unattended-word*, *attend-word*, *encode-word*, and *type-word*. Queues in the motor servers allow skilled typing performance to be modeled in the unit of a word rather than a letter. Reading comprehension was modeled following similar models in ACT-R (J. R. Anderson et al., 2001). When reading a word in a sentence, the model retrieves the semantic meaning of the word. After the retrieval, the meaning is stored in a slot of a chunk representing sentence semantics. After finishing a sentence, the model memorizes the sentence semantics in declarative memory. Several comprehension questions in the form of propositions are asked after the model finishes reading a passage. For each proposition, the model searches its declarative memory for any chunk encoding the related semantic information. The model can answer the question correctly if the retrieval succeeds, but it cannot answer if the retrieval fails. Since memory limitations such as forgetting are modeled in the sub-symbolic computations of the declarative module, the model can capture comprehension errors caused by forgetting. For the concurrent task phenomena, the secondary auditory-pedal task was modeled following the Sperling task in ACT-R's tutorial. When a tone is presented, the model first detects the sound and then responds by issuing a pedal-pressing action. The goal chunk of this secondary task co-exists with the goal chunk of the primary typing task in the goal buffer. The queueing mechanism in the production module coordinates multiple tasks without the need to define any multitask-specific knowledge or any executive process. Since the primary typing task was stressed by instructions in the original experiment, the typing task was assigned a higher priority than the auditory-pedal task in the queueing mechanism. All related parameters used values from previous studies (Table 6), with all other parameters at their default values. As shown in Equation 1, the parameters A_i, B_i, α_i, and N_i affect how the motor servers' processing time decreases with practice. The other parameters are integrated from the ACT-R architecture, and Table 6 provides a brief description (see ACT-R reference manual for details, ACT-R Group, 2011). The Nelson-Denny Reading Test was used in the simulation as in the empirical experiments.

Table 5. Summary of transcription typing phenomena modeled in QN-ACTR (from Cao & Liu, 2013c).

Phenomena description	Empirical human results	QN-ACTR simulation results
	Phenomena involving complex cognition	
Typing is slower than reading.	Reading speed = 253 words-per-minute (wpm); typing speed = 58 wpm; comprehension accuracy is 58.1% for reading only and 44.7% for reading while typing. Results from 74 typists typing or reading passages with about 1200 characters. (Salthouse, 1984)	Reading speed = 267 wpm (absolute percentage error, *APE* = 5.4%); typing speed = 69 wpm (*APE* = 19.0%); comprehension accuracy is 55.3% (*APE* = 4.9%) for reading only and 44.9% (*APE* = 0.4%) for typing and reading.
Typing skill and comprehension are independent.	No significant correlation (i.e., r = - 0.169, p > 0.15). 74 typists. Passage with about 1200 characters. (Salthouse, 1986)	The correlation r = 0.042, p = 0.80 > 0.15.
	Dual-task phenomena	
A concurrent task does not affect typing performance.	The interkey time in the concurrent task situation (185 ms) was not significantly longer than that in single-task typing (181ms). 40 typists. Passage with about 1250 characters. (Salthouse & Saults, 1987)	Interkey time in the concurrent task situation was 177 ms (*APE* = 4.3%), similar to typing only (172 ms).
	Other phenomena	
Typing is faster than choice reaction time.	Typing interkey time (median) = 177ms for skilled typists; choice reaction time = 560 ms. 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	Typing interkey time (median) = 182 ms (*APE* = 2.8%); choice reaction time = 495 ms (*APE* = 11.7%).
Typing rate is independent of word order.	Qualitative phenomena (Wu & Liu, 2008b)	Typing interkey time = 172 ms for normal order and 172 ms for randomized word order, t (6601) = 0.001, p = 0.999.
Typing speed is slower with random character order.	Interkey time in typing increased to 454 ms when typing materials composed of words with random characters. 5 subjects (3 typists). 220 words. (Hershman & Hillix, 1965)	Typing interkey time = 172 ms for normal order and 373 ms for random character order (*APE* = 17.8%).
Typing rate is impaired by restricted preview.	Typing rate decreases with smaller preview window of the material to be typed. 8 typists. 6 passages each with about 74 words. (Inhoff & Wang, 1992)	R^2 of simulated interkey time is 0.98 (*APE* = 9.2%).
Alternate-hand advantage	Alternate-hand keystrokes are about 45ms faster than the same-hand keystrokes (Wu & Liu, 2008b)	78 ms faster (*APE* = 73.6%).
Digram frequency effect	Digram (letter pairs) that occur more frequently in normal language are typed faster than less frequent digram. 45 typists. Passage with about 1250 characters. (Salthouse, 1984)	Significantly faster, t (197) = -11.062, p < 0.001.
Interkey time is independent of word length.	No significant difference between long and short words. 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	No significance, t (387) = 0.381, p = 0.70.
Word initiation effect	The first keystroke in a word is about 45ms slower than the subsequent keystrokes. 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	53 ms slower (*APE* = 17.1%).

Phenomena description	Empirical human results	QN-ACTR simulation results
Context phenomenon	The time for a keystroke is dependent on the specific context in which the character appears, especially keyboard topography (Wu & Liu, 2008b)	Interkey time for the same key can range from around 60 ms to 200 ms, depending on the context of the previous key.
Copying span	14.6 characters. 29 typists. 8 sentences, each about 75 characters. (Salthouse, 1985)	10.9 characters (*APE* = 25.5%).
Stopping span	2.16 characters. 12 secretaries. 300 sentences, each about 28 characters. (Logan, 1982)	1.8 characters (*APE* = 16.7%).
Eye-hand span	5.25 characters, averaged from multiple studies (for details, see Salthouse, 1986).	6.1 characters (*APE* = 16.0%).
Eye-hand span is smaller for randomly ordered letters.	1.75 characters. 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	1.0 characters (*APE* = 42.9%).
Replacement span	2.8 characters. 85 typists. Passages with about 1200 characters. (Salthouse & Saults, 1987)	4.5 characters (*APE* = 60.7%).
Detection span	8.1 characters. 85 typists. Passages with about 1200 characters. (Salthouse & Saults, 1987)	9.1 characters (*APE* = 12.3%).
Two-hand digrams or two-finger digrams exhibit greater changes with skills than do one-finger digrams.	The slope of the regression equations relating the digram interval (ms) to typing speed of two-hand digrams (-2.08) and two-finger digrams (-2.38) were greater than that of one-finger digrams (-1.38). 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	Two-hand (-6.05) and two-finger (-4.06) are greater than one finger (-2.92 on average).
Repetitive tapping rate increases with skill.	Significant positive correlation between the tapping rate and the net typing speed ($p < 0.01$). 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	$r = 0.81$ ($p < 0.01$).
The variability of interkey time decreases with the skills.	Inter-keystroke variability correlated - 0.69 with the net typing speed; intra-keystroke variability correlated - 0.71 with the net typing speed. 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	Inter-key $r = - 0.85$ ($p < 0.05$); intra-key $r = - 0.90$ ($p < 0.05$)
Eye-hand span is larger with increased skills.	The correlation between the eye-hand span and net words-per-minute was significant with $p < 0.01$. 74 typists. Passage with about 1200 characters. (Salthouse, 1984)	$r = 0.99$ ($p < 0.05$).
Replacement span is larger with more skills.	The correlation between net words per minute and the replacement span was 0.80 ($p < 0.01$). 29 typists. 8 sentences, each about 75 characters. (Salthouse, 1985)	$r = 0.61$ ($p < 0.05$).
Interkey time decreases with practice.	Qualitative phenomena (see Gentner, 1983)	$R^2 = 0.94$ with significant correlation, $p < 0.05$.
Eye gaze duration-per-character decreases with increased preview window size.	(see Figure 2 in Wu & Liu, 2008b).	R^2 of the simulated fixation time is 0.97 (*APE* = 21.6%).
Eye saccade size	4 characters, averaged from multiple studies (for details, see Rayner, 1998).	4.1 characters (*APE* = 2.5%).
Eye fixation duration	400 ms, averaged from multiple studies (for details, see Rayner, 1998).	705 ms (*APE* = 76.3%).

Table 6. Descriptions, values, and sources of parameters used in the transcription typing model. The first four parameters are from the QN-MHP architecture, whereas the other parameters are from the ACT-R architecture (adapted from Cao & Liu, 2012c).

Parameter	Description	Value and source
α_i	Learning rate alpha	0.001 (Heathcote et al., 2000; Wu & Liu, 2008b)
A_i	Expected minimal processing time after intensive practice	21.5 ms (J. R. Anderson & Lebiere, 1998; Card et al., 1983)
B_i	Change of expected processing time from the beginning to the end of practice	50 ms (J. R. Anderson & Lebiere, 1998; Card et al., 1983)
N_i	Total number of digrams that have been processed	15,000,000 for skilled typists (Wu & Liu, 2008b)
:imaginal-delay	Determine the time for the imaginal module to form a chunk of imaginal representation.	0.100 s (Mehlhorn & Marewski, 2011)
:lf	Latency factor for declarative retrieval time. Larger values lead to longer memory retrieval time.	0.003 (Budiu & Anderson, 2004)
:bll	Base level learning parameter for chunk activation. Larger values lead to faster activation decay.	0.3 (Pavlik & Anderson, 2005)
:rt	Retrieval threshold. Set the minimum activation a chunk must have to be able to be retrieved.	-0.704 (Pavlik & Anderson, 2005)
:ans	Set the instantaneous noise added to chunk activation.	0.5 (J. R. Anderson & Matessa, 1997)
:tone-recode-delay	Determine the auditory perception time to recode a tone sound.	0.05 s (Byrne & Anderson, 2001)
saccade duration	Saccade movement duration	20 ms for saccade execution, plus an additional 2 ms for each degree of visual angle (Salvucci, 2001)

3.2 Results

The model simulated various transcription typing tasks and generated performance results that can be compared with the human results. While the model is performing the task, the task visualization feature in QN-ACTR can show the simulated typing behavior in real time, as illustrated in the screenshot of Figure 5. Modeling results include text output traces for module activities, typing performance such as finger movement and eye movement, and reading comprehension performance such as reading speed and comprehension accuracy. The absolute percentage error (*APE*) and the coefficient of determination (R^2) were computed between QN-ACTR's results and the human results. These results are summarized in Table 5. The modeling results were similar to the human results from the experiments. It is particularly important to note that the modeling results captured the phenomena involving reading comprehension (i.e., typing is slower than reading; typing skill and comprehension are independent) that is difficult to model by the QN architecture alone and the phenomena involving concurrent tasks and skilled typing (e.g., a concurrent task does not affect typing performance) that is difficult to model by ACT-R alone. To examine whether an ACT-R model without queues can produce similar results, I repeated the QN-ACTR simulation without the QN mechanisms that were introduced previously in the method section. In this case, the reduced version of QN-ACTR became the same as ACT-R. This simulation not using the QN mechanisms produced a much longer typing interkey time of 500 ms. In comparison, the result from the simulation using the QN mechanisms was just 182 ms, much closer to the human result of 177 ms. Removing the QN mechanisms did not change choice reaction time (still 495 ms).

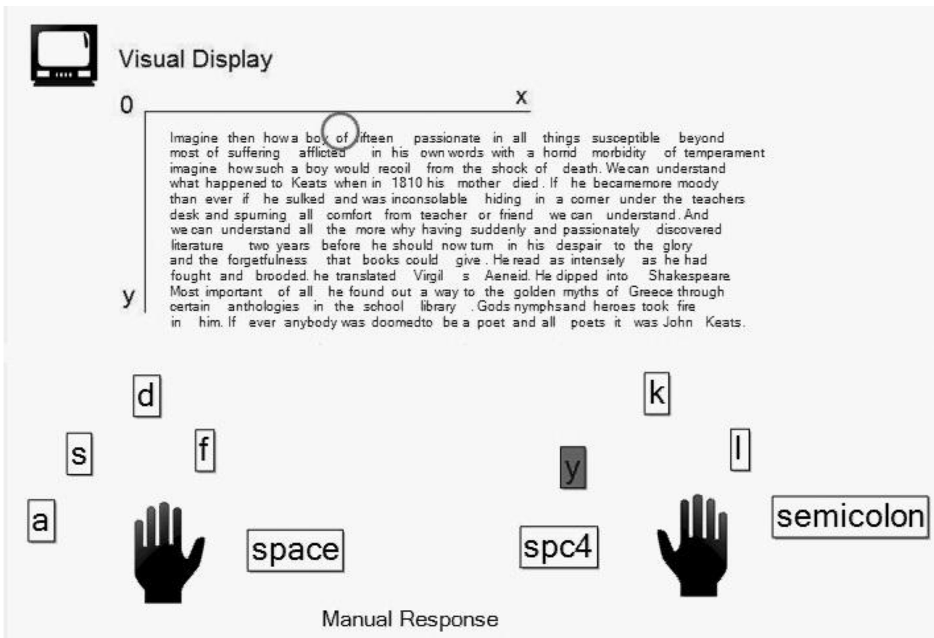


Figure 5. Visualization of the task interaction in QN-ACTR. The visual display section shows the texts on the screen and the location of visual attention (represented by a circle). The manual response section shows that the index finger of the right hand is pressing key "y" while other fingers are resting at the home locations (from Cao & Liu, 2013c).

4. Discussion

QN-ACTR is an integrated cognitive architecture that unifies the QN mathematical architecture and the ACT-R symbolic architecture. Regarding programming and implementation, model verification showed that QN-ACTR has successfully ported ACT-R from Lisp to C#, providing an alternative programming platform for modelers. The translation to C# will support future integration between QN-ACTR and IMPRINT, and the use of the discrete event simulation software Micro Saint® Sharp provides practical and useful features including visualization of entity flows and mental status as well as a build-in tool for parameter optimization. However, since this simulation software works only for Windows systems, future work is needed to port QN-ACTR to other languages such as Java to support cross-platform programming and applications. Continuing along the line of model integration, QN-ACTR combines QN's unique queueing mechanisms and hybrid server network and ACT-R's unique symbolic knowledge representations and sub-symbolic computations. This integration allows QN-ACTR to model a wider range of tasks, especially complex cognitive and multi-task scenarios.

The current study is the initial step of QN-ACTR work. Before this study, it was unclear whether such integration is feasible. I focus on the verification of the integration and the demonstration of the improved modeling capability in the simulation of transcription typing and reading comprehension scenarios. The results show that the integrated QN-ACTR is able to model what have been modeled by ACT-R alone (e.g., paired learning) and QN-MHP alone (e.g., skilled typing). In addition, QN-ACTR can also model concurrent performance involving both typing and reading comprehension, which previous methods have difficulty to model. In this study, reading comprehension was modeled following previous ACT-R models of sentence comprehension, which utilized ACT-R's advantages of the declarative memory and subsymbolic computations. Skilled typing was modeled following previous QN-MHP models, which utilized QN's advantages of queues and hybrid server network.

I also find that without the QN mechanisms, it is difficult for ACT-R mechanisms alone to simulate skilled typing performance. Without any queue in the motor subnetwork, the production module can only send motor typing commands in the unit of a single letter and wait for the completion of the previous typing action before issuing the next motor command. As a result, the typing interkey time became much longer (500 ms) than the human result (177 ms) and was similar to choice reaction time (495 ms for models; 560 ms for human), where responses were made one at a time. This comparison demonstrated the added value of QN-ACTR to human performance modeling and simulation.

The verification results in the current study provide support for future work to examine other novel aspects of the integration, such as the use of queues in the perceptual modules. In addition, future studies can further investigate the cognitive and neurological bases of queues. There may be a connection between short term sensory storages and the queues of the perceptual modules, in the sense that the queues can temporarily store sensory information when the stimuli just disappear and the modules are still busy. The models in the current study did not use the queues in the visual and auditory modules (i.e., the visual and auditory modules worked the same as in ACT-R), but the queues in the motor subnetwork were used in the simulation of typing and reading comprehension to store time-ordered motor commands of finger movement. As in previous QN modeling work, QN-ACTR currently does not apply any constraint to the capacity of queues. The models under this assumption have produced results similar to the human results. Future studies are needed to investigate the capacity of queues and its implication to human performance modeling. In addition, the current study focused on typing time performance and did not simulate typing errors. Future study can continue to test models to capture typing error phenomena.

In conclusion, QN-ACTR is an integrated cognitive architecture and a computerized simulation program that combines the benefits of QN and ACT-R. ACT-R's modules and buffers are implemented as QN servers with their processing logics identical to the corresponding ACT-R algorithms. This QN representation of ACT-R has been verified in the simulation of 20 typical tasks from the ACT-R literature using the same task setups and codes from the original ACT-R models. From the QN perspective, three unique QN positions have also been implemented in QN-ACTR, including the queueing mechanisms to coordinate multi-task performance at the local server level, the hybrid server network to model parallel processing between individual motor effectors, and the mathematical function to model motor skill learning. The benefits of the integration have been demonstrated in the simulation of 29 transcription typing phenomena. In particular, QN-ACTR accounted for both the phenomena involving the complex cognitive activities of reading comprehension and the phenomena involving concurrent tasks and skilled typing, showing the improved modeling capability in complex cognitive and multi-task scenarios that have not been modeled by either QN or ACT-R.

Chapter 3. An Experimental Investigation of Effects of Concurrent Tasks on Diagnostic Decision Making

Chapter Summary

Physicians' decision making performance is one of the most important human factors in healthcare system engineering. Multitasking and interruptions while making diagnostic decisions have been frequently observed in the healthcare work environment. However, little evidence from controlled experiments is available to determine whether physician multitasking affects the quality and timely performance of diagnostic decisions. In the current study, a diagnostic task was designed to examine the effects of concurrent tasks on diagnostic decision making using a controlled laboratory experiment, in which potential confounding factors were controlled to allow the quantification of diagnostic performance and strategies. The results showed that diagnostic performance was negatively affected (in terms of increased decision time) by a complex concurrent memorization task that required participants to listen to verbal updates and remember information about other patients while performing the diagnostic task. In contrast, a simple concurrent sound monitoring task did not affect diagnostic performance. Both types of concurrent tasks significantly increased mental workload. Diagnostic decision strategies were not significantly different between the single- and dual-task conditions. These findings provide new insights into the cognitive mechanisms underlying diagnostic decision and physician multitasking and can serve as the human data for the validation of computational human performance models. Implications for the control and improvement of healthcare quality are discussed.

1. Introduction

Physicians' decision making performance is one of the most important human factors in healthcare system engineering. As the recent prevalence of multitasking and interruptions observed in the healthcare work environment, a growing concern is how the multitasking working style affects decision making performance. Although field observations and cognitive psychology theories suggest that increased mental workload from multitasking may degrade decision making performance, there is still a lack of controlled experimental studies rigorously examining such effects in diagnostic decision making scenarios. To fill this gap, I designed a diagnostic decision making task and conducted an experiment to investigate the effects of concurrent tasks on diagnostic decision making. The results and their implications are reported and discussed in this chapter. These results also provide the human data to test and examine QN-ACTR models, which will be introduced in the next chapter.

Crowding in hospitals, for example in emergency departments (EDs), has been frequently reported in recent years and created a significant challenge to physicians and nurses. The total number of ED visits in the U.S. in 2005 was estimated to be more than 109 million (Owens et al., 2010). Facing the excessive number of patients waiting for care, ED professionals often must perform multiple tasks at the same time. Typically, an ED attending physician's tasks include stabilizing patients, ordering tests, diagnosing diseases, conducting treatment, teaching, and discussing issues with other physicians. In general, multitasking refers to performing two or more tasks simultaneously or switching frequently between them. For example, a physician may be "inserting a central venous line while answering a nurse's question about another patient" (Chisholm, Collison, Nelson, & Cordell, 2000, p. 1240). Observation-based studies have shown that multitasking and interruptions are prevalent in the healthcare work environment (Alvarez & Coiera, 2005; Coiera, Jayasuriya, Hardy, Bannan, & Thorpe, 2002; Collins, Currie, Patel, Bakken, & Cimino, 2007). It has been reported in some cases that an ED physician simultaneously attended to 5.1 patients on average and was interrupted every 9 minutes

(Chisholm et al., 2000; Laxmisan et al., 2007). Frequent interruptions have also been reported in hospital nursing (Tucker & Spear, 2006). The most frequent sources of nurses' work interruptions include nurse colleagues and system or equipment failures (Biron, Lavoie-Tremblay, & Loiselle, 2009). How to properly manage the increased workload caused by multitasking and interruptions has become a major challenge for both healthcare providers and researchers (Carayon et al., 2011).

The goal of the current study is to increase the knowledge base on healthcare human factors by examining the effects of concurrent tasks on diagnostic decision making using a controlled experiment. I examined three important aspects of human factors in diagnostic decision making including strategy, performance, and mental workload. Strategy may vary between different decision makers or for the same person in different task conditions. Strategy affects decision making performance, and performance is often measured by both the time used to reach a decision and the correctness of the decision. Both decision time and correctness need to be considered because one measure alone is not sufficient to examine performance due to the existence of speed-accuracy tradeoff (Wickens, Lee, Liu, & Becker, 2004). Decision correctness is often quantitatively measured by the rate of diagnostic errors, where an error can be operationally defined as

> "a diagnosis that was unintentionally delayed (sufficient information was available earlier), wrong (another diagnosis was made before the correct one), or missed (no diagnosis was ever made), as judged from the eventual appreciation of more definitive information." (Graber, Franklin, & Gordon, 2005, p. 1493)

Since the use of the term "human error" has been criticized to be vague in terms of denoting both causes and results of human actions (Hollnagel, 2007), it is important to note that the above operational definition of diagnostic error focuses on the undesired results of an erroneous action without implying anything about the cause, as in the definition of erroneous action by Hollnagel,

> "an erroneous action is an action which fails to produce the expected result and which may lead to unwanted consequences." (Hollnagel, 1993, p. 2)

Finally, mental workload is another important measure different from performance. Performance is about work accomplished utilizing mental resources, whereas mental workload is about the relative amount of resources utilized in mental processing.

Diagnostic decision strategies are commonly categorized into two groups – analytical strategies and heuristics – based on the widely adopted dual-process (or dual-system) model of decision making (Evans, 2008; Kahneman, 2003; Norman, 2009; Reyna, 2008; Shiffrin & Schneider, 1977; Sloman, 1996). The model proposed two distinct cognitive processes that are often labeled as System 1 and System 2. System 1 represents the automatic process that is intuitive, fast, parallel, and heuristic, whereas System 2 represents the controlled process that is deliberate, slow, serial, and analytic. Skill development theories assumed that learning is a progress from the controlled to the automatic process (J. R. Anderson, 1982; Fitts, 1964). Novice medical students characteristically use the deliberate and controlled analytical strategies, examining a large number of hypotheses before making a diagnostic decision. Experienced physicians, in contrast, tend to use heuristics or short-cuts such as learned patterns of symptoms to reduce the number of hypotheses and generate diagnoses with less effort (Elstein & Schwarz, 2002). Heuristics may speed up decision making but do not guarantee an optimal diagnosis, because heuristics reflect personal experience, rules-of-thumb, and assumptions that do not necessarily follow probability rules or have statistical significance. Although many researchers have emphasized the failure of heuristics (Brewer, Chapman, Schwartz, & Bergus, 2007; Kovacs & Croskerry, 1999; Wallsten, 1981), it is important to note that heuristics can be both erroneous and useful in diagnostic decision making (Eva, Hatala, LeBlanc, & Brooks, 2007; Kulatunga-Moruzi, Brooks, & Norman, 2001; Norman, 2009). For example, the tendency to test hypotheses that are expected to have the property of interest rather than those expected to lack the property can be very useful, thus labeled as a positive test strategy; the same tendency, however, can lead to systematic errors, thus labeled as confirmation bias (Klayman & Ha, 1987).

Cognitive factors have been identified to be able to affect decision-making strategies. Some of these factors, including aging, experience, stress, and time pressure, have been found to be able to shift strategies toward more use of automatic and heuristic thinking (Keinan, 1987; Patel, Gutnik, Karlin, & Pusic, 2008; Peters, Diefenbach, Hess, & Västfjäll, 2008; Wright, 1974). But few studies have been conducted to examine the effects of concurrent tasks. A concurrent task increases the overall task demands and therefore may increase perceived stress and time pressure. I hypothesized that a concurrent task would affect the selection of strategies similar to these cognitive factors such as time pressure. When decision makers realize the increased task load, they may choose to use more heuristics that can help them reach a diagnosis faster.

In addition to the potential effect on diagnostic strategies, concurrent tasks may also have effects on diagnostic performance (Burgess, 2010). Human cognitive system has limited working memory and attentional resources for controlled processing (Fougnie, 2008; Wickens, 2008). When two tasks competing for the same resources are performed concurrently, the performance of one or both tasks is likely to suffer. Such dual-task interference has been found in numerous experiments in the fields of cognitive psychology and human factors (Engström, Johansson, & Östlund, 2005; Lamble, Kauranen, Laakso, & Summala, 1999; McKnight & McKnight, 1993; Pashler, 1994; Welford, 1952; Wu & Liu, 2008a). For example, texting messages is very difficult to be performed simultaneously while driving without impairing driving performance, because the two tasks both require significant amounts of visual attentional resources (Drews, Yazdani, Godfrey, Cooper, & Strayer, 2009). In contrast, a concurrent speech comprehension task has little effect on the primary driving task, because the two tasks use different resources from the visual and auditory channels (Cao & Liu, 2013a). In other cases when two tasks significantly compete for the same processing resources such as working memory, people have to switch between the tasks and focus on one task at a time. Task switching has also been found to reduce performance, because reconfiguration of task-related information in the mind takes extra time and also suffers from interference (Altmann & Gray, 2008; Monsell, 2003). Studies have shown that practice can significantly reduce dual-task interference (Hazeltine, Teague, & Ivry, 2002; Oberauer & Kliegl, 2004; Ruthruff, Johnston, & Van Selst, 2001). An explanation is that after practice, the mental processes become more automatic and require less mental resources, thus reducing the conflict between concurrent tasks. Strong negative interference exists between two controlled processes but not between a controlled and an automatic process (Schneider & Shiffrin, 1977).

Previous studies of dual-task performance mostly focused on perceptual-motor, working memory, speech generation, and speech comprehension tasks. Only a few studies have examined decision making performance in a dual-task paradigm. These studies found that multitasking or interruptions had detrimental effects on decision performance (Gonzalez, 2005; Speier, Valacich, & Vessey, 1999; Speier, Vessey, & Valacich, 2003). But it is difficult to tell which mental mechanism causes the interference, that is, whether the conflict lies in the visual perception process or the central cognitive process. Since both tasks in these dual-task experiments used visual stimuli, the existence of central cognitive conflict cannot be clearly tested. Moreover, these previous studies did not examine any diagnostic decision making task. In the present study, I focused on the examination of cognitive interference involving diagnostic decision making. The two concurrent tasks in the dual-task paradigm were separately presented in the visual and auditory channels, avoiding perceptual interference. Based on the theories about automatic and controlled processes described above, I hypothesized that a complex concurrent task would degrade diagnostic performance, but a simple task would not. A simple task refers to one that requires little controlled processing and can be easily learned to become automatic. In contrast, a complex task refers to one that requires significant amounts of controlled processing, working memory, and attentional resources.

Finally, I hypothesized that concurrent tasks would increase mental workload experienced by decision makers. Mental workload is often operationally defined as the ratio of task demands to the capacity of human information processing (Hancock & Meshkati, 1988; Xie & Salvendy, 2000). Performing multiple tasks concurrently increases task demands, so it should increase mental workload by its definition. It is important to measure and control mental workload, because if task demands exceed the cognitive capacity, a situation referred to as mental overload, some tasks will be delayed or processed with reduced quality. Prolonged work under high levels of mental workload also accelerates the accumulation of fatigue and adversely affects subsequent performance. Mental workload measures include performance-based measures, physiological measures, and subjective ratings (Wierwille & Eggemeier, 1993). Among these measures, subjective assessment measures are often favored by many researchers because of their demonstrated sensitivity to demand manipulations, low intrusion to the primary task, and convenience to use. One of the most widely used mental workload measures is the NASA-Task Load Index (NASA-TLX; Hart & Staveland, 1988). It assesses mental workload on six scales: mental demand, physical demand, temporal demand, performance, effort, and frustration level. The overall mental workload can then be quantified as the weighted average score over all these scales. NASA-TLX has been shown to be sensitive to the change of task demands in a wide range of tasks, from laboratory cognitive judgment tasks to real-world driving and aviation tasks (Cao & Liu, 2013a; Colle & Reid, 1998; Hart, 2006; Lee & Liu, 2003; Wu & Liu, 2007). A recent study has shown that NASA-TLX is also reliable and valid to measure mental workload in the healthcare domain (Hoonakker et al., 2011). The current experiment used the raw NASA-TLX measure, which simplifies the rating procedure by eliminating the weighting process and using the sum of ratings from the six scales as the estimation of overall workload (Byers, Bittner, & Hill, 1989; Hart, 2006; Hoonakker et al., 2011).

In summary, the current experiment examined the effects of concurrent tasks on diagnostic decision strategies, performance, and mental workload. I also expected the dependence of these effects on the type of the concurrent task – a simple task (automatic process) or a complex task (controlled process). Since the goal of this study is to examine the general mechanism of diagnostic multitasking independent of domain-specific knowledge, I used a controlled experiment with abstract diagnostic tests. Using laboratory tasks has the advantages of providing control of potential confounding factors such as individual knowledge and experience and allowing the quantification of decision strategies and performance that previous field observation studies cannot offer.

2. Methods

2.1 Participants

Thirty adults (20 males and 10 females, mean age = 21.4 years, standard deviation of age = 2.3 years), all of whom were students or recent graduates recruited from a university in the United States, were paid for their participation. They were informed that if their performance was among the top 20% of all the participants of this experiment, they could receive an extra monetary bonus. Six of the 30 participants received the bonus after all of them completed the experiment.

2.2 Tasks and materials

The experiment was computerized as a Personal Computer (PC) program using C# and had five task conditions – three single-task conditions and two dual-task conditions. The three single-tasks included an abstract diagnostic decision making task (visual), a sound monitoring task (auditory), and a memorization task (auditory). In each trial of the decision task, participants needed to diagnose what kind of disease a simulated patient had. Patients' identification number and gender were visually displayed for each trial. Each patient must have one of the eight diseases represented by eight numbers (1-4 and 6-9), as shown in Figure 6. Participants reported their diagnostic decision by clicking a number using a computer mouse. Three diagnostic tests were provided to facilitate their decision. The Small-Large test examined whether the true disease number was in the range of the smaller (1, 2, 3, and 4) or larger (6, 7, 8, and 9) half. The Odd-Even test examined whether the true disease number was odd (1, 3, 7, and 9) or even (2, 4, 6, and 8). The Blue-Red test examined whether the true disease number was in color blue (1, 2, 6, and 7) or red (3, 4, 8, and 9). For example, if the true disease number is 6, the results of the three tests would be Large, Even, and Blue. Only one test result could be displayed at a time, and there was a brief delay (four seconds) between issuing a test and the appearance of its result. Tests could be performed in any order or repeated. A decision could be made with or without the completion of all three tests. The three tests are an abstract representation of a physician asking three questions, for example, whether the patient has a high or low blood pressure, a positive or negative fungal infection test result, and high or low white blood cell count.

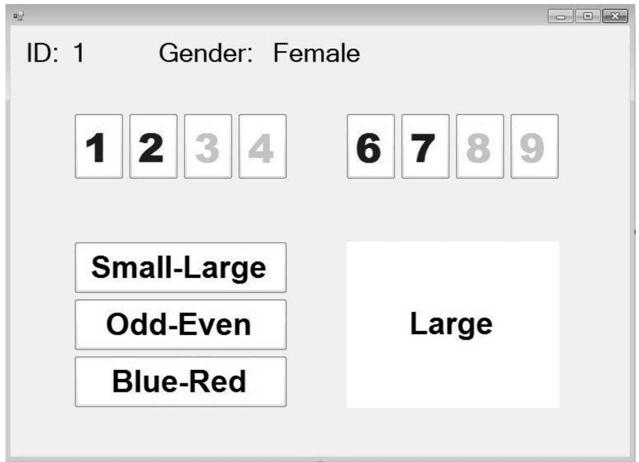


Figure 6. Display of the diagnostic decision making task. The three tests each reveal a property of the number representing the true disease. Number 1, 2, 6, and 7 are blue. Number 3, 4, 8, and 9 are red (from Cao & Liu, in press).

One of the benefits of using an abstract diagnostic task is to rigorously examine the effects of multitasking while minimizing the effects of potential confounding factors such as domain-specific knowledge and experience. If real disease names and real diagnostic tests were used, the difference in familiarity and personal experience with these diseases and tests would become very difficult to control and confound the interpretation of the results, because heuristics depend heavily on individual experience that may vary significantly from person to person. Another benefit of using an abstract diagnostic task is to allow the quantification of decision strategies and performance. In contrast, diagnostic strategies and performance in real-world healthcare settings often cannot be quantified easily due to the lack of a consensus on what constitutes best practice and the lack of definitive methods to identify the true causes of symptoms.

In the abstract diagnostic task used in this study, strong links between patients' gender and the true disease were embedded in the program but not explicitly disclosed to the participants. It was designed that 90% of male patients all had one typical disease (e.g., Number 7), and 90% of female patients all shared a different typical disease (e.g., Number 8). For the remaining 10% of patients in either gender group, the probability of the true disease was uniformly distributed among the other seven atypical diseases. The actual numbers representing typical diseases were randomly selected for each participant. This design created two types of strategies. First, the controlled and analytic strategy was to complete all three tests, gradually eliminate alternatives, remember the remaining choices, and finally reach a diagnosis. On the other hand, participants might learn the strong links hidden between the gender and the true disease and form their heuristics. The resulting automatic and heuristic strategy was to take a short-cut and directly choose a diagnosis based on personal experience without completing all three tests. Instructions told participants that their top priority was to make their diagnosis as accurate as possible. Secondly, they should diagnose as many patients as possible within the limited time period. After they gave their diagnostic decision for a trial, the next trial would start shortly in three seconds.

The sound monitoring task required participants to monitor the vital signs of a simulated patient in an intensive care unit by listening to an auditory display. There were two types of sounds – Normal and Target. Both of them resemble warning signals from patient monitoring systems. The Normal sound was composed of consecutive beeps that resemble the beeping sound from a cardiac monitor. The Target sound was a long and flat tone that is often used to indicate a stopped pulse. Participants were asked to respond correctly as fast as possible by pressing the space bar on a keyboard when a target was presented and not to press it when no target was presented. Pressing the key during Normal sound was recorded as an incorrect response. After a correct response, which must be made during Target sound, the sound would turn back to Normal. The interval before the onset of the next Target was uniformly distributed between 4 and 14 seconds.

The memorization task required participants to remember the emergency levels of three simulated patients named Alpha, Bravo, and Charlie in an intensive care unit. The emergency levels included the three levels of Low, Medium, and High. All patients' status started at the Medium level and would be constantly updated verbally through a speaker in every 4 to 14 seconds. This task represented the verbal communication between physicians and nurses. Speech used in the experiment was synthesized from texts using Microsoft Speech Platform. For example, participants might hear: "Update. The emergency level of Patient Charlie has changed to High." After a random number (one to three, uniformly distributed) of status updates, a question would be asked about a patient's status. For example, participants might hear: "Question. Is the emergency level of Patient Bravo at the Low level?" Patient names and their status in both updates and questions were randomly selected. Participants were asked to respond correctly as fast as possible by pressing keys (d and f) as their answers for yes and no (key order balanced across subjects).

The two dual-task conditions were formed by combining the decision task with each of the two auditory tasks. In the decision-and-monitoring condition, the diagnostic decision task (visual) and the sound monitoring task (auditory) were presented simultaneously. In the decision-and-memorization condition, the diagnostic decision task (visual) and the patient status memorization task were presented simultaneously. The instruction for dual-tasks was to treat both tasks as equally important.

2.3 Design and measures

The experiment used a within-subject design. Each participant experienced all five conditions in an order balanced across subjects. The independent variable was task condition. For the diagnostic decision task, the number of tests issued in a trial (min = 0 and max = 3) was recorded. If all three tests were consulted before a decision, the strategy was categorized as analytic. If less than three tests were used, the strategy was categorized as heuristic. The rate of using the analytic strategy was used as a dependent variable. Correct decision/reaction rate and decision/reaction time were recorded for the decision task as well as the monitoring and memorization tasks. Mental workload was measured by raw NASA-TLX for each experimental condition. The raw NASA-TLX overall score was examined as a dependent variable. Statistical analyses using repeated measures ANOVA and paired t-test were conducted in SPSS.

2.4 Procedure

Participants completed a consent form and a demographic questionnaire before the experiment. They first adjusted the sound volume to a comfortable level and practiced each single- and dual-task condition following instructions displayed on a computer screen. To pass the practice, they must reach a correct decision/reaction rate of no less than 0.8 for each task. The practice section took approximately 35 minutes. During the practice, immediate feedback was given to each participant response, telling the participants whether their response was correct or wrong. Such immediate feedback has also been used in real-world healthcare training. For example, new methods such as interactive simulated training can give immediate feedback on trainees' diagnostic performance (Olsen & Sticha, 2005). In real-world healthcare practice, there is also feedback but delayed feedback. For example, when an earlier diagnosis did not lead to successful treatment, a later re-evaluation might correct the earlier misdiagnosis and serve as feedback to a doctor.

Each participant practiced a total of at least 60 trials of the diagnostic decision making task. After the practice, participants completed the formal experiment section for the five task conditions in a balanced order. No feedback was provided during the formal experiment section. Among the decision making trials in both practice and formal sections, 90% of the trials were typical cases where patients' gender could be used as the indicator of the true disease, and the remaining 10% were atypical cases where the true disease was randomly distributed among the seven diseases other than the typical one. The practice section with feedback provided all participants with the same experience that the link between gender and the true disease was not perfectly reliable. They were able to experience both successes and failures for using the gender as a short-cut of diagnosis. After each task condition in the formal section, participants completed a raw NASA-TLX mental workload assessment for the task condition. Short breaks were given between task conditions, and the formal experiment section took approximately 45 minutes. Incentives were used to motivate high performance in both single- and dual-task conditions. Participants were informed that they could win an extra bonus if their performance was ranked among the top 20% of all participants.

3. Results

3.1 Effects on strategy

Concurrent tasks had no significant effect on the strategy of diagnostic decision in this experiment, as shown in Table 7, in terms of neither significant difference in the rate of using the analytic strategy ($F(2, 58) = 1.225$, $p = 0.301$, $\eta^2 = 0.040$) nor significant difference in the number of tests used ($F(2, 58) = 2.452$, $p = 0.095$, $\eta^2 = 0.078$). On average over all trials from all participants, 69% of the diagnostic strategies were categorized as the analytic strategy, and 2.48 of the three diagnostic tests were issued before making a decision. Thirteen participants (43% of all participants) always completed all three diagnostic tests before making their decisions (i.e., categorized as the analytic strategy group); the rest 17 participants used only the heuristic strategy or a mix of the analytic and heuristic strategies (i.e., categorized as the heuristic strategy group).

Table 7. Diagnostic decision results in different task conditions (from Cao & Liu, in press).

Mean ± Standard deviation	**Decision-only**	**Decision and monitoring**	**Decision and memorization**	***p***	$\boldsymbol{\eta^2}$
Analytic strategy rate	0.68 ± 0.39	0.67 ± 0.40	0.72 ± 0.40	0.301	0.040
Number of tests used	2.45 ± 0.80	2.43 ± 0.87	2.55 ± 0.77	0.095	0.078
Decision time (s)	12.98 ± 3.89	12.98 ± 4.26	16.12 ± 5.28	**< 0.001**	0.510
Correct rate (ranging from 0 to 1)	0.96 ± 0.05	0.96 ± 0.06	0.97 ± 0.04	0.123	0.079

3.2 Effects on performance

As shown in Table 7, the effect of concurrent tasks on diagnostic decision time was significant ($F(1.428, 41.398) = 30.222$, $p < 0.001$, $\eta^2 = 0.510$; using the Greenhouse-Geisser correction $\hat{\varepsilon} = 0.714$, because Mauchly's Test showed that the Sphericity assumption was violated, $p = 0.001$). Pairwise comparisons showed that decision time was significantly longer by about 3.1 s in the decision-memorization dual-task condition than both the decision-only and decision-monitoring conditions (all p values < 0.001). Decision times were not significantly different between the decision-only and decision-monitoring conditions ($p = 1.000$). Concurrent tasks showed no significant effect on the rate of correct diagnostic decision ($F(2, 58) = 2.170$, $p = 0.123$, $\eta^2 = 0.079$).

Regarding the performance of the monitoring and memorization tasks, paired t-test revealed significant effects of task condition, showing impaired performance in the dual-task conditions. For the monitoring task, reaction time increased from 0.57 s in the single-task condition to 0.65 s in the dual-task condition ($t(29) = 8.616$, $p < 0.001$, $d = 0.907$), but the correct response rates were not significantly different between the single- (0.99) and the dual-task conditions (0.98; $t(29) = 1.507$, $p = 0.143$, $d = 0.277$). For the memorization task, reaction time significantly increased from 5.04 s in the single-task condition to 5.38 s in the dual-task condition ($t(29) = 5.379$, $p < 0.001$, $d = 0.793$), whereas the correct response rate decreased from 0.97 in the single-task condition to 0.93 in the dual-task condition ($t(29) = 2.337$, $p = 0.027$, $d = 0.634$).

3.3 Effects on mental workload

As shown in Figure 7, the effect of task condition on mental workload (measured by the raw NASA-TLX overall rating) was significant ($F(4, 116) = 58.037$, $p < 0.001$, $\eta^2 = 0.667$). Pairwise comparisons showed that mental workload was significantly different for all the condition pairs (all p values < 0.05) except for the pair of the decision-only and monitoring-only conditions ($p = 1.000$). When comparing between the analytic strategy group (13 participants) and the heuristic strategy group (17 participants; defined previously in Section 3.1), no significant difference was found for mental workload in any of the five task conditions (t-test p values ≥ 0.248).

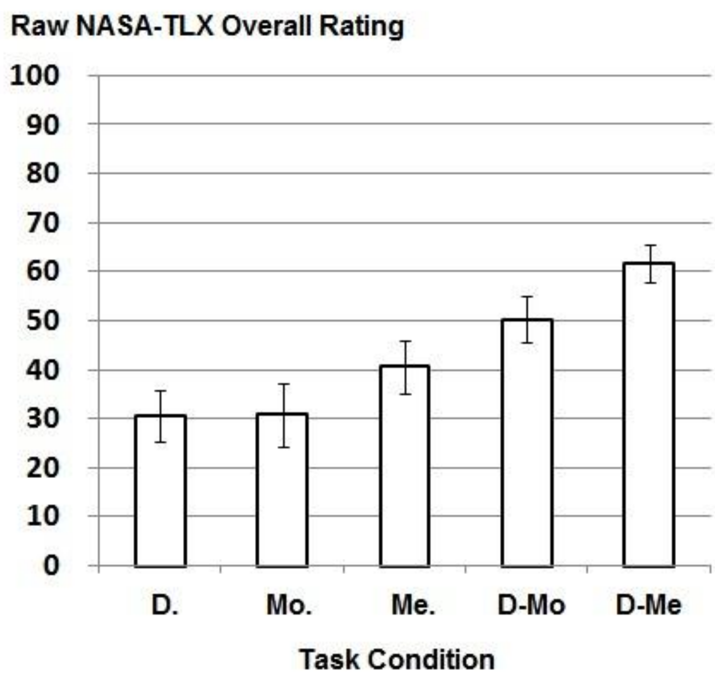


Figure 7. Raw NASA-TLX mental workload overall rating in different task conditions. D. = Decision-only; Mo. = Monitoring-only; Me. = Memorization-only; D-Mo = Decision-monitoring dual-task; D-Me = Decision-memorization dual-task. Error bars represent 95% confidence intervals (from Cao & Liu, in press).

4. Discussion

The quality and timely performance of diagnostic decisions is a critical factor in healthcare delivery. The prevalence of multitasking and interruptions observed in the healthcare work environment has raised concerns about the potential detrimental effects of the multitasking working style. Although dual-task effects on human performance have been extensively studied in cognitive psychology and human factors, there is a lack of controlled experimental studies rigorously examining such effects in diagnostic decision making scenarios. In the current study, I designed a diagnostic decision making task in order to control potential confounding factors such as previous knowledge about diseases and allow the quantification of decision strategies and performance. The effects of performing concurrent tasks on diagnostic decision making were examined in a controlled laboratory setting. There were two different types of tasks to be paired with the decision task in the dual-task paradigm. The sound monitoring task represented simple tasks that mainly use automatic processing, whereas the memorization task represented complex tasks that mainly use controlled processing.

This experiment generated several important results about the characteristics of diagnostic decision making and concurrent task performance. First, the experiment results confirmed the existence of a strong tendency of decision makers to rely on heuristics in diagnostic decision making. The instructions emphasized that the top priority should be given to diagnostic accuracy rather than speed. Since there was a 10% chance that the heuristics would fail, as the participants learned in the practice, the normative and rational strategy would be to follow the standard procedures and complete all three tests before making a decision. However, only 69% of the diagnoses were made by following this analytic and controlled strategy. This finding indicates that participants' reliance on heuristics in diagnostic decision is a strong disposition inherent in the human cognitive system.

Second, diagnostic strategies did not significantly differ between the single- and dual-task conditions. This result is different from our expectation. It indicates that concurrent tasks may affect diagnostic decision in a way different from stress and time pressure that have been found to be able to change decision strategies, as introduced previously in the introduction section. It might also be possible that participants in this experiment did not have the need to change strategies, because the relatively short duration of the experiment (about 80 minutes) was not very fatiguing or the relatively simple diagnostic task has only a limited number of factors to consider.

Third, the concurrent memorization task significantly reduced diagnostic performance, but the concurrent sound monitoring task did not. Since diagnostic strategies did not significantly differ between conditions, the concurrent task was likely to be the only factor that was identified to account for the reduced diagnostic performance. These results are in line with our hypothesis and provide evidence supporting the two-system theory from the decision making and cognitive psychology literature (e.g., Kahneman, 2003). Both the memorization task and the diagnostic decision task required controlled processing. Performing both tasks at the same time created the competition for the limited working memory and attentional resources (Wickens, 2008). The detrimental effects were indicated by the increased decision time, increased memorization response time, and reduced memorization correctness. On the contrary, the simple sound monitoring task that required little controlled processing did not affect diagnostic performance. This finding may provide some guidance to the design of multitasking and interruption policies in the medical care work environment. When a physician is actively focusing on a time-critical diagnostic decision, other information such as warnings, reminders, or situation updates should be delivered through automatic processing as much as possible. The results from this study show that verbal communication regarding other patients' information while decision making will lead to decreased performance in both tasks. Determining the best way and the best time to interrupt is difficult, especially when the communication is not face-to-face but mediated via machine interfaces such as broadcasts, pagers, and mobile phones. Medical professionals and interface designers should be aware of the cognitive mechanisms of multitasking interference.

Finally, performing a concurrent task significantly increased mental workload measured by raw NASA-TLX; no matter whether the concurrent task was a simple or a complex one. The highest level of mental workload among all task conditions came from the decision-memorization dual-task, which was about twice as high as the mental workload in the decision-only condition. It is interesting to note that the concurrent sound monitoring task also significantly increased mental workload, although it did not affect diagnostic performance. Increased mental workload may aggravate the accumulation of fatigue and reduce performance in prolonged work. This finding reveals the potential latent effects of multitasking on diagnostic decision making, which should also be considered for the control and improvement of healthcare quality. In addition, no significant difference was found for mental workload between the analytic strategy group and the heuristic strategy group of participants. A reason may be that the diagnostic decision task in the current study did not use fixed trial duration. Instead, after a decision trial was finished, the next trial would start shortly in three seconds. Each task condition had a fixed total duration, so participants who completed each trial faster would experience more trials. Future studies may use fixed trial duration to test the hypothesis that participants using the heuristic strategy and completing trials faster may experience lower mental workload.

A limitation of this study is that the abstract diagnostic decision task is simplified for a laboratory experiment setting, whereas medical diagnostic decision making in the real-world is much more complex due to, for example, a large number of possible diseases, uncertainty in diagnostic tests, and the requirement for complex cognitive and perceptual skills. Some diagnostic tasks, such as interpreting an x-ray image, require perceptual pattern recognition skills that were not examined in the current study. However, the abstraction and simplification used in this experiment is necessary to rigorously control potential confounding factors such as domain-specific knowledge and experience that vary significantly from person to person and allow the quantification of decision strategies and performance. Another limitation of this study is that the participants were not medical professionals and were generally younger than doctors. The participants' lack of medical-domain-specific knowledge was not a concern in this experiment because of the use of an abstract diagnostic task, but there may be some problem-solving skills that are special in healthcare personnel, and there is also a possibility that an elder population may have different multitasking capabilities. Based on the findings of the current study, future studies are needed to further investigate the effects of concurrent tasks on diagnostic decision in medical professionals from a specific field of practice using domain-specific materials from the field.

5. Conclusions

This study examined the effects of concurrent tasks on diagnostic decision making using a controlled laboratory experiment. The results showed that the diagnostic strategies used by the participants did not significantly differ between the single- and dual-task conditions. Diagnostic performance was negatively affected by a complex memorization task that required participants to listen to verbal updates and remember the emergency levels of three other patients concurrently while performing the diagnostic decision task. In contrast, a simple sound monitoring task did not affect diagnostic performance. Both types of concurrent tasks significantly increased mental workload measured by raw NASA-TLX. These findings provide new insights into the cognitive mechanisms underlying diagnostic decisions and physician multitasking, which should be considered in the design of multitasking and interruption policies for the control and improvement of healthcare quality. Future studies can further investigate the effects of concurrent tasks on diagnostic decision in medical professionals from a specific field of practice using domain-specific materials. The next chapter will introduce QN-ACTR computational models built to model and simulate the human results collected from this experiment.

Chapter 4. Modeling Cognitive Multitasking Performance in Diagnostic Decision Making Tasks

Chapter Summary

An important question in the research field of human performance modeling is how to model concurrent task performance and mental workload involving complex cognitive activities. Particularly in the healthcare domain, there is a strong need for computational methods to model and simulate diagnostic decision making performance and the effects of concurrent tasks. In this study, QN-ACTR computational models were built to simulate concurrent-task performance involving multiple controlled processes in need of dedicated cognitive resources, which previous methods have difficulties to model. The key concept is a filtering discipline implemented in QN-ACTR cognitive architecture that allows cognitive resources to be exclusively occupied by one of the concurrent tasks when necessary, instead of switching between the tasks as frequently as possible. In the simulations of dual-tasks involving diagnostic decision making and patient status tracking, I found that the new discipline seems to be necessary to model human performance and mental workload. Implications and practical applications of QN-ACTR in system evaluation and design are discussed.

1. Introduction

An important and challenging topic of human performance modeling is the modeling of concurrent multi-tasks involving complex cognitive activities. Multi-task models must have mechanisms explaining how limited mental resources are scheduled and utilized to meet multi-task demands. In this chapter, I describe a new filtering discipline from the QN theory that schedules mental resources in complex cognitive dual-tasks. I have found that this mechanism is needed to model human results in concurrent tasks involving multiple controlled processes that existing methods have difficulties to model. Controlled processes refer to deliberate mental processing that is different from automatic processing. As described in the dual-process (or dual-system) model (Kahneman, 2003), there are two distinct cognitive processes—System 1 and System 2. System 1 represents the automatic process that is intuitive, fast, parallel, and heuristic, whereas System 2 represents the controlled process that is deliberate, slow, serial, and analytic. Since automatic processing requires very few mental resources, dual-tasks that consist of two automatic tasks, or an automatic task paired with a controlled task, are less likely to interfere. However, strong negative interference may exist between two controlled processes.

Using queueing mechanisms or similar concepts, previous studies have modeled performance in dual-tasks that consist of two automatic tasks or an automatic and a controlled task, including the psychological refractory period (Wu & Liu, 2008a), driving and map reading (Liu et al., 2006), and tracking and choice tasks (Salvucci & Taatgen, 2008). From the queueing network (QN) perspective, multiple goals can co-exist in the goal buffer, and multi-task performance emerges as the behavior of multiple streams of information flowing through a network, with no need for multitask-specific goals to interleave production rules into a serial program or for an executive process to interactively control task processes (Liu, 1997; Liu et al., 2006). However, few studies have modeled dual-tasks that both require controlled cognitive processing. To fill this gap, an approach is to combine the QN's benefit in modeling multitasking performance and the benefit of Adaptive Control of Thought-Rational (ACT-R, J. R. Anderson et al., 2004) in modeling complex cognitive activities, as introduced in previous chapters.

In QN-ACTR, the goal buffer can hold multiple entities simultaneously representing multiple concurrent tasks, which are coordinated by a queue sorting these tasks based on their waiting time. If multiple tasks are competing for limited mental resources, the task that is the closest to the front of the queue (i.e., longest waiting time) will receive priority. This simple queueing mechanism leads to alternation between tasks whenever it is possible to switch tasks. When multiple production rules for different tasks are matched at the same time, the production module will choose the rule that processes the task different from the last processed one. This mechanism has been shown to be appropriate for dual-tasks with at least one automatic component, but as described in the modeling work reported in this chapter, it has difficulty to model controlled dual-tasks in which both tasks require a significant amount of cognitive processing (e.g., imaginal and declarative processing).

Controlled cognitive processing usually requires a series of production rules to be executed as a continuous stream without interruption. For instance, in a single-digit addition task (ACT-R tutorial, http://act-r.psy.cmu.edu/), the model needs to retrieve the meaning of numbers, form a representation in the imaginal buffer, and manipulate the representation until an answer is found. If interruption happens, for example, the declarative retrieval result is harvested by another concurrent task, or the problem representation is changed to a different representation for another concurrent task, the controlled processing of the interrupted task will fail to reach its goal. As a result, the simple sorted queue scheduling method is not suitable to model dual-tasks that both require controlled processing because the frequent switching between tasks will interrupt the continuous flow of a controlled process.

To address this interruption issue and improve the modeling capability of QN-ACTR, I introduce a filtering discipline that allows cognitive resources to be exclusively occupied by one of the concurrent tasks when necessary. First, production rules are categorized into two groups—the ones that need follow-up processing and the ones do not. For example, if a production rule's action part has a retrieval request to fetch a chunk from the declarative memory and put the chunk in the retrieval buffer, it requires follow-up processing carried out by another rule to use the retrieved chunk for the same task. Then, if a rule that needs follow-up is executed, the next selected rule must process the same task as the previous rule. This filtering discipline is implemented in QN-ACTR's production module. After a rule requiring follow-up is processed, the production module will start to exclusively accepting only the rules that follow up the same task. If there is no such rule matched and available, the module will be enforced to idle and ignore other rules. Such exclusive processing continues until a rule that does not require follow-up is processed. Later in this chapter, I will show that models without this new discipline can only model dual-tasks with at least one automatic component but cannot model dual-tasks both involving controlled processing. In contrast, models with the new discipline can model all the scenarios.

How to determine whether a rule requires follow-up? Based on simulation results, I have identified three conditions. A rule may be categorized as requiring follow-up, if at least one of the following three conditions is met. First, the rule's action part has a declarative retrieval request. Second, the rule's action part has an imaginal request, i.e., to create a chunk (problem state) in the imaginal buffer. Third, the rule processes aural information from a continuous stream of important audio stimuli, such as a question sentence. All these three conditions have been found to be necessary to model human performance in the simulation work of this study.

This filtering discipline described above can help model dual-task performance where both tasks compete for the problem state (imaginal buffer) and the declarative (memory retrieval) resources. Each task can obtain a period of exclusive use of mental resources to complete its controlled cognitive process, avoiding endless task switching that happens when the simple queueing is the only scheduling discipline.

The following sections describe a simulation using QN-ACTR to model diagnostic decision making dual-task scenarios. I first briefly describe the experiment conducted to collect human data and then describe the modeling results of both performance and mental workload. Implications and applications from the results are discussed.

2. Method

The human experiment – diagnostic decision making and concurrent tasks – was described in the previous chapter. This section focuses on the introduction of the modeling method. Since there were two types of diagnostic decision strategies, two models that each corresponded to a type of strategy were built in QN-ACTR to simulate the human results. The analytic model used all three tests before making a decision. The heuristic model followed a common short-cut strategy observed from the human results, that is, after two tests, heuristically choosing a diagnosis based on the gender information. Production rules were defined for each task individually without using any executive control rule. In dual-task scenarios, a model simply used the rules combined from the two sets of rules for the two single-tasks. These rules, as shown in Figure 8, followed the model building principles and conventions in previous ACT-R and QN-ACTR studies. All parameters were in their default values, except for *:lf* (latency factor) that was set to 0.106 (J. R. Anderson & Matessa, 1997). Mental workload was modeled by overall server utilization, which has been shown to have a linear relationship to NASA-TLX (Cao & Liu, 2011b). The previously described filtering discipline and the follow-up principles were implemented in QN-ACTR. The production rules that required follow-up are marked in Table 8. I also tested model performance without this new discipline in comparison.

Table 8. Procedures and production rules for the diagnostic decision, sound monitoring, and patient status tracking tasks. The definition of these production rules follows the principles used in previous cognitive models (e.g., see models from http://act-r.psy.cmu.edu/). Source: (Cao & Liu, 2013b).

Step	Task procedure	Production rules
	Diagnostic decision: Analytic strategy (i.e., using all tests)	
1	At the start of a new trial, create a problem state that all diseases are possible.	start-trial
2	Find the visual location of gender; move the mouse cursor to the first test button as preparation.	find-patient-gender
3	Visually attend the gender information.	attend-gender
4	Visually encode the gender information.	encode-gender
5	Decide to issue the next test.	issue-test-small-large(odd-even, or blue-red)
6	Find the visual location of the next test button.	decide-test-small-large(odd-even, or blue-red)
7	Move the mouse curser to the button.	move-mouse
8	Click the button.	click-mouse
9	Find the visual location of the test result.	find-test-result / find-test-result-again (if failed)
10	Visually attend the test result location (may fail if no word has been shown yet).	attend-test-result
11	See the word “wait.”	pre-encode-test-result-wait
12	In case this is the last test of the three, move the mouse curser closer to the answer buttons as preparation.	prepare-mouse-final-answer-find & prepare-mouse-final-answer-move
13	In case the problem state in the imaginal buffer is empty or not for this task (e.g., caused by interruption from another concurrent task), retrieve the problem state from declarative memory.	decision-attending-test-retrieval-imaginal*
14	When retrieved, recreate the problem state in the imaginal buffer.	decision-attending-test-recreate-imaginal*
15	Visually encode the test result; update the problem state by eliminating unsupported diseases.	encode-test-result-large(small, even, odd, red, or blue)
16	If there is still any remaining test to do, visually attend the next test button. (i.e., back to Step 10.)	attend-test-result
17	If this is the last step, notice that all tests have been done.	all-test-done
18	In case the problem state is empty or not for this task (e.g., caused by interruption from another concurrent task), retrieve the problem state from declarative memory.	decision-decide-number-retrieval-imaginal*
19	When retrieved, recreate the problem state in the imaginal buffer.	decision-decide-number-recreate-imaginal*

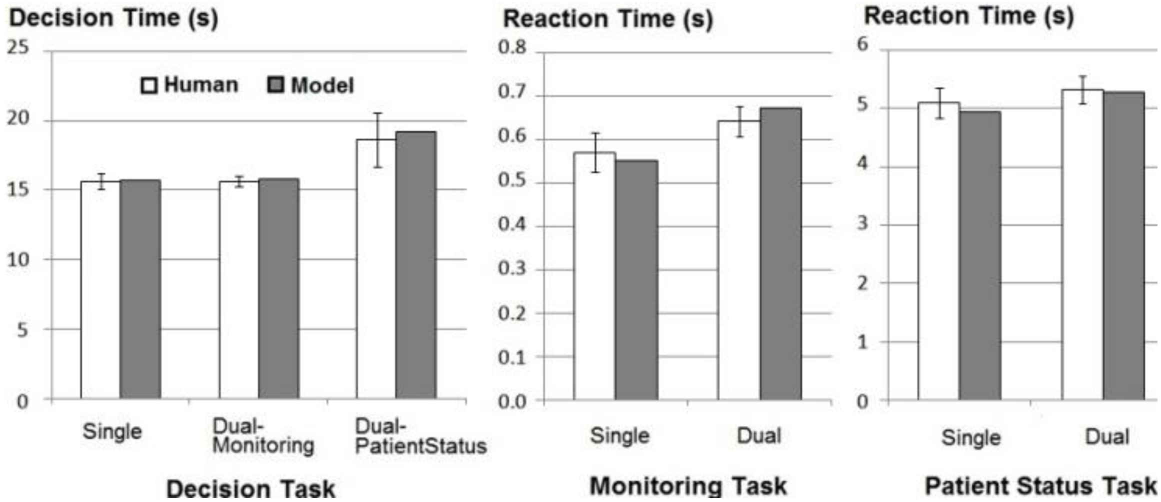
20	Make the decision based on the problem state (with only one possible disease remaining); find the visual location of the decision number button.	decide-number-1 / decide-number-2 / … / decide-number-8
21	Move the mouse cursor to the button. (same rule as Step 7)	move-mouse
22	Click the button. (same rule as Step 8)	click-mouse
23	Finish a trial; restart the goal for the next trial.	trial-done
	Diagnostic decision: Additional procedure and rules for heuristic strategies	
A1	Heuristic decision. After completing the first two tests, heuristically diagnose based on the gender information.	issue-test-blue-red-competing-short-cut-male / issue-test- blue-red -competing-short-cut-female
	Sound monitoring	
1	Whenever a sound is detected, start processing it in the aural module.	detected-tone
2	Hear a target sound	encode-target-tone
3	Manually press the key as a response.	respond-target-tone
	Patient status tracking (memorization)	
1	Create the initial problem state that all patients have a medium emergency level.	memory-pre-start
2	Whenever perceptual, imaginal, and retrieval modules are free and buffers are empty, rehearse the problem state; retrieve the state from the declarative memory.	memory-rehearse-memory*
3	Rehearse the problem state; put the retrieved state back to the declarative module.	memory-rehearse-memory-return
4	Whenever a sound is detected, start processing it in the aural module.	memory-hear-sound*
5	Start hearing the update of patient status.	memory-hear-update
6	Hear the name of the patient.	memory-hear-update-patient*
7	In case the problem state in the imaginal buffer is empty or not for this task, retrieve the problem state from declarative memory.	memory-hear-update-retrieval-imaginal*
8	When retrieved, recreate the problem state in the imaginal buffer.	memory-hear-update-recreate-imaginal*
9	Hear the updated level for a patient; update the problem state accordingly.	memory-hear-update-a(b, or c)-level
10	Start hearing the question of patient status.	memory-hear-question*
11	Hear the name of the patient.	memory-hear-question-patient*
12	In case the problem state in the imaginal buffer is empty or not for this task, retrieve the problem state from declarative memory.	memory-hear-question-retrieval-imaginal*
13	When retrieved, recreate the problem state in the imaginal buffer.	memory-hear-question-recreate-imaginal*
14	If the patient status is same as the one in the problem state, answer yes; if not the same, answer no.	memory-hear-question-a(b, or c)-level-yes(or no)
15	Press the corresponding key as the answer.	memory-press-key-yes / memory-press-key-no

*: marks the production rules require follow-up.

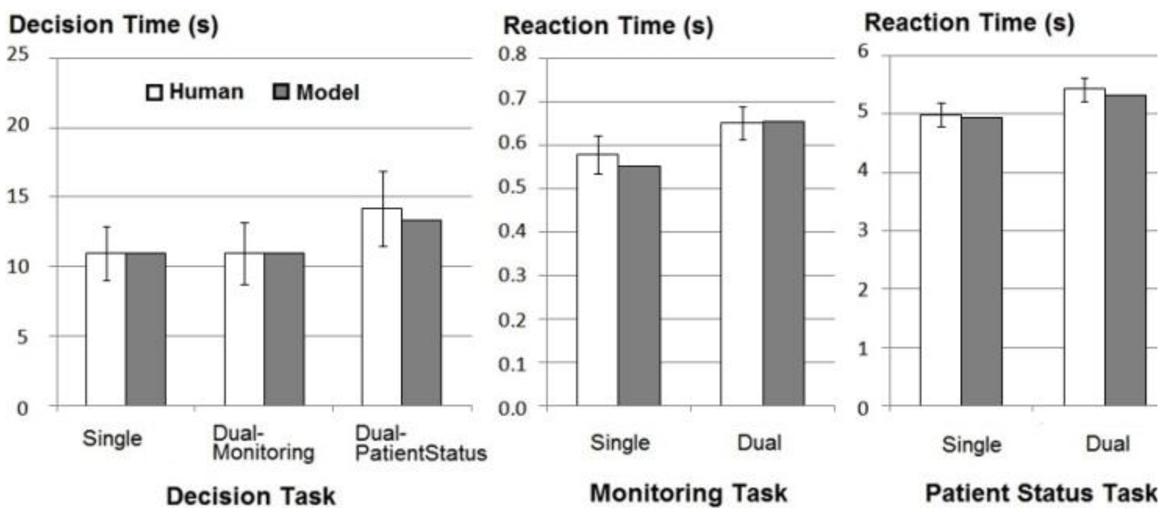
3. Results

The human results showed that the effect of concurrent tasks on diagnostic decision time was significant ($F(1.428, 41.398) = 30.222$, $p < 0.001$, $\eta^2 = 0.510$). Decision time was significantly longer in the decision-memorization (i.e., the decision and patient status tracking) dual-task condition than other conditions (ps < 0.001). The monitoring and memorization tasks also had impaired performance in terms of time delay in the dual-task conditions. On average, 69% of the diagnoses were made following the analytic strategy (i.e., use all three tests). For participants used heuristic short-cuts, 2.1 of the three tests, on average, were issued before making a decision. The effect of task condition on mental workload (measured by the raw NASA-TLX overall rating) was significant ($F(4, 116) = 58.037$, $p < 0.001$, $\eta^2 = 0.667$). Pairwise comparisons showed that mental workload was significantly different between all condition pairs (ps < 0.05) except for the pair of decision-only and monitoring-only (p = 1.000).

The simulation was repeated six times, reaching a 95% confidence interval smaller than 5% for each measurement. The model results were similar to the human results for both types of strategies, as shown in Figure 8. For the analytic strategy, *mean absolute percentage error* (*MAPE*) = 2.31%, *root mean square error* (*RMSE*) = 0.24 s. For the heuristic strategy, *MAPE* = 2.01%, *RMSE* = 0.32 s. In contrast, models without the filtering discipline could only simulate the decision-monitoring task but could not complete the decision-memorization task, because the controlled processes were interrupted by task switching. All three follow-up principles were necessary. Without the retrieval request principle, models would stop in the middle of simulation, because the retrieval buffer was filled with an inappropriate chunk for the other task that prevented further processing. Without the imaginal request principle, the two tasks would endlessly compete and recreate the imaginal buffer chunk and could not complete any task. Without the auditory stream principle, models can only perform the decision task but would stop performing the memorization task, because they missed important auditory information.



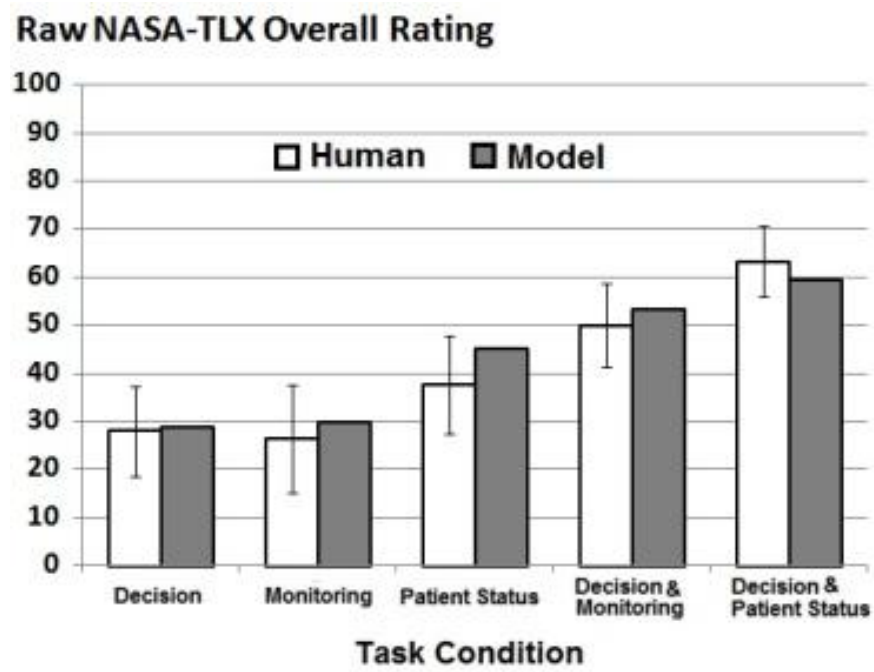
(a) Analytic strategy (use all three tests)



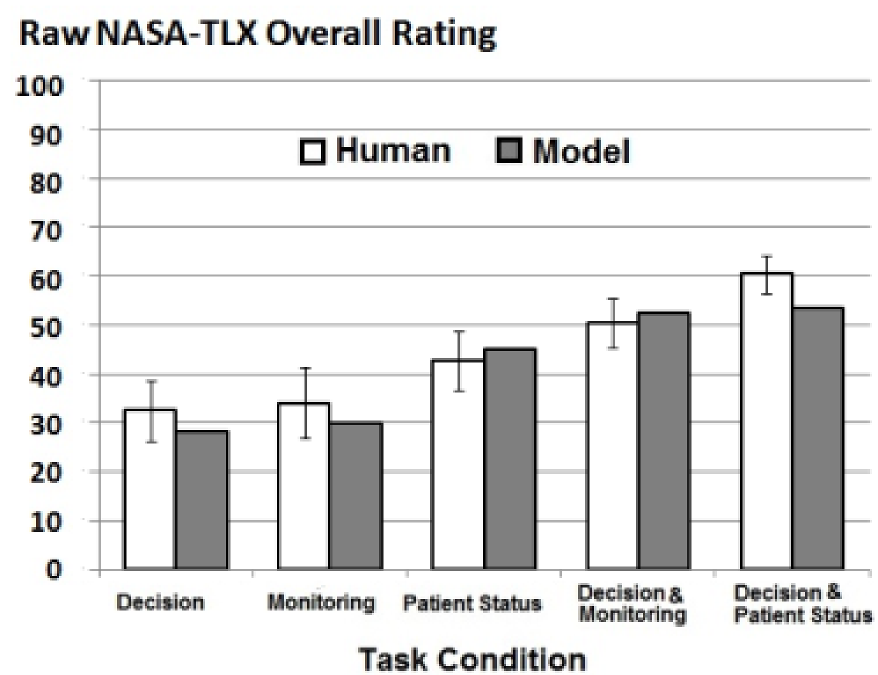
(b) Heuristic strategy (use two tests and short-cut)

Figure 8. Effects of concurrent tasks on the decision/reaction time performance of each task component for both the human and model results. Error bars represent 95% confidence intervals (from Cao & Liu, 2013b).

Using overall utilization, the simulation also captured the difference of mental workload between task conditions, as shown in Figure 9, *MAPE* = 9.5%, β = 0.938 (p < 0.001), *intercept* = 6.3, s*lope* = 437.1, R^2 = 0.879 (p < 0.001).



(a) Analytic strategy (use all three tests)



(b) Heuristic strategy (use two tests and short-cut)

Figure 9. Effects of concurrent tasks on mental workload for both the human and model results. Error bars represent 95% confidence intervals (from Cao & Liu, 2013b).

4. Discussion

The modeling results from this study showed that the proposed filtering discipline implemented in QN-ACTR is necessary to successfully model human performance and mental workload in concurrent tasks involving two or more controlled processes. This filtering discipline allows cognitive resources to be exclusively utilized by one of the concurrent tasks when necessary, resolving the interruption issue that exists when the simple queueing is the only scheduling mechanism. Currently, three conditions have been identified regarding which production rules require follow-up processing. Since multi-task modeling is still a new topic in human performance modeling, more studies are needed to examine whether these mechanisms are sufficient and whether additional mechanisms are needed to model a wider range of cognitive multi-task scenarios.

A limitation of the present study is that only the overall NASA-TLX rating is modeled but not each individual dimension of NASA-TLX scores. As previously introduced, NASA-TLX has six dimensions of workload rating scores. In particular, the effort dimension is described as how hard a person has to work (mentally and physically) to accomplish the level of performance. This effort issue is currently not included in the QN-ACTR architecture, but intuitively, working harder may be modeled as assigning more processing resources to a task. Future research may test the use of a mind-wandering concurrent process (i.e., task-unrelated thought) to model effort. Hypothetically, the mind-wandering concurrent process, as an extra task not required by formal task instructions, may compete with other concurrent tasks but does not create mental workload related to any required task. Higher effort may reduce the proportion of mind-wandering and therefore increase resource utilization and mental workload of the required tasks.

Modeling methods capable of simulating cognitive multi-task scenarios are valuable for the design, evaluation, and improvement of complex human-machine systems, such as healthcare, transportation, and plant operation control. These complex and concurrent tasks are characterized by having complicated dynamics and multiple components that require controlled processing. Human performance modeling using QN-ACTR can support the selection of design alternatives and the examination of what-if scenarios. The mental workload modeling capability of QN-ACTR can also support the development of workload management systems that help control mental workload within a suitable range to improve system performance and efficiency, while controlling fatigue that can be caused by prolonged work.

Chapter 5. An Experimental Investigation of Concurrent Processing of Vehicle Lane Keeping and Speech Comprehension Tasks

Chapter Summary

To further examine QN-ACTR's modeling capability in complex multi-task scenarios, the work reported in this chapter collected detailed human performance and mental workload results in a driving and speech comprehension dual-task scenario. Then the next chapter will introduce the QN-ACTR model to simulate these human results.

With the growing prevalence of using in-vehicle devices and mobile devices while driving, a major concern is their impact on driving performance and safety. However, the effects of cognitive load such as conversation on driving performance are still controversial and not well understood. In this study, an experiment was conducted to investigate the concurrent performance of vehicle lane keeping and speech comprehension tasks with improved experimental control of the confounding factors identified in previous studies. The results showed that the standard deviation of lane position (*SDLP*) was increased when the driving speed was faster (0.30 m at 36 km/h; 0.36 m at 72 km/h). The concurrent comprehension task had no significant effect on *SDLP* (0.34 m on average) or the standard deviation of steering wheel angle (*SDSWA*; 5.20 degree on average). The correct rate of the comprehension task was reduced in the dual-task condition (from 93.4% to 91.3%) compared with the comprehension single-task condition. Mental workload was significantly higher in the dual-task condition compared with the single-task conditions. Implications for driving safety were discussed.

1. Introduction

The use of in-vehicle information systems and mobile devices has increased rapidly over the past few decades. For a long time, drivers' interaction with in-vehicle systems has been limited to radio and air-condition controls by pressing buttons and turning knobs. Then drivers started to use cell phones for telephone conversation while driving. With the growing prevalence of mobile devices or smartphones in recent years, drivers are surrounded by more distractions than ever before. A major concern about the use of in-vehicle devices and mobile devices is their impact on driving performance and safety. The work reported in this paper is an experimental investigation of the impact of concurrent speech comprehension on vehicle lane keeping performance using improved experimental control of the confounding factors identified in previous studies.

Using in-vehicle or mobile devices while driving can create two types of load, including visual load and cognitive load. Visual load is produced by tasks that require drivers to move their visual attention away from the driving scene, for example, text messaging. Such visual tasks compete with driving for the limited visual attention resource. The effects of concurrent visual tasks on driving performance have been relatively well-established. Since driving requires continuous visual processing, it is not surprising that a visual task almost always degrades driving performance to some extent. Numerous studies have found negative effects of visual load, including increased reaction time and decreased correct rate in response to traffic events (Lamble et al., 1999; McKnight & McKnight, 1993), increased lateral position variation (Engström et al., 2005), and degraded car following performance (Drews et al., 2009). With converging evidence, 39 states in the U.S. have banned text messaging for all drivers up till November 2012.

Compared with visual load, the effects of cognitive load on driving performance are still controversial and not well understood. Cognitive load in this research field often refers to the mental demand from a concurrent auditory task, such as voice control and telephone conversation. As predicted by multiple resource theory (Wickens, 2008), a secondary task using the auditory channel (e.g., speech conversation) should have less interference with the primary task using the visual channel (e.g., driving), compared with a secondary task that also uses the visual channel (e.g., text messaging). Still, epidemiological surveys of traffic accidents have found association between increased cell phone calls and higher risk of accidents (Redelmeier & Tibshirani, 1997; Violanti & Marshall, 1996). To examine the effects of cognitive load, numerous experiments have been conducted in both simulated and real-world driving scenarios. The results of traffic event reaction performance showed mostly negative effects of cognitive load, but the results of speed control (i.e., longitudinal control) and lane keeping (i.e., lateral control) are mixed and inconclusive.

Effects of cognitive load on event reaction performance. Most studies have shown that cognitive load degrades event reaction performance. Recarte and Nunes (2003) found that conversations, no matter by phone or with a passenger, impaired visual detection of flashing targets in the driving scene. McKnight and McKnight (1993) asked their subjects to respond to video-recorded traffic situations and found that cell phone conversations significantly reduced the number of vehicle control responses. In addition, both abstract mental tasks (Alm & Nilsson, 1995; Lamble et al., 1999) and naturalistic conversations (Strayer, Drews, & Johnston, 2003) performed over the phone have led to increased brake reaction time to the braking maneuver from the lead vehicle. There are a few studies that failed to find any significant effect. In these cases, the phone task was often less demanding, requiring only passive listening without the need for immediate action (Recarte & Nunes, 2003; Strayer & Johnston, 2001). Overall, as suggested by Horrey and Wickens (2006) in a meta-analysis study, the negative effects of cell phone conversations on traffic event reaction performance are significant and relatively well-established.

Effects of cognitive load on speed control performance. Few studies have examined whether conversations affect speed control performance. Rakauskas, Gugerty, and Ward (2004) found that conversations caused larger variations in both accelerator pedal position and driving speed (i.e., degraded performance). Similarly, Kubose et al. (2006) found more variable velocity in both concurrent speech production and comprehension conditions compared with driving only. However, a recent study found the opposite effect – that is, drivers exhibited smaller variability in velocity (i.e., improved performance) when driving with concurrent speech tasks (Becic et al., 2010).

Regarding average driving speed, some studies found that conversations led to slower driving speed (Rakauskas et al., 2004), but other studies found no significant effect (Engström et al., 2005). These mixed results indicate the existence of confounding factors such as the strategic tradeoff between driving and conversation tasks. A better understanding of this issue requires further experiments with these confounding factors controlled, which will be discussed in more detail later in this paper.

Effects of cognitive load on lane keeping performance. Another important component of driving is to steer the vehicle and maintain lane position. In contrast to the discrete nature of traffic event reaction, lane keeping is a continuous task that requires uninterrupted visual-manual control. The cognitive load involved in lane keeping may be high for novice drivers but can decrease with the development of driving skills (Groeger, 2000). After fully mastering the skills, experienced drivers may perform lane keeping automatically with very little conscious control and attention. One may then expect little or no effect of conversations on lane keeping performance; however, it is difficult to draw any conclusion from existing empirical findings, which are mixed with seemingly contradictory results. There have been studies showing that concurrent cognitive load improved lane keeping performance (Becic et al., 2010; Brookhuis, de Vries, & de Waard, 1991; Engström et al., 2005; Kubose et al., 2006; Liang & Lee, 2010), degraded lane keeping performance (Just, Keller, & Cynkar, 2008; Strayer & Johnston, 2001), or had no significant effect (Alm & Nilsson, 1995; Kubose et al., 2006; Rakauskas et al., 2004). With a closer look, I have identified several confounding or uncontrolled factors (as summarized in Table 9) that may offer explanations to these contradictory results. A confounding factor is a variable that is not included in an experimental design but may vary systematically between different experimental conditions and affect a dependent variable.

Table 9. Comparison of studies investigating effects of concurrent vehicle lane keeping and cognitive tasks (from Cao & Liu, 2013a).

Study	**Driving task component**	**Lane keeping difficulty**	**Cognitive task**	**Priority instruction**	**Incentive**	**Cognitive performance***	**Lane keeping performance****	***SDLP* (m)****	***SDSWA* (degree)****	**Mental workload****
This study	Lane keeping only (steering)	Zigzag curves, fixed-base simulator (f-sim.) (36 and 72 km/h)	Speech comprehension	Safe driving first	Used	Examined	Examined	Examined	Examined	NASA-TLX Examined
(Brookhuis et al., 1991)	Multiple	Driving calmly on a quiet motorway, field (95 km/h)	Paced serial addition	-	-	-	Improved	From about 0.21 to 0.20	No sig. About 1.6	Physiological and subjective. Increased
(Engström et al., 2005)	Multiple	Motorway, f-sim., m-sim., & field (110 km/h)	Auditory continuous memory	-	-	-	Improved (f-sim. & m-sim.); No sig. (field)	From about 0.30 to 0.25 (f-sim.); 0.23 to 0.20 (m-sim.); No sig. (field)	No sig.	Physiological. No sig.
(Becic et al., 2010)	Multiple	Straight road, f-sim. (48 km/h)	Story retelling and recall	-	-	Degraded. Accuracy from 68% to 62%	Improved	From 0.16 to 0.10	-	-
(Liang & Lee, 2010)	Lane keeping only (steering)	Straight road with continuous external disturbance force, f-sim. (72 km/h)	Speech comprehension	-	-	-	Improved	From about 0.24 to 0.22	-	-
(Kubose et al., 2006)	Multiple	Straight road with random side wind, f-sim. (89 km/h)	Speech production; Speech comprehension	Safe driving first	-	No sig. Accuracy 88% (production), 86% (compreh.)	Improved (production); No sig. (compreh.)	From 0.38 to 0.35 (production); No sig. 0.35 (compreh.)	-	-
(Alm & Nilsson, 1995)	Multiple	Rather straight and easy-to-follow road, m-sim. (90 km/h)	Working memory span test with sentence judgement	-	-	-	No sig.	-	-	NASA-RawTLX Increased
(Rakauskas et al., 2004)	Multiple	Circle, rural road, f-sim. (72 km/h)	Speech production	-	-	-	No sig.	-	No sig. About 2.1	Rating Scale of Mental Effort. Increased
(Strayer & Johnston, 2001)	Lane keeping only (joystick tracking)	Unpredictable sin wave movement	Word generation	-	-	-	Degraded	-	-	-
(Just et al., 2008)	Lane keeping only (mouse or trackball)	A curving virtual road	Speech comprehension	Attend equally to both tasks	-	-	Degraded	-	-	-

-: not reported
*: how cognitive performance is affected by the introduction of the concurrent lane keeping task
**: how it is affected by the introduction of the concurrent cognitive task
SDLP: Standard deviation of lane position
SDSWA: Standard deviation of steering wheel angle
f-sim.: fixed-base simulator; m-sim.: motion-base simulator; field: real-world field driving

The first potential confounding factor is the strategic tradeoff between different tasks. Driving is a task that has multiple components by itself, including traffic event reaction, speed control, and lane keeping. It is possible that drivers choose different strategies and allocate attention resources differently among these components in different driving scenarios. As shown in Table 9, most of the existing experimental designs used driving tasks with multiple components, and the potential strategic tradeoffs were not controlled. A method to control such tradeoffs is to confine driving to a single-task of lane keeping only, while vehicle speed is automatically control like in cruise control modes. In addition, the potential strategic tradeoff between the driving task and the phone task also needs to be controlled, because the strategy about which task should take priority may reasonably affect the performance of each task. However, most of the existing experiments did not report the instructions regarding the assignment of priority and did not examine the performance of the speech task, both of which are necessary for improved experimental control.

The second factor is lane keeping difficulty, which is determined by both driving speed and the type of roads used in an experiment. If a road is straight and easy to follow, as in some of the previous experiments, a lane keeping task may not require frequent steering corrections, and therefore its performance may become insensitive to (i.e., not affected by) a concurrent conversation task. When a lane keeping task is very easy and the resulting mental workload is very low, the performance may also be low because of the lack of excitement and motivation (White, 1959), which may explain why lane keeping performance was found to be improved by a concurrent task in some of the previous studies. To examine drivers' performance capability, the lane keeping task in the current study needs to be sufficiently difficult. The difficulty level also needs to be consistent between the driving-only and dual-task conditions, because otherwise the change in lane keeping performance may be due to the change in lane keeping difficulty rather than the interference from the concurrent cognitive task. This requires vehicle speed to be controlled, because slower speed simplifies the lane keeping task, while faster speed makes it more difficult.

The third factor is the effort to actively process the cognitive task. Some of the previous studies did not report cognitive task performance. To properly evaluate the effect of speech comprehension on lane keeping, an experiment needs to show sufficiently high performance of the comprehension task in the dual-task condition, in order to ensure that drivers are indeed actively engaged in speech comprehension. Another issue is the type of cognitive tasks. There are mainly three types of cognitive tasks used in the previous studies: numerical calculation, speech production, and speech comprehension, which may involve different brain mechanisms. Studies have suggested that numerical calculation and speech or language skills rely on different neural bases (Gelman & Butterworth, 2005). Brain imaging results have also shown that language production and comprehension involve different brain regions (Price, 1998). These different mechanisms may not interact with the concurrent lane keeping task in the same way, which may be a cause of the contradictory results in the previous studies. Experiments are needed to examine these different types of tasks separately.

The fourth factor is motivation. Several previous studies found a counter-intuitive result: lane keeping performance was improved by a concurrent cognitive task. As discussed by Becic et al. (2010), an explanation of these results may be the lack of motivation in the driving-only condition, especially when the task difficulty was low and no incentive was used to promote high performance. A concurrent cognitive task may act as an excitement to increase drivers' motivation and effort in the dual-task condition, thereby improving performance. To examine drivers' multi-task capability, incentives should be used to promote best performance in both single and dual-task conditions.

The goal of the present study is to carefully investigate the effects of concurrent vehicle lane keeping and cognitive tasks with improved experimental control of the confounding factors previously described. Table 9 compares the experimental control in this study with the previous studies. In the current experiment, the strategic tradeoff between different driving components was controlled by confining driving to a single task of lane keeping only, while driving speed was automatically kept constant. Instructions requiring the subjects to give priority to safe driving (i.e., lane keeping) were explicitly given in order to control the tradeoff between the two concurrent tasks. The lane keeping task was designed to be sufficiently difficult and challenging by using zigzag curves that require frequent steering corrections. Driving speed was kept the same between the single and dual-task conditions so that it could not affect the difficulty of lane keeping. Among different types of cognitive tasks, this experiment focused on speech comprehension, because it is one of the most common tasks involved in the interaction with in-vehicle and mobile devices and a more natural task in contrast to numerical calculation. Knowledge about the effect of speech comprehension on lane keeping can help designers develop safety counter-measures to control the amount and level of in-vehicle speech interaction, for example, by controlling when and how information should be verbalized to drivers. Finally, this experiment used monetary incentives to promote best performance in order to investigate drivers' multi-task performance capability rather than satisfactory performance.

The specific research question of the present study is:

What are the dual-task effects of concurrent vehicle lane keeping and speech comprehension tasks?

These effects include the dual-task effects on lane keeping performance, comprehension performance, and mental workload. In recent years, as the prevalence of cruise control and adaptive cruise control technologies (Jenness, Lerner, Mazor, Osberg, & Tefft, 2008), driving on highways and rural roads has often been reduced to a single task of lane keeping. Since workload is reduced, a question is whether drivers could or should perform a concurrent task while driving. Drivers have the desire to improve productivity while commuting. For example, a student may want to listen to a recorded lecture while driving. An office worker may want to check emails through a speech interface while driving. An investor may want to listen to news about the stock market while driving. The results of this study can help further develop the empirical data and knowledge base in answering this question. The following sections describe the experiment, report the results, and discuss the new insights that can be drawn from the results.

2. Method

2.1. Participants

Twenty-four participants (17 males, 7 females, mean age = 29.6 years, standard deviation of age = 6.7 years), all of whom were native Mandarin Chinese speakers recruited in China, were paid for their participation. They all had a valid driver license, at least 8000 km of driving mileage (mean driving mileage = 44800 km), and normal or corrected-to-normal visual and auditory acuities.

2.2. Tasks and materials

The simulated driving environment was developed with Animator3D simulation in Micro Saint® Sharp (www.maad.com). Simulation was performed on a PC computer with a 24” LCD monitor and a force-feedback steering wheel (fixed-base simulation). In order to confine the driving task to lane keeping only, driving speed was automatically controlled at constant, and no mirror or speedometer was displayed. The simulated road had two lanes in the same direction, and the driver’s car started in the center of the right lane. Each lane was occupied by a program-controlled car driving in front of the driver’s car at the same constant speed. The road had abundant left and right curves with different curvatures so that frequent steering corrections were necessary to maintain lane position. As shown in Figure 10, the road has the following sequence of segments: 100 m straight, three left curves of 80 m each (with radii of 458 m, 229 m, and 458 m respectively), 20 m straight, three right curves of 80 m each (with the same radii of the three left curves), and repeated correspondingly. The task was to steer the car and maintain as close to the center of the lane as possible.

Figure 10. Section of the road map used in this experiment (from Cao & Liu, 2013a).

The speech comprehension task and its sentence materials were developed following a previous study (J. R. Anderson, 1974). The task required participants to listen to a series of sentence pairs and judge whether the two sentences in a pair (called the input sentence and the probe sentence) had the same meaning by pressing buttons on the steering wheel. A previous study has shown that the motions of pressing buttons on the steering wheel did not affect the performance of lane keeping (Kubose et al., 2006). Each sentence contained a subject noun, a verb, and an object noun. Depending on the voice of the two sentences, either active or passive, and the order of the nouns, there were eight types of combinations of input and probe sentences. Table 7 illustrates these combinations and the correct response for each combination. The materials were presented to the participants in Mandarin Chinese, and the English translations are as follows. The verbs included *follow*, *block*, *pass*, and *hit*. The nouns included *bus*, *car*, *van*, *SUV*, *truck*, *bicycle*, *motorbike*, *ambulance*, *fire truck*, *touring coach*, and *minibus*. These words and the sentences formed by them were carefully selected so that all sentences within the same voice category contained the same number of syllables and were relevant to driving. For example, "*the bicycle blocked the car*." Microsoft Speech Platform was used to synthesize speeches from texts. Each trial of the comprehension task took seven seconds. First, a low frequency tone was played, followed by an input sentence. Then a high frequency tone was played to signal the start of the corresponding probe sentence. A correct response must be made after the high tone and before the end of the trial, which created a response window of 3.5 s. There was no delay between consecutive trials. In the comprehension single-task condition, a white screen was shown with engine noise played through loud speakers in the same volume as in the corresponding dual-task conditions.

Table 10. Types of combinations of input and probe sentences. A and B represent nouns (from Cao & Liu, 2013a).

Input	**Probe**	**Correct response**
The A verbed the B.	The A verbed the B.	Yes
The A verbed the B.	The B was verbed by the A.	Yes
The A verbed the B.	The B verbed the A.	No
The A verbed the B.	The A was verbed by the B.	No
The B was verbed by the A.	The A verbed the B.	Yes
The B was verbed by the A.	The B was verbed by the A.	Yes
The B was verbed by the A.	The B verbed the A.	No
The B was verbed by the A.	The A was verbed by the B.	No

2.3. Design and measures

This experiment used a within-subject design. An independent variable was task condition with two levels: single-task and dual-task. The single-task condition could be either lane keeping or speech comprehension. Another independent variable was driving speed with two levels: 36 and 72 km/h. Table 11 shows the sequence of experimental conditions experienced by each participant. Each block (either single or dual-task) lasted two minutes, and the 24 blocks were divided into four sessions, two with low speed and two with high speed. The order of speed levels was balanced across subjects. Each experimental condition (e.g., lane keeping single-task at the lower speed level) had four repetitions, whose results were averaged before analyzing the effects of independent variables. In the speech comprehension task, the correspondence between buttons (left or right) and responses (yes or no) was also balanced across subjects.

Table 11. Sequence of experimental conditions experienced by each participant (from Cao & Liu, 2013a).

Session	**Vehicle speed**	**Block 1**	**Block 2**	**Block 3**	**Block 4**	**Block 5**	**Block 6**
1	Level 1	LK	SC	LK+SC	LK+SC	SC	LK
2	Level 2	LK	SC	LK+SC	LK+SC	SC	LK
3	Level 2	LK	SC	LK+SC	LK+SC	SC	LK
4	Level 1	LK	SC	LK+SC	LK+SC	SC	LK

Level 1 and 2: 36 and 72 km/h, or 72 and 36 km/h (order balanced across subjects)
LK: Lane keeping single-task
SC: Speech comprehension single-task
LK+SC: Dual-task condition

Lane keeping performance was measured by both the standard deviation of lane position (*SDLP*), i.e., lateral deviation, and the standard deviation of steering wheel angle (*SDSWA*). *SDLP* and *SDSWA* values averaged from each experimental session were also analyzed to examine the carryover effect, e.g., learning and fatigue. Speech comprehension performance was measured by both reaction time and correct rate (i.e., accuracy). Mental workload was measured by NASA-Task Load Index (NASA-TLX; Hart & Staveland, 1988) for each experimental condition.

2.4. Procedure

Participants completed a consent form and demographic and general driving background questionnaires before the experiment. For the lane keeping task, participants were instructed to remain as close to the center of the lane as possible. For the speech comprehension task, they were instructed to give the correct answer as quickly as possible. For the dual-task conditions, they were instructed to first ensure safe driving and then complete the comprehension task as well as they can, assigning priority to the lane keeping task. Incentives were used to motivate high performance in both single and dual-task conditions. Participants were informed that they could win an extra bonus (equivalent to about 25 U.S. dollars) in addition to their compensation (equivalent to about 20 U.S. dollars), if their performance was ranked among the top 20% of all participants. The overall ranking, which was used to determine the top 20%, was averaged from the ranking of individual task conditions. Lane keeping performance was ranked by smaller *SDLP*. Comprehension performance was ranked by the average of reaction time ranking and correct rate ranking. In the dual-task condition, performance was ranked first by lane keeping and second by comprehension, because priority was given to lane keeping. Each participant spent about six minutes in practicing all task conditions. During the practice, immediate feedback was given to each comprehension response, telling the participants whether their response was correct or incorrect. However, no feedback was provided during the formal experiment. Participants completed a NASA-TLX survey for each experimental condition and a debriefing questionnaire at the completion of all task procedures. Short breaks were given between sessions, and the whole experiment took about 80 minutes.

2.5. Statistical analysis

Repeated measures analysis of variance (ANOVA) was carried out using SPSS version 20.0. All the independent variables (i.e., task condition and driving speed) were within-subject factors, and there was no between-subject factor. Full factorial models were used to test all possible main effects and interactions. When Mauchly's Test showed that the Sphericity assumption was violated, the Huynh-Feldt correction and the Greenhouse-Geisser correction were consulted (Field, 2009). Effect sizes were measured by eta-squared (η^2), which is defined as the proportion of the variance associated with a factor among the total variance of all factors (i.e., main effects, interactions, and errors) in an ANOVA study (Field, 2009).

3. Results

3.1. Lane keeping performance (SDLP)

Across all conditions, the participants' standard deviation of lane position (*SDLP*) had a mean value of 0.34 m with a standard deviation of 0.16 m. As shown in Figure 11, repeated measures ANOVA revealed a significant effect of vehicle speed on *SDLP* ($F(1, 23) = 16.181$, $p = 0.001$, $\eta^2 = 0.224$). Average *SDLP* at the high speed (0.36 m) was larger than the value at the low speed (0.30 m). The effects of task condition ($F(1, 23) = 3.303$, $p = 0.082$, $\eta^2 = 0.046$) and interaction ($F(1, 23) = 0.730$, $p = 0.402$, $\eta^2 = 0.003$) were not significant. Note that the *p* value of the task condition factor was 0.082, which is close to the significance level of 0.05. One may speculate that removing any outlier from the data may make this effect significant. However, there is no firm reason to remove any data point, because all participants' *SDLP* values were within the range of three times the standard deviation. Nevertheless, I re-analyzed the data with the largest *SDLP* data point (0.74 m) removed, and the results were still the same, with the *p* value of the task condition factor increased to 0.099.

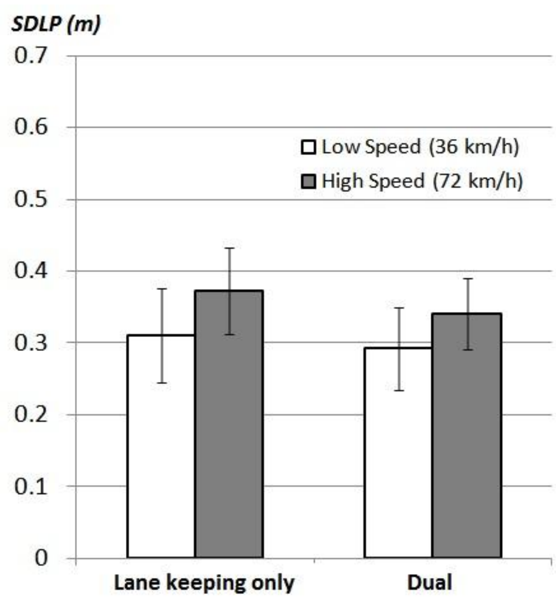


Figure 11. Effects of task condition and vehicle speed on lane keeping performance (standard deviation of lane position, *SDLP*). Error bars represent 95% confidence intervals (from Cao & Liu, 2013a).

3.2. Steering wheel control (SDSWA)

Repeated measures ANOVA found no significant effect on SDSWA, including task condition ($F(1, 23) = 1.942$, $p = 0.177$, $\eta^2 = 0.021$), vehicle speed ($F(1, 23) = 2.426$, $p = 0.133$, $\eta^2 = 0.041$), or interaction ($F(1, 23) = 0.079$, $p = 0.781$, $\eta^2 = 0.001$). Overall, SDSWA had a mean value of 5.20 degree and a standard deviation of 1.35 degree.

3.3. Carryover effect on driving performance

The analyses using repeated measures ANOVA found no significant carryover effect on any of the driving performance measures. SDLP values from the four experimental sessions were not significantly different ($F(2.675, 61.527) = 0.088$, $p = 0.955$, $\eta^2 = 0.003$; using the Huynh-Feldt correction $\hat{\varepsilon} = 0.892$, because Mauchly's Test showed that the Sphericity assumption was violated, $p = 0.030$). SDSWA values from the four sessions were also not significantly different ($F(2.082, 47.884) = 0.545$, $p = 0.590$, $\eta^2 = 0.023$; using the Huynh-Feldt correction $\hat{\varepsilon} = 0.694$, because Mauchly's Test showed that the Sphericity assumption was violated, $p = 0.001$).

3.4. Speech comprehension performance

Repeated measures ANOVA found no significant effect of task condition, vehicle speed, or interaction on the reaction time performance (all p values >= 0.135, η^2 <= 0.043), as shown in Figure 12 (a). The overall average reaction time was 1.951 s. For correct rate, only the effect of task condition was significant ($F(1, 23) = 8.168$, $p = 0.009$, $\eta^2 = 0.098$) , as shown in Figure 12 (b). Correct rate in the dual-task condition (91.3%) was reduced by 2.1% compared with the comprehension-only condition (93.4%). No other effect was significant (all p values >= 0.507, $\eta^2 < 0.001$).

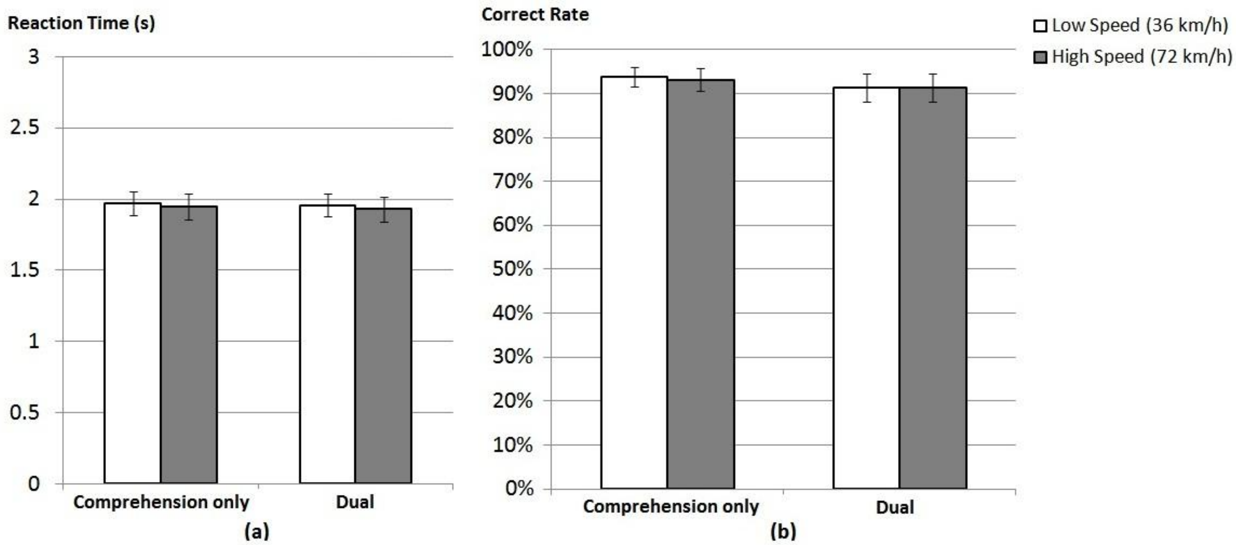


Figure 12. Effects of task condition and vehicle speed on speech comprehension performances of (a) reaction time and (b) correct rate. Error bars represent 95% confidence intervals (from Cao & Liu, 2013a).

3.5. Mental workload

Before the repeated measures ANOVA, Mauchly's Test showed that the Sphericity assumption for the task condition factor was violated (p = 0.023). With the Huynh-Feldt correction ($\hat{\varepsilon}$ = 0.820), the effect of task condition on mental workload (NASA-TLX overall rating) was found to be significant (F(1.641, 37.738) = 39.381, p < 0.001, η^2 = 0.574), as shown in Figure 13. The result from the more conservative Greenhouse-Geisser correction ($\hat{\varepsilon}$ = 0.775) was also significant (p < 0.001). Pairwise comparisons showed that mental workload was significantly different between any two conditions of lane keeping only (29.9), comprehension only (50.3), and dual-task (60.6; all p values <= 0.001). No other effect was significant (driving speed, F(1, 23) = 2.307, p = 0.142, η^2 = 0.003; interaction, F(2, 46) = 0.185, p = 0.832, η^2 < 0.001).

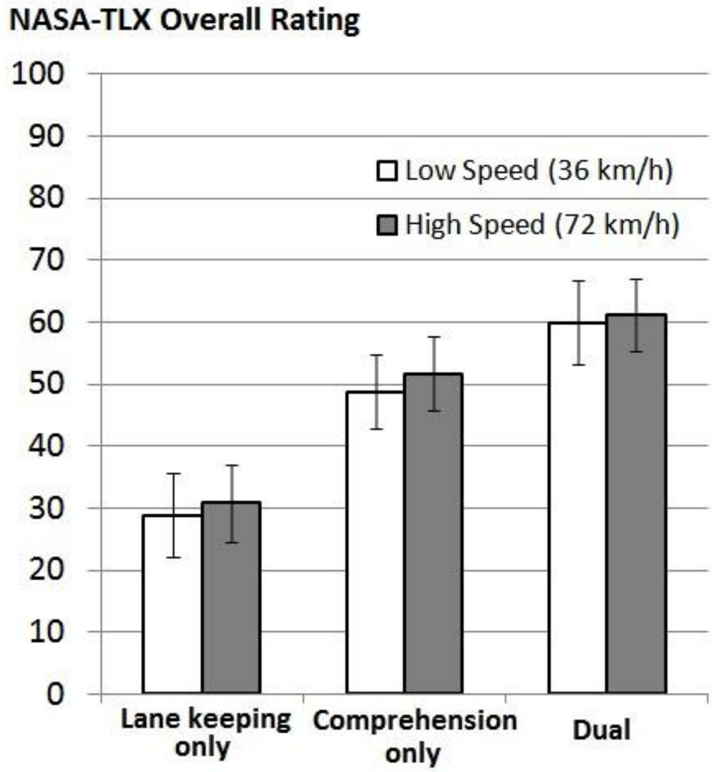


Figure 13. Effects of task condition and vehicle speed on mental workload (NASA-TLX). Error bars represent 95% confidence intervals (from Cao & Liu, 2013a).

4. Discussion

4.1. Experimental control

The current experiment examined drivers' capability to concurrently perform a vehicle lane keeping task and a speech comprehension task. Improved experimental control was used to address four potential confounding factors including strategic tradeoff, lane keeping difficulty, effort in the cognitive task, and motivation.

First, the potential strategic tradeoff between different components of the driving task was eliminated by confining driving to lane keeping only. The tradeoff strategy between lane keeping and speech comprehension was controlled by instructions telling the participants to give priority to lane keeping in the dual-task condition, which follows the common requirement of driving safety in real-world driving. It is difficult to guarantee that all participants have followed the instruction in the experiment as in the real-world where many car accidents were caused by drivers failing to focus on driving but distracted by other tasks. But at least regarding the experimental design, all participants were explicitly instructed to use the same strategy of prioritizing on lane keeping in this experiment.

Second, the lane keeping difficulty was guaranteed to be the same in both the single and dual-task conditions, because the same road and the same vehicle speed were used. Vehicle speed was controlled automatically at either 36 or 72 km/h. The results showed that SDLP was significantly larger when the speed was faster, which proves the necessity to control vehicle speed for equal lane keeping difficulty.

Third, the effort to actively process the reading comprehension task was demonstrated by the high correct rate (91.3%) in the dual-task condition. The comprehension task was designed to require frequent and active processing. The participants' reading comprehension was examined by a question every seven seconds. Mental workload results showed that the comprehension task produced a workload level much higher than the lane keeping single task. These evidences indicate that the participants were actively engaged in speech comprehension concurrently with lane keeping in the dual-task condition.

Fourth, motivation was controlled by incentives to promote best performance in both single and dual-task conditions. Although it is difficult to measure motivation directly, the amount of effort spent on steering control is related to the standard deviation of steering wheel angle (*SDSWA*). When the standard deviation of lane position (*SDLP*) is the same, a larger *SDSWA* value often means more steering corrections. The results showed that both *SDLP* and *SDSWA* were not significantly different between the single and dual-task conditions. This fact indicates that the participants tried equally hard to control the steering wheel in both conditions.

In addition, the results showed no significant carryover effect on any of the driving performance measures, which indicates that the effect of learning or fatigue is none or minimal and should not affect the interpretation of the effects of current tasks.

4.2. Representing real-world driving

The goal of this study is to examine drivers' dual-task processing capability with a strictly controlled experiment. Since the experiment needs to ask participants to perform difficult tasks under high workload conditions, it is dangerous and unethical to test them in real-world driving. Driving simulation also makes it possible to control experimental conditions (e.g., minimal traffic and constant driving speed) and to design a road that contains balanced left and right curves with different curvatures. The driving task in this study contains only the component of lane keeping, but it is not necessarily easy to perform. The results showed an average *SDLP* of 0.34 m and an average *SDSWA* of 5.20 degree, both of which are at the larger ends of the values found in the previous studies (Table 9). The lane keeping task used in this experiment represents a major component in real-world driving. Particularly, it represents driving while using cruise control or adaptive cruise control. With these technologies, driving on low-traffic highways and rural roads has often been reduced to a single task of lane keeping.

Several studies have compared driving performance measured in simulators with real-world driving (Blaauw, 1982; Engström et al., 2005; Reed & Green, 1999). In general, they found that performance was poorer in simulated driving (e.g., larger SDLP). One explanation is the lack of driver motivation in simulation studies because of the lack of risk. In the current study, I used monetary incentives (about $25 awarded to the top 20% participants) to enhance motivation in driving simulation. Since the overall ranking determining who could receive the award was averaged across the rankings in all task conditions, the participants had a reasonable motivation to perform well in both single and dual-task conditions.

In this experiment, I designed a zigzag road to increase the sensitivity of identifying the effect of a concurrent task on lane keeping performance, because driving on such a road requires more steering corrections and continuous attention than driving on a straight road. The route, as shown in Figure 10, is more difficult to follow but not unrealistic. In comparison, previous driving simulation studies often used the external disturbance force method, for example, adding crosswind force that follows a combination of three sine waves. The zigzag road method should be more realistic. Participants may not have any experience with the continuous and unpredictable disturbance force, but they probably have experience with zigzag roads. The disturbance force cannot be seen and therefore is hard to predict. It may cause driver frustration when large disturbance is needed to create a difficult task. In contrast, a zigzag road can be seen and predicted by drivers. Based on the above discussion, the experiment used in this study can represent real-world driving.

4.3. Comparison with previous findings

Before comparing the results from this study with previous studies, it is necessary to point out that although these studies were designed to examine similar research questions, they are often different in many details, including experimental scenarios (e.g., road, speed, and other vehicles) and equipment configurations (e.g., data sampling rate, vehicle dynamics, and motion feedback). These factors may be potential causes of difference in the results, so one must be aware of them and be cautious when interpreting the results. On the other hand, if some common trends can be found from different studies, the variation in these factors can strengthen the validity of the results. With these considerations in mind, I compared the results from this study with previous studies and found several interesting trends.

First, as shown in Table 9, most of the studies found lane keeping performance not affected or improved by a concurrent cognitive task. The two studies that found degraded lane keeping performance both used unconventional driving devices – that is, joystick, mouse, or trackball devices rather than a steering wheel that was used in the other studies including the present study. While drivers are very familiar with a steering wheel and can use automatic processing, they may have to use controlled processing with the unconventional devices. Then the dual-process model (Kahneman, 2003; Shiffrin & Schneider, 1977) may provide an explanation to the degraded lane keeping performance. As it predicts, strong interferences exist between a controlled process (comprehension) and another control process (lane keeping with unconventional devices) but not between a controlled process and an automatic process (lane keeping with a steering wheel). In addition, the instruction used by Just et al. (2008) is also a potential explanation for the degraded performance. Since participants were required to attend to both tasks equally, some of their mental processing resources could be shifted to the comprehension task.

Second, among the studies that found unaffected or improved lane keeping performance by a concurrent speech comprehension task, the magnitude of difference is relatively small compared with the effects of other factors such as vehicle speed, a concurrent visual task, or alcohol consumption. For example, the improvement of SDLP by concurrent speech comprehension is small, in the magnitude of 0.02 m (Liang & Lee, 2010), 0.01 m (Brookhuis et al., 1991), or not significant (the current study, and Kubose et al., 2006). In contrast, the degradation of *SDLP* caused by some established risk factors is much larger, in the magnitude of 0.06 m by faster speed (the current study), 0.08 m by a concurrent visual-manual task (Tsimhoni, Smith, & Green, 2004), and 0.06 m by alcohol (Lenné et al., 2010). There are two simulation studies that found larger improvement of *SDLP* by concurrent speech comprehension in the magnitude of 0.06 m (Becic et al., 2010) and around 0.04 m (Engström et al., 2005). However, these two results should be interpreted with caution, because the experiments used relatively easy and straight roads possibly without external disturbance force and did not introduce any control of motivation or potential strategic tradeoffs. The previously discussed issue of the lack of risk and driver motivation in simulation studies may explain these two larger values of SDLP improvement, especially considering that the same study conducted by Engström et al. (2005) did not find any significant effect in a real-world driving test. Besides SDLP, SDSWA was also not affected by concurrent speech comprehension in the current study as well as in the previous studies (Table 9). In summary, the converging evidence seems to show that a concurrent speech comprehension task has a very small or no effect on lane keeping performance.

Third, most of the studies found that mental workload is increased by the concurrent cognitive task, as shown in Table 9. This effect was reproduced in the current study. The absolute values of overall NASA-TLX ratings are greater in the current study than previous studies. In the current study, the overall NASA-TLX ratings ranged from 29.9 (lane keeping only) to 60.6 (dual-task condition), whereas the values in two previous studies were around 3.5 to 10.8 in a field study of driving and cell phone use (Matthews, Legg, & Charlton, 2003) and around 20 to 40 in a simulation study of driving and button pressing (Wu & Liu, 2007). The larger NASA-TLX ratings in the current study indicate that the design to make the tasks difficult is effective. Therefore, the absence of strong interference in this study is not likely to be caused by a low difficulty level or a lack of work demand.

4.4. Implications

The results obtained from this study provide some insights into the mechanism of drivers' concurrent task performance and the design of driver workload management systems. In recent years, as the prevalence of cruise control and adaptive cruise control technologies, driving on highways and rural roads has often been reduced to a single task of lane keeping. Since workload is reduced, a question is whether drivers could or should perform a concurrent task while driving. Drivers sometimes choose to perform a secondary task concurrently because of the desire to increase productivity (e.g., checking emails through a speech interface) or to counter drowsiness (e.g., listening to the car radio). The current study suggests that performing a secondary speech comprehension task for a limited period of time may not immediately affect the primary task performance (i.e., lane keeping), but it increases drivers' mental workload and reduces drivers' capability to comprehend speech. No significant effect of fatigue was found during the current experiment that lasted about 80 minutes; since fatigue is not the focus of the current study, I did not test the participants for a longer period of time. Nevertheless, the increased mental workload indicates the potential of increased risk when the dual-task is performed for a longer time period, because increased mental workload may aggravate the accumulation of fatigue and reduce driving performance. A proper workload management strategy should consider all the factors including the effects on the performances of the primary task (i.e., driving) and the secondary task (e.g., speech comprehension) as well as the effects on mental workload and fatigue.

In addition, drivers should always be aware of the change of driving scenarios and give priority to driving. When the work demand of the driving task increases, for example, when the driving task requires not only lane keeping but also lane changing, route selection, and speed control, drivers should stop any secondary task, because many studies have shown that cognitive load degrades event reaction performance and may also affect speed control performance. The requirements of situation awareness and being able to stop the secondary task when necessary create a new task of workload management, which is the third one in addition to driving and the secondary task. The development of intelligent workload management systems and driver assistance systems may help reduce mental workload caused by multi-task scheduling (Wu & Liu, 2007). The use of automation is more crucial for the tasks that cannot be performed concurrently by a human operator without significantly losing performance. In this regard, drivers may benefit more from collision avoidance warning and blind spot assistance systems that help detect hazard events than from lane departure assistance systems that help maintain lane position.

The results from the present study accumulate empirical data for the examination of human concurrent processing capability, which has been one of the research focuses of cognitive psychology. Previous studies in this field of cognitive psychology research have focused on discrete tasks such as simple reaction tests (Wu & Liu, 2008a). In contrast, the present study examined a continuous task of lane keeping and found that its performance was not affected by a concurrent speech comprehension task. The results from the current study and previous studies together support a multi-dimensional view of human concurrent processing capabilities (e.g., Wickens, 2008). Future studies are needed to identify all major factors that affect the concurrency of multi-task processing.

4.5. Future studies

Future studies are needed to examine drivers' workload management capability, for example, switching back from performing multiple tasks to focusing on driving. In addition, future studies with improved experimental control are also needed to examine the effect of cognitive tasks on longitudinal control (e.g., speed control and car following) and the potentially different effects between different types of cognitive tasks (e.g., numerical computation, speech comprehension and production). Conventional wisdom may believe that concurrent tasks would always degrade driving performance; however, the results from the current study, combined with previous study findings, showed that the whole story may depend on the kinds of concurrent tasks and the types of driving performance measures. Although the present study of speech comprehension effects on lane keeping performance did not find any significant effect of comprehension on lane keeping performance, future studies, for example, investigating the effects of speech production on hazard response performance, may find significant effects of concurrent tasks as suggested by previous studies. Another potential reason for the lack of interference in the current results may be the use of non-spatial coding of the speech contents. Future research using spatial coding in speech tasks may find stronger interference with driving that also involves spatial coding. As described in the introduction section, previous studies have mixed results, indicating the need for more research and proper experimental control of potential confounding factors such as strategic tradeoff and task difficulty. Future studies can also examine the combined effects of fatigue and concurrent tasks using experiments with fatigue manipulation.

5. Conclusions

This study examined the concurrent performance of vehicle lane keeping and speech comprehension tasks with improved experimental control of the confounding factors identified in previous studies. The results suggest that performing a secondary speech comprehension task for a limited period of time may not immediately affect the primary task performance (i.e., lane keeping), but it increases drivers' mental workload and reduces drivers' capability to comprehend speech. The significantly increased mental workload during the concurrent task condition indicates the potential of increased risk when the dual-task is performed for a longer time period. These findings provide some new insights into the mechanism of drivers' concurrent task performance and the design of driver workload management systems.

Chapter 6. QN-ACTR Modeling and Simulation of Lane Keeping and Speech Comprehension Dual-task Performance

Chapter Summary

This chapter introduces the QN-ACTR model built to simulate the human results collected from the lane keeping and speech comprehension dual-task reported in the previous chapter. I want to demonstrate that QN-ACTR is a generic method that can be applied to a wide range of human factors domains also including transportation human factors. Through the modeling of the driving dual-task, it has been confirmed that the QN disciplines previously examined in modeling the medical diagnostic decision multitasking performance are also necessary in modeling driving dual-task performance.

1. Introduction

The previous chapters have demonstrated the advantages of integrating QN and ACT-R in modeling cognitive multitasking performance in the human-computer interaction domain and the healthcare domain. Another domain to test and examine the modeling capability of QN-ACTR is ground transportation, where the potential interference and distraction from using in-vehicle devices have been a focus of human factors research with increasing importance, as the usage of such devices while driving become more and more prevalent.

Several previous studies have tested computational simulation models of human driving performance. Salvucci (2006) proposed and examined a driver model in ACT-R, utilizing a two-level control model (Salvucci & Gray, 2004) that determines the change of the steering angle ($\Delta\varphi$) based on the near point visual angle (θ_{near}) and the far point visual angle (θ_{far}) as in,

$$\Delta\varphi = k_{far}\Delta\theta_{far} + k_{near}\Delta\theta_{near} + k_I \min(\theta_{near}, \theta_{n\max})\Delta t\,, \tag{2}$$

where θ_{nmax} (set to 0.07) controlled the maximum of θ_{near}. The near point is set as the middle point of the lane 10 m ahead, and the far point is one of (a) the vanishing point of a straight road, or (b) the inner tangent point of an upcoming curve, or (c) the lead vehicle when there is one. In a typical control cycle (Δt), three production rules fire consecutively in 150 ms, perceive visual information and issue a motor control action turning the steering wheel to maintain lateral position in the center of the lane.

This driving model has been demonstrated to be able to model driving performance in lane keeping and lane change single-tasks (Salvucci, 2006) and also model dual-task performance of lane keeping and digit rehearsal (Salvucci & Beltowska, 2008). Using similar methods, a recent study also modeled the effect of driving experience on lane keeping performance by assigning a closer visual attention focus point (i.e., far point) to the novice-driver model than the experienced-driver model (Cao, Qin, & Shen, 2013).

Computational simulation models of human driving performance have also been proposed and examined in QN-MHP. Liu, Feyen, and Tsimhoni (2006) developed a QN-MHP model of driving and map reading. The modeling algorithms simulating lane keeping/steering control performance were similar but different from Salvucci's model in ACT-R (2006). Instead of using a continuous equation (Equation 2) to describe the relationship between visual perceptual input and manual steering action, the QN-MHP model used three discrete levels of steering actions including no action, normal action, and imminent action.

> "Steering actions are selected based on the orientation of the vehicle within a look-ahead time (a parameter currently defined as 1 s) as calculated in server F using the following logic: If the vehicle's orientation within the look-ahead time is close to the center of the lane (±0.1 m), no action is taken. Otherwise, if it is within the lane boundaries, a normal

> steering action is initiated, or if it is outside the lane boundaries, an imminent steering action is initiated." (Liu et al., 2006, p. 56)

In addition to modeling lane keeping and map reading, this QN-MHP model has also simulated lane keeping and button pressing dual-task scenarios and been demonstrated to be able to integrate with physical digital human models to simulate both cognitive and physical performance (Fuller, Reed, & Liu, 2012; Wu & Liu, 2007)

Previous studies have modeled driving single-task and simple dual-task scenarios; however, there is still a lack of computational simulation models that can model dual-task performance involving more complex cognitive tasks such as speech comprehension. Since drivers in the real-world not only perform perceptual-motor tasks such as map icon reading and button pressing but also complex cognitive tasks such as speech comprehension and decision making, it is important to model and simulate driver performance in such complex dual-tasks for the analysis and evaluation of driving safety issues regarding the use of in-vehicle devices. To fill this research gap in the field of HPM, I built a computational model in QN-ACTR to simulate human performance and mental workload in the dual-task of lane keeping and speech comprehension. The human experiment and results to be simulated have been introduced in the previous chapter. Since the simulated driving environment was developed using Animator3D simulation in Micro Saint® Sharp that is also the implementation platform of QN-ACTR, QN-ACTR models can be seamlessly connected with the simulated driving environment to test and examine simulated performance and mental workload. This integration of human experiments with model simulation on the same platform allows the model and the human to perform and be compared in the same tasks with identical interfaces, with no need to replicate the real world experiment system in the modeling platform for the model to interact with. It provides the benefit of avoiding any potential discrepancy between human and model tests due to the experiment setup.

Modeling complex multi-task scenarios in the transportation domain is a further demonstration that QN-ACTR is a generic computational simulation theory and method that can be applied to a wide range of human factors domains. The unique features of QN-ACTR were tested and examined in this lane keeping and speech comprehension dual-task that contains important and practical human factors issues for driving performance and safety.

2. Method

The human experiment – lane keeping and speech comprehension dual-task – was described in the previous chapter. The experiment results showed that the standard deviation of lane position (*SDLP*) was increased when the driving speed was faster (0.30 m at 36 km/h; 0.36 m at 72 km/h). The concurrent comprehension task had no significant effect on *SDLP* (0.34 m on average) or the standard deviation of steering wheel angle (*SDSWA*; 5.20 degree on average). The correct rate of the comprehension task was reduced in the dual-task condition (from 93.4% to 91.3%) compared with the comprehension single-task condition. The reaction time of the comprehension task was not significantly different between the single- and dual-task conditions (1.95 s on average). Mental workload was significantly higher in the dual-task condition compared with the single-task conditions.

This section focuses on the introduction of the QN-ACTR modeling method. Since the model can directly interact with the experiment platform used in the human study, the task setup was identical as the human experiment, just substituting human steering wheel and button press control signals with the model's simulated control signals. The task-specific knowledge and parameters are introduced below in details. Following the same principle of using generic production rules for dual-task modeling used in previous QN-ACTR dual-task models, the production rules were defined for each task individually without using any executive control rule in this modeling work. It means that the knowledge of dual-task model is simply the combination of the knowledge of the two single-task models.

The modeling of the lane keeping task closely followed previous ACT-R models of driving performance (Salvucci, 2006). In particular, three production rules fires consecutively to complete a control cycle including (1) look for the visual-location of a near-point; (2) update near-point information in the goal buffer, and look for the visual-location of a far-point; and (3) update far-point information in the goal buffer, and send a manual command to steer the wheel based on Equation 2. Since the human driving task focused on lane keeping only without other driving components such as mirror checking or lane changing, as described in the previous chapter, the model of lane keeping performance did not include any other production rules. English descriptions of the rules are summarized in Table 12.

The modeling of the speech comprehension task closely followed previous ACT-R models of reading comprehension (J. R. Anderson et al., 2001), which was also one of the bases of the typing and reading comprehension model introduced in previous chapters. In essence, the model's comprehension process of the input and probe sentences used in the task can be summarized as follows. When the input sentence is played through the loud speaker, the model attends to each audio event and encodes the meaning of each word. The meanings of the words are stored in a sentence representation formed in the imaginal buffer. Each sentence representation has the following slots or attributes including *subject*, *aux*, *verb*, and *object*. The *aux* slot represents the auxiliary verb, for example "was," in a passive sentence. A sentence with an active voice will have no auxiliary verb so an empty *aux* slot. After the input sentence is fully encoded, the model stores the sentence representation in its declarative memory. When a high pitch tone is played to signal the start of the probe sentence, the model retrieves the sentence representation from its declarative memory, which can be erroneous. Then the model examines whether the subject from the probe sentence matches the subject from the input sentence, and also whether the auxiliaries from the two sentences match. Since the two nouns are always the same in both the input and probe sentences, responses can be made without checking whether the second nouns match or not. For example, if the input sentence is "the car hit the bicycle" and the probe sentence is "the bicycle was hit by the car," the facts that (1) the subjects do not match (car vs. bicycle) and (2) the auxiliaries do not match either (hit vs. was hit) imply that the two sentences must have the same meaning, without the need to further check the objects (bicycle vs. car). This strategy was used to built the cognitive models and successfully modeled human results in the previous study (J. R. Anderson et al., 2001) and also reflected in the comments from the subjects' debrief survey in the human experiment reported in the previous chapter. Finally, the model retrieves the key association from the declarative memory for yes or no and presses the key as a response. English descriptions of the rules are summarized in Table 12. In dual-task scenarios, a model simply used the production rules combined from the two sets of rules for the two single-tasks.

The model parameters are summarized in Table 13. All other model parameters were in their default values. For the parameters used in this modeling work, all the values followed the values used in previous studies, except for one parameter estimated to fit the human lane keeping performance data and two parameters estimated to fit the human data of speed-accuracy tradeoff in the speech comprehension task (i.e., single-task vs. dual-task condition speed-accuracy tradeoff). I assumed that the human results of unaffected reaction time and reduced correct rate were caused by a speed-accuracy tradeoff in the dual-task. Human factors studies have identified speed-accuracy tradeoff effects as a common characteristic of human performance that people may sacrifice response accuracy for faster speed of reaction, or vice versa (Fitts, 1954; Fitts & Peterson, 1964; David E. Meyer, Abrams, Kornblum, Wright, & Keith Smith, 1988; Woodworth, 1899). Under high workload or time stress, people often sacrifice response accuracy for faster speed of reaction to meet task time requirement. I assumed that the lane keeping task component in the dual-task scenario would have increased comprehension response time, but the response time was compensated by a speed-accuracy tradeoff that reduced the processing time of speech comprehension at the cost of the reduced correct rate.

This speed-accuracy tradeoff effect was modeled by adjusting a set of architecture parameters in QN-ACTR. Such method of modeling behavior moderators has been used in previous ACT-R literature and was called the overlay method. “An overlay is a technique for including a theory of how a behavioural moderator, such as stress, influences cognition across all models within a cognitive architecture” (Ritter & Reifers, 2007). At the implementation level, the overlay technique adjusts architecture parameters to model the effects of behavioural moderators and does not modify any task-specific knowledge. The speed-accuracy tradeoff effect is modelled in QN-ACTR using the overlay method, as shown in Table 13 for the dual-task (speed-accuracy tradeoff). In addition, mental workload was modeled by overall server utilization, which has been shown to have a linear relationship to NASA-TLX (Cao & Liu, 2011b).

Table 12. Procedures and production rules for lane keeping and speech comprehension tasks. The definition of these production rules follows the principles used in previous cognitive models (e.g., see models from http://act-r.psy.cmu.edu/).

#	Task procedure	Production rules
	Lane keeping	
1	At the start of a steering control cycle, look for visual-location of near-point.	drive-control-attend-near
2	If a near-point is focused in visual-location buffer, update near-point information in the goal buffer, and look for visual-location of far-point.	drive-control-process-near-attend-far
3	If a far-point is focused in visual-location buffer, update far-point information in the goal buffer, and send a manual command to steer the wheel.	drive-control-process-far
	Speech comprehension	
1	If there is an audio-event in aural-location buffer, encode the event in the aural buffer.	detected-sound
2	If a low frequency tone is encoded in aural buffer, create a sentence representation in the imaginal buffer, enter the input phase.	heard-low-beep*
3	If a word is encoded in aural buffer, retrieve the meaning of the word.	heard-word-sound-retrieve-meaning*
4	If retrieval is done, attend to the next audio-event.	find-next-sound
5	If it is the input phase, a noun is retrieved, and the subject slot of sentence representation in the imaginal buffer is empty, store the noun’s meaning in the subject slot, and attend to the next audio-event.	find-next-sound-phase-input-subject
6	If it is the input phase, an auxiliary verb is retrieved, and the aux slot of sentence representation in the imaginal buffer is empty, store the auxiliary verb’s meaning in the aux slot, and attend to the next audio-event.	find-next-sound-phase-input-aux
7	If it is the input phase, a verb is retrieved, and the verb slot of sentence representation in the imaginal buffer is empty, store the verb’s meaning in the verb slot, and attend to the next audio-event.	find-next-sound-phase-input-verb
8	If it is the input phase, a noun is retrieved, and the subject slot of sentence representation in the imaginal buffer is not empty, but the object slot is empty, then store the noun’s meaning in the object slot, and attend to the next audio-event.	find-next-sound-phase-input-object
9	If a high frequency tone is encoded in aural buffer, retrieve a sentence representation.	heard-high-beep*
10	If a sentence representation is retrieved, create a copy of the representation in the imaginal buffer, enter the probe phase.	start-recognition*
11	If it is the probe phase, subject has not been tested, a noun is retrieved, the noun’s meaning is the same as the subject’s meaning of the representation in the imaginal buffer, then check the goal’s subject-match as yes, and attend to the next audio-event.	find-next-sound-phase-probe-subject-match-yes*
12	If it is the probe phase, subject has not been tested, a noun is retrieved, the noun’s meaning is not the same as the subject’s meaning of the representation in the imaginal buffer, then check the goal’s subject-match as no, and attend to the next audio-event.	find-next-sound-phase-probe-subject-match-no*
13	If it is the probe phase, auxiliary has not been tested, an auxiliary verb is retrieved, the auxiliary verb’s meaning is the same as the auxiliary verb’s meaning of the representation in the imaginal buffer, then check the goal’s aux-match as yes, and attend to the next audio-event.	find-next-sound-phase-probe-aux-match-yes-type-1*
14	If it is the probe phase, auxiliary has not been tested, an auxiliary verb is retrieved, the auxiliary verb’s meaning is not the same as the auxiliary verb’s meaning of the representation in the imaginal buffer, then check the goal’s aux-match as no, and attend to the next audio-event.	find-next-sound-phase-probe-aux-match-no-1*
15	If it is the probe phase, auxiliary has not been tested, a verb is retrieved (which means that there is no auxiliary verb in the probe sentence), and the sentence representation in the imaginal buffer does not have an auxiliary verb, then check the goal’s aux-match as yes, and attend to the next audio-event.	find-next-sound-phase-probe-aux-match-yes-type-2*
16	If it is the probe phase, auxiliary has not been tested, a verb is retrieved (which means that there is no auxiliary verb in the probe sentence), and the sentence representation in the imaginal buffer has an auxiliary verb, then check the goal’s aux-match as no, and attend to the next audio-event.	find-next-sound-phase-probe-aux-match-no-1*
17	If the goal’s subject-match value is the same as aux-match, retrieve the key for yes.	response-yes
18	If the goal’s subject-match value is not the same as aux-match, retrieve the key for no.	response-no
19	If the key is retrieved, press the key.	press-key

*: marks the production rules require follow-up.

Table 13. Descriptions, values, and sources of parameters used in the lane keeping and speech comprehension model.

Parameter	Description	Value and source
	Lane keeping	
k_{far}	Parameter in the steering control equation (Equation 2)	20 (Salvucci & Gray, 2004)
k_{near}	Parameter in the steering control equation (Equation 2)	9 (Salvucci & Gray, 2004)
k_I	Parameter in the steering control equation (Equation 2)	1.5, estimated to fit the human data
	Speech comprehension (single-task)	
:lf	Retrieval time latency factor. Affect retrieval time.	0.3 (J. R. Anderson et al., 2001)
:bll	Chunk base-level learning decay. Affect the rate of declarative activation decay.	0.5 (J. R. Anderson et al., 2001)
:rt	Chunk retrieval threshold. The minimum activation a chunk must have to be able to be retrieved.	-1.5 (J. R. Anderson et al., 2001)
:ans	Chunk activation noise. Affect the noise of chunk activation.	0.05 (Jones & Ritter, 1998)
sdp *:references*	Set the reference history of chunks. Affect chunk activation.	1000 (J. R. Anderson et al., 2001)
sdp *:creation-time*	Set the initial creation time (s) of chunks. Affect chunk activation.	-10000 (J. R. Anderson et al., 2001)
spp start-recognition *:at*	Set the production execution time duration (s) of the rule.	0.2 (J. R. Anderson et al., 2001)
spp find-next-sound-phase-probe-subject-match-yes/no *:at*	Set the production execution time duration (s) of the rule.	0.1 (J. R. Anderson et al., 2001)
spp find-next-sound-phase-probe-aux-match-yes-1/2 *:at*	Set the production execution time duration (s) of the rule.	0.1 (J. R. Anderson et al., 2001)
spp find-next-sound-phase-probe-aux-match-no-1*:at*	Set the production execution time duration (s) of the rule.	0.2 (J. R. Anderson et al., 2001)
spp find-next-sound-phase-probe-aux-match-no-2*:at*	Set the production execution time duration (s) of the rule.	0.15 (J. R. Anderson et al., 2001)
	Dual-task (speed-accuracy tradeoff)	
:lf	Retrieval time latency factor. Affect retrieval time.	0.01, estimated to fit the human data
:ans	Chunk activation noise. Affect the noise of chunk activation.	0.20, estimated to fit the human data

The model formed by simply combining the two sets of single-task knowledge and parameters with the mechanisms described so far could simulate single-task performance; however, it could not model the dual-task performance properly. The major reason is that the two task components significantly interfere with each other in the production module so that both tasks are frequently interrupted. This observation showed that the basic queueing mechanism used in previous studies, i.e., first-in-first-out queueing, is not sufficient to model complex cognitive multitasking performance. As a result, I used two more sophisticated cognitive mechanisms in the model of this lane keeping and speech comprehension dual-task. The first mechanism is the visual-motor pathway used in previous QN-MHP studies, and the second mechanism is the filtering discipline used in the previous chapters of QN-ACTR work to model the diagnostic decision and current task scenarios.

In the framework of QN-MHP, there is an information pathway/link between Server A (visuospatial sketchpad) and Server V (Sensorimotor integration) (Liu et al., 2006; Wu & Liu, 2008a). The assumption is that some part of routine performance, such as skilled typing or driving, may be performed through this pathway without the need of central executive processing. Adopting the same assumption in QN-ACTR, I assume that the rule *drive-control-process-far* (#3) in the lane keeping model could be performed without using the production module resource. Instead, information will pass directly from the visual module to the motor module to issue the motor command of steering control.

The other mechanism that is important to the simulation of this dual-task performance is the filtering discipline. Recall this mechanism that has been described in previous chapters. First, it categorizes production rules into two groups—the ones that need follow-up processing and the ones do not. After a rule requiring follow-up is processed, the production module will start to exclusively accepting only the rules that follow up the same task (i.e., to work as a filter). If there is no such rule matched and available, the module will be enforced to idle and ignore other rules. Such exclusive processing continues until a rule that does not require follow-up is processed. A rule may be categorized as requiring follow-up, if at least one of the following three conditions is met. First, the rule's action part has a declarative retrieval request. Second, the rule's action part has an imaginal request, i.e., to create a chunk (problem state) in the imaginal buffer. Third, the rule processes aural information from a continuous stream of important audio stimuli, such as a question sentence. After applying this filtering discipline, several rules were categorized as requiring follow-up, as marked in Table 12. Models without the two extra QN mechanisms were also tested as a comparison.

3. Results

The simulation in QN-ACTR was repeated 15 times, reaching a 95% confidence interval within a ±5% range for each measurement. Across all conditions, the model's standard deviation of lane position (*SDLP*) had a mean value of 0.34 m same as the human value (0.34 m). As shown in Figure 14, the model's average *SDLP* at the high speed (0.37 m) was larger than the value at the low speed (0.30 m). The model results were very similar to the human results, producing *SDLP* results that showed only difference between the low and high speed conditions but not between the single- and dual-task conditions. The mean absolute percentage error (*MAPE*) was 3.8%, and the root mean square error (*RMSE*) was 0.02 m.

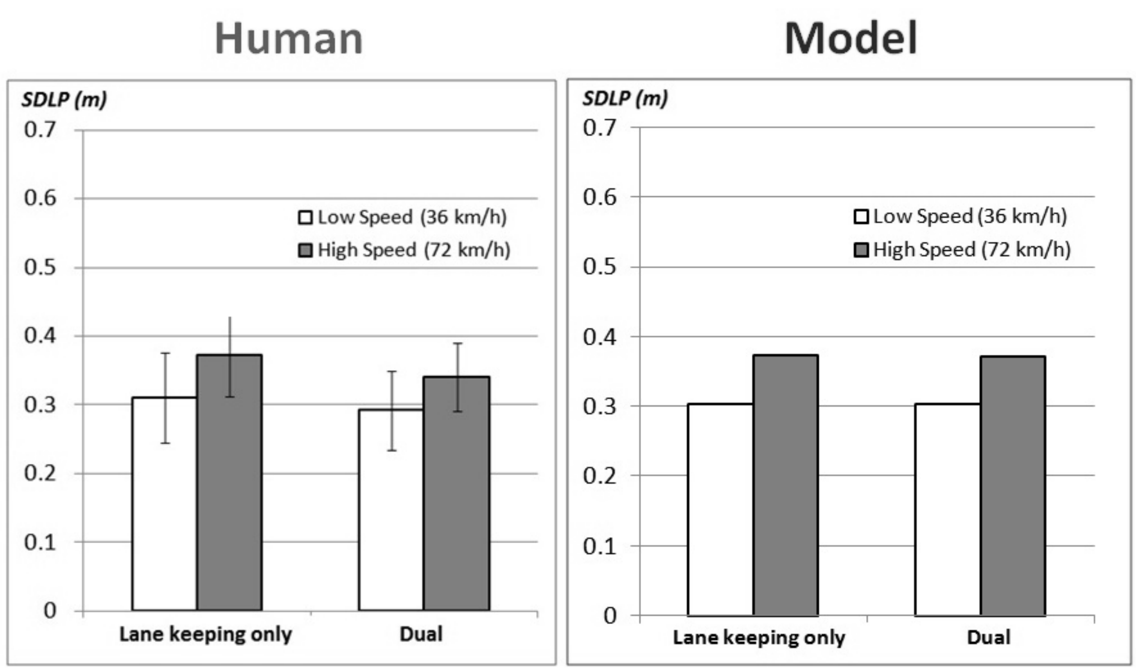


Figure 14. Model simulation results of lane keeping performance (standard deviation of lane position, *SDLP*) in comparison to the human results. Error bars represent 95% confidence intervals of the human results.

The model results of standard deviation of steering wheel angle (*SDSWA*) showed no difference between all task conditions (low speed driving only, 3.9 degree; low speed dual, 3.9 degree; high speed driving only, 3.8 degree; high speed dual, 3.9 degree), which was the same as the no significant effect of the human results, although the overall average *SDSWA* of the model (3.9 degree) was smaller than the human value (5.2 degree). *MAPE* was 25.0%, and *RMSE* was 1.3 degree.

The model simulation results of speech comprehension performance also have values very similar to the human results. The overall speech comprehension reaction time was 1.916 s for the model, similar to the human results of 1.951 s, as shown in Figure 15. *MAPE* was 1.8%, and *RMSE* was 0.04 s. The speech comprehension correct rate in the dual-task condition (89.0%) was reduced by 5.4% compared with the comprehension-only condition (94.4%) for the model, which was also similar to the reduced correct rate in the human results, as shown in Figure 16. *MAPE* was 2.0%, and *RMSE* was 2%.

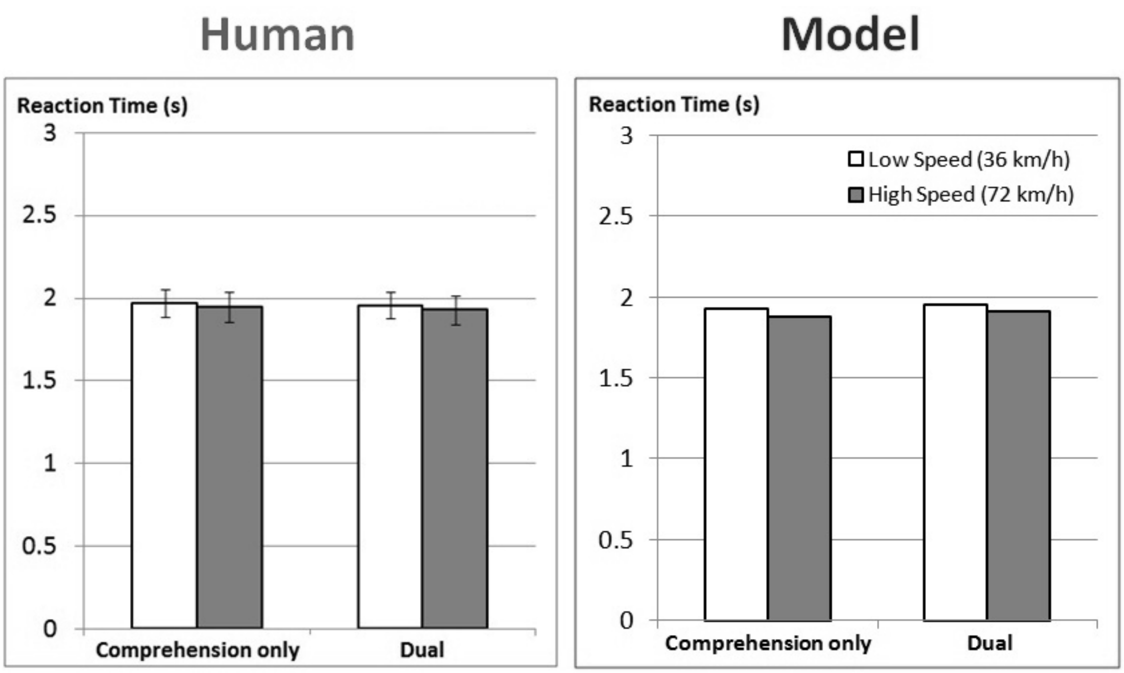


Figure 15. Model simulation results of speech comprehension reaction time in comparison to the human results. Error bars represent 95% confidence intervals of the human results.

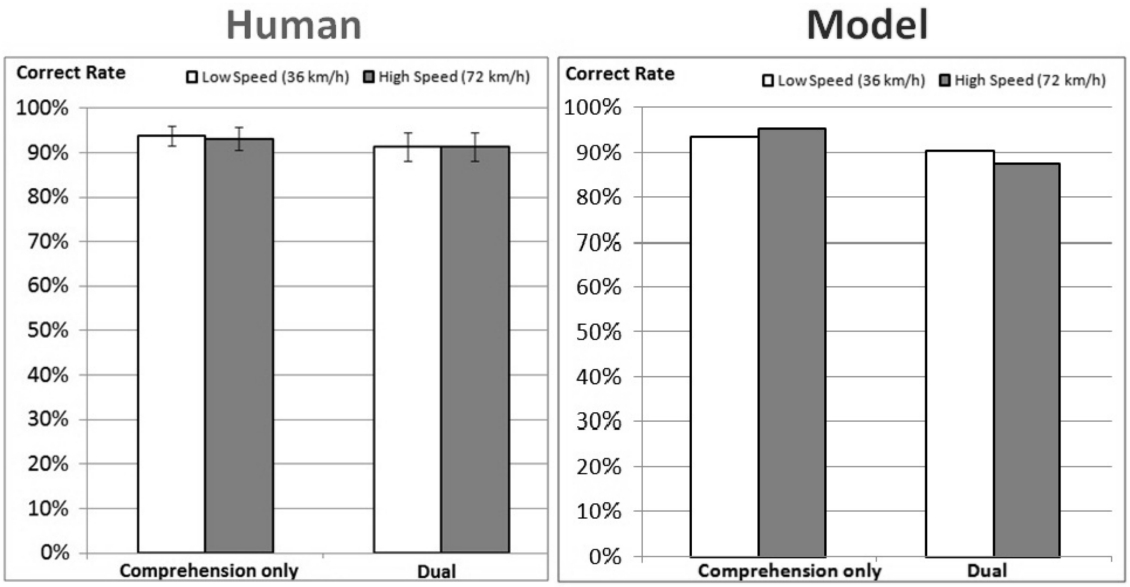


Figure 16. Model simulation results of speech comprehension correct rate in comparison to the human results. Error bars represent 95% confidence intervals of the human results.

The model simulation results of mental workload were 32.0 for the lane keeping only, 46.3 for the comprehension only, and 62.4 for the dual-task conditions, which were very similar to the human results, as shown in Figure 17. The linear relationship between overall utilization and overall workload is

$$\textit{Overall Workload} = \textit{Overall Utilization} * 228.6 + 6.7 \qquad (3).$$

The linear relationship was significant ($\beta = 0.972$, $t(4) = 8.329$, $p = 0.001$) and also explained a significant proportion of variance in NASA-TLX overall workload scores ($R^2 = 0.945$, $F(1, 4) = 69.371$, $p = 0.001$).

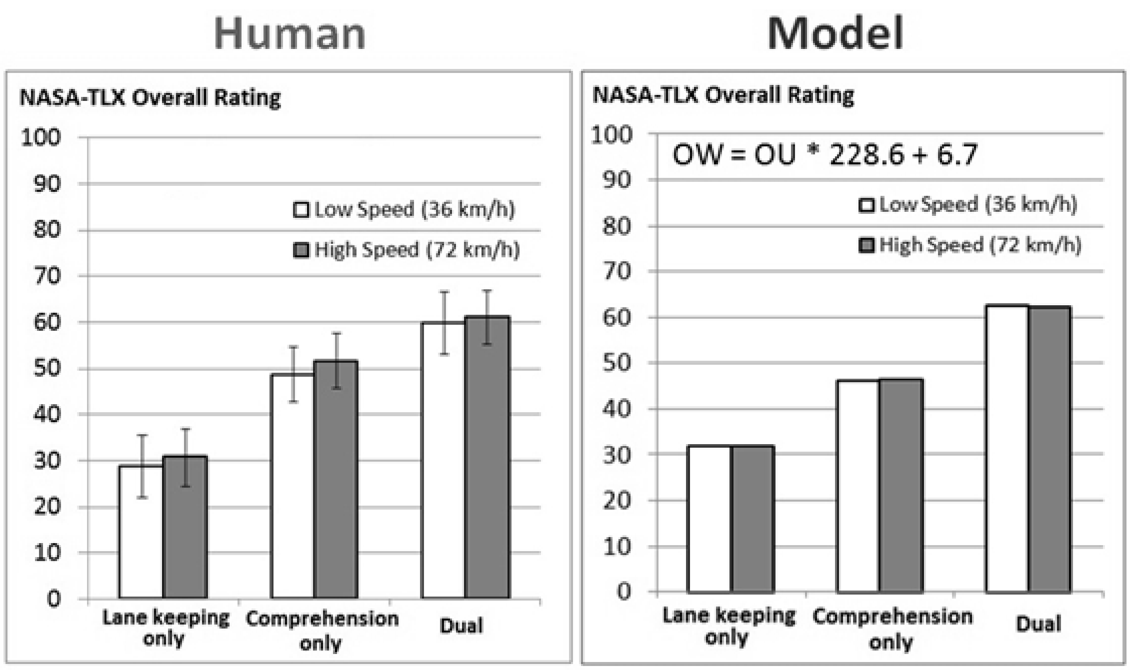


Figure 17. Model simulation results of mental workload in comparison to the human results. Error bars represent 95% confidence intervals of the human results.

In addition, I also examined model performance without using the visual-motor pathway or the filtering discipline. When only the visual-motor pathway mechanism was removed, the comprehension reaction time at the low speed condition became much longer (2.146 s), showing strong interference caused by the lane keeping task component that did not occur in the human results. At the high speed condition, the model without the visual-motor pathway drove out of the road and failed to complete the task, again showing strong interference that did not occur in the human results. On the other hand, when only the filtering discipline was removed, the comprehension reaction times were also much longer than the human results. The values were increased to 2.436 s in the low speed condition and 2.369 s in the high speed condition, showing strong interference that did not occur in the human results. These results demonstrated that both the visual-motor pathway and the filtering discipline from the QN perspective were necessary to model this dual-task performance of lane keeping and speech comprehension.

4. Discussion

In this chapter, I further demonstrated the benefits of integrating QN and ACT-R in the simulation of a dual-task scenario involving lane keeping and speech comprehension. The QN-ACTR computational model successfully produced similar results as the human participants in both performance and mental workload results. The modeling results also showed that both the visual-motor pathway mechanism, previously used in QN-MHP, and the filtering discipline, proposed and examined in previous QN-ACTR research (Chapter 4) , were both necessary to model the dual-task performance of lane keeping and speech comprehension.

Together with previous QN-ACTR models built for the simulation of skilled transcription typing and reading comprehension concurrent tasks (human-computer interaction) and medical diagnostic decision making concurrent tasks, this modeling work of driving with a secondary speech comprehension task (transportation) further demonstrated that QN-ACTR is a generic method and can be applied to a wide range of human factors domains containing important and practical human performance and mental workload issues. In addition, it is important to note that the filtering discipline was demonstrated again to be necessary in the simulation of complex cognitive multitasking performance and mental workload.

A limitation of the current model of lane keeping is that the overall average *SDSWA* results produced by the model were smaller than the human value. This difference may be caused by the fact that the model did not use any parameter to account for random movement errors in manual movements, which is also the case in previous QN and ACT-R driving simulation studies. To address this limitation, future research can add parameters to the motor module mechanisms to account for the errors of steering wheel control movement.

Chapter 7. Usability Development of Queueing Network-ACTR for Cognitive Engineering Applications

Chapter Summary

The increasing complexity of computational cognitive architectures may increase the difficulty for industrial and human factors engineers to learn and use them as cognitive engineering tools. This chapter reports the work to enhance the usability and the cognitive engineering applicability of QN-ACTR. The aim is to provide an easy-to-use interface and intuitive modeling that support both inexperienced and experienced users in using this complex and powerful architecture. The process of model development is greatly simplified with improved visualization and validation methods. The results were examined using heuristic evaluation. The benefits and practice implications are discussed.

1. Introduction

Cognitive models can be used to support cognitive engineering. Compared with other forms of cognitive models such as verbal frameworks and pure mathematical models, cognitive architectures are particularly useful for complex cognitive engineering applications, because they unify a wide range of cognitive theories (Newell, 1990) and can computationally simulate human-machine interactions (Byrne & Pew, 2009; Schunn & Gray, 2002).

In the recent years, cognitive architectures are becoming increasingly integrated and complex in terms of having more components and interactions between components, requiring the use of knowledge description languages, and involving a large number of parameters. This complexity may increase both modeling capabilities and the difficulty to learn and use them as cognitive engineering tools. For example, building useful models in ACT-R requires a considerable amount of training and practice. The basic theory and syntaxes of ACT-R can be learned by reading a seven-unit tutorial and practicing with examples, which are often covered in a seven-day short course. The model description of displays and controls is written in the Lisp language, and therefore a modeler must also gain reasonable Lisp programming skills. Adjusting model parameters could also be very difficult, because the effect of changing a parameter may be buried deep in the text-based output traces. Currently, most users of cognitive architectures are expert researchers of cognitive modeling. The usage among cognitive engineers in the industry is very limited.

Emphasizing the need for the usability development of cognitive architectures for cognitive engineering, Pew (2008) pointed out three challenges for researchers in this field, including the needs for (1) simplified model development, (2) better capabilities for articulating and visualizing how the models work, and (3) model validation.

Recently, several efforts have been made to address these challenges. G2A (Amant, Freed, & Ritter, 2005) and ACT-Simple (Salvucci & Lee, 2003) were developed to automatically translate GOMS (Goals, Operators, Methods, and Selection rules) style operators into ACT-R production rules. Incorporating ACT-Simple, CogTool (John, Prevas, Salvucci, & Koedinger, 2004) simplified the construction of human-computer interaction tasks, allowing the modeling of web browsing tasks by user demonstration with the mouse and the keyboard. Integrating ACT-Simple and an ACT-R driving model (Salvucci, 2006), Distract-R (Salvucci, 2009) simplified the construction of models for human interaction with in-vehicle devices in driving scenarios. Using Visual Basic Application in Excel, a click-and-select user interface has been developed in Queueing Network-Model Human Processor (QN-MHP, Wu & Liu, 2008a). It allows users to build QN-MHP models without learning any simulation language. Usability tests showed that this click-and-select interface can save time and reduce errors in model development (Wu & Liu, 2009). In addition, easy-to-use user interfaces have also been developed in E-GOMS (Gil, 2010) and SANLab-CM (Patton & Gray, 2010).

The previous efforts have simplified model development, reducing or eliminating the need for learning a modeling language. However, each of them still has limitation in one or more of the following aspects.

- The simplification of modeling work comes at the cost of limiting the task displays and controls that can be model to a limited set of tasks.
- The flexibility to construct customized models with customized parameters is limited, for example, not being able to define the road curvature of each road segment as in a human driving experiment.
- Human information processing that can be modeled is limited to procedural and perceptual-motor processes, lacking the capabilities to model complex cognitive tasks such as learning, decision making, and sentence comprehension.
- The visualization of how the model works can be improved to include both the visualization of mental information processing and task interaction.
- Model simulation and its validation with human experiments use different platforms. The potential discrepancies between human and model tests (display and control mechanisms) may confound the validation results.

Considering these aspects where improvements can be made, the work reported in this chapter develops the usability of QN-ACTR and addresses Pew's three challenges (2008), including (1) the two methods of building a model to simplify model development, (2) the visualization of modeling results to articulate how the model works, and (3) an integrated human experiment interface for model validation. These features are evaluated using Nielsen's ten heuristics for user interface design (1994), which is one of the best-known methods of usability assessment (Hollingsed & Novick, 2007). The benefits and practice implications are discussed.

2. Method

Further developed on the basis of a previous version (Cao & Liu, 2011a), QN-ACTR is implemented in Micro Saint® Sharp, a C#-based discrete event simulation software package, which is also the platform for IMPRINT (Allender et al., 1997). Micro Saint® Sharp provides built-in features that support the usability development of QN-ACTR such as easy-to-use model setup and visualization.

2.1 Simplified model development

Two methods for model setup have been developed in QN-ACTR: a basic method using text-based syntaxes and a click-and-select interface.

The basic syntax method supports fast and direct model editing (i.e., copy and paste), which is designed for advanced users. Among the three parts of a QN-ACTR model (i.e., the task, the knowledge, and the parameters), syntaxes for the knowledge and the parameters are the same as in ACT-R. ACT-R syntaxes for the task are essentially Lisp codes, which supports the modeling of a wide range of tasks but requires users to learn the Lisp programming language. To simplify task building while keeping the capability to model a wide range of tasks, I have developed QN-ACTR task syntaxes that include different templates to model typical tasks. A task template is a general description for a type of experiments. A modeler can build a task by setting the template's parameters according to the experiment setup. For example, the Day-Block-Trial template can be used to define discrete-event experiments. This template has been used to replicate 17 models from the ACT-R 6.0 (v1.3) tutorial and three dual-task models from threaded cognition, as introduced in chapter 2. The World3D template can be used to define dynamic tasks such as driving. When combined, the two templates can define a dual-task of driving and another mental task such as arithmetic computation. These modeling syntaxes cover both static tasks, where display stimuli are not affected by previous responses, and dynamic tasks, where display stimuli are affected by previous responses.

Although the syntax method has its advantages, it may take a long time for novice users to learn. To support novice users, I have developed a click-and-select interface named Model Setup Assistant (MSA) that can help user generate model syntaxes. MSA is programmed in C# and added to Micro Saint® Sharp as a plug-in module. Providing both the syntax and the click-and-select interfaces can meet the needs of both novice and advanced users. This dual-mode method has been implemented in scientific software such as SPSS®.

Similar to the user interface in QN-MHP (Wu & Liu, 2009), MSA in QN-ACTR allows users to define a model by writing texts in tables and selecting options from menus. Users start from selecting the single or dual task scenario or loading demonstrations (Figure 18). Previous research models are included as demos (samples), such as ACT-R Tutorial models, driving models, and a transcription typing and reading dual-task model. New models can be made by modifying existing demos.

When a single task is selected with a template such as the Day-Block-Trial template for discrete-event experiments or the World3D template for driving, a task setup window will appear, asking users for the information needed in the experiment setup. When a dual-task is selected, two windows will appear each defining a task. For example, the Day-Block-Trial template in MSA asks for configuration settings such as whether a display stage in a trial will be terminated when all responses are detected (Figure 19) and setup details such as the number of trials in a block and the type of a display item (e.g., text, line, button, and tone) (Figure 20). The World3D template defining a driving task asks for road and car details such as lane width, road type, road length, and other cars' speed (Figure 21).

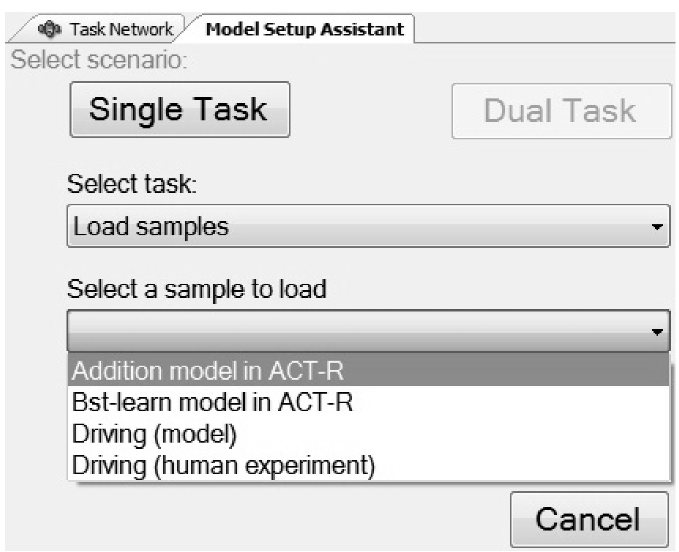


Figure 18. Screenshot of selecting a task using Model Setup Assistant (from Cao & Liu, 2012b).

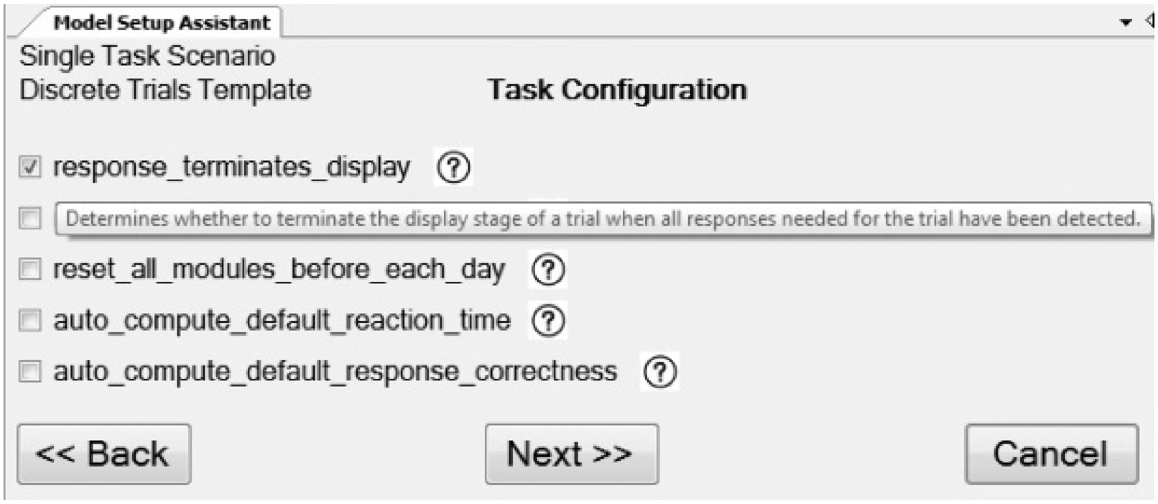


Figure 19. Screenshot of selecting task configuration options using Model Setup Assistant (from Cao & Liu, 2012b).

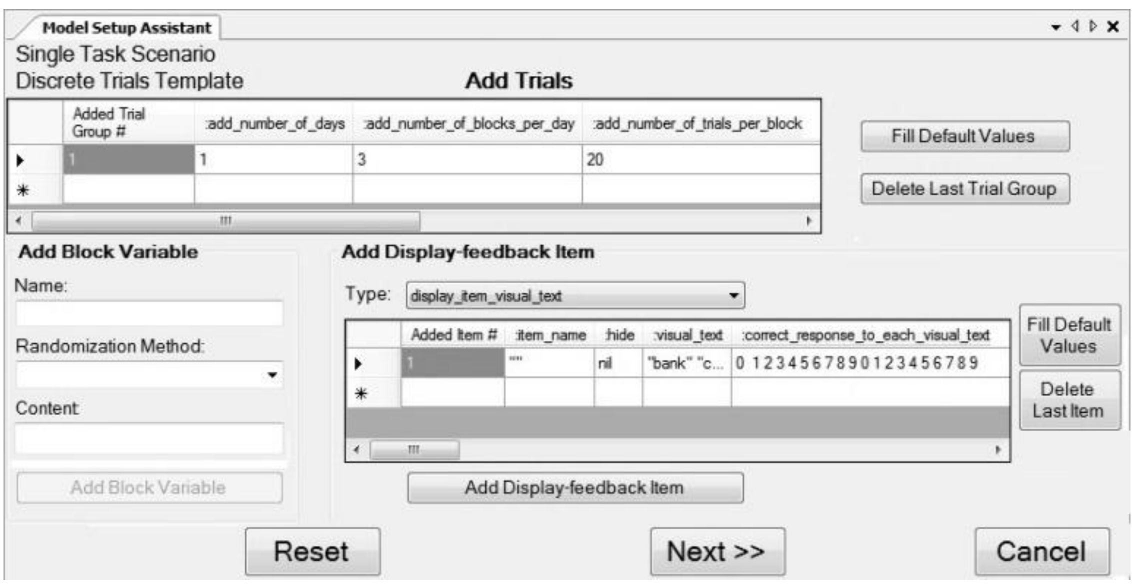


Figure 20. Screenshot of defining a discrete task using Model Setup Assistant (from Cao & Liu, 2012b).

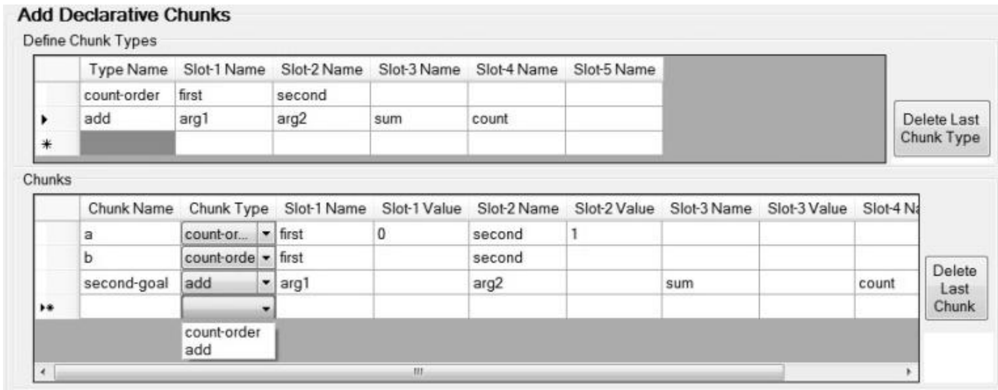
Figure 21. Screenshot of defining a driving task using Model Setup Assistant (from Cao & Liu, 2012b).

After defining the task, MSA can also assist users to define the knowledge and the parameter parts of a model. Since these syntaxes in QN-ACTR are the same as in ACT-R, if existing ACT-R codes are available, users can simply copy the ACT-R codes and paste into a QN-ACTR model. If no existing code available, users can define the knowledge, including chunks (Figure 22a) and production rules (Figure 22b), and set parameters (Figure 22c) with the assistance of MSA, by filling in tables and selecting from lists without the need to learn the knowledge description language used in ACT-R.

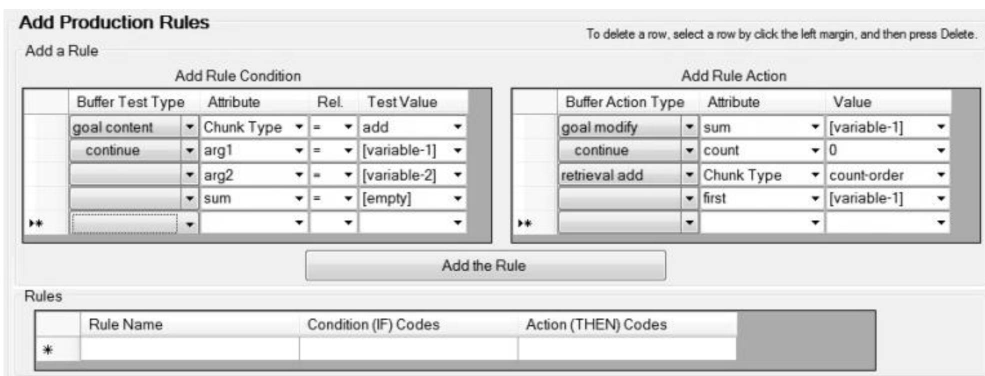
The model generated by MSA is also written in syntaxes. The resulted syntaxes can be saved, edited, or directly used to run the model. Simple modification of a model such as changing a few parameters can be easily achieved by directly editing the syntax file.

2.2 Visualization of 3D dynamic tasks

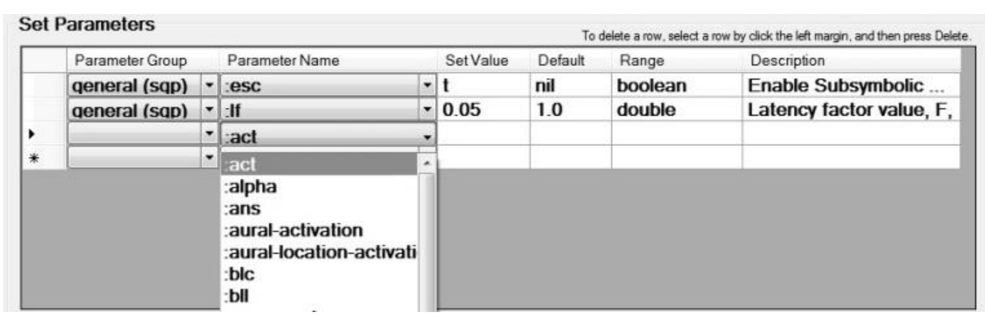
Previous work has developed the visualization of mental information processing, discrete experiment displays and controls, and the multi-dimensional mental workload (Cao & Liu, 2011a, 2011b). A new feature added to the visualization capabilities of QN-ACTR in this study is visualizing 3D dynamic tasks.



(a)



(b)



(c)

Figure 22. Screenshot of defining the knowledge and the parameter parts of a model using Model Setup Assistant, including (a) chunks, (b) production rules, and (c) parameters (from Cao & Liu, 2012b).

Using Animator3D in Micro Saint® Sharp, 3D dynamic tasks such as driving in single or dual task scenarios can be visualized in real time while the model is performing the task, which allows intuitive observation of model performance. The system refresh rate can be set by the user (10 ms by default). System dynamics such as speed, steering angle, and lateral deviation are visualized and recorded. Figure 23 illustrates that the model is driving a car while performing an arithmetic addition task. The model is following the car in the right lane and is visually focusing on the car. At the same time, the model is speaking “three” in response to the question of “1 + 2,” which is displayed through the auditory channel and visualized on the right hand side of the figure.

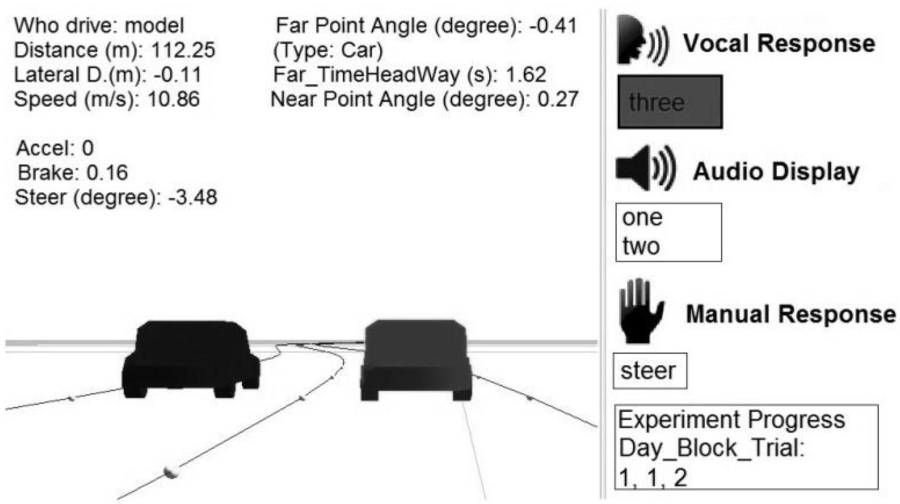


Figure 23. Visualization of a driving and arithmetic addition dual-task in QN- ACTR (from Cao & Liu, 2012b).

2.3 Integrated human experiment interface

The same task interface with which the model interacts can also serve as the interface for human participants to complete the same tasks. I have developed a human driving interface in QN-ACTR that supports simulated driving experiments with steering wheels and pedals. This feature allows the model and the human to perform and be compared in the same tasks with identical interfaces, with no need to replicate the real world experiment system in the modeling platform for the model to interact with (e.g., Salvucci, 2006). Using the same experiment platform avoids any discrepancy between human and model tests due to the experiment setup.

3. Findings

The usability development of QN-ACTR is evaluated using Nielsen's ten heuristics for user interface design (1994).

Visibility of system status. MSA always shows the stage of model development at the top-left corner. The visualization of the knowledge and the task keeps users informed about what is going on in the model during the simulation. Buttons in MSA are dimmed and disabled when their actions cannot be performed in some cases. Program responses and feedbacks are immediate with no delay.

Match between system and the real world. All the column headers in MSA tables and the items in menus use self-explanatory phrases without abbreviation. The steps of modeling in MSA follow the logical order shown in Figure 4. Full names and detailed descriptions are shown for each abbreviated ACT-R parameter name (Figure 22c).

User control and freedom. MSA supports undo (e.g., change the road name, delete a chunk, and reset a table) and redo (e.g., go back to the previous stage, and then go next again). A cancel button is provided at each stage to exit the setup at any time, and then users can restart MSA if needed.

Consistency and standards. Definitions and names are used consistently throughout all modeling steps. Tables and menus follow similar layouts and styles. Button position is the same between templates and stages.

Error prevention. The use of menu selection in MSA tables prevents the input of invalid items. Table cells automatically perform validation check, and users are notified when an input is of an invalid type or out of the valid range. Duplicated names assigned by users (e.g., chunk names) are automatically revised to prevent run-time errors. Syntax errors are also reported before the simulation starts.

Recognition rather than recall. MSA provides menus for users to select their options and tables to fill in. Model developing knowledge is provided to users in the interface. For example, users do not have to learn any modeling syntax. Instead, they can describe the model in natural language and fill in blanks or select items (Figure 22a, b). The default value, valid range, and description of model parameters are displayed for the users (Figure 22c).

Flexibility and efficiency of use. The syntax method and MSA cater to both inexperienced and experienced users. Experienced user can speed up the modeling work by directly copying and editing syntaxes. Syntaxes for the knowledge and the parameters can also be directly copied from ACT-R codes.

Aesthetic and minimalist design. MSA tables and menus are organized and aligned in groups. Introductions and explanations are concise.

Help users recognize, diagnose, and recover from errors. Error messages are expressed in plain language (no codes) and precisely indicate the problem. For example, "Error! Set General Parameters needs para_name: :lf to be a double rather than: nil."

Help and documentation. Help information is embedded in MSA. For example, the corresponding help information is shown when the mouse rests on the question mark beside a task configuration item (Figure 19). A QN-ACTR user manual has also been developed to provide detailed instructions.

4. Discussion

A cognitive architecture is both an integrated theory of cognition and a computerized simulation platform that can be used for cognitive engineering applications. The complexity of cognitive architectures allows the modeling of complex cognitive mechanisms, but at the same time, increases the difficulty to learn and use them as cognitive engineering tools. The work reported in this chapter developed the usability of an integrated general-purpose cognitive architecture, QN-ACTR, which integrates two complementary cognitive architectures QN and ACT-R. The aim is to provide easy-to-use modeling for both inexperienced and experienced users while keeping the capability to model a wide range of tasks.

This chapter reports the usability development in QN-ACTR that addresses the three challenges in human performance modeling (Pew, 2008). The completed work includes (1) the two methods of building a model to simplify model development, (2) the visualization of modeling results to articulate how the model works, and (3) an integrated human experiment interface for model validation.

Evaluated using heuristic evaluation for user interface design (Nielsen, 1994), the usability development of QN-ACTR has met the usability guidelines. In particular, Model Setup Assistant, a click-and-select user interface, simplifies model development and allows model setup by selecting from menu items and filling in blanks. Users can describe the model following natural language and experiment logic without the need to learn any programming or cognitive engineering language. Auto-check functions have also been developed to prevent modeling errors. In addition, a new feature of visualizing 3D dynamic tasks is added to the visualization capabilities of QN-ACTR. Dynamic tasks such as driving in single or dual task scenarios can be visualized in real time, which allows intuitive observation of model performance.

A human experiment interface has also been integrated into QN-ACTR, which allows the model and the human to perform and be compared in the same tasks with identical display and control setups. This method helps eliminate the need to replicate the real world experiment system in the modeling platform for the model to interact with and therefore avoids any discrepancy between human and model tests due to the experiment setup.

In conclusion, the usability development of QN-ACTR supports easy-to-use modeling for cognitive engineering applications. It allows users who are not experts of cognitive modeling to explore its application in human factors tests and evaluation. Since QN-ACTR uses the same syntaxes as ACT-R to define chunks and production rules and set parameters, Model Setup Assistant in QN-ACTR can also generate modeling codes for ACT-R models and help simplify the model development in ACT-R. Future studies will further improve the usability of QN-ACTR as a cognitive engineering tool and examine the results with empirical usability tests.

Chapter 8. Conclusions and Future Research

1. Summary of Thesis

As the complexity of human-machine systems grows rapidly, there is an increasing need for human factors theories and computational methods that can quantitatively model and simulate human performance and mental workload in complex multi-task scenarios. In response to this need, I have developed and evaluated an integrated cognitive architecture named QN-ACTR, which integrates two previously isolated but complementary cognitive architectures – Queueing Network (QN) and Adaptive Control of Thought-Rational (ACT-R). Combining their advantages and overcoming the limitations of each method, QN-ACTR possesses the benefits of modeling a wider range of tasks including multi-tasks with complex cognitive activities that existing methods have difficulty to model.

QN-ACTR is currently implemented in Micro Saint® Sharp, a general purpose discrete event simulation platform based on C# language. The source codes currently have about 30,000 lines. The implementation of ACT-R functionality has been verified in the simulation of 20 typical cognitive tasks that have been modeled by ACT-R. It has been shown that QN-ACTR can produce the same performance results as ACT-R did. In addition, QN-ACTR has successfully modeled skilled transcription typing that was previously modeled by QN-MHP and, more importantly, modeled multitasking performance involved in typing and reading comprehension or a concurrent foot pedal reaction task. These results showed that QN-ACTR can model what have been modeled by ACT-R or QN-MHP and have advantages in modeling cognitive multitasking performance that each previous method alone has difficulty to model.

After the programming and verification of QN-ACTR, I conducted a laboratory experiment to examine medical diagnostic decision making with concurrent information tasks, which contains critical and practical human factors issues in the healthcare work environment. The results showed that diagnostic performance was negatively affected by a complex concurrent memorization task that required participants to listen to verbal updates and remember information about other patients while performing the diagnostic task. In contrast, a simple concurrent sound monitoring task did not affect diagnostic performance. Both types of concurrent tasks significantly increased mental workload.

The human results were then compared with QN-ACTR simulation results of the same tasks. During the development of computational models for the diagnostic decision multi-tasks, a new concept from the QN perspective was implemented in QN-ACTR to allow the modeling of the concurrent tasks that involve multiple controlled processes. The key concept is a filtering discipline that allows cognitive resources to be exclusively occupied by one of the concurrent tasks when necessary, instead of switching between the tasks as frequently as possible as used in previous modeling methods. The simulations of dual-tasks involving diagnostic decision making and patient status tracking showed that the new discipline is necessary to model human performance and mental workload, demonstrating the improved modeling capability of QN-ACTR.

The benefits of QN-ACTR in modeling cognitive multitasking performance were further demonstrated in the simulation of lane keeping and speech comprehension dual-task scenarios from the ground transportation domain. The modeling results were similar to the human results for both performance and mental workload. The previously proposed filtering discipline from the QN perspective was again shown to be necessary to model complex cognitive multitasking performance.

All together, these models that covered a wide range of human factors domains including human-computer interaction, healthcare, and transportation demonstrated the advantages of integrating QN and ACT-R, the two previously isolated but complementary cognitive architectures. ACT-R's symbolic mechanisms provided the bases to model complex cognitive activities such as diagnostic decision and reading comprehension, whereas QN mechanisms provided the bases to model multi-task scheduling at the local server level without the need of any executive control rule.

Theoretically, there has been a long debate regarding the use of executive control rules in modeling human performance, especially dual-task performance (Byrne & Anderson, 2001; D. E. Meyer & Kieras, 1997; Wu & Liu, 2008a). Since human experimental results have not been sufficient to verify the existence of executive control rules, a simpler theory that can explain the same amount or more human results without using the assumption of task-specific executive control rules is generally preferred.

In particular, the filtering discipline, which has not been used in cognitive modeling or human performance modeling before this dissertation study, was validated in both the diagnostic decision making dual-task and the driving dual-task scenarios. In essence, the filtering discipline allows mental resources to exclusively process one of the multiple concurrent tasks when necessary. It is different from the first-in-first-out queueing discipline used in some previous studies that always alternates the allocation of mental resources between concurrent tasks when possible. This filtering discipline is the only new mechanism proposed and examined in this dissertation study that was not previously used in either QN or ACT-R cognitive architectures. Future research is needed to collect more evidence and further examine this filtering discipline, especially establishing a set of consistent principles regarding how to determine the rules that require follow-up processing. Nevertheless, as examined and demonstrated in modeling both the diagnostic decision dual-task and the driving dual-task with complex cognitive activities, the filtering discipline has been shown to be necessary to successfully simulate human performance and mental workload in this dissertation study. Besides the cognitive engineering values of the QN perspective in terms of human performance modeling, the findings from this dissertation also support a theoretical speculation that queueing mechanisms may have their deeper roots in human brain neurology. The QN perspective may guide future studies to look for potential neurological evidence with the collection of neurobiological data such as neuroimaging and electroencephalography. Although very limited previous work has been conducted to examine queueing mechanisms in the human brain, a recent study has shown that at the micro level of synapses and vesicles, the human brain indeed exhibits some QN mechanisms (Holt & Jahn, 2004).

Finally, to support industrial applications of QN-ACTR, such as interface evaluation and rapid design prototyping, and to allow industrial engineers who are not experts of human performance modeling to use this promising computational modeling method, I have developed the usability features of QN-ACTR to facilitate the use of QN-ACTR as a cognitive engineering tool.

2. Conclusions

- Literature review of previous human performance modeling methods identified the need for integrated cognitive architectures to model cognitive multitasking performance involved in human-machine interactions that are becoming increasingly complex and demanding.
- QN-ACTR, which integrates two previously isolated but complementary cognitive architectures – Queueing Network (QN) and Adaptive Control of Thought-Rational (ACT-R), was developed and programmed to address the challenge of how to computationally model cognitive multitasking performance.
- Verification process showed that QN-ACTR can model cognitive single-tasks that have been modeled by ACT-R and perceptual-motor multi-tasks that have been modeled by QN-MHP. QN-ACTR can also model cognitive multitasking performance involved in typing and reading comprehension that previous methods have difficulty to model.
- A laboratory experiment was designed, programmed, and conducted to examine a common cognitive multitasking scenario in the healthcare domain, i.e., medical diagnostic decision making with concurrent information task performance. The results provided new insights into the cognitive mechanisms underlying diagnostic decision and physician multitasking, which have important implications for the control and improvement of healthcare quality. The results also provided human data to test and examine QN-ACTR cognitive models.
- Computational models built in QN-ACTR have successfully modeled the human data collected from the diagnostic decision multi-task experiment. QN theory played an important role in the modeling of cognitive multitasking performance and mental workload.

- To further examine QN-ACTR's modeling capability in complex multi-task scenarios, a simulated driving experiment was conducted to collected detailed human performance and mental workload results in a lane keeping and speech comprehension dual-task scenario. The results showed that the concurrent comprehension task had no significant effect on lane keeping performance, whereas the correct rate of the comprehension task was reduced in the dual-task condition compared with the comprehension single-task condition. Mental workload was significantly higher in the dual-task condition compared with the single-task conditions.
- A QN-ACTR model simulated the human results collected from the lane keeping and speech comprehension experiment, producing results similar to the human data for both performance and mental workload measures.
- The filtering discipline from the QN perspective proposed and examined in this dissertation study has been shown to be necessary to model complex cognitive multitasking performance, as demonstrated in both the diagnostic decision dual-task and driving dual-task scenarios. It is a more detailed account for multi-task scheduling at the local server level than the previously used simple first-in-first-out queueing discipline. The results also suggest that QN mechanisms may be one of the fundamental mechanisms of cognitive psychology. The QN-ACTR architecture evaluated in this dissertation could be used to guide future research looking for neurological evidence of information queues in the brain.
- The integration of QN and ACT-R moves one step forward towards a unified theory of cognition (Newell, 1990), which is essential for understanding all aspects of the human mind as a whole.
- Usability features of QN-ACTR have also been developed and programmed to support industrial applications of QN-ACTR for industrial engineers to use this promising computational modeling method as a cognitive engineering and design tool.

3. Future Research

3.1 Modeling human performance in other domains

This thesis has tested and examined QN-ACTR in two domains – human-computer interaction and healthcare. Other human factors domains can also be the arena to apply this computational modeling method, because QN-ACTR is a generic method without any specific domain. The task and knowledge used to setup a model can be from any domain and any interface with which human operators are able to interact.

In ground transportation, future research can connect QN-ACTR with driving simulation programs such as TORCS (http://torcs.sourceforge.net/) that has been used to conduct human driving experiments (Cao et al., 2013). A platform integrating TORCS and QN-ACTR will allow humans and models to perform and be compared in identical driving scenarios.

In the area of uninhabited vehicle supervisory control, there is also a need for computational modeling methods to explain cognitive mechanisms underlying human performance and evaluate designs and new policies. Future research can connect QN-ACTR with uninhabited vehicle simulators such as RESCHU (MIT Humans and Automation Laboratory, 2009). QN-ACTR's capability in modeling complex cognitive multitasking can be used to model operators' performance and mental workload in the uninhabited vehicle supervisory control tasks, which are often very demanding and stressful.

QN-ACTR can also be applied in aerospace cockpit control scenarios to model and simulate the performance and mental workload of pilots and astronauts. In such high-stake safety domains, erroneous actions and reaction in the scale of milliseconds are very critical to system performance and safety.

In all these human factors domains, domain-specific tools can be developed and designed to meet the need of human factors engineers working in the specific domain. For example, future work can establish connection protocols between QN-ACTR and websites to allow rapid prototyping and evaluation of website interface design.

3.2 Modeling human-human and human-automation trust in multi-operator systems

The current version of QN-ACTR contains only one mental structure representing one human operator. Future research can make another duplication of the mental structure and form a two-operator framework. Such multi-operator framework will allow the examination and simulation of human-human communication and human-automation trust in an integrated computational architecture.

3.3 Modeling haptic/tactile perception and proprioception

The current version of QN-ACTR, as well as other cognitive architectures, is limited in perceptual channels, containing only the visual and auditory perceptual modules. The lack of haptic/tactile modeling mechanisms becomes an increasingly important issue, because more and more empirical studies have been conducted to explore the haptic/tactile channel as an additional information pathway to deliver information to operators, when their visual and auditory channels are occupied by other necessary information. In addition, proprioception is critical to the control and execution of accurate body movements but is also not covered in QN-ACTR. The lack of proprioception mechanisms may also be a reason of the remaining difference between the model and human results. There is a strong need for these other mechanisms beyond traditional cognitive aspects to be added in this integrated cognitive architecture.

3.4 Integrating cognitive and physical models

The current version of QN-ACTR focuses on the cognitive aspects of human factors and ergonomics. Another important aspect is physical ergonomics, which includes research areas such as anthropometry and biomechanics. Previous research has demonstrated the possibility and initial work to integrated QN cognitive models with physical ergonomic models (Fuller, Reed, & Liu, 2010). Future research can integrated QN-ACTR with physical ergonomics models to form a holistic view of the entire human-machine interaction.

3.5 QN-Java project

The current implementation of QN-ACTR is in a commercial program. Although it provided the convenience for mental structure setup and some visualization features, it is not ideal for the future academic development of QN-ACTR, which needs to be publicly and freely available and accessible for all researchers around the world. We are in the process of porting QN-ACTR to a Java implementation. Java is selected because it is open-sourced, free, and cross-platform. Many researchers around the world have used and developed their human factors projects in Java, such as the uninhabited vehicle supervisory control simulator RESCHU (MIT Humans and Automation Laboratory, 2009). There is also an open-sourced QN simulation program available in Java called Java Modelling Tools (http://jmt.sourceforge.net/), which serves as the foundation to build QN-ACTR cognitive architecture and implement cognitive algorithms.

3.6 Looking for neurological evidence of queueing mechanisms in the brain

The modeling results from this dissertation showed the necessity of queueing mechanisms to simulate human behavioral performance. A reasonable theoretical speculation is whether such queueing mechanisms exist in the brain from the cognitive-neuroscience point of view. QN-ACTR with its QN perspective may serve as a theoretical framework to guide future studies in the search for potential neurological evidence of queueing mechanisms with the collection of neurobiological data such as neuroimaging and electroencephalography data. Future studies along this direction have important values for cognitive psychology. Although QN has been used to model and simulate human behavioral performance, there is still a lack of neurological studies examining queueing mechanisms in the brain.

3.7 Modeling effort and mind-wandering

Effort was not included in the experimental designs of this dissertation but may have effects on performance and mental workload. Future research may test the use of a concurrent process of mind-wandering (i.e., task-unrelated thought) to model effort. Hypothetically, the mind-wandering concurrent process, as an extra task not required by formal task instructions, may compete with other concurrent tasks but does not create mental workload related to any required task. Higher effort may reduce the proportion of mind-wandering and therefore increase resource utilization and mental workload of the required tasks.

References

ACT-R Group. (2011). *ACT-R 6.0 Tutorial*: ACT-R Web site: http://act-r.psy.cmu.edu/.

Allender, L., Kelley, T., Archer, S., & Adkins, R. (1997). IMPRINT: The transition and further development of a soldier-system analysis tool. *MANPRINT Quarterly, 5*(1), 1-7.

Alm, H., & Nilsson, L. (1995). The effects of a mobile telephone task on driver behaviour in a car following situation. *Accident Analysis and Prevention, 27*(5), 707-715.

Altmann, E. M., & Gray, W. D. (2008). An integrated model of cognitive control in task switching. *Psychological Review, 115*(3), 602-639.

Alvarez, G., & Coiera, E. (2005). Interruptive communication patterns in the intensive care unit ward round. *International Journal of Medical Informatics, 74*(10), 791-796.

Amant, R. S., Freed, A. R., & Ritter, F. E. (2005). Specifying ACT-R models of user interaction with a GOMS language. *Cognitive Systems Research, 6*, 71–88.

Anderson, A. M., Mirka, G. A., Joines, S. M. B., & Kaber, D. B. (2009). Analysis of alternative keyboards using learning curves. *Human Factors, 51*(1), 35-45.

Anderson, J. R. (1974). Verbatim and propositional representation of sentences in immediate and long-term memory. *Journal of Verbal Learning and Verbal Behavior, 13*, 149-162.

Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review, 89*(4), 369–406.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111*(4), 1036–1060.

Anderson, J. R., Budiu, R., & Reder, L. M. (2001). A theory of sentence memory as part of a general theory of memory. *Journal of Memory and Language, 45*(3), 337–367.

Anderson, J. R., Fincham, J. M., Qin, Y., & Stocco, A. (2008). A central circuit of the mind. *Trends in Cognitive Sciences, 12*(4), 136-143.

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

Anderson, J. R., & Matessa, M. (1997). A production system theory of serial memory. *Psychological Review, 104*(4), 728-748.

Anderson, J. R., Reder, L. M., & Lebiere, C. (1996). Working memory: Activation limitations on retrieval. *Cognitive Psychology, 30*(3), 221-256.

Becic, E., Dell, G. S., Bock, K., Garnsey, S. M., Kubose, T., & Kramer, A. F. (2010). Driving impairs talking. *Psychonomic Bulletin & Review, 17*(1), 15-21.

Biron, A. D., Lavoie-Tremblay, M., & Loiselle, C. G. (2009). Characteristics of work interruptions during medication administration. *Journal of Nursing Scholarship, 41*(4), 330-336.

Blaauw, G. J. (1982). Driving experience and task demands in simulator and instrumented car: A validation study. *Human Factors, 24*(4), 473–486.
Brewer, N. T., Chapman, G. B., Schwartz, J. A., & Bergus, G. R. (2007). The influence of irrelevant anchors on the judgments and choices of doctors and patients. *Medical Decision Making, 27*(2), 203-211.
Brookhuis, K. A., de Vries, G., & de Waard, D. (1991). The effects of mobile telephoning on driving performance. *Accident Analysis and Prevention, 23*(4), 309-316.
Budiu, R., & Anderson, J. R. (2004). Interpretation-based processing: A unified theory of semantic sentence comprehension. *Cognitive Science*, *28*(1), 1–44.
Burgess, D. J. (2010). Are providers more likely to contribute to healthcare disparities under high levels of cognitive load? How features of the healthcare setting may lead to biases in medical decision making. *Medical Decision Making, 30*(2), 246-257.
Byers, J. C., Bittner, A. C., & Hill, S. G. (1989). Traditional and raw task load index (TLX) correlations: Are paired comparisons necessary? In A. Mital (Ed.), *Advances in Industrial Ergonomics and Safety* (pp. 481-485): Taylor & Francis.
Byrne, M. D., & Anderson, J. R. (1998). Perception and action. In J. R. Anderson & C. Lebiere (Eds.), *The Atomic Components of Thought* (pp. 167-200). Mahwah, NJ: Lawrence Erlbaum.
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.
Byrne, M. D., & Pew, R. W. (2009). A history and primer of human performance modeling. *Reviews of Human Factors and Ergonomics, 5*, 225-263.
Cao, S., & Liu, Y. (2011a). Integrating Queueing Network and ACT-R cognitive architectures. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 55*(1), 836-840.
Cao, S., & Liu, Y. (2011b). Mental workload modeling in an integrated cognitive architecture. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *55*(1), 2083-2087.
Cao, S., & Liu, Y. (2012a). ACTR-QN: Integrating Queueing Network and ACT-R cognitive architectures. In W. D. Gray (Organizer), Symposium on human performance modeling. *The 11th International Conference on Cognitive Modeling*, Berlin, Germany.
Cao, S., & Liu, Y. (2012b). An integrated cognitive architecture for cognitive engineering applications. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *56*(1), 323-327.
Cao, S., & Liu, Y. (2012c). QN-ACTR modeling of multitask performance of dynamic and complex cognitive tasks. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 56*(1), 1015-1019.
Cao, S., & Liu, Y. (2013a). Concurrent processing of vehicle lane keeping and speech comprehension tasks. *Accident Analysis and Prevention, 59*, 46-54.
Cao, S., & Liu, Y. (2013b). Queueing Network-ACTR modeling of concurrent tasks involving multiple controlled processes. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 57*(1), 768-772.

Cao, S., & Liu, Y. (2013c). Queueing network-adaptive control of thought rational (QN-ACTR): An integrated cognitive architecture for modelling complex cognitive and multi-task performance. *International Journal of Human Factors Modelling and Simulation 4*(1), 63-86.

Cao, S., & Liu, Y. (in press). Effects of concurrent tasks on diagnostic decision making: An experimental investigation. *IIE Transactions on Healthcare Systems Engineering*.

Cao, S., Qin, Y., & Shen, M. (2013). Modeling the effect of driving experience on lane keeping performance using ACT-R cognitive architecture. *Chinese Science Bulletin (Chinese Version), 58*(21), 2078-2086.

Carayon, P., Bass, E. J., Bellandi, T., Gurses, A. P., Hallbeck, M. S., & Mollo, V. (2011). Sociotechnical systems analysis in health care: A research agenda. *IIE Transactions on Healthcare Systems Engineering*, *1*(3), 145-160.

Card, S., Moran, T. P., & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Chisholm, C. D., Collison, E. K., Nelson, D. R., & Cordell, W. H. (2000). Emergency department workplace interruptions: Are emergency physicians "interrupt-driven" and "Multitasking"? *Academic Emergency Medicine*, *7*, 11.

Coiera, E. W., Jayasuriya, R. A., Hardy, J., Bannan, A., & Thorpe, M. E. C. (2002). Communication loads on clinical staff in the emergency department. *Medical Journal of Australia*, *176*(9), 415-418.

Colle, H. A., & Reid, G. B. (1998). Context effects in subjective mental workload ratings. *Human Factors, 40*(4), 591-600.

Collins, S., Currie, L., Patel, V., Bakken, S., & Cimino, J. J. (2007). Multitasking by clinicians in the context of CPOE and CIS use. *Studies in Health Technology and Informatics, 129*(Pt 2), 958-962.

Darwin, C. J., Turvey, M. T., & Crowder, R. G. (1972). An auditory analogue of the sperling partial report procedure: Evidence for brief auditory storage. *Cognitive Psychology*, *3*(2), 255-267.

Dick, A. O. (1974). Iconic memory and its relation to perceptual processing and other memory mechanisms. *Perception and Psychophysics, 16*(3), 575-596.

Drews, F. A., Yazdani, H., Godfrey, C. N., Cooper, J. M., & Strayer, D. L. (2009). Text messaging during simulated driving. *Human Factors*, *51*, 762-700.

Elkind, J. I., Card, S. K., Hochberg, J., & Huey, B. M. (1989). *Human Performance Models for Computer-Aided Engineering*. Washington, D.C.: National Academy Press.

Elstein, A. S., & Schwarz, A. (2002). Clinical problem solving and diagnostic decision making: Selective review of the cognitive literature. *BMJ (Clinical Research Ed.), 324*(7339), 729-732.

Engström, J., Johansson, E., & Östlund, J. (2005). Effects of visual and cognitive load in real and simulated motorway driving. *Transportation Research Part F, 8*, 97–120.

Eva, K. W., Hatala, R. M., LeBlanc, V. R., & Brooks, L. R. (2007). Teaching from the clinical reasoning literature: Combined reasoning strategies help novice diagnosticians overcome misleading information. *Medical Education, 41*(12), 1152-1158.

Evans, J. S. (2008). Dual-processing accounts of reasoning, judgment, and social cognition. *Annual Review of Psychology, 59*, 255-278.

Feyen, R. (2002). *Modeling human performance using the queuing network-model human processor (QN-MHP).* Doctoral Dissertation. Department of Industrial and Operations Engineering. University of Michigan.

Field, A. (2009). *Discovering Statistics Using SPSS* (3rd ed.). London: SAGE Publications.

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.

Fitts, P. M. (1964). Perceptual-motor skill learning. In A. W. Melton (Ed.), *Categories of human learning*. New York: Academic Press.

Fitts, P. M., & Peterson, J. R. (1964). Information capacity of discrete motor responses. *Journal of Experimental Psychology, 67*(2), 103-112.

Fougnie, D. (2008). The relationship between attention and working memory. In N. B. Johansen (Ed.), *New Research on Short-Term Memory*: Nova Science Publishers, Inc.

Fu, W.-T., & Anderson, J. R. (2006). From recurrent choice to skill learning: A reinforcement-learning model. *Journal of Experimental Psychology: General, 135*(2), 184-206.

Fuller, H. J. A., Reed, M. P., & Liu, Y. (2010). Integrating physical and cognitive human models to represent driving behavior. *Human Factors and Ergonomics Society Annual Meeting Proceedings, 54*, 1042-1046.

Fuller, H. J. A., Reed, M. P., & Liu, Y. (2012). Integration of physical and cognitive human models to simulate driving with a secondary in-vehicle task. *IEEE Transactions on Intelligent Transportation Systems, 13*(2), 967-972.

Gelman, R., & Butterworth, B. (2005). Number and language: How are they related? *Trends in Cognitive Sciences, 9*(1), 7-10.

Gentner, D. R. (1983). The acquisition of typewriting skill. *Acta Psychologica*, *54*(1-3), 233–248.

Gil, G.-H. (2010). *An accessible cognitive modeling tool for evaluation of human-automation interaction in the systems design process.* Doctoral Dissertation. North Carolina State University.

Gonzalez, C. (2005). Task workload and cognitive abilities in dynamic decision making. *Human Factors, 47*(1), 92-101.

Graber, M. L., Franklin, N., & Gordon, R. (2005). Diagnostic error in internal medicine. *Archives of Internal Medicine, 165*(13), 1493-1499.

Groeger, J. A. (2000). *Understanding Driving: Applying Cognitive Psychology to a Complex Everyday Task*. Philadelphia: Psychology Press.

Hancock, P. A., & Meshkati, N. (1988). *Human mental workload*. New York: North-Holland.

Hart, S. G. (2006). NASA-task load index (NASA-TLX): 20 years later. *Human Factors and Ergonomics Society Annual Meeting Proceedings, 50*, 904-908.

Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of experimental and theoretical research. In P. A. Hancock & N. Meshkati (Eds.), *Human mental workload* (pp. 139-183). New York: North-Holland.

Hazeltine, E., Teague, D., & Ivry, R. B. (2002). Simultaneous dual-task performance reveals parallel response selection after practice. *Journal of Experimental Psychology: Human Perception and Performance*, *28*(3), 527-545.

Heathcote, A., Brown, S., & Mewhort, D. J. K. (2000). The power law repealed: The case for an exponential law of practice. *Psychonomic Bulletin and Review, 7*(2), 185–207.

Hershman, R. L., & Hillix, W. A. (1965). Data processing in typing: Typing rate as a function of kind of material and amount exposed. *Human Factors, 7*(5), 483-492.

Hollingsed, T., & Novick, D. G. (2007). Usability inspection methods after 15 years of research and practice. *SIGDOC '07: Proceedings of the 25th Annual ACM International Conference on Design of Communication*, 249-255.

Hollnagel, E. (1993). The phenotype of erroneous actions. *International Journal of Man-Machine Studies, 39*(1), 1-32.

Hollnagel, E. (2007). Human Error: Trick or Treat? In *Handbook of applied cognition* (pp. 219-238). Chichester, UK: Wiley.

Holt, M., & Jahn, R. (2004). Synaptic vesicles in the fast lane. *Science, 303*(5666), 1986-1987.

Hoonakker, P., Carayon, P., Gurses, A. P., Brown, R., Khunlertkit, A., McGuire, K., et al. (2011). Measuring workload of ICU nurses with a questionnaire survey: The NASA Task Load Index (TLX). *IIE Transactions on Healthcare Systems Engineering, 1*(2), 131-143.

Horrey, W. J., & Wickens, C. D. (2006). Examining the impact of cell phone conversations on driving using meta-analytic techniques. *Human Factors, 48*(1), 196–205.

Inhoff, A. W., & Wang, J. (1992). Encoding of text, manual movement planning, and eye-hand coordination during copytyping. *Journal of Experimental Psychology: Human Perception and Performance, 18*(2), 437-448.

Jenness, J. W., Lerner, N. D., Mazor, S., Osberg, J. S., & Tefft, B. C. (2008). *Use of Advanced In-Vehicle Technology by Young and Older Early Adopters. Survey Results on Adaptive Cruise Control Systems (DOT HS 810 917)*: National Highway Traffic Safety Administration.

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

John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). *Predictive human performance modeling made easy*. Paper presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, Vienna, Austria.

Jones, G., & Ritter, F. E. (1998). Simulating development by modifying architectures. *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society.* , 543-548.

Just, M. A., Keller, T. A., & Cynkar, J. (2008). A decrease in brain activation associated with driving when listening to someone speak. *Brain Research, 1205*, 70-80.

Kahneman, D. (2003). A perspective on judgment and choice: Mapping bounded rationality. *American Psychologist, 58*(9), 697–720.

Keinan, G. (1987). Decision making under stress: Scanning of alternatives under controllable and uncontrollable threats. *Journal of Personality and Social Psychology, 52*(3), 639-644.
Klayman, J., & Ha, Y.-w. (1987). Confirmation, disconfirmation, and information in hypothesis testing. *Psychological Review, 94*(2), 211-228.
Kovacs, G., & Croskerry, P. (1999). Clinical decision making: An emergency medicine perspective. *Academic Emergency Medicine, 6*(9), 947-952.
Kubose, T. T., Bock, K., Dell, G. S., Garnsey, S. M., Kramer, A. F., & Mayhugh, J. (2006). The effects of speech production and speech comprehension on simulated driving performance. *Applied Cognitive Psychology, 20*, 43-63.
Kulatunga-Moruzi, C., Brooks, L. R., & Norman, G. R. (2001). Coordination of analytic and similarity-based processing strategies and expertise in dermatological diagnosis. *Teaching and Learning in Medicine, 13*(2), 110-116.
Laird, J. E. (2008). Extending the Soar Cognitive Architecture. *Proceedings of the First Conference on Artificial General Intelligence (AGI-08).*
Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence, 33*(1), 1–64.
Lamble, D., Kauranen, T., Laakso, M., & Summala, H. (1999). Cognitive load and detection thresholds in car following situations: Safety implications for using mobile (cellular) telephones while driving. *Accident Analysis and Prevention, 31,* 617–623.
Laxmisan, A., Hakimzada, F., Sayan, O. R., Green, R. A., Zhang, J., & Patel, V. L. (2007). The multitasking clinician: Decision-making and cognitive demand during and after team handoffs in emergency care. *International Journal of Medical Informatics, 76,* 801–811.
Lee, Y.-H., & Liu, B.-S. (2003). Inflight workload assessment: Comparison of subjective and physiological measurements. *Aviation, Space, and Environmental Medicine, 74*(10), 1078-1084.
Lenné, M. G., Dietze, P. M., Triggs, T. J., Walmsley, S., Murphy, B., & Redman, J. R. (2010). The effects of cannabis and alcohol on simulated arterial driving: Influences of driving experience and task demand. *Accident Analysis and Prevention, 42,* 859-866.
Lewis, R. L., & Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science, 29*(3), 375–419.
Liang, Y., & Lee, J. D. (2010). Combining cognitive and visual distraction: Less than the sum of its parts. *Accident Analysis and Prevention, 42*, 881–890.
Liu, Y. (1996). Queueing network modeling of elementary mental processes. *Psychological Review, 103*(1), 116–136.
Liu, Y. (1997). Queueing network modeling of human performance of concurrent spatial and verbal tasks. *IEEE Transactions on Systems, Man, Cybernetics, 27*(2), 195-207.
Liu, Y. (2009). QN-ACES: Integrating queueing network and ACT-R, CAPS, EPIC, and Soar architectures for multitask cognitive modeling. *International Journal of Human-Computer Interaction, 25*(6), 554–581.
Liu, Y., Feyen, R., & Tsimhoni, O. (2006). Queueing Network-Model Human Processor (QN-MHP): A computational architecture for multitask performance in human-

machine systems. *ACM Transactions on Computer-Human Interaction, 13*(1), 37–70.
Logan, G. D. (1982). On the ability to inhibit complex movements: A stop-signal study of typewriting. *Journal of Experimental Psychology: Human Perception and Performance, 8*(6), 778-792.
Matthews, R., Legg, S., & Charlton, S. (2003). The effect of cell phone type on drivers subjective workload during concurrent driving and conversing. *Accident Analysis and Prevention, 35*, 451-457.
McGeoch, C. (1992). Analyzing algorithms by simulation: Variance reduction techniques and simulation speedups. *ACM Computing Surveys*, *24*(2), 195-212.
McKnight, A. J., & McKnight, A. S. (1993). The effect of cellular phone use upon driver attention. *Accident Analysis and Prevention, 25*(3), 259-265.
Mehlhorn, K., & Marewski, J. N. (2011). Racing for the city: The recognition heuristic and compensatory alternatives *Proceedings of the Interdisciplinary Workshop on Cognitive Neuroscience*, Educational Research and Cognitive Modeling. Delmenhorst, Germany.
Meyer, D. E., Abrams, R. A., Kornblum, S., Wright, C. E., & Keith Smith, J. E. (1988). Optimality in human motor performance: Ideal control of rapid aimed movements. *Psychological Review, 95*(3), 340-370.
Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance: Part 1. Basic mechanisms. *Psychological Review, 104*(1), 3–65.
Monsell, S. (2003). Task switching. *Trends in Cognitive Sciences, 7*(3), 134-140.
Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.
Newell, A. (Ed.). (1973). *You can't play 20 questions with nature and win: Projective comments on the papers of this symposium*. New York: Academic Press.
Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1-55). Hillsdale, NJ: Erlbaum.
Nielsen, J. (1994). Heuristic evaluation. In J. Nielsen & R. L. Mack (Eds.), *Usability Inspection Methods*. New York, NY: John Wiley & Sons.
Norman, G. (2009). Dual processing and diagnostic errors. *Advances in Health Sciences Education, 14*(1), 37-49.
Oberauer, K., & Kliegl, R. (2004). Simultaneous Cognitive Operations in Working Memory After Dual-Task Practice. *Journal of Experimental Psychology: Human Perception and Performance, 30*(4), 689-707.
Olsen, D. E., & Sticha, D. (2005). Interactive simulation training: Computer simulated standardized patients for medical diagnosis. In *Studies in Health Technology and Informatics. Medicine Meets Virtual Reality 14: Accelerating Change in Healthcare: Next Medical Toolkit* (Vol. 119, pp. 413-415): IOS Press.
Owens, P. L., Barrett, M. L., Gibson, T. B., Andrews, R. M., Weinick, R. M., & Mutter, R. L. (2010). Emergency department care in the United States: A profile of national data sources. *Annals of Emergency Medicine, 56*(2), 150-165.
Pashler, H. (1994). Dual-task interference in simple tasks: Data and theory. *Psychological Bulletin, 116*(2), 220-244.

Patel, V. L., Gutnik, L. A., Karlin, D. R., & Pusic, M. (2008). Calibrating urgency: Triage decision-making in a pediatric emergency department. *Advances in Health Sciences Education*, *13*(4), 503-520.

Patton, E. W., & Gray, W. D. (2010). SANLab-CM: A tool for incorporating stochastic operations into activity network modeling. *Behavior Research Methods, 42*(3), 877-883.

Pavlik, P. I., & Anderson, J. R. (2005). Practice and forgetting effects on vocabulary memory: An activation-based model of the spacing effect. *Cognitive Science, 29*(4), 559–586.

Peters, E., Diefenbach, M. A., Hess, T. M., & Västfjäll, D. (2008). Age differences in dual information-processing modes. *Cancer*, *113*(S12), 3556-3567.

Pew, R. W. (2008). Foreword. In D. C. Foyle & B. L. Hooey (Eds.), *Human Performance Modeling in Aviation*. Boca Raton, FL: CRC Press/Taylor & Francis Group.

Price, C. J. (1998). The functional anatomy of word comprehension and production. *Trends in Cognitive Sciences, 2*(8), 281-287.

Rakauskas, M. E., Gugerty, L. J., & Ward, N. J. (2004). Effects of naturalistic cell phone conversations on driving performance. *Journal of Safety Research, 35*, 453– 464.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin, 124*(3), 372–422.

Recarte, M. A., & Nunes, L. M. (2003). Mental workload while driving: Effects on visual search, discrimination, and decision making. *Journal of Experimental Psychology: Applied, 9*(2), 119–137.

Redelmeier, D. A., & Tibshirani, R. J. (1997). Association between cellular-telephone calls and motor vehicle collisons. *The New England Journal of Medicine, 336*(7), 453-458.

Reed, M. P., & Green, P. A. (1999). Comparison of driving performance on-road and in a low-cost simulator using a concurrent telephone dialling task. *Ergonomics, 42*(8), 1015-1037.

Reyna, V. F. (2008). Theories of medical decision making and health: An evidence-based approach. *Medical Decision Making, 28*(6), 829-833.

Ritter, F. E., & Reifers, A. L. (2007). Lessons from defining theories of stress. In W. D. Gray (Ed.), *Integrated Models of Cognitive Systems (IMoCS)* (pp. 254-262). New York: Oxford University Press.

Roland, P. (1993). *Brain Activation*. New York: Wiley.

Roland, P. E., Larsen, B., Lassen, N. A., & Skinhøj, E. (1980). Supplementary motor area and other cortical areas in organization of voluntary movements in man. *Journal of Neurophysiology, 43*(1), 118-136.

Ruthruff, E., Johnston, J. C., & Van Selst, M. (2001). Why practice reduces dual-task interference. *Journal of Experimental Psychology: Human Perception and Performance, 27*(1), 3-21.

Salthouse, T. A. (1984). Effects of age and skill in typing. *Journal of Experimental Psychology: General*, *113*(3), 345-371.

Salthouse, T. A. (1985). Anticipatory processing in transcription typing. *Journal of Applied Psychology, 70*(2), 264-271.

Salthouse, T. A. (1986). Perceptual, cognitive, and motoric aspects of transcription typing. *Psychological Bulletin, 99*(3), 303-319.

Salthouse, T. A., & Saults, J. S. (1987). Multiple spans in transcription typing. *Journal of Applied Psychology, 72*(2), 187-196.

Salvucci, D. D. (2001). An integrated model of eye movements and visual encoding. *Cognitive Systems Research, 1*(4), 201-220.

Salvucci, D. D. (2006). Modeling driver behavior in a cognitive architecture. *Human Factors, 48*(2), 362–380.

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

Salvucci, D. D., & Beltowska, J. (2008). Effects of memory rehearsal on driver performance: Experiment and theoretical account. *Human Factors, 50*(5), 834-844.

Salvucci, D. D., & Gray, R. (2004). A two-point visual control model of steering. *Perception, 33*(10), 1233-1248.

Salvucci, D. D., & Lee, F. J. (2003). *Simple cognitive modeling in a complex cognitive architecture.* Paper presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, Ft. Lauderdale, Florida, USA.

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

Schneider, W., & Shiffrin, R. M. (1977). Controlled and automatic human information processing: I. Detection, search, and attention. *Psychological Review, 84*(1), 1-66.

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

Schunn, C. D., & Gray, W. D. (2002). Introduction to the special issue on computational cognitive modeling. *Cognitive Systems Research, 3,* 1–3.

Shiffrin, R. M., & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending and a general theory. *Psychological Review, 84*(2), 127-190.

Sloman, S. A. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin, 119*(1), 3-22.

Speier, C., Valacich, J. S., & Vessey, I. (1999). The influence of task interruption on individual decision making: An information overload perspective. *Decision Sciences, 30*(2), 337-360.

Speier, C., Vessey, I., & Valacich, J. S. (2003). The effects of interruptions, task complexity, and information presentation on computer-supported decision-making performance. *Decision Sciences, 34*(4), 771-797.

Squire, L. R. (2004). Memory systems of the brain: A brief history and current perspective. *Neurobiology of Learning and Memory*, *82*(3), 171–177.

Strayer, D. L., Drews, F. A., & Johnston, W. A. (2003). Cell phone-induced failures of visual attention during simulated driving. *Journal of Experimental Psychology: Applied*, *9*(1), 23-32.

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.

Taatgen, N. A., & Anderson, J. R. (2002). Why do children learn to say “Broke”? A model of learning the past tense without feedback. *Cognition, 86*(2), 123–155.
Tsimhoni, O., Smith, D., & Green, P. (2004). Address entry while driving: Speech recognition versus a touch-screen keyboard. *Human Factors, 46*(4), 600-610.
Tucker, A. L., & Spear, S. J. (2006). Operational failures and interruptions in hospital nursing. *Health Services Research, 41*(3, Part I), 643-662.
Violanti, J. M., & Marshall, J. R. (1996). Cellular phones and traffic accidents: An epidemiological approach. *Accident Analysis and Prevention, 28*(2), 265-270.
Wallsten, T. S. (1981). Physician and medical student bias in evaluating diagnostic information. *Medical Decision Making, 1*(2), 145-164.
Welford, A. T. (1952). The Psychological refractory period and the timing of high-speed performance - a review and a theory. *British Journal of Psychology, 43*(1), 2-19.
White, R. W. (1959). Motivation reconsidered: The concept of competence. *Psychological Review, 66*(5), 297-333.
Wickens, C. D. (2008). Multiple resources and mental workload. *Human Factors, 50*(3), 449-455.
Wickens, C. D., Lee, J. D., Liu, Y., & Becker, S. (2004). *Introduction to Human Factors Engineering* (2nd ed.): Prentice Hall.
Wierwille, W. W., & Eggemeier, F. T. (1993). Recommendations for mental workload measurement in a test and evaluation environment. *Human Factors, 35*(2), 263-281.
Woodworth, R. S. (1899). Accuracy of voluntary movement. *The Psychological Review: Monograph Supplements, 3*(3), i-114.
Wright, P. (1974). The harassed decision maker: Time pressures, distractions, and the use of evidence. *Journal of Applied Psychology, 59*(5), 555-561.
Wu, C., & Liu, Y. (2007). Queuing Network modeling of driver workload and performance. *IEEE Transactions on Intelligent Transportation Systems, 8*(3), 528-537.
Wu, C., & Liu, Y. (2008a). Queuing network modeling of the psychological refractory period (PRP). *Psychological Review, 115*(4), 913–954.
Wu, C., & Liu, Y. (2008b). Queuing Network modeling of transcription typing. *ACM Transactions on Computer-Human Interaction, 15*(1), 6: 1-45.
Wu, C., & Liu, Y. (2009). Development and evaluation of an ergonomic software package for predicting multiple-task human performance and mental workload in human–machine interface design and evaluation. *Computers and Industrial Engineering, 56*, 323–333.
Wu, C., Liu, Y., & Quinn-Walsh, C. M. (2008). Queuing network modeling of a real-time psychophysiological index of mental workload—P300 in event-related potential (ERP). *IEEE Transactions on systems, Man, and Cybernetics-Part A: Systems and Humans, 38*(5), 1068-1084.
Xie, B., & Salvendy, G. (2000). Review and reappraisal of modelling and predicting mental workload in single- and multi-task environments. *Work and Stress, 14*(1), 74–99.