CHAPTER 3

Modeling Human Performance for Human–Robot Systems

Caroline E. Harriott & Julie A. Adams

Human performance plays an important role in system performance and is influenced by a wide set of internal, organizational, and environmental factors. Human–robot systems are being developed for a large number of domains, including in-home care, military deployments, emergency response, industrial robots, and therapeutic applications. Human performance modeling provides the opportunity to investigate multiple system prototypes and theories regarding human performance that can be further evaluated in human-in-the-loop experiments, to quickly adapt to rapidly changing robotic technology, and to represent human behavior in extreme conditions. Interacting with robots also influences human performance, and models of the human–robot system provide an avenue to investigate the impact on human performance that the human–robot teaming elicits. This chapter provides an overview of examples representing common methods for assessing human performance in human–robot systems. The topics are summarized in Table 3.2 and include a description of human performance modeling, a summary of commonly used human performance modeling tools, examples of human performance modeling with a focus on its application in human–robot systems, information regarding the validation of models, and guidelines for implementing human performance modeling techniques for robotics.

INTRODUCTION

The performance of humans interacting with a system influences the effectiveness of the system. It is common knowledge that human performance varies and that a number of factors influence human performance. The vast majority of human–robot systems literature over the last decade has focused on evaluating systems and measuring human performance. It is important not only to measure but also to characterize and model human performance and how it impacts overall system goals and objectives. A critical aspect of future human–robot system design is understanding how human performance can influence interaction with a robot, such that the overall system is more robust in highly uncertain and dynamic domains. Modeling of human performance can provide important insight into how to design systems to support human–robot interaction, how to design robots and better interfaces, and how the system can adapt to human performance degradations. Human performance modeling has a long history in general system design but has only recently begun to receive attention in relation to human–robot systems.
Human performance modeling allows system designers to understand the implications of systems design on human performance and to develop more usable, safe, and robust prototypes prior to spending time, effort, and money on system development. In specific, human performance modeling allows designers to identify and analyze factors likely to impact performance with the system, such as environmental variables (e.g., weather, ambient noise), stressors (e.g., fatigue), task demands (e.g., multitasking, workload), and associated behavioral implications. During the traditional system conceptualization, design, and evaluation process, it is often not entirely clear what is required to fully support the user. The integration of human performance modeling with traditional human–robot system development creates a method by which to improve the technology for both expected and unexpected usage. As robots become more common, everyday elements in society (e.g., industrial robots that move beyond caged environments, service robots in the medical and elder care sectors), it is critical to increase the effectiveness, efficiency, and safety of interactions between humans and such increasingly autonomous systems. Robots are complex systems that can function in uncertain and dynamic environments with humans who may not be fully aware of the robots’ capabilities. Thus, it is increasingly important to model the impact of future robot capabilities on human performance and usage.

Current robot technology is advancing rapidly. Robots are being designed for and analyzed in an increasing number of specialized tasks that involve human–robot systems, such as in-home care (e.g., Michaud et al., 2010), military (e.g., Deutsch, 2006), emergency response (e.g., Murphy et al., 2008; Harriott, Zhang & Adams, 2011a), industrial (e.g., Barcellini et al., 2012), or therapeutic (e.g., Mazzei et al., 2011) domains. The development of each of these types of robots is a complex, time-consuming, and expensive task that can involve extensive market research, user evaluations, and prototypes before such systems can be developed and reliably used by human operators or partners. The development of human performance models offers numerous benefits that apply to such robotic domains.

This chapter provides an overview of assessing human performance in human–robot systems. The primary focus is human performance modeling. The chapter is intended not to provide a detailed review of all available performance assessment tools but simply to identify and describe the range of tools that have been implemented. First, an overview explains what human performance modeling is and why it is important, followed by an overview of the common tools for modeling human performance. The human performance modeling literature is reviewed, including a high-level review of applications to non-robotics systems that have relevance to robotics, which grounds the robotics review. A primary focus is a review of robotics literature related to human performance modeling. Further, the chapter presents methods for verifying human performance models and presents examples from robotics and non-robotics systems. Finally, guidelines for modeling and verifying human performance for robotics are provided.

Overview of Assessing Human Performance

A central question motivating this chapter is how does human performance modeling differ from human performance measurement? Thus far, the majority of human–robot systems research has focused on measuring human performance with well-structured user evaluations (e.g., Akgun, Cakmak, Yoo, & Thomaz, 2012; Harriott et al., 2011a),
semistructured field evaluations (e.g., Lee et al., 2012; Murphy et al., 2008), and case studies (e.g., Feil-Seifer & Mataric, 2011; Robins et al., 2012). Irrespective of the type of evaluation, the focus is frequently on how well a user is able to employ a robot to complete a task, how the interface impacts the user’s cognitive workload (e.g., Boles, Burks, Phillips, & Perdelwitz, 2007; Gawron, 2008; Hart & Staveland, 1988), situation awareness (e.g., Endsley & Garland, 2000), and so on. The human–robot interaction community has worked to formalize methods for assessing human performance (e.g., Scholtz, 2003; Steinfield et al., 2006). The last decade has seen this aspect of human–robot systems research mature and provide insights into the design of systems (e.g., Chen, Barnes, & Harper-Sciarini, 2011; Coradeschi et al., 2011; Crandall, Cummings, Penna, & De Jong, 2011; Gielen & Thomaz, 2012; Harden & Goodrich, 2009; Harriott et al., 2011a; Humphrey & Adams, 2010; Micire et al., 2011; Mutlu, Shiwa, Kanda, Ishiguro, & Hagita, 2009).

The breadth of systems being evaluated has also grown tremendously, and evaluation techniques and metrics have been adapted from other fields, such as the human factors and human–computer interaction communities. As robot capabilities continue to improve and provide additional multimodal interactions, metrics from other areas have become increasingly important, such as nonverbal measures (e.g., Manusov, 2005), physiological metrics (e.g., Conn, Liu, Sarkar, Stone, & Warren, 2008; Parasuraman, Barnes, & Consenzo, 2007), applied attention metrics (e.g., Wickens & McCarley 2008), and natural language metrics (e.g., Berthold & Jameson, 1999; Lively, Pisoni, Van Summers, & Bernacki, 1993).

While the measurement of human performance when interacting and using robots is important, human performance assessment cannot be mistaken for performance modeling. Human performance modeling is intended to provide a prediction of interaction outcomes for specific tasks. The outcomes can include cognitive workload, stress, and so on that can also be captured via human performance metrics during evaluations. Data collected during user evaluations may serve as bases for improving human performance models. As well, user evaluations are necessary tools for validating results of human performance models (e.g., Allender et al., 1995; Harriott et al., 2011a; Harriott, Zhang, & Adams, 2011b; Pew & Mavor, 1998).

**HUMAN PERFORMANCE MODELING**

Human performance modeling is designed to capture human behaviors via analytical or computational means (Pew & Mavor, 1998). Typically, models are designed to replicate human behavior(s) for a range of tasks. This section provides a definition and motivation for human performance modeling, outlines potential impacts of modeling, and details a set of commonly used human performance modeling tools.

**What Is Human Performance Modeling?**

Human performance models can be employed to predict performance attainable between a human and a system, or models can be used to describe performance resulting with adjustments to model parameters to match actual user data (Baron, Kruser, & Messick...
Huey, 1990; Booher & Minninger, 2003; Pew & Mavor, 2007). The purpose of modeling is to generate results that can be used by decision makers to answer specific questions, but such models must represent both the human and the system. Performance models can be used to understand the implications of decisions on human performance or to develop theories about performance. Models can also be used to inform system design and evaluate systems.

An important aspect of human performance modeling is the understanding that performance occurs on a continuum. Human performance modeling requires identification of a particular domain in order to analyze performance across environmental factors, task requirements and demands, stressors, and so on. This cross-factor analysis allows for development of an understanding of the ergonomic impact on human performance, that is, how the system, task, and environment interact to either positively or negatively impact human performance. System capabilities that do not match human expectations can negatively impact performance. Properly designed system capabilities can serve to facilitate the human interaction and task performance needs, which can lead to improved speed, accuracy, and so on.

Another aspect of human performance modeling is the need to ensure the selected modeling tool can accommodate the range of performance factors (see Silverman, Johns, Shin, & Weaver, 2002, for a summary of performance factors) one seeks to analyze. Available modeling tools are not created equally. Modeling capabilities and represented performance factors differ across tools. While modeling tools are often developed with the intention to be general and apply across a number of situations, they often have a particular focus. For example, some tools have a cognitive focus, some have an inference focus, some have a perceptual focus, and so on. More specifically, the basic human behaviors and capabilities that tools model the best will vary, including errors, memory, learning, visual attention, workload, and so on (Foyle & Hooey, 2008). It is up to the modeler to determine the necessary level of model fidelity and the set of performance factors for an analysis, which may also serve as a basis for selection among available modeling tools and techniques.

Yet another important aspect of the performance modeling process is verification of the model (Allender et al., 1995; Pew & Mavor, 1998). It is necessary to verify that a model provides predictions within an expected range. Often, models are developed for a baseline condition for which data exist, thus allowing model outcomes to be compared with the existing data set. Additionally, a model can be verified by conducting user evaluations, either simultaneously or after modeling, which captures a representative data set for verification. Either approach can result in the modeler “tuning” the model as needed. Another approach is simple trial and error.

Why Model Human Performance, and What Are the Uses?

Human performance modeling facilitates analysis of existing and future systems. Additionally, it allows for the prediction of expected impacts of human–system usage based on the contextual aspects of the task and environment. Human performance modeling also provides the opportunity to conduct investigations of a broader set of conditions than may be possible with human-in-the-loop evaluations.
System prototypes can be expensive to develop or simply may not exist. Human performance models provide a means of exploring system influences on human performance prior to the existence of prototypes. Modeling efforts also permit the exploration of new and alternative human–system interactions for less investment, since real systems or dynamic prototypes are not required (Baron et al., 1990; Booher & Minninger, 2003; Pew & Mavor, 2007). More specifically, modeling a system provides the capacity to understand the system in ways that observation may not support (Baines & Benedetti, 2007; Foyle & Hooey, 2008). For example, modeling often allows for manipulation of system, task, and environment in ways that actual experimentation may not support.

Another important application of human performance modeling is the development of theories related to human performance that can be further investigated via human-in-the-loop evaluations (Baron et al., 1990; Harriott et al., 2011a, 2011b; Rothrock & Narayanan, 2011). Models can be developed to represent hypotheses on human cognition or behavior under typical conditions of system use, such as time sharing or multitasking (Baron et al., 1990; Booher & Minninger, 2003; Pew & Mavor, 2007).

We live in a time when technology and associated capabilities change rapidly; thus it is not always possible to know exactly how humans will use and interact with future systems. While it is true that earlier versions of systems exist, the advent of technological advances can limit the application of knowledge regarding human performance with existing systems to future systems (Booher, 2003). Modeling allows for a representation of the generic technology and human interaction with the technology as well as a capacity to explore potential future system capabilities and usage. Modeling activities during early system planning and concept design often lead to the development of measures of effectiveness and performance that can be refined during the design process and applied throughout system development and testing (Booher, 2003). Early modeling activities also provide insight into potential performance with a new system that provides a basis for human-in-the-loop evaluations during later design iterations. Performance modeling supports an understanding of the operation of future systems, which can impact system design, facilitate cost-benefit analysis, and aid conceptual interpretations (Dahn & Laughery, 1997). The modeling of future systems can provide improved understanding of current design alternatives that can lead to logical reductions in the set of design alternatives and tractable human-in-the-loop evaluations of alternatives.

Modeling can also be used to analyze human performance in extreme conditions, for either future or existing systems. For example, modeling can be used to understand trade-offs between system modifications and human performance. When multiple system designs exist for handling extreme conditions, the analysis of alternatives can provide insight into costs necessary to remedy the situation. While this approach may not be preferred for analysis, it can be effective in times when there are insufficient resources to develop actual system prototypes and conduct human-in-the–loop system evaluations. This approach also provides a safer alternative to exposing humans to extreme conditions (Dahn & Laughery, 1997).

Such modeling activities yield predictions and should be coupled with human-in-the-loop evaluations for verification. Predictive model results should be identifiable in data collected from user evaluations. However, modeling can be used to identify new or emergent behaviors that were previously not recognized or performance patterns that are...
difficult to identify in a data set. Fundamentally, the overall reason to model human performance is to better understand human behavior and performance with a particular system. One purpose of conducting performance modeling is to understand how variations in a task or environment impact human information processing, including decision making and situation management, in order to minimize human error and improve system safety (Kontogiannis, 2005).

Human performance modeling is, however, not without its limitations. As with any modeling method, one must ensure that the model is not too simplistic and accurately represents the system and human capabilities. The development of more complex models, such as an increased number of parameters necessary to represent human performance, occurs with associated costs. The range of performance factors can be quite large, and factors can have a broad range of values, thus complicating accurate specification of model parameters. As the complexity of a model increases, so does the process for validating the model. In addition to these limitations, humans exhibit individual differences, and capturing and understanding such individual differences via human performance models can be difficult. While it may not be feasible to represent all individual differences, models can help to understand a system’s sensitivity to individual differences by manipulating model parameters.

Tools for Modeling Human Performance

A variety of modeling tools have been developed to represent human performance under a range of conditions. Most tools incorporate a method of representing the environment, the tasks, and human responses. Cognitive architectures represent theories on components of cognition, including perception and response execution, and are typically developed to be context independent. Other modeling tools aim to represent systems at a higher level and do not delve into lower-level cognitive constructs. When choosing a modeling tool, it is important to weigh the level of detail necessary for the analysis.

The Atomic Components of Thought–Rational (ACT-R) is a modeling system representing human cognition (Anderson, 2007; Anderson & Lebiere, 1998). Referred to as a cognitive architecture, ACT-R provides a method of incorporating assumptions about a task or domain in a model along with general assumptions on the human mind and perceptual systems. As described in Anderson et al. (2004), ACT-R is composed of

- perceptual-motor modules that represent the interface with the world,
- memory modules containing knowledge of facts and procedures,
- buffers for each of the modules representing the current state, and
- a pattern matcher to match the current state to a known method of information processing.

Models are written in production rules that represent knowledge of how to accomplish tasks. ACT-R models focus on the details of perception and cognition and can produce accurate predictions of task timing and accuracy that can be directly compared to experimental data.
The Soar cognitive architecture is similar to ACT-R in that knowledge representation is combined with a rule-based production system for actions on knowledge stores (Laird, 2012; Laird, Newell, & Rosenbloom, 1987). Soar is used to model cognitive behaviors and explore artificial intelligence (Wray & Jones, 2005). Soar models are based on the concept that behaviors are dictated by an architecture and knowledge. The Soar architecture does not generate behaviors per se but, rather, provides a method of emulating behavior with given knowledge content (Lehman, Laird, & Rosenbloom, 1996).

Cassimatis (2002, 2006) created the Polyscheme cognitive architecture to emulate human-level intelligence. The three basic principles of Polyscheme are that “specialists” represent knowledge and make inferences on specific aspects of situations, the specialists must communicate with each other using complete and relevant information, and the focus of attention is based upon inference schemes. Using these three principles, Polyscheme models represent human intelligence learning and decision making.

Cognition as a Network of Tasks (COGNET) is also a cognitive architecture and includes components representing perception, memory, cognition, and motor action (Zachary et al., 2005). COGNET models represent knowledge in five areas of expertise: declarative, which contains state information; procedural, which contains goals and methods of achieving the goals; action, which includes motor system information; perceptual, which processes external environment information; and metacognitive, which selects and executes procedural knowledge. COGNET includes the knowledge that humans can perform tasks concurrently and attention focuses on tasks with the highest priority.

The Distributed Operator Model Architecture (D-OMAR) is a cognitive modeling tool that provides the capacity to represent rule-based behaviors, proactive goal-oriented behaviors, memory, multitasking, differing skill levels of human agents, and parallel execution of behaviors (Deutsch, Cramer, Keith & Freeman, 1999). D-OMAR represents the human agent in a system and can interface with simulations of a physical system, such as aircraft, or a specific task environment. The aspect of D-OMAR that is distributed refers to a network-based configuration designed for execution in multiple middleware environments.

Cognitive architectures focus on modeling human behavior with major components of perception, cognition, and action. These modeling tools incorporate details on memory and decision making to create realistic representations of the mind. Soar and Polyscheme are designed to replicate human behavior for improving artificial intelligence. While ACT-R has artificial intelligence applications, ACT-R and COGNET focus on representing human behavior for analyzing system design and predicting human performance.

Rather than replicating the details of human behavior, human performance modeling tools often focus instead on larger system interactions and analysis. The fine-grained detail and analysis of behavior provided by cognitive architectures is not always the focus of human performance modeling tools. Instead, these tools are designed to compare, for example, personnel configurations, workload levels, or task assignments without explicitly specifying many of the step-by-step cognitive processes or procedure selection rules involved in performing the task.
Discrete event simulation tools are often used in human performance modeling. They are used to predict system outcomes based on time-based tasks or events (Robinson, 2004). Discrete event simulation tools are composed of the following elements:

- a clock to keep track of time within the model,
- an events list to queue executable events,
- random-number generators for variable values,
- statistics to track important changes in variables, and
- an ending condition to ensure that the model does not continue indefinitely.

Micro Saint Sharp is an example of a discrete simulation tool and has been integrated in some of the following human-performance modeling tools (Bloechle & Schunk, 2003).

The Improved Performance Research Integration Tool (IMPRINT) is designed to provide timing and personnel data for large-scale systems. It does not focus on the individual steps of cognition and perception, like ACT-R. IMPRINT represents the order and length of discrete tasks, while ACT-R models information-processing steps. IMPRINT was developed by the United States Army Research Laboratory Human Research and Engineering Directorate to evaluate human–system configurations based on task time, number of crew members, training, equipment, internal and external stressors, and operator workload (Archer, Gosakan, Shorter, & Lockett, 2005). The IMPRINT modeling system incorporates Micro Saint Sharp, the former Manpower and Personnel Integration models (MANPRINT), and the Hardware versus Manpower tool (HARDMAN) (Allender et al., 1995). Rather than detailing all of the human cognition and perception steps, a user can base task time input on micromodels of human behavior built into the IMPRINT system (U.S. Army Research Laboratory [ARL], 2009). Operator workload can also be modeled using the tool’s built-in guidelines along seven channels of demand: auditory, visual, cognitive, fine motor, gross motor, speech, and tactile. Task timing and accuracy can be modified through a set of internal and external stressors (e.g., whole-body vibration, weather conditions, sleep loss). IMPRINT Pro offers the capacity to input customized stressors based on established data from the literature or human-in-the-loop experimental results; custom stressors can change over time. All stressors influence task timing and accuracy based on their weights along the channels of demand.

Complicated human–machine systems can also be modeled and analyzed using the Man-Machine Integration Design and Analysis System (MIDAS), which was developed by the NASA Ames Research Center for human factors analysis (Tyler, Neukom, Logan, & Shively, 1998). MIDAS simulates a human performing goal-related behaviors and has been used in aeronautics and space exploration mission research (Gore & Jarvis, 2005). Human behavior predictions with MIDAS are based on knowledge of the perceptual (visual, auditory, central processing), memory, and attention systems. Capacities include predicting human error, providing insight into the best practice for system design, three-dimensional visual simulation, predictions of cognitive load along seven channels, situation awareness estimates, and demonstration of the effect of factors influencing human performance (such as microgravity). MIDAS incorporates Jack, a model of human physical characteristics (Badler, Phillips, & Webber, 1993), and Micro Saint Sharp, a discrete event simulation tool.
The Integrated Performance Modeling Environment (IPME) incorporates the Micro Saint Sharp discrete event simulator with a human operator simulator to analyze human–system performance (Dahn & Belyavin, 1997). Fowles-Winkler (2003) describes IPME’s goals as twofold: (a) providing a flexible modeling environment with the capacity to choose which system components are necessary for the analysis and (b) interacting with other simulation software. Each system modeled with IPME can include an environment representation, a crew model, performance shaping factors, a task network model, an external model, and an experimental suite allowing for manipulation of independent variables settings before running the model. Like IMPRINT and MIDAS, operator workload can be modeled in IPME.

Human reliability analysis uses knowledge of human state (e.g., sleep loss) to compute the likelihood of risk in a workplace. The Standardized Plant Analysis Risk–Human Reliability Analysis method (SPAR-H) is used to predict human failure events using known human error probabilities and performance-shaping factors with values set by the user according to guidelines (Gertman, Blackman, Marble, Byers, & Smith, 2005). With SPAR-H, one considers human behavior in two categories: action (e.g., calibrating a sensor) and diagnosis (e.g., prioritizing activities).

The Goals, Operators, Methods, and Selection Rules (GOMS) model is a well-known model created for interaction with computers (Card, Moran, & Newell, 1983). GOMS is based on small atomic steps of information processing, cognition, and action but is not a cognitive architecture. Goals include all tasks and subtasks that need to be completed. Operators are atomic actions taken to achieve goals, using methods or procedures for how to perform operators. Selection rules help determine what methods are appropriate given the current state of the system. Actions, including keystrokes, mouse clicks, or drag-and-drop, have associated timing are computed by considering the user’s goals and the ways used to achieve them. Error recovery can also be represented in GOMS models. Additionally, the GOMS Language Evaluation and Analysis (GLEAN) tool is a computer-based implementation of GOMS (Kieras, Wood, Abotel, & Hornof, 1995). GOMS Language (GOMSL) models are constrained by a simplified version of the Executive Process Interactive Control (EPIC) cognitive architecture developed by Kieras and Meyer (1997). EPIC provides the ability to perform tedious time and accuracy calculations of a GOMS model with cognitive plausibility.

Performance moderator functions, such as workload, fatigue, level of training, or environmental conditions, can be important in capturing the “full picture” of a modeled system. The Performance Moderator Function Server (PMFServ) system can model human behavior that takes into account physiology and stress, emotion, cognitive processes, and group behavior (Silverman, 2004). The PMFServ incorporates the effects that these states have on behavior using experimentally validated data (Silverman et al., 2002) and creates an emotional subjective utility that factors into decision making (Pelechano, O’Brien, Silverman, & Badler, 2005). Examples of the data used to modify human behavior include the quantifiable difference in time when making decisions under stress and the effect of sleep loss on reaction time.

Human performance modeling tools, such as IMPRINT, MIDAS, IPME, SPAR-H, and PMFServ, support task analysis while taking into account influences from internal and external stressors, system configurations, experience, and/or teammates. IMPRINT,
MIDAS, and IPME incorporate Micro Saint Sharp for discrete event simulation and focus on human–system interaction. SPAR-H is very specific to workplace error but takes into account internal state and environmental conditions when calculating risk. GOMS provides a good method of developing predictions for timing in human–computer interaction but does not account for the human’s ability to perform concurrent actions. PMFServ provides the additional representation of emotion and social phenomena that are not available in the other modeling tools.

Overall, modeling human performance can be achieved in many different ways. Depending on the scale of detail needed and the system analysis to be performed, some modeling tools are more appropriate for given tasks than others.

**APPLICATIONS OF HUMAN PERFORMANCE MODELING**

This section begins with a brief review of human performance modeling application examples for non-robotic domains. The primary focus of this section is a review of human performance modeling for robotics applications, with relevant examples discussed in the second subsection.

**Human Performance Modeling for Non-Robotic Domains**

Human performance modeling has been applied to a large number of domains that include aviation (e.g., Foyle & Hooey, 2008), simulating human agents (e.g., Best & Lebiere, 2006; Silverman, 2004; Silverman et al., 2002), manufacturing systems (e.g., Baines et al., 2005), job design (Kontogiannis, 2005), driving (Salvucci, 2006), and military systems (e.g., Booher, 2003; Pew & Mavor, 1998), to name a few. The purposes of the resulting models vary from understanding how to develop future systems, to understanding sources of human error and performance problems, to understanding human behavior in different situations.

Robots vary in level of autonomous behavior and humans must be able to understand such behaviors as a basis for interaction with the robots. Very few robot systems exist that permit interaction across the spectrum of levels of autonomy (Endsley & Kaber, 1999; Parasuraman, Wickens, & Sheridan, 2000) or that can adapt interactions and behaviors based on human interactions, understanding, or performance. Future robotic systems may require such capacities as they are deployed in real-world domains. Duric et al. (2002) focused on developing and validating a model of embodied cognition using ACT-R/Perceptual-Motor (Byrne & Anderson, 1998) for adaptive human–computer interaction. The model was used to adapt computer interaction with the human user based on human performance. The model demonstrated the usefulness of performance modeling in the human–computer interaction domain and the potential application to human–robotic system interaction.

Salvucci (2006) developed a cognitive behavioral model of driving. The ACT-R model was designed to represent a broad spectrum of tasks related to driving, simulate realistic controls and vehicle dynamics, and perform tasks using “cognitive” processes integrated
with perceptual and motor processes. The model supported analysis of serial versus parallel processes, limited attention, and individual differences in driving abilities for distracted driving. While humans will not likely “drive” robots as they do cars, Salvucci’s model demonstrates the use of such tools for assessing multitasking and future interactions when working with robotic partners or teammates.

Silverman and his group extended PMFServ for investigation of emergent human behaviors based on modifying human performance parameters (Silverman, 2004; van Lent et al., 2004). They developed realistic and socially intelligent software agents (Silverman, 2004). This effort differs significantly from traditional human performance modeling in that the researchers sought to understand the impact of personal values, emotions, physiology, and stress on human decision making, as individuals and in groups (Pelechano, O’Brien, Silverman, & Badler, 2005; Silverman, 2004). This research demonstrates the potential usefulness of human performance modeling for understanding social implications of introducing robotic systems (with increasing functionality) into society, where emotions and personal values can impact system usage. Additionally, incorporating realistic human agents with nonrational decision-making capacities may help provide more accurate model predictions.

Best and Lebiere (2006) developed a cognitive model for software agents acting in a game-based environment. This effort focused on understanding how humans navigate and represent space in a simulated military operation in an urban terrain training environment. Another aspect of the research focused on developing robotic behaviors based on simulated humans under the premise that humans expect robots to behave in a similar manner. This research demonstrated a link between human performance modeling and robotic systems design, which is the focus of the next subsection.

**Human Performance Modeling for Robotics**

Relationships between humans and robots may be unique and should be accounted for when developing human performance models. The purpose of this section is to present a review of modeling paradigms that have been used and modified for different types of human–robot systems interaction. Table 3.1 provides a summary of the reviewed research. The table details the focus of models (e.g., physical behavior, cognition), the purpose of using human performance modeling in each project (e.g., predicting performance, system design), validation metrics, and the modeling tool chosen. Modeling tools appearing in italics are those presented in the review of human performance modeling tools.

**General models.** Goodrich and Boer (2003) analyzed the interaction between a human and an automated cruise control and braking system. The authors argued that models of human behavior are important to human-centered design of automated systems. In general, there are four factors considered to affect human use of automation: (a) limitations of automation and human awareness, (b) responsibility transfer between the human and automation, (c) acceptability and predictability of automated system “behavior” dynamics, and (d) the effect of use of automation on overall system efficiency. The effect an automated system had on human driving was analyzed with respect
<table>
<thead>
<tr>
<th>Authors</th>
<th>Model Focus</th>
<th>Purpose of Modeling</th>
<th>Validation Metrics</th>
<th>Modeling Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goodrich &amp; Boer, 2003</td>
<td>Cognitive</td>
<td>Identify human behaviors</td>
<td>Car velocities, pedal state, cruise control state</td>
<td>Mental models</td>
</tr>
<tr>
<td>Olsen &amp; Goodrich, 2003</td>
<td>Cognitive</td>
<td>Predict performance</td>
<td>Task effectiveness, neglect tolerance, robot attention demand, free time, fan-out, interaction effort</td>
<td>Mathematical models</td>
</tr>
<tr>
<td>Crandall, Nielsen, &amp; Goodrich, 2003</td>
<td>Timing</td>
<td>Predict performance</td>
<td>Secondary task performance, neglect tolerance, interface efficiency, time to completion</td>
<td>Mathematical models</td>
</tr>
<tr>
<td>Best &amp; Lebiere, 2006</td>
<td>Cognitive</td>
<td>Versatile agent development</td>
<td>—</td>
<td>ACT-R</td>
</tr>
<tr>
<td>Haddadin, Albu-Schäffer, &amp; Hirzinger, 2007a, 2007b</td>
<td>Physical responses</td>
<td>Predict performance</td>
<td>Robot movement speeds, crash test dummy injuries</td>
<td>Safety tree chart</td>
</tr>
<tr>
<td>Howard, 2007</td>
<td>Workload</td>
<td>Predict performance</td>
<td>Execution time, composite task scores</td>
<td>Fuzzy logic model, HumAnS-3D</td>
</tr>
<tr>
<td>Trafton et al., 2005</td>
<td>Cognitive</td>
<td>System design</td>
<td>Task success</td>
<td>Polyscheme</td>
</tr>
<tr>
<td>Gluck et al., 2005</td>
<td>Cognitive</td>
<td>System design, personnel training</td>
<td>Degree of correctness, spatial interference effect in visual working memory</td>
<td>ACT-R, VERBOSE</td>
</tr>
<tr>
<td>Hunn &amp; Heuckeroth, 2006</td>
<td>Workload, timing</td>
<td>Predict performance</td>
<td>Surveys of subject matter experts</td>
<td>IMPRINT</td>
</tr>
<tr>
<td>Petkosek, Warfield, &amp; Carretta, 2005</td>
<td>Workload</td>
<td>Predict performance</td>
<td>—</td>
<td>CART</td>
</tr>
<tr>
<td>Deutsch, 2006</td>
<td>Workload, timing</td>
<td>Predict performance</td>
<td>—</td>
<td>D-OMAR</td>
</tr>
</tbody>
</table>

(continued)
<table>
<thead>
<tr>
<th>Authors</th>
<th>Model Focus</th>
<th>Purpose of Modeling</th>
<th>Validation Metrics</th>
<th>Modeling Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ritter, Van Rooy, St. Amant, &amp; Simpson, 2006</td>
<td>Cognitive, physical</td>
<td>System design</td>
<td>Lane deviation, total time before crash</td>
<td>ACT-R and SegMan (DUMAS)</td>
</tr>
<tr>
<td>Kaber, Wang, &amp; Kim, 2006</td>
<td>Cognitive</td>
<td>Predict performance</td>
<td>Execution time, path error</td>
<td>GOMS</td>
</tr>
<tr>
<td>Liu, Rani, &amp; Sarkar, 2006</td>
<td>Affect</td>
<td>Predict performance</td>
<td>Anxiety levels, satisfaction with game, game scores, perceived challenge of game</td>
<td>Affective model</td>
</tr>
<tr>
<td>Trafton, Bugajska, Fransen, &amp; Ratwani, 2008</td>
<td>Verbal</td>
<td>System design</td>
<td>Subjective ratings of conversation naturalness</td>
<td>ACT-R, ACT-R/E</td>
</tr>
<tr>
<td>Crandall, Cummings, &amp; Nehme, 2009; Crandall, Cummings, Penna, &amp; De Jong, 2011</td>
<td>Attention allocation</td>
<td>Predict performance</td>
<td>Number of objects found, number of robots lost in explosion</td>
<td>RESCU</td>
</tr>
<tr>
<td>Reitter &amp; Lebiere, 2010</td>
<td>Cognitive</td>
<td>Predict performance</td>
<td>Normalized area between chosen travel paths</td>
<td>ACT-R</td>
</tr>
</tbody>
</table>

*Note.* ACT-R = Atomic Components of Thought–Rational; ACT-R/E = ACT-R/Embodied; IMPRINT = Improved Performance Research Integration Tool; CART = Combat Automation Requirements Testbed; D-OMAR = Distributed Operator Model Architecture; DUMAS = Driving User Model in ACT-R and SegMan; GOMS = Goals, Operators, Methods, and Selection Rules; RESCU = Research Environment for Supervisory Control of Unmanned Vehicles.
to these four factors. The authors also developed mental models of the processes in order to gain insight on human behaviors during driving and interaction with the cruise control system. The driving task was divided into three categories of subtasks: speed regulation, time-headway regulation, and active braking. Speed regulation involved the choice to accelerate, time-headway regulation determined following distance, and active braking occurred when the driver or system perceived the need to decelerate and press the brake pedal. Determining the benefit of added automation involved emulating drivers switching between each of the three subtasks, with switches preceded by perceptual cues. A divide between the braking and regulation tasks was found when comparing these metrics between for models and participants. The models provided insight on why cruise control affecting speed regulation was acceptable but identified active braking automation as a poor choice. The four principles of automation interaction, and the idea that modeling human responses may inform design choices, can be applied to other work.

Metrics such as the amount of cognitive effort expended and the amount of free time available for a secondary task are relevant during human–robot system assessment and design. Such information can be used to determine how much time robot operators have to complete tasks in addition to robot control, which may increase overall productivity. Olsen and Goodrich (2003) devised six interrelated metrics for analyzing human–robot systems: (a) task effectiveness, (b) neglect tolerance, (c) robot attention demand, (d) free time, (e) fan-out, and (f) interaction effort. Task effectiveness measures how well a human–robot team achieves the task goal. Neglect tolerance reflects the level of a robot’s effectiveness over time, in the absence of human teammate attention, and is a mechanism for assessing robot autonomy. Robot attention demand is the proportion of deployment time spent attending to the robot. Free time is the proportion of time spent not attending to the robot and can be measured using performance on a secondary task. Fan-out is a measure of the approximate number of robots a person can effectively operate and is the reciprocal of the robot attention demand. Finally, interaction effort is related to time spent interacting or engaged with a robot. Interaction effort is not directly measurable, as recording the exact time when a human is cognitively focused on a task is difficult. Instead, interaction effort can be determined using relationships among other metrics, such as neglect tolerance, secondary task performance, and fan-out.

Overall, the system goals may be to increase task effectiveness, neglect tolerance, and free time, while reducing interaction effort. Team organization and task assignments can be modeled by considering these metrics and may prevent the deployment of an ineffective team. These metrics can predict the performance of a potential team and can be used as metrics when modeling human–robot systems. They have been applied in robotic system models and evaluations (e.g., Crandall, Nielsen, & Goodrich, 2003; Elara, Calderon, Zhou, & Wijesoma, 2010; Olsen & Wood, 2004).

Crandall et al. (2003) expanded upon Olsen and Goodrich’s (2003) idea by creating an algorithm to predict the performance of a single-human, multiple-robot team and experimentally validated the results. The neglect tolerance and interface efficiency (the effectiveness of the interaction between a robot and a human) of human–robot teams were measured in multiple user studies. This information fed into an algorithm that output time to completion for various system configurations of one human and three robots.
These completion times were experimentally validated and shown to be good predictions of actual completion times, with the exception of configurations that were very difficult; teams performed faster than the predictions in difficult configurations. Data from previous user studies can be used to create algorithms that successfully predict team performance in subsequent tasks.

While the previous topics focused on more general principles of modeling in human system interaction, specific systems have been deployed that were developed based on modeling techniques for a single human interacting with a single robot either directly or through remote (tele-) operation. Additionally, humans interacting with multiple-robot systems have employed modeling techniques to glean information about human–robot systems. The following subsections focus on these systems.

**Single-robot interaction.** Jung, Choi, Park, Shin, and Myaeng (2007) sought to improve single-human, single-robot interaction. The authors believed that robots lack “common sense” and wanted robots to use prior experiences to evolve the way they respond to situations through cognitive human–robot interaction. Singh’s (2005) EM-ONE cognitive architecture includes human reactive, reflective, and retrospective layers of thinking. This architecture was modified and integrated into a semiautonomous script-generating software system that was used in robot path planning. The software maintained task-related information, such as pre- and postconditions for possible actions within the current scenario. Using models based on human cognition and task-related information, the authors were able to modify robot behavior and system design to reflect human behavior. The system was used to script robot behaviors and interactions with humans based on a model of human information processing and learning.

Direct and immediate contact between humans and robots or automation has been examined, evaluated, and incorporated into human performance models. Domains where modeling has been tested include urban military scenarios, industrial robotics, first response, space exploration, and video games. The following cases provide examples of modeling teamwork, injury, workload, completion time, affect, and perspective taking.

Urban military scenarios can involve soldiers and robots acting as teammates to achieve goals according to structured protocols. Best and Lebiere (2006) used the ACT-R cognitive architecture to develop human and robotic agents interacting as a team in an urban military operation with clear rules and structured goals. The objective was to create agents that may be used in simulations of team tasks in real and virtual environments. Robotic agents were developed based on the Pioneer 3DX general-use robot incorporating visual and navigation systems. Teamwork between the agents involved sharing plans, communication of the steps involved to complete a plan, and a method of sharing a representation of the surrounding space. The authors suggested that it is possible to extend cognitive models that are not tied to lower-level environment details to other domains. The resulting agents were used to help with training for military personnel for urban terrain operation.

Some models of human–robot systems have been developed to quantify outcomes of physical interaction. Haddadin, Albu-Schäffer, and Hirzinger (2007a, 2007b) studied safety considerations in physical human–robot interaction, specifically with industrial robots. The authors first evaluated incidents of robots colliding with the human head, neck, and chest. Such tests cannot be performed using actual humans; therefore, crash-test dummies...
were used. Equations outlining the predicted severity of an injury to the head, neck, and chest were adapted from the automobile industry but were shown to have limited applicability to the slower speeds of movement in physical human–robot interaction. New models of physical human–robot interaction were necessary, and the authors later developed tests and simulations to investigate the impact of robot mass and velocity on human injuries. A safety tree model was developed to classify possible injuries and predict worst-case scenarios. The model is a chart of worst-case scenarios and possible injuries that can occur during human–robot interaction. Users of this model can identify conditions of a scenario, find matching conditions in the model, and determine levels of risk during physical human–robot interaction. It was concluded that blunt impacts to the head or chest were not life threatening no matter how large the robot is as long as the human is not pinned against a hard surface. Such findings can impact the design of human–robot systems by allowing for more accurate predictions of injury. Using this model, system designers can evaluate the trade-off between increasing robot speed and size and possible injury to those working with robots.

An individual’s performance can affect an entire team’s performance. Predicting human performance and workload levels can aid task assignment and team organization decisions. While some functions predicting human performance have been validated for human–human scenarios, a limited number have been evaluated for human–robot teams. Harriott et al. (2011a, 2011b) used the IMPRINT Pro modeling software to predict whether performance predictions for human workload also apply to human–robot teams. The authors focused on a first-response scenario, as the addition of robots to this domain decreases risk to human responders and increases manpower in the field. Models of a human–human team and a human–robot team, performing the same set of triage steps, were created. The chosen scenario related to search and rescue with participants representing uninjured victims in a contaminated performing medical triage with instructions from their human or robot partner. Participants performed 11 triage assessments during a trial, each with one of three triage levels designed to represent a within-subjects manipulation of workload. The triage steps that participants performed were the same between participants and between the human–human and human–robot team conditions. Interactions between the human or robot partner and the participants were based on a verbal script used in all trials.

The different models were based on the verbal script created for the evaluation; one model represented the human–human team, and the other was the human–robot team performing the same set of tasks. Since the two models were created before performing the evaluation, the models were not biased by the outcomes. Each step of the planned evaluation was divided into atomic tasks. IMPRINT Pro requires inputs of task times and workload values. Task times were based on the provided micromodels of human behavior that dictate, for example, average human walking and speech speeds. The human–robot model was different from the human–human model as a result of slower robot speech and movement tasks, as the robot spoke and moved more slowly than a human. Workload values for each atomic task were assigned according to IMPRINT Pro’s guidelines and were approved by a subject matter expert. The models output workload values for each time point. Weighted mean workload values were computed for each triage assessment in addition to the predicted time to perform each assessment.
Results revealed that when taking into account the differences in robot movement and speech speed, the same guidelines for determining workload apply for both human–human and human–robot teams. The modeling technique, not originally developed for human systems, was shown to be extensible to human–robot systems. The ability to predict human performance, task times, and workload levels can impact task assignment in human–robot teams. Additionally, robot system design can be optimized to reduce task time (e.g., increase robot movement speed) or decrease human workload levels (e.g., increase robot capacity).

A second modeling and evaluation activity investigated the effect of a collaborative relationship on workload and reaction time predictions (Adams, Harriott, Zhuang, & DeLoach, 2012). Results indicated more complex and dynamic relationships between a human and robot partner may make IMPRINT Pro’s predictions less accurate. Participants were paired with either a human or robot partner to search a hallway of an academic building for hazardous materials related to a bomb threat. The relationship between the partners was dynamic and collaborative. The teams proceeded through the hallway and two laboratories. Experimenters placed a set of 19 suspicious or nonsuspicious items in the hallway and rooms (e.g., suspicious blueprints or fake pipe bombs, unsuspicious textbooks) and incorporated items already present in the environment (e.g., laboratory warning signs). The same items were placed in the same locations for both conditions.

Each condition of the hazardous materials search was also modeled in IMPRINT Pro. Again, the human–human and human–robot models only differed in the timing of robot speech and movement tasks. Workload values were assigned for each task based on IMPRINT Pro guidelines and subject matter expert insight. Item reaction time was modeled by combining individual elements of reaction time. The predicted item reaction time encompassed eye fixation and movement time, head movement time, decision time, and search time.

Space exploration also involves human–robot teams with missions that involve detailed planning before execution. Howard (2007) worked to optimize role allocation for such human–robot team systems by incorporating the process of alternating attention between tasks and assessing the workload levels of human teammates. Team performance was optimized by intelligently choosing whether the human or robot is assigned each role. Increasing the time interval between the completion of a task and a subsequent stimulus for the next task was shown to reduce task-switching cost. Switching between similar tasks had a lower cost, but switching from a set of similar tasks to an unrelated task had a higher cost. A fuzzy-logic model of these observations was created and used to develop set of algorithms to determine optimal system performance based on human workload and expected performance. Using the virtual test environment HumAnS-3D, the authors compared a fitness function based solely on individual performance values to the new methodology incorporating switching costs. The new task-switching model resulted in faster completion times, showing that using customized models to aid in role allocation for team scenarios can improve system performance.

Programming of robot behavior can use models of human behavior to facilitate more natural interaction within humans. Trafton et al. (2005) argue that modeling human–robot systems based upon human–human interaction is effective and used the Polyscheme cognitive architecture (Cassimatis, 2002) in order to improve robot behavior.
Three conceptual guidelines to consider in human–robot systems are as follows: (a) Robot perception, reasoning, and representation should be similar to human systems; (b) cognitive systems should integrate cognitive architectures; and (c) the use of heuristics similar to those that humans use is effective. The authors designed a robot that was able to generate a representation of the visual perspective of the human with whom it was communicating to make the human–robot conversation more like one held between humans. After analyzing videos of human–human interaction during astronaut training, the authors created a model using Polyscheme to represent the human behavior of simulating another visual perspective. Using Polyscheme to create a representation of the environment from the human’s point of view aided the robots by helping to disambiguate commands from and actions taken by the human. The model was integrated in the programming of a robot that was given ambiguous commands from a human to select objects in the room. The robot was not able to solve ambiguities before the perspective-taking model was added. Integrating the model based on human–human interaction aided the human–robot conversation.

Human performance models and predictions are useful in situations in which humans directly interact with robots for tasks such as role allocation, robot behavior design, and workload estimation. Similarly, interacting with a robot in a remote location can also be improved by modeling human performance patterns and limitations.

Military use of unmanned aerial vehicles has become a common type of teleoperation. Such robots are often used to perform surveillance or reconnaissance. As mentioned earlier, spatial awareness and orientation are important when working with a remotely located robot. Gluck et al. (2005) developed cognitive models in the ACT-R software to simulate human accuracy in three-dimensional spatial orientation tasks, with the eventual purpose of using these models to aid system design and provide additional training opportunities. Experimental validation showed that these models provided good predictions of human behavior. Models simulated control of Predator unmanned aerial vehicles in the Predator synthetic task environment (Schreiber, Lyon, Martin, & Confer, 2002). The models were shown to be good predictors of expert pilot performance in basic maneuvering tasks. Interfacing the synthetic task environment with ACT-R was achieved by reprogramming the visual display for the task environment. Extending this research is the Verbalization Between Operators and Synthetic Entities (VERBOSE) project, which aims to create simulated agents that can use verbal communication as a computational cognitive linguistic system (Ball, 2004). Simulation language skills and task performance may provide a valuable training tool for pilots of unmanned aerial vehicles.

Some unmanned aerial vehicles require multiple operators. Specifically, the Shadow 200 System was determined to require approximately 22 to 28 personnel by an IMPRINT model developed by Hunn and Heuckeroth (2006). The model’s goal was to explore task assignments to the operators of the Shadow 200 and associated workload outcomes. The model inputs included task time and number of people required, and frequency of each subtask was determined by surveying military personnel with training and experience with unmanned aerial vehicles. Workload values for each subtask were assigned according to IMPRINT’s built-in guidelines. The model was used to provide estimates of crew workload during the mission, individual crew member workload, baseline levels of workload for each crew member, how long the mission would take, and how many personnel it takes...
to complete the task. Using data collected from field-experienced personnel and IMPRINT’s workload level assignment guidelines, the model was able to answer all of these questions. The model was not validated with an empirical evaluation, but the use of subject matter expert data to create the model resulted in useful information for the authors.

Petkosek, Warfield, and Carretta (2005) conducted a similar investigation on unmanned aerial vehicle operation by creating models using the Combat Automation Requirements Testbed (CART) to model task completion time and workload levels for operators. CART is based on the IMPRINT tool. The task scenario involved surveillance of a potentially dangerous situation including refueling a number of grounded passenger planes. The models served as an investigation tool for task decomposition and workload estimation but were not experimentally validated.

Deutsch (2006) used a variant of the same scenario and created three human performance models of each of three unmanned vehicle operators. Using the Distributed Operator Model Architecture (D-OMAR), the authors developed a test bed for the unmanned aerial vehicles and were able to assess the impact of tasks on each of the three operator roles: sensor operator, aerial vehicle operator, and multifunction operator. The model’s goals included analyzing workplace design and operating procedures, improving model robustness, and reducing training time. Team performance featuring these three operators can be simulated with the model in order to test potential task assignments and to identify minimal staffing combinations before spending the time and money for tests with actual unmanned aerial vehicles and personnel.

Other than military domains, urban search and rescue is a burgeoning avenue of research for remotely located semiautonomous or teleoperated robots. Human–robot interaction during an urban search-and-rescue environment is difficult and requires multiple tasks often performed under time pressure, for example, steering the robot and monitoring sensor feedback. Ritter, Van Rooy, St. Amant, and Simpson (2006) aimed to create an operator model in order to improve human–robot system designs. As a starting point, a model of a person driving a car was created using the ACT-R cognitive architecture and the SegMan eye and hand simulator (St. Amant, Horton, & Ritter, 2004). This model was called the Driving User Model in ACT-R and SegMan (DUMAS). After the experimental validation, some limitations of the model were determined, including the lack of a complex interface in the test and the model’s visual system. The model was extended to process visual information in real time in order to more accurately represent the human visual system and was tasked with teleoperation (Ritter, Kukreja, & St. Amant, 2007). A study compared human performance to the model using metrics of lane deviation and total time before crashing. The model was shown to be a reasonable predictor of human performance. This work demonstrated the ability for a model of human behavior to interact with a real-world robotic system in real time while processing complex visual displays. Using a model of a human user like DUMAS can aid in the development of more efficient interfaces with remotely located robots.

Kaber, Wang, and Kim (2006) extended a GOMS Language model for a human–robot system to predict human performance while navigating a robot. The GOMS Language model code was fed into a compiler called Error-Extended GOMS Language Evaluation and Analysis (EGLEAN). The model emulated human control of a single small ground vehicle traveling on a simple path at a fixed speed. Data gathered from a single operator
executing the task guided the model’s creation. The model predicted an execution time that was approximately 10 times the actual human performance time and with one tenth of the human error on path navigation. A revised model, relaxing a path tracking accuracy criterion, yielded model performance times only twice as long as human operators and with errors rates at one fifth of the human error. While the time and accuracy predictions were not spot-on, this work demonstrates the ability to use GOMS models to represent human behavior in a human–robot system. The revised model Kaber et al. developed could be further relaxed in terms of control accuracy to better represent actual operator rover control behavior. In addition to GOMS, other human performance modeling methods focusing on parallel operation execution may provide more accurate timing or error rate predictions.

The previously described systems are traditional examples of human performance modeling. A broader definition of human performance modeling can include affective models. Models of performance can be developed during interaction with a system rather than created as a prediction of the interaction with a system. Models created during interaction can be used to adjust system parameters for aiding the individual human working with the system. Along with workload levels and task completion time, a person’s affect can offer insight into performance. Rani, Sarkar, Smith, and Kirby (2004) developed a way to monitor human affective state through physiological responses while interacting with a robot and used signals from the human to modify the robot’s behavior. Liu, Rani, and Sarkar (2006) expanded upon this methodology and altered movements of a robotic basketball hoop according to participant anxiety levels. Participants threw basketballs into the robotic basketball hoop under low, medium, or high levels of motion associated with high, medium, and low anxiety. The robotic basketball hoop was able to produce lower levels of anxiety and higher levels of performance when responding to the participant’s current affect. These affective models of human anxiety map how changes to the experimental situation influence the state of human game play. An individual model of affect was required for each participant.

Additionally, Liu, Agrawal, Sarkar, and Chen (2009) used physiological measures to create a model of human affect usable during human–automation interaction. Participants wore a physiological monitoring system that recorded anxiety levels when interacting with a video game. The authors compared interactions using information regarding the user’s affective state to modify the game’s difficulty and compared it to games in which difficulty was modified only by participant task performance levels. Each participant went through a training period, and the data were used to create an affective model. Alternate methods of creating an affective model were tested, and a regression tree–based affective model provided the most accurate predictions (Rani, Sarkar, & Adams, 2007). The affective model–based game modification resulted in higher performance levels for most participants, participants’ rating of the game as more challenging and satisfying, and lower participant-perceived anxiety. While a limitation of this technique includes the extensive training period to build an affective model for each participant, the benefit to performance of developing these systems is apparent. Modeling the internal state of each human benefited the system’s overall performance. Internal factors, such as workload, interaction effort, and anxiety, affect performance.
Additionally, human performance modeling can be considered when using a cognitive architecture to emulate human behaviors by a robot. Trafton, Bugajska, Fransen, and Ratwani (2008) modeled a robot's behavior during conversation with a human based upon behaviors present in human–human interaction. People speaking in group conversations typically look at the person who is talking but wait approximately 500 ms to switch to a new speaker. ACT-R (Anderson & Lebiere, 1998) was adapted into ACT-R/Embodied for an embodied robotic conversation member by connecting the audition and vision modules to a real robot and adding pedal and spatial modules for moving about and representing the environment, respectively. The robot was tasked to look at whichever person was currently speaking, even in the presence of challenges such as ambient noise and speaker interruptions. An evaluation rated the naturalness of the robot looking at each speaker with and without the 500-ms pause before switching gaze; participants subjectively rated the naturalness of conversation in the two conditions. The addition of the human-like pausing period was perceived as much more natural compared to the system that did not wait, indicating that more human-like behavior is seen as more natural. The system design was improved by including a model of human behavior.

Multiple-robot interaction. Interactions with multiple robots have also been analyzed through human performance modeling. General models of interaction will first be presented, followed by specific systems that have implemented human performance modeling in multiple-robot systems.

Determining how many robots to use, robot autonomy levels, and appropriate overall system design is a complex task. Crandall and Cummings (2007) proposed a set of metrics for the control of multiple robots performing independent tasks in remote locations. Metrics are measurements and criteria that allow for assessment of the effectiveness of a system and should (a) contain key performance parameters of a team, (b) identify limits of agents in the team, and (c) be able to generalize to other situations. Supervisory control of multiple robots differs from a single-robot case in that human attention is divided between robots. A metric specific to multiple-robot interaction is attention allocation efficiency, which encompasses global situational awareness, switching times, selection strategies, percentage time spent following an optimal policy, and wait times due to loss of situation awareness. The interaction efficiency and neglect efficiency are two metric classes that are relevant to both single-robot and multiple-robot control. Interaction efficiency measures the effectiveness of the team, and neglect efficiency represents the efficiency of a single robot when neglected by the human.

Metrics for multiple-robot control have been extended to create models of interaction based on human behavior and robot performance (Crandall, Cummings, & Nehme, 2009). The need for division of human attention among multiple robots was modeled in an emergency response scenario using the Research Environment for Supervisory Control of Unmanned Vehicles (RESCU). The scenario involved a search-and-rescue mission whereby a human supervised multiple robots and was tasked to find as many objects as possible from a building 8 min before the building exploded. The model predicted both the number of objects found and the number of robots lost in the explosion. The difference between the number of robots that were left in the building to explode and the number of objects found resulted in an overall score for the mission. Two interface types and two levels of
autonomy of the robots were tested. Results revealed the models to successfully predict the mission score. The model has been shown to apply only to single-human, multiple-robot control in which the robots do not collaborate and the operator cannot interact with groups of robots together. Despite these limitations, the model provides a good prediction of team performance within the specified assumptions.

As indicated, when a human supervises a team of multiple robots, attention must be divided between the robots. Crandall et al.’s (2009) research revealed the need to improve attention allocation representation in human performance models. Crandall et al. (2011) created a system to determine what human operators can attend to at a given time, based upon a model of optimal attention allocation using the prior scenario. Each robot’s state was evaluated and categorized according to the priority of that state based on the current time. For example, in the first minute, replanning paths and assigning tasks took top priority, but during the last minute, it was more important to move idle robots and pick up objects. Model predictions indicated that optimizing attention allocation would increase system performance. A second user evaluation focused on the impact of attending to specifically selected robots via manual selection, automated selections, or a guided mode that provided suggestions. Overall scores did not improve with the automated selection, and participants preferred the guided mode. This work revealed that an optimal attention allocation strategy may not be necessary, and aiming for satisficing selections with flexibility may be ideal for human operators.

An important aspect of moving about any physical environment is path planning. Human–robot systems that include remotely located semiautonomous robots sometimes require an operator to designate a path for the robot to follow. Reitter and Lebiere (2010) created an ACT-R cognitive model of the human path-planning process by incorporating information about the environment retrieved from declarative memory and visual search. The memory components of the model defined location memory chunks that represented a state of the system and path memory chunks that represented transitions between location chunks. Production rules facilitated model use of subgoals to move the robot between locations using memories of paths to the goal. The visual aspect of the model allowed for selection of paths to subgoals within a short visual range using straight lines and without using memory chunks or production rules. Incorporating the two strategies allowed the model to successfully navigate mazes.

The memory-based strategy model and the visual-based strategy model were each validated separately before using a combination of the two in a third evaluation in which participants controlled 4-, 8- and 12-robot teams through a computer interface. The goal of the task was to investigate the layout of an unknown building. Participants created itineraries of robots traveling through the building. The model was run using the start and end points of these participant-created itineraries. The difference in itineraries between model-created itineraries and the participant data was computed using trajectory area normalization, which determines the area of the space between the two itineraries. The model was able to predict the robot itineraries created by participants very closely. These results validated the two-part cognitive model of human path planning.

Human performance modeling has been applied to robotics and automation interaction in a variety of ways: interacting directly with a single robot, remote interaction with both single and multiple semiautonomous or teleoperated robots, and teamwork. Metrics
have been designed for both single-human, single-robot interaction and a single human supervising multiple semiautonomous robots. Human performance can be emulated with robot behavior or analyzed to identify trends resulting from work with robots as opposed to humans. System design can improve based upon model and user performance comparisons; workload levels from common missions can be analyzed using human performance models. In general, introducing human performance modeling techniques to robotic systems can occur in many different, but valuable, ways.

**HUMAN PERFORMANCE MODELING VERIFICATION TECHNIQUES**

The verification of human performance models, especially when applied to robotics, is critical. The utilization of modeling predictions is not possible without verifying that the model applies to the real-world system. Evaluating the degree to which a model represents real system performance typically involves a user evaluation. After demonstrating that a model is representative of a real-world task, one may use the outcomes of variations on the model to predict system performance, including manipulations of human performance moderator functions. For example, if a model has been validated for a human–robot team building a table together, the model can be adapted for adding a third team member, performing the task outside with wind, or adding the assumption that the human is very fatigued. It is important to remember that adapting a validated model in this way assumes that the new adjustments to the original model are based on accepted research findings.

Creating a database of validated human performance models for human–robot systems can reduce the need to verify each model. Such a collection of adaptable validated human performance models does not yet exist. Thus, validation is an important step in current applications of human performance modeling for human–robot systems.

Validation metrics from prior human performance modeling for robotics are provided in Table 3.1. Gawron (2008) proposes 12 considerations for performing an experiment, including the following: (a) Define the question to be answered with the experiment (e.g., Does the model predict accurate task times? Is timing different between Task A and Task B?); (b) check for qualifiers that restrict the generalizability of results; (c) specify the conditions (e.g., Task A, modeled; and Task A, experimental) and whether they are between- or within-subject conditions; (d) select performance measures, data collection equipment, and data recording equipment; and (e) match trials of the same length, level of difficulty, and environmental conditions.

Once the question is defined and a scenario is outlined, the model is designed to include what the participants experienced throughout the evaluation and how the robot or automated system is incorporated in order to match the environmental conditions in the model and evaluation in Consideration (e). Consideration (d) is crucial to the verification of models through experimentation. A modeling tool should be chosen that will output metrics measurable in the evaluation. John and Newell (1989) assert that a model is valid if it does not deviate more than 20% from actual human performance. Possible metrics include primary task performance, secondary task performance, physiological measures, and subjective responses.
Task performance and task timing measures can be directly compared between model and experimental results. Steinfeld et al. (2006) offered a set of metrics specifically for human–robot systems. While robot performance alone is not the focus of a human performance model, the robot’s effect on the human and metrics such as fan-out and time for the human to notice a robot’s request can be modeled and compared with experimental results. The relationship between the human and robot is also an important consideration. During teleoperation, other metrics, such as obstacles avoided or percentage of time on correct path, can be relevant validation metrics between the model and experimental data.

Salvucci’s (2006) ACT-R model of driving behavior, for example, was validated by directly comparing steering angles at specific time points along curving roads with given curvature, lane positions during lane changing, and proportion of eye movements spent on specific areas to match the model profiles to human data. The model was shown to accurately predict human behavior in these three categories, demonstrated through significant coefficients of determination.

Quantitative value comparisons between models and experimental results are often achieved through timing data predictions and performance scores (Howard, 2007). Howard’s work compared time data for tasks executed by a human teleoperating the robot and the robot executing the tasks while moving autonomously. The HumAnS-3D system model predicted timing information for many different task assignments between teleoperation and autonomous navigation. These predicted times were compared with tasks allocated between team agents using a traditional method, and the times were shown to be the same or faster using the HumAnS-3D system.

Ritter et al.’s (2007) ACT-R model) used a simulation to directly interact with the graphical user interface, using a model of human vision. The validation of this model compared performance metrics (e.g., number of mouse clicks), durations of actions (e.g., the time to drive the robot to the object and back), and error occurrences between the model outcomes and human participant performance.

Secondary tasks have also been shown to represent spare mental capacity, particularly in relation to workload (Gawron, 2008). Secondary tasks are separate from the primary task and can include, for example, responding to verbal memory questions or completing math problems. Response time and accuracy of responses to recall and recognition questions have provided insight into human workload levels and spare mental capacity (Adams et al., 2012; Harriott et al., 2011a, 2011b). Metrics including correctness and speed to respond can indicate levels of participant performance and workload. Using a subset of the metrics by Olsen and Goodrich (2003), neglect tolerance was modeled and subsequently measured via user study (Crandall, Goodrich, Olsen, & Nielsen, 2005). The two secondary tasks included controlling a second robot and performing simple arithmetic problems on the robot control display. Secondary task performance informed the calculation of neglect tolerance, interface efficiency, and interaction times.

Another way to validate human performance models includes measuring human physiological responses. Heart rate, respiration rate, and heart rate variability have been analyzed for comparison to workload, effort, and physical activity (Aasman, Mulder, & Mulder, 1987; Poh, McDuff, & Picard, 2011; Roscoe, 1992; Vicente, Thornton, & Moray, 1987). Heart rate variability, for instance, was shown to decrease as cognitive load increases.
Modeling cognitive load and comparing trends to heart rate variability is a means of validating the model. Physiological measures can be recorded using a variety of methods, including electrocardiogram systems, chest strap heart rate monitors, and so on (Vicente et al., 1987) or even by computationally analyzing video of participant faces (Poh, Kim, Goessling, Swenson, & Picard, 2011; Poh, Mcduff, et al., 2011). Other metrics, such as skin conductance, skin temperature, vector magnitude, and pedometer data, can correlate with human task performance that can be relevant for comparison with a model (e.g., Steele, Holt, Belza, Ferris, & Lakshminaryan, 2000).

Subjective responses can be compared to modeled predictions as well. Many of the human performance modeling tools include a method of representing workload. Workload is an important aspect of human performance, and fluctuations can reflect system changes (Wickens, Lee, Liu, & Gordon Becker, 2003). Workload can be measured subjectively, for example, with the NASA Task Load Index survey (Hart & Staveland 1988) or the Multiple Resources Questionnaire (Boles et al., 2007). Subjective workload measurement can provide insight into how participants perceive demand but typically must take place after completing a task or subtask, which can limit the granularity of knowledge about how workload changes over time.

Harriott et al. (2011a, 2011b) focused on determining whether IMPRINT Pro’s method of assigning workload values are applicable in a human–robot peer-based team. The validation involved physiological measures and subjective responses in addition to primary and secondary task performance. During the evaluation, participants wore a chest strap heart rate monitor, answered memory-related secondary task questions, and subjectively rated workload. Physiological measures recorded by the heart rate monitor included heart rate, respiration rate, and heart rate variability. Secondary task questions required participants to recall a list of names provided during the training. Throughout the evaluation, participants were asked to recall whether or not a given name was or was not on a provided list. Subjective workload ratings were gathered by periodically asking participants to rate experienced demand on a scale from 1 to 5 along six channels (cognitive, auditory, visual, speech, motor, and tactile). Modeled workload was not significantly different from experimental workload in either condition. Model results were accurate with respect to the timing of tasks. The accuracy of the models indicated that IMPRINT Pro can be used to model human–robot peer-based teaming scenarios.

In order to use a model to predict human performance and behavior under a specific set of conditions, the model must first be validated for the target domain. Models have been verified in a variety of ways (Crandall et al., 2005; Harriott et al., 2011a, 2011b; Howard, 2007; Kaber et al., 2006; Reitter & Lebiere, 2010; Ritter et al., 2007; Salvucci, 2006). Again, it is important to know what question is being asked in the evaluation and to derive the correct measurements in order to evaluate whether the model truly represents human behavior. In general, when attempting to validate a model of human performance, it is important to identify model outcomes/metrics that can be directly compared with human responses, whether they are physiological measures, subjective ratings, time data, or task performance. This section presented multiple examples of research that investigated the applicability of human performance models and validated the capacity to represent human behavior.
Human performance modeling can be instrumental to design and analysis of human–robot systems. The creation of human performance models can, for example, provide insight into the step-by-step breakdown of tasks, offer ways to compare system configurations and task assignments, estimate the length of time a task will take, or improve a human’s interaction with the system in real time. This chapter focused on modeling human performance within human–robot systems. The chapter presented general motivation for modeling human performance, common modeling tools and rationale for choosing the most appropriate tool, examples of successful human performance modeling of non-robotic systems, and applications of modeling human–robot systems. Finally, model verification techniques were presented to help ensure accuracy and applicability to the real world. Table 3.2 summarizes the major points of each section of this chapter.

### GUIDELINES AND LESSONS LEARNED

Human performance modeling can be instrumental to design and analysis of human–robot systems. The creation of human performance models can, for example, provide insight into the step-by-step breakdown of tasks, offer ways to compare system configurations and task assignments, estimate the length of time a task will take, or improve a human’s interaction with the system in real time. This chapter focused on modeling human performance within human–robot systems. The chapter presented general motivation for modeling human performance, common modeling tools and rationale for choosing the most appropriate tool, examples of successful human performance modeling of non-robotic systems, and applications of modeling human–robot systems. Finally, model verification techniques were presented to help ensure accuracy and applicability to the real world. Table 3.2 summarizes the major points of each section of this chapter.

Many modeling tools and validation methods were presented in this chapter. This section outlines a set of guidelines for determining when modeling is beneficial and the step-by-step modeling process, including tool selection and model validation. First, it is important to explicitly identify why human–robot system research benefits from the development of valid human performance models.

Determining when modeling is beneficial can depend on the resources available and the developing organization’s structure. Booher and Minninger (2003) developed a list of 10 important human–system integration factors that should be considered when...
an organization or group is deciding whether and how to develop human performance models for military applications. The factors were determined using case studies of military system modeling. Analysis of these factors helps identify the potential benefit and influence of modeling as well as the preparedness of the organization or group to create a model. The factors are generalized for consideration here:

1. **Top-level support** is the level of understanding of modeling capacities and supporting resources (e.g., funding, manpower, time) for human performance modeling provided by the project manager, principle investigator, or funding source.
2. The pursuit of human-centered design motivates human performance modeling. System design can benefit from modeling predictions when emphasis is placed on human performance and experience while working with the system.
3. Determining an appropriate project funding source selection policy, if options are available and sponsorship is needed, places a high weight of importance on model results. This step will ensure models will be allotted necessary resources.
4. Organizational integration implies that more than one domain should be considered, and associated experts consulted, when assessing the desired influence of modeling, for example, safety, design, or speed.
5. System documentation integration is required and entails feeding model and system documentation directly into systems requirements documents and verification and evaluation planning documents.
6. Quantitative human performance analyses should be accurately represented in the model and used for comparisons with model outcomes in verification evaluations. A wide library of human performance moderator function data exists in the military and human factors domains that can be utilized for modeling. Qualitative data (e.g., open-ended questions) can be difficult to interpret and compare to a model.
7. Technology involved in the modeling process should be analyzed. Selecting an appropriate modeling tool is critical. Additionally, representing system technology accurately can impact the human performance predictions with the model.
8. Integrated test and evaluation refers to model verification and validation evaluations. It is critical to identify metrics of performance and human–system effectiveness to determine the utility of the model.
9. Practitioners are the skilled model creators and evaluators necessary for creating and using human performance models. Practitioners must have expert knowledge of the tasks, modeling tools, human performance metrics, and evaluation design.
10. Education and training are essential for ensuring that model developers have necessary knowledge and skills. Training can encompass formal courses specific to the chosen modeling tool, expert knowledge of the domain to be modeled, and academic resources. Along with consideration of the availability of trained personnel, an organization must consider the availability of education and training for model development.

These 10 factors represent concerns for organizational preparedness for modeling and analysis to benefit the design process. These factors are relevant to model development in any domain, not just military-focused domains. The development of a
human performance model requires careful consideration of each of these factors throughout the model creation process. Creating a model of a human–robot system involves executing a series of necessary steps. During the execution of each of the model creation steps, it is important to reflect on the 10 organizational human–system integration factors. The rest of this section will delve into each of the 10 steps of model creation and use.

1. **Identify the primary question.** First, a primary investigation question must be identified. It is important to know exactly what modeling should achieve, and answering the primary question is the main goal of the model. The primary question also helps to define the scope of the model.

2. **Identify a scenario that will assess the primary question.** Models cannot include every aspect of a task scenario that may affect human performance. It is necessary to analyze the scenario and select the most important factors present in the physical environment, team relationships, task details, human internal states, and/or robot design. Some situations may depend heavily on weather conditions, while other scenarios occur indoors. Another situation may involve carrying a large payload, or performance may be more strongly influenced by human height.

3. **Choose a modeling tool.** This step involves (a) the selection of parameters relevant to the modeled system, (b) determining whether the selected parameters are supported by the modeling tool or can be added, (c) the level of decision making performed by the human, (d) the presence of teamwork or adversarial relationships, and (e) the desired metrics produced as model output. The data necessary to incorporate all relevant system parameters will not always be available for the specific scenario, and the chosen modeling tool may require extensive work to address specific performance parameters. Data and modeling tools may be available for human–human or human–system interaction, but it cannot be assumed that the same rules governing human performance outcomes produce the same results in human–robot systems. Experimentally determining the presence (or lack) of quantifiable changes to human performance due to interaction with robots will create a more accurate and more useful model.

Each of the presented modeling tools offers a unique set of advantages. Cognitive architectures offer the benefits of realistic representations of decision making and memory. Tools such as ACT-R, Polyscheme, or Soar are optimal in situations in which a human perceives the current state of the environment and uses a set of condition-based rules to perform actions upon the environment. Model outputs can include the set of decisions made (e.g., turns made in a rover motion path; Reitter & Lebiere, 2010) as well as timing information. Cognitive architectures offer the unique advantage of providing outputting agent decisions.

Human performance modeling tools, such as MIDAS and IMPRINT Pro, require a general set of tasks in a predetermined order, as they include the discrete event simulation tool Micro Saint Sharp. These tools can include conditional actions, but each condition must be specified. The tools work best for situations involving structured tasks with a known sequence of events and are best for providing comparative results for a set of alternate
systems (e.g., System Configuration A provides faster task completion times than System Configuration B during daylight hours, but the reverse is true at night). IMPRINT Pro, for example, also provides the means to add customized performance-influencing parameters (ARL, 2009). Many modeling tools provide the means to model internal factors (e.g., workload and situation awareness) and external influences (e.g., number of team members and system design). PMFServ not only offers a large database of performance moderator functions but is unique in that it provides the capacity to model emergent phenomena from crowds and social behavior (Silverman et al., 2002). However, support for modeling human–robot systems by PMFServ is minimal.

4. Decompose the tasks. Once a scenario and modeling tool are chosen, a set of tasks must be outlined for modeling. The level of granularity of the model is dependent on the results of Steps 1 through 3. For example, using a modeling tool such as ACT-R to evaluate visual perception of changes in lighting would have a smaller granularity of tasks in comparison to use of a tool such as PMFServ to evaluate the influence of weather on crowd formation. Task analysis is necessary to break down the steps a human will perform in completing the task to be modeled. Each subtask will generally have a goal and a necessary action for completion. Walking 10 feet, for example, can be broken down further by identifying the subtasks of individual leg lifts, foot plants, knee bends, and so on.

5. Create the model. Using the set of tasks identified in Step 4, a model can be created. The method of inserting individual tasks into the model highly depends on the chosen modeling tool. For example, it is sometimes necessary, before and during model creation, to consult subject matter experts in order to confirm variable assignments, task decompositions, and timing values.

6. Plan and execute a comparable verification study. The accuracy of models and their applicability to real-world scenarios are determined through validation and verification evaluations. The evaluations must focus on measurement of the model output metrics. Multiple physiological, observational, and subjective metrics can be compared directly to model outcomes. It is important to experimentally measure more than one metric appearing in the model output (e.g., task completion time and task completion accuracy) in order establish the validity of aspects of the model and to identify which characteristics are not realistic.

7. Assess the accuracy and applicability of model. The model can be validated by a number of methods, including (a) demonstrating the influence of independent variables on dependent measurements from both the model and evaluation and (b) computing the delta or percentage difference between modeled and evaluation results. Method (a) can indicate trends in the data using statistical tests. For example, if participant visual acuity reveals a significant correlation with task completion accuracy in evaluation results, but such a trend is not present in the modeled results, the model may not be a good predictor. Method (b) demonstrates the actual difference between the model and evaluation results. This difference can be computed as a percentage and compared against John and Newell’s (1989) guideline that a model within 20% of experimental

---

Downloaded from rev.sagepub.com by guest on July 14, 2015
results is valid. The difference can also be evaluated using an analysis of variance or other statistical tests to determine whether the model provides significantly different results than the evaluation provides.

8. If the model is determined to not be representative of the comparable evaluation, return to Step 4. The model may not represent a crucial aspect of the real-world system. Human performance moderator functions may have been missed (e.g., the presence of wind was not included) or inaccurately represented in the model (e.g., physical workload was too low). There is a missing model parameter that needs to be identified, adjusted, and validated again.

9. Identify adjustments necessary for the next iteration of the system, and use the model for predictive purposes. A model must first be verified before use in prediction. Once the model is shown to accurately represent a currently evaluated system, the model can be used to predict changes to the system, more extreme operating conditions, and so on. These model manipulations will be based on other validated factors in human performance (e.g., the documented effects of cold weather on task accuracy). Using model predictions can be integral to improving system design and human–system performance.

10. Improve system design using human performance modeling. The final step in the model creation process is to use model predictions from Step 9 to make changes to the design of the system or robot. Model results can be used to compare performance of alternative system designs and help to choose the system that ensures the best performance. For example, human perceptual system models can identify the optimal placement or size of displays on a robot. A model can help determine how accurately humans can use a touch screen interface to teleoperate a robot or provide clear voice input to a robot in the presence of background noise. Model results can help determine display and input modes (e.g., auditory, visual, tactile) or sizes, optimal team task configurations, expected task execution times, and suboptimal operating procedures. Once the model predicts human performance in various conditions with a system, decisions can be made regarding which configuration may result in the best human performance.

The utilization of both the organizational human–systems factors list and the list of 10 model creation steps can ensure that the modeling process and results are allocated proper resources and answer the right questions about the given system. Human performance models have been shown to provide avenues to technology advancements by adapting to the state of the art quickly, enhancing system design by evaluating design effects on human performance, increasing safety in military domains, and providing large returns on investment (e.g., thousand-fold returns for some military projects; Booher & Minninger, 2003). This chapter presented a variety of examples of improvements to human–robot systems facilitated by the use of human performance modeling. Examples include the following:

- Metrics for assessing interactions have been established to objectively gauge levels of system performance (e.g., Crandall & Cummings, 2007; Olsen & Goodrich, 2003).
Best practice for assigning tasks to teams of unmanned aerial vehicle operators have been investigated (e.g., Gluck et al., 2005; Hunn & Hueckeroth, 2006; Petkosek et al., 2005).

System performance is improved when robot behavior is based upon principles of human performance (e.g., Trafton et al., 2005).

A human performance model will never be perfect, but when the proper considerations for selecting both a modeling tool and verification techniques are made, a model can be a useful tool. Human–robot systems are a diverse and quickly expanding field without many human performance modeling standards. Human–robot systems have been improved through the use of human performance modeling, including development of robots that emulate aspects of human behavior, comparisons of alternate system designs before implementation, assessments of levels of human workload, and optimizations of task timing by modeling task assignments. The utilities of human performance modeling to human interaction with robots are many. Adapting and verifying human performance modeling techniques for human–robot systems is worthwhile. In general, models offer a safe time- and cost-saving method of simulating behavior.

REFERENCES


### ABOUT THE AUTHORS

**Caroline E. Harriott** is a graduate student in the Human Machine Teaming Laboratory in the Department of Electrical Engineering and Computer Science at Vanderbilt.
University. She received her BA in computer science from the University of Virginia in 2009 and her MS degree in computer science from Vanderbilt University in 2011. Her research focuses on investigating human performance in human–robot teams.

Julie A. Adams is an associate professor of computer science and computer engineering in the Electrical Engineering and Computer Science Department at Vanderbilt University, where she directs the Human Machine Teaming Laboratory. Her research focuses on distributed artificially intelligent algorithms for autonomous multiple robot coalition formation and the development of complex human–machine systems for large human and robotic teams. She worked in human factors for Honeywell, Inc., and the Eastman Kodak Company (1995–2000) and was an assistant professor of computer science at Rochester Institute of Technology (2000–2003). She is the recipient of the NSF CAREER Award. She received her bachelor of science in computer science and her bachelor of business administration in accounting from Siena College. She received her MSE (1993) and her PhD (1995) in computer and information sciences from the University of Pennsylvania.