INTERNATIONAL JOURNAL OF COGNITIVE ERGONOMICS, 2001, *5*(1), 1–21

Using Time Windows to Evaluate Operator Performance

Ling Rothrock
Department of Biomedical, Industrial, and Human Factors Engineering
Wright State University

ABSTRACT

A proper understanding of human performance characteristics is a prerequisite for designers of complex systems. Although human factors texts provide some insights into basic performance issues, the emergence of highly automated computing systems have fundamentally altered the way humans work. The purpose of this article is to present a research approach to quantify and analyze human performance within a complex, time-critical system. The approach is centered on a measurement construct, called a time window, which enables a functional relation between constraints on operator activities and time availability.

A blackboard model is developed as the mechanism to generate, maintain, and complete time windows. Moreover, an object-oriented methodology is described that implements the blackboard model within a realistic task context. To demonstrate the utility of time windows, an existing implementation in a real-time human-in-the-loop simulation is also described. Using time window outcomes, some cursory analyses are completed to exhibit the potential of the construct.

1. INTRODUCTION

The emergence of highly automated computing systems has fundamentally altered the way humans work. As these systems have increasingly become mediators between human operators and the work environment, human understanding of how work is accomplished has greatly diminished. Remarks of “What happened?” or “Why did it do that?” are not uncommon as operators seek to understand the processes of systems designed to improve their work. Rather than serving the purpose of being tools for human use, these systems have come to be regarded as autonomous agents to which humans must adapt in the workplace.

To investigate human decision making in these highly automated systems, researchers have had to rethink the applicability of traditional laboratory methods such as ex-

---

Requests for reprints should be sent to Ling Rothrock, Department of Biomedical, Industrial, and Human Factors Engineering, Wright State University, 207 Russ Engineering Center, 3640 Colonel Glenn Highway, Dayton, OH 45435–0001. E-mail: lrothroc@cs.wright.edu

pected utility theory (Beach & Lipshitz, 1993). The use of traditional methods assumed that findings from the laboratory environment—where highly cognitive, single-choice tasks were conducted—could be applied to more realistic settings. The premise that findings from a static, forced-choice task can be extended to an operational environment has been called into question (Hammond, 1986). In fact, some researchers have even suggested that the entire approach toward the factoring of processes into events occurring inside and outside the mental system is misdirected (Suchman, 1987; Vicente, 1997; Vicente & Kirlik, 1992). These researchers have recommended that studies of human operators must occur in settings that are representative of the actual environment. More important, the search for emergent properties of operator activities in situated contexts should be favored over the study of isolated factors influencing cognition (Suchman, 1987).

The purpose of this article is to present a research approach to quantify and analyze human performance within a dynamic task context. The key concept introduced here is the notion of a time window that provides a functional relation between constraints on operator activities and time availability. A methodology is proposed to evaluate time windows as well as to assess operator attunement to them. A simulation environment is used as a tool (Brehmer, Leplat, & Rasmussen, 1991; Howie & Vicente, 1998) to bridge the gap between controlled, unrepresentative laboratory environments and unconstrained real-world settings.

The proposed research methodology is shown in Figure 1. First, the researcher extracts those situations and time and task constraints that contribute toward meeting an operator's objective in the operational domain. The extracted information is then implemented in a simulation using object-oriented and blackboard methodologies. Data is then collected on activities within the human-in-the-loop simulation. Lastly, performance evaluation is conducted by determining the extent of operator attunement to constraints. To illustrate the proposed methodology, results from an existing implementation are discussed.

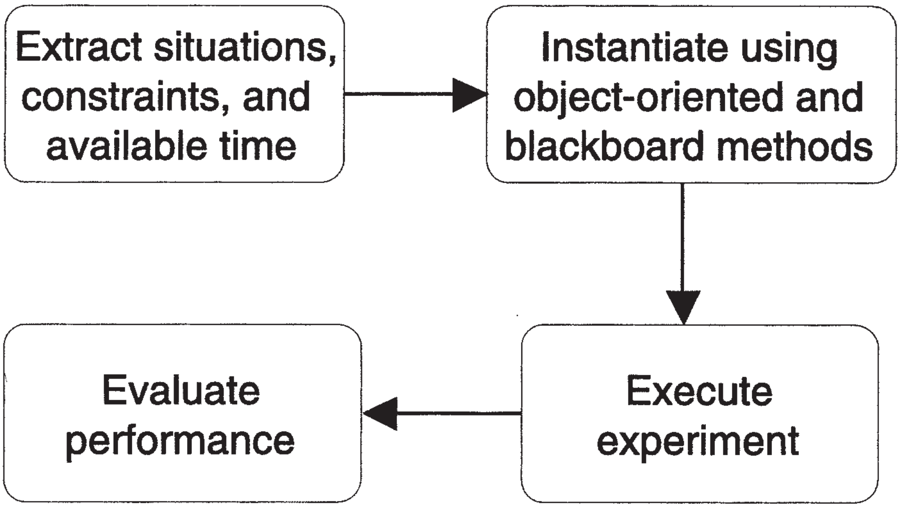

FIGURE 1 Proposed research process.

2. SITUATIONS, CONSTRAINTS, AND TIME WINDOWS

2.1. Situativity Theory

To extract situations, constraints, and available time, these terms must first be clearly defined. The meaning of the terms *situation* and *constraint* as they have been used thus far is consistent with the interpretation provided by Greeno and Moore (1993) and Greeno (1998). They introduced a theory of situativity in which cognitive processes are analyzed as relations between operators and other subsystems in the environment. The theory is powerful because it stipulates that a functional relation exists between an operator's decision-making activities and the task environment. For instance, consider the act of answering the telephone. An individual's action of answering the telephone results in the situation of talking on the telephone. The dependency relation between an action and the resultant situation—also known as a constraint—contains the following form:

<<answering the telephone>> => <<talking on the telephone>>

Greeno (1994) presented a class of more generalized constraint as follows:

<<action by operator>> => <<good effects in situation>>

where the good effects are outcomes that are required for a broader activity to be successful.

2.2. Time Window Extension to Situativity Theory

The notion of a time window is an extension to situativity theory. Situativity theory is conceptually appealing; however, work remains to implement it as a mathematically rigorous model (Greeno, 1998). To computationally implement the time window extension, therefore, a greater degree of definitional precision is required. Accordingly, the definition of time windows conveys the concepts of situativity theory while relying on temporal logic (Allen, 1983; Gabbay, Hodkinson, & Reynolds, 1994) to provide the basic foundation for a computational model.

2.2.1. Definition of Time Windows. Let us begin by defining a time window as a construct that specifies a functional relation between a required situation and a time interval that specifies availability for action. A time window does not specify what action must be taken, but only that there exists an action that will result in the required situation. In the course of operator activity within a dynamic task, n time windows are denoted as w_i for $i = 1$ to n.

At the onset of operator interaction, all time windows are designated as inactive and represented by the set U_0. Until a time window is designated as open, it remains inactive. Time windows are designated as open if the availability for action exists for a required situation at the current point in time space. The set of open time windows at time t is designated as O_t. When a required situation no longer exists, the corresponding time window is designated as closed. The set of closed time windows at time t is denoted as C_t. The membership of U, O, and C is defined to be persistent over time and will remain the same (i.e., $U_{t+1} = U_t$, $O_{t+1} = O_t$, and $C_{t+1} = C_t$) unless designated otherwise. Methods to extract conditions specifying the opening and closing of time windows are covered in Section 2.3.

2.2.2. Relation between Operator Actions and Time Windows. To complete the constraint specified by situativity theory in a temporal context, one must define operator action and the relation between action and time window. An operator action is defined here as a two-tuple that includes a detectable act performed by the operator at a specific point in time. In the course of operator interaction within a dynamic task environment, m actions are denoted as b_j for $j = 1$ to m. The relation between action and time window can be described by two Boolean indicator functions, I_w^l, such that, for l = 1, the function evaluates whether an action meets the required situation specified by a time window, and for l = 2, the function evaluates the relevance of an action toward a time window.

Thus,

$$I_w^1(\mathbf{b}) = \left\{ \begin{matrix} 1 \; if \; \mathbf{b} \; meets \; situation \; specified \; in \; w \\ 0 \; if \; \mathbf{b} \; does \; not \; meet \; situation \end{matrix} \right\} \text{ and}$$

$$I_w^2(\mathbf{b}) = \left\{ \begin{matrix} 1 \; if \; \mathbf{b} \; is \; relevant \; toward \; w \\ 0 \; if \; \mathbf{b} \; is \; not \; relevant \; toward \; w \end{matrix} \right\}$$

Six predicates, $M_T^k(w_i, \mathbf{b}_j)$ for $k = 1$ to 6, are now constructed to characterize fundamental relations between time windows and operators actions over a time interval T. In particular, the truth value, $||M^k(w_i, \mathbf{b}_j)||_{T+T-}$, of each predicate is evaluated for a time interval that starts when operator interaction in the task begins (T+) and ends when operator interaction ceases (T). Given that b_j occurs at time s, equations to evaluate the first five predicates are listed as follows:

- An on-time action that results in a required situation, $M_T^1(w_i, \mathbf{b}_j)$, is formally defined as

$$||M^1(w_i, \mathbf{b}_j)||_{T+,T-} = 1 \; iff \; \exists i \; such \; that \left[I_{w_i}^1(\mathbf{b}_j) = 1 \right] \wedge (w_i \in O_s) \tag{1}$$

- An early action that results in a required situation, $M_T^2(w_i, \mathbf{b}_j)$, is defined as

$$||M^2(w_\iota, \mathbf{b}_j)||_{T+,T-} = 1 \; iff \; \exists i \; such \; that \left[I_{w_i}^1(\mathbf{b}_j) = 1 \right] \wedge (w_i \in U_s) \tag{2}$$

- A late action that results in a required situation, $M_T^3(w_i, \mathbf{b}_j)$, is defined as

$$||M^3(w_\iota, \mathbf{b}_j)||_{T+,T-} = 1 \; iff \; \exists i \; such \; that \left[I_{w_i}^1(\mathbf{b}_j) = 1 \right] \wedge (w_i \in C_s) \tag{3}$$

- An action that is relevant toward a required situation, but does not result in it, $M_T^4(w_i, \mathbf{b}_j)$, is defined as

$$||M^4(w_i, \mathbf{b}_j)||_{T+,T-} = 1 \, iff \; \exists i \; such \; that \left[I^1_{wi}(\mathbf{b}_j) = 0\right] \wedge \left[I^2_{w_i}(\mathbf{b}_j) = 1\right] \tag{4}$$

- An action with no corresponding time window, $M^5_T(\mathbf{b}_j)$, is defined as

$$||M^5(\mathbf{b}_j)||_{T+,T-} = 1 \, iff \; \forall i, (I^2_{w_i}(\mathbf{b}_j) = 0) \tag{5}$$

Because the sixth predicate is based on a time window instead of action, the equation to evaluate it is defined separately as follows:

- A time window that has been missed, $M^6_T(w_i)$, is defined as

$$||M^6(w_i)||_{T+,T-} = 1 \, iff \; \forall j, (I^2_{w_i}(\mathbf{b}_j) = 0) \tag{6}$$

Because of their reliance on temporal logic, Equations 1 through 5 offer a more explicit description of constraints than the conceptual distinctions offered by situativity theory. Specifically, the time window framework can now be utilized as a dependency relation between an action and a required situation that is also bound by time.

2.3. Extracting Time Window Information

To extract time window information, one must view operator decision making in its experiential context and avoid normative descriptions of analytical decision mechanisms (e.g., expected utility theory). The focus of the extraction is, therefore, on the use of analysis methods to discover mappings between operator actions and situations required to meet system objectives.

Three techniques meet the criteria for extracting time window information. Because each technique focuses on a slightly different information source, the most effective approach is one that integrates the advantages of all three. One method, cognitive task analysis (e.g., Militello & Hutton, 1998), is based on human input. Cognitive task analysis focuses on experienced practitioners in operational contexts to extract information they deem diagnostic to successfully operate in the task environment. The two other methods rely on theoretical and empirical studies of the environment in which the task is performed. Cognitive work analysis utilizes theoretical expertise and engineering analyses of system dynamics to identify conceptual distinctions within a work domain that can later be used as modeling tools (Vicente, 1999). Ecological task analysis is focused on analysis of the task environment to determine empirical regularities in environmental behavior (Kirlik, 1995). Time window information extracted through the integrated method should therefore be (a) valid from an operator's perspective, (b) consistent with system dynamics, and (c) true to the availability of action within the task environment. Consider, for example, the process of extracting time window information in an air traffic control (ATC) domain. Cognitive task analysis is used to determine normal operator courses of actions to reach established objectives. Cognitive work analysis is used to ascertain static and kinematic constraints in the ATC domain that affect the operator's ability to reach the objectives (e.g., radar range). Ecological task analysis is used to discover constraints in the

ATC environment (e.g., appropriate regulations) and empirical regularities to which good controllers must be sensitive.

Once time window information has been extracted, the next step in the proposed research methodology is to implement the construct. The next section presents an object-oriented simulation architecture that includes a time window generation and maintenance system based on the blackboard model.

3. BLACKBOARD MODEL IN OBJECT-ORIENTED SIMULATIONS

The blackboard model was first developed in the early 1970s as a tool for speech understanding (Erman, Hayes-Roth, Lesser, & Reddy, 1980). Since then, it has been implemented in many domains for multiple purposes. For example, Vranes, Lucin, Stanojevic, Stevanovic, and Subasic (1991) used it as a tool to conduct military planning, and Rubin, Jones, and Mitchell (1988) used it as a framework to construct an operator’s associate in a supervisory control task. More recently, Adeli and Yu (1995) used it to develop an integrated computing environment to solve complex engineering problems. Although it has been implemented in vastly different forms, the blackboard model approach to problem solving remains the same. In essence, the blackboard model of problem solving is a reasoning scheme that applies pieces of knowledge at the most opportune time to construct a solution to the problem.

A blackboard model consists of three major components (Nii, 1986): knowledge sources, the blackboard data structure, and control. The knowledge sources contain knowledge required to solve the problem. The blackboard data structure is a global database in which partial and full solutions are kept. The blackboard control is an opportunistic reasoning model that guides problem solving by choosing and activating appropriate knowledge sources.

3.1. The Blackboard and Time Windows

To illustrate the use of blackboard model to open, maintain, and close time windows, consider the following example: In a real-time simulation, a human operator assumes the role of an air traffic controller monitoring aircraft entering and leaving Country X’s airspace (Figure 2). The operator has been given specific instructions to identify all unknown aircraft entering the airspace and to establish radio contact with all aircraft that come within radio range. An unknown aircraft, traveling along the trajectory indicated by the direction vector, enters Country X airspace at point A, enters and leaves range to establish radio contact at point B, and leaves Country X airspace at point C.

In the context of time windows, the blackboard knowledge sources include operators who act on the environment and entities that produce situations. These sources contribute not only actions and situations to the blackboard, but also temporal information that defines constraints within the environment in which the task is performed.

In the example, the knowledge sources include the air traffic controller and the unknown aircraft. Moreover, the unknown aircraft also reveals constraints that dictate expected air traffic controller actions. At point A, w_1 is designated as open so that $w_1 \in O_{t_a}$ with the specification that the situation of a correctly identified aircraft be re-

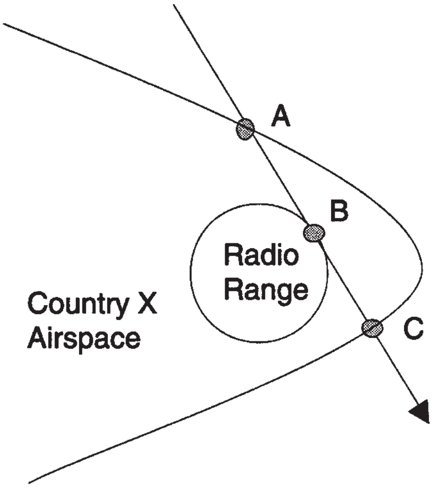

FIGURE 2 ATC example. An unknown aircraft enters Country X airspace at point A, enters, and leaves range to establish radio contact at point B, and leaves Country X airspace at point C.

quired. The time at which the aircraft reaches point A is designated as t_a. At point B, a second time window, w_2, is designated as open to specify the situation of established radio contact at time t_b so that $W_2 \in O_{t_b}$. Because the trajectory of the aircraft is tangential to the curve bounding the radio contact area, the available time interval for the air traffic controller to establish radio contact is instantaneous. Therefore, w_2 is also designated as closed at time t_b so that $W_2 \in C_{t_b}$. At point C, the aircraft exits Country X airspace and triggers the closing of w_1 so that $w_1 \in C_{t_c}$.

The blackboard data structure holds time window information in the form of computational and solution-state data. Each time window represents a structural means-ends hierarchy (Vicente, 1999) where the required situation (ends) is achieved by an expected operator action (means).

Although the knowledge sources provide necessary information to generate and maintain time windows within the blackboard architecture, the activities on the blackboard are monitored and controlled by the blackboard control. The control uses opportunistic reasoning to apply backward-reasoning as well as forward-reasoning models to maintain time window information. Backward reasoning is applied at the point of a required situation to determine if the expected operator action has been taken, whereas forward reasoning starts at an operator action to determine if the action outcome meets any required situations.

Returning to the ATC example, assume that the operator takes three actions. The first action, b_1, incorrectly identifies the aircraft at time t_1, where t_1 is before t_a (i.e., $t_1 < t_a$). The second action, b_2, correctly identifies the aircraft at time t_2 where $t_a < t_2 < t_c$. The third action, b_3, alerts Country X's border patrol at time t_3 where $t_b < t_3 < t_c$.

Using backward reasoning, the blackboard control examines all open time windows to determine if any have been met. At time t_a, the control assesses b_1 as applicable to w_1 so that $I^2_{w_1}(\mathbf{b}_1) = 1$, but does not result in the required situation so that $I^1_{w_1}(\mathbf{b}_1) = 0$. Thus, Equation 4 is satisfied and the action is deemed irrelevant. At time t_2, the control determines that b_2 is consistent with the expected operator action specified by w_1 so that

$I^{1}_{w_1}(\mathbf{b}_2) = 1$. Moreover, because $w_1 \in O_{t_2}$, the control evaluates w_1 and b_2 to satisfy Equation 1 and assesses b_2 an on-time, required action.

Applying forward reasoning, the control examines all current actions to determine if they address any required situations. At time t_3, the control determines that b_3 is not relevant toward any time window so that $\forall i, I^{2}_{w_1}(\mathbf{b}_3) = 0$. The control does not, however, make a determination on the action at this point. Rather, it seeks resolution of the action's status by checking backward-reasoning results to ensure that the action is not early for a later time window. Nevertheless, the third action was eventually determined to be irrelevant.

3.2. Blackboard Models in a Real-Time, Object-Oriented Simulation

Conceptually, the use of time windows in a blackboard model has been demonstrated. To illustrate the utility of time windows in a simulation environment, the implementation of time windows via a blackboard model is now presented. The simulation architecture developed at the Georgia Institute of Technology (Chu, Jones, & Mitchell, 1991; Jones, Chu, & Mitchell, 1995) is used as a baseline for discussion. The integration of the blackboard model within the simulation architecture is depicted in Figure 3.

The active simulation object (ASO) is used as a base class so that events can be scheduled by methods contained in its subclasses. The display class contains parameters as well as methods for generating the graphical user interface. The simulator class contains methods to control the experimental simulation. The platform class represents physical platforms (e.g., airplanes) that exist in the simulation environment and contains methods that allow those objects to act on the environment. The blackboard class contains the knowledge sources within the blackboard data structures. It also contains methods to control the blackboard by opening time windows, closing time windows, updating and reconciling time windows, conduct forward-chaining reasoning, or execute backward-chaining reasoning.

An illustration of the blackboard and time window implementation within an object-oriented simulation framework is represented in the form of a sequence diagram in Fig-

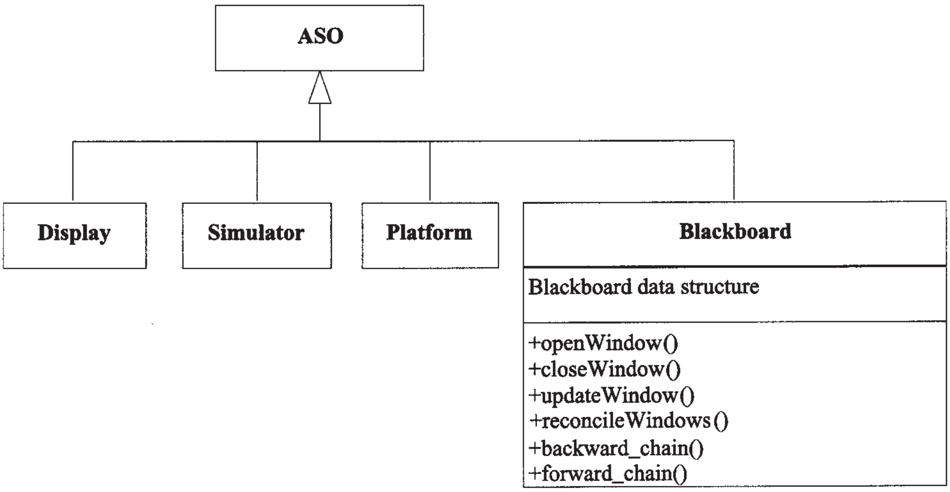

FIGURE 3 Simulation class diagram.

ure 4. A sequence diagram is a model that describes how groups of objects collaborate in some behavior (Booch, Rumbaugh, & Jacobson, 1999; Fowler & Kendall, 2000). Each box above the diagram represents an object. Each vertical line represents the object's life during the interaction. The flow of events is chronologically ordered from top to bottom. Methods labeled with an asterisk are iterative.

Revisiting the air traffic control example, the event flow of operator actions and aircraft movements is reflected in Figure 4. A chronologically-ordered narration on the sequence of events follows:

1. The flight of the unknown aircraft along the southeasterly trajectory is accomplished by the iterative call of the modify Position() method.
2. The first operator action, $\boldsymbol{b}_1$, of incorrectly identifying the aircraft (as a jet) is posted to the blackboard.
3. When the control detects the aircraft entering the airspace of Country X, w_1 is designated as open.
4. The backward-chaining model reasons that b_1 is an incorrect identification that has been taken early. Thus, $||M^4(w_1, \mathbf{b}_1)||_{T+,T-} = 1$.
5. The second operator action, b_2, of correctly identifying the aircraft (as a propeller-driven aircraft) is posted to the blackboard.
6. The backward-chaining model determines that a correct identification action has been taken on time. Therefore, $||> M^1(w_1, \mathbf{b}_2)||_{T+,T-} = 1$.
7. When the control detects the aircraft entering radio range, w_2 is designated as open.
8. The control immediately detects the aircraft leaving radio range and closes w_2.
9. The third operator action to alert the border patrol, b_3, is posted to the blackboard. The forward-chaining model determines that no time window specifies the need for b_3.

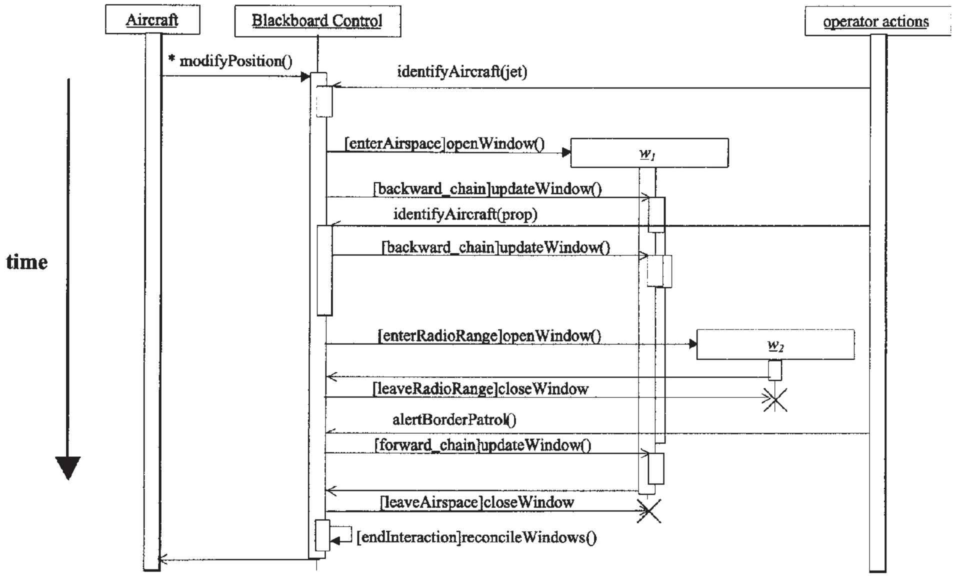

FIGURE 4 Time window sequence diagram.

Moreover, the action does not serve any required situation—radio contact or correctly identified aircraft. Therefore, the action is classified as irrelevant. Thus, $||M^5(\mathbf{b}_3)||_{T+,T-} = 1$.

10. When the control detects the aircraft leaving Country X airspace, w_1 is closed.
11. Operator interaction ceases as the aircraft leaves Country X airspace. At this point, the backward-chaining model reconciles the blackboard by closing all open windows and assessing if windows have been missed. The only window in question is w_2 and is assessed to be missed so that $||M^6(w_2)||_{T+,T-} = 1$.

3.3. Possible Time Window Outcomes

The utility of a time window is not only in its temporal and functional descriptions but also in the richness of the possible outcomes. Some time window outcomes have already been described. Not surprisingly, the complete space of possible time window outcomes (Figure 5) is represented by the fundamental relations between time windows and operator actions outlined in Equations 1 through 6. In itself, the existence of a required situation does not impact system performance. It is the presence of operator action in a temporal context that specifies whether performance is good or poor. An incorrect, early action (first ATC operator action) is represented as Equation 4. An on-time, accurate action (second ATC operator action) is represented as Equation 1. An action with no corresponding required situation (third ATC operator action) is categorized as Equation 5. A nonaction for an existing situation requirement (no attempt to establish radio contact) is characterized as a miss and is represented as Equation 6.

It has been shown that time window is a viable construct, both conceptually as well as in an implemented mechanism within a simulation framework. However, the value of imple-

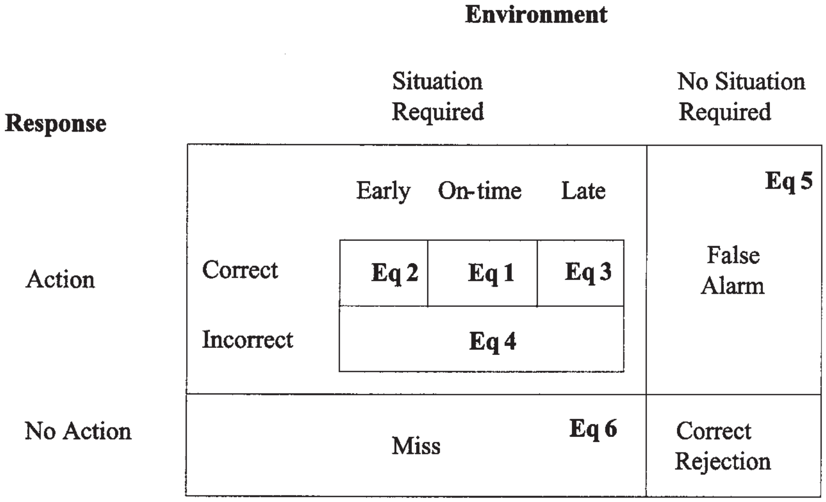

FIGURE 5 Possible time window outcomes. The environment is delineated in terms of situation required (time window exists) or no situation is required (time window does not exist). Equations 1 through 4 represent actions that are relevant to a time window. Equations 1 through 3 represent actions that result in the required situation (correct actions). Equation 4 represents actions that do not meet the required situation (incorrect actions) even though they are relevant.

menting time windows in a research effort has yet to be discussed. The following section discusses the implications of applying time windows toward human performance measurement and evaluation.

4. TIME WINDOWS AND HUMAN PERFORMANCE

4.1. Implications Toward Measurement

Wickens and Holland (2000) observed that most performance measures are associated with one of the following categories of raw data:

1. Measure of speed or time (e.g., how fast can an operator reach for a lever?).
2. Measure of accuracy or error (e.g., how many typing mistakes are made?).
3. Measure of workload or capacity demands (e.g., how difficult is this task?).
4. Measure of preference (e.g., is one display preferred over another?).

In most cases, the use of a particular type of measure is dependent on the real-world task to which the results of the laboratory task generalize. The emphasis, therefore, is on finding methods that analyze factors in isolation. However, it has already been noted that research on dynamic and complex environments should take place in representative settings. Recognizing the problem, researchers have sought to develop techniques to measure performance in tasks that are more representative of the operational environment. Sanderson, Verhage, and Fuld (1989) focused on the use of verbal protocol data in operational tasks. Howie and Vicente (1998) used automated log files to construct a number of measures to assess operator performance in a microworld setting. Still other researchers (Laudeman & Palmer, 1995; Moray, Dessouky, Kijowski, & Adapathya, 1991; Raby & Wickens, 1994) focused on recorded data in time-critical task environment.

Even with a focus on the environment, however, the majority of measurement efforts remain guided strictly by operator actions and do not adequately consider the environment in which the task is situated. One notable exception is the research of Laudeman and Palmer (1995). Laudeman and Palmer derived a window of opportunity measure to evaluate operator response to different task demands. Using their measure, they were able to generate workload profiles for aircraft crews based on their ability to complete a specified set of tasks.

The time window construct represents a fundamental shift from existing performance measurement approaches. It is not focused solely on whether a certain task is completed, or how fast a certain button is pushed, or what percentage of error is detected. Rather, it provides a computational framework to dynamically evaluate heterogeneous situation demands and operator abilities to meet them in a complex domain. The benefit of the framework is the functional link between operator actions and the domain with which he or she interacts.

4.2. Implications Toward Evaluation

As shown in Figure 5, utilization of the time window construct leads to a multidimensional space of possible outcomes. As yet, no mathematical formalism exists to comprehensively evaluate operator performance based on all dimensions. Instead, two methods are proposed to provide different perspectives on operator attunement to the constraints. The first method, factor analysis, is designed to determine correlations among different types of time windows and time window outcomes. The second method depends on the use of signal detection theory (SDT) to determine the sensitivity of operator actions to situation requirements.

4.2.1. Use of factor analysis. Factor analysis is a data reduction technique that attempts to find a smaller number of dimensions, or factors, while retaining most of the information in the original space (Green, 1978).The intent, therefore, is to evaluate which situations and operator actions can be aggregated into higher order factors. The analysis process proceeds in three major steps:

1. Rotate original data (i.e., variables consisting of the different time window outcomes in different types of required situations) to a new orientation that exhibits dimensions with maximal variance.
2. Reduce the dimension of the transformed data space.
3. Identify the new dimensions, or factors, in terms of variables that show high association with each factor.

The reader is referred to any multivariate statistics text for details on Steps 1 and 2. To identify underlying factors, a technique called the scree test (Cattell, 1966) is suggested. In essence, the scree test requires plotting the variance accounted by each factor extracted and then finding elbow in the curve of the plot. To identify which variables belong to the selected factors, factor loadings (i.e., correlation between the variable with a factor) are recommended.

4.2.2. Use of signal detection theory. Signal detection theory is a formulation that has been widely used to assess human ability to detect signals (Green & Swets, 1966; Swets, 1996). The premise of the paradigm is that there are two states of the world (signal vs. noise) and two possible human responses (I detect a signal vs. I do not detect a signal). The possible resulting states produces a 2 × 2 stimulus–response matrix shown in Figure 6.

A key theoretical representation of signal detection theory is the receiver operating characteristic (ROC; Swets, 1996). The standard graphical depiction of the ROC is known as the ROC curve (Figure 7). The curve reveals two important sources of information about operator performance: an individual’s decision criterion (the amount of evidence required to detect a signal) and the sensitivity of an individual’s detection performance (the individual’s ability to discriminate between signal and noise).

To apply SDT to the sensitivity analysis of time window outcomes, one must develop methods that do not violate assumptions of either formulation. In particular, the following three issues must be addressed: conversion of time window outcomes to SDT outcomes, calculating the probability of a false alarm in time window outcomes, and the development of a sensitivity measure without distribution assumptions.

The conversion of time window outcomes (Figure 5) to SDT outcomes is dependent on a common definition of a hit. If a hit is defined to be an on-time and accurate action, so that Equation 1 holds, then conversions from time window outcomes to SDT outcomes can readily be made. Table 1 shows the conversion from time window outcomes to SDT outcomes. If an action is not executed on time, it is considered a false alarm. Therefore, a signal

	State of the world	
Response	Signal	Noise
Detected	Hit	False Alarm
Not Detected	Miss	Correct Rejection

FIGURE 6 Signal detection theory outcomes.

is only considered valid and detectable during a specified time interval in which the associated time window is designated open.

The original SDT formulation required forced-choice tasks primarily to ensure that correct rejections were accurate assessments of the absence of a signal. However, the decision environments for which time windows are intended as dynamic and interactive, and operators are not forced to take action. To calculate the probability of false alarm, which requires the number of false alarms and correct rejections, an accurate accounting method for correct rejections is needed. In fact, one method to measure correct rejections in these "free response" (Wickens & Kessel, 1979) environments has already been developed. Wickens and Kessel computed the probability of false alarms as the number of false alarms divided by the number of false-alarm intervals. In their formulation, equal-valued intervals that span the detection task are separated into those that contain hits and those that do not—called false-alarm intervals. Based on this concept, a false-alarm interval can be defined in the time

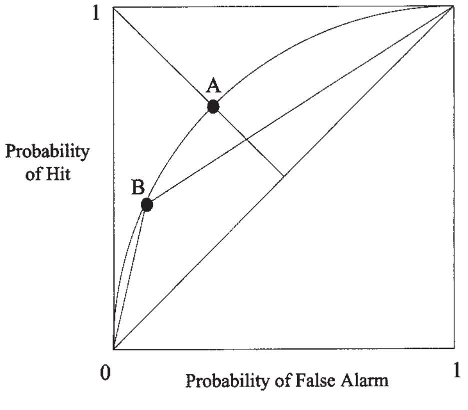

FIGURE 7 The ROC curve under different distribution assumptions. If the distributions of signal and noise are normal, the sensitivity, *d'*, is determined by the distance of a point on the curve, point A, from the upper-left diagonal. If no assumptions on the distributions can be made, the sensitivity can be approximated by the area under the ROC (e.g., point B).

TABLE 1
Conversion Between Time Window Outcomes and
Signal Detection Theory (SDT) Outcomes

Time Window Outcome	*SDT Outcome*
$\| M^1 (w_i, \mathbf{b}_j) \|_{T+,T-=} 1$	Hit
$\| M^2 (w_i, \mathbf{b}_j) \|_{T+,T-} = 1$	False alarm
$\| M^3 (w_i, \mathbf{b}_j) \|_{T+,T-} 1$	
$\| M^4 (w_i, \mathbf{b}_j) \|_{T+,T-} = 1$	
$\| M^5 (\mathbf{b}_j) \|_{T+,T-} = 1$	
$\| M^6 (w_i) \|_{T+,T-} = 1$	Miss
Correct rejection	Correct rejection

window context. Consider the duration of a time window, T, over the lifetime of a simulation, Ts. The number of false-alarm intervals can simply be formulated as

$$FAI = \frac{T_s}{T} - 1 \tag{7}$$

The third issue to be addressed is the need for an appropriate sensitivity measure. If the distributions of the signal and noise are normal, the determination of the sensitivity, *d'*, can be visually determined from the ROC curve. In Figure 7, for instance, the closer point A is from the upper-left corner, the higher the sensitivity value. However, no assumptions can be readily made about distributions of signal and noise in dynamic domains. Therefore, one must rely on nonparametric measures of sensitivity. Wickens and Hollands (2000) recommended a simple measure based on area under a ROC. The measure, first considered by Green and Swets (1966), is formulated as follows:

$$A_G = \frac{P(H) + [1 - P(FA)]}{2} \tag{8}$$

If only one point is acquired on the ROC, such as Point B in Figure 7, a sensitivity value can now be calculated. Although these measures are still dependent on distributional assumptions (Caldeira, 1980), they nevertheless serve as a good first approximation (Craig, 1979).

The research methodology proposed here was implemented in a study to investigate tactical decision-making performance under stress. The study is now presented to demonstrate the viability of the methodology and the utility of the time window construct. Only salient aspects of the study are discussed to highlight the research approach. For experiment details, see Hodge et al. (1995).

5. EMPIRICAL STUDY

5.1. Method

The participants were 72 male undergraduate university students. The basic experimental equipment per participant included two personal computers. One of the computers hosted a discrete-event, human-in-the-loop simulation called the Georgia Tech Aegis Simulation Platform (GT–ASP). The other computer hosted a sound player that provided auditory stimulus consistent with the simulation. Six training conditions were used that varied the form of post-scenario feedback and online decision aiding. Each participant ran 18 thirty-min scenarios. The 15 earlier scenarios were designed to train the participants to a criterion level on the assigned training condition. The final three scenarios were used to evaluate transfer of learning to a nonaided version of the simulation.

Participants were briefed that they were assuming the role of an air defense officer on board a naval warship. They were told to follow nine rules in a military context and were given one general order to protect their own ship. Each of the nine rules is represented as constraints in Table 2.

The GT–ASP simulation was constructed in a manner consistent with the architecture reflected in Figure 3. As participants ran the simulation, the blackboard generated, maintained, and completed all time windows associated with the nine rules. All time window information was logged for further analysis.

5.2. Analysis

To highlight the use of factor analysis, participant performance data from just the first transfer scenario was used. The variables examined included the following:

1. (R1ERR–R9ERR) Number of errors committed while acting on time windows associated with GT–ASP Rules 1 through 9 (compiled using Equations 4, 5, and 6).
2. (EARLY) Number of early actions (compiled using Equation 2).
3. (ONTIME) Number of on-time actions (compiled using Equation 1).
4. (LATE) Number of late actions (compiled using Equation 3).
5. (FA) Number of false alarms (compiled using Equation 4).
6. (MISS) Number of required situations missed (compiled using Equation 6).
7. (TOTERR) Number of total errors committed.
8. (MTACT) Mean time required for operator to act after a window has been opened, regardless of GT–ASP rule.

TABLE 2
GT–ASP Rules in Constraint Format

Expected Action	*Required Situation*	*Condition of Availability*
Engage hostile track	Hostile track engaged	Hostile track within 20 nm of own ship
Illuminate hostile track	Hostile track targeted	Hostile track within 30 nm of own ship
Issue level 1 warning	Hostile track warned (level 1)	Hostile track within 50 nm of own ship
Issue level 2 warning	Hostile track warned (level 2)	Hostile track within 50 nm of own ship
Issue level 3 warning	Hostile track warned (level 3)	Hostile track within 30 nm of own ship
Recall own aircraft	Recall own aircraft	Own aircraft further than 256 nm
Dispatch own aircraft	Own aircraft leaving	Own aircraft closer than 20 nm
Assign primary ID to unknown aircraft	Correctly identified aircraft	Unknown aircraft is detectable
Assign designation to unknown aircraft	Correctly designated aircraft	Undesignated aircraft is detectable

9. (R1MT–R9MT) Mean time required for operator to act after a time window corresponding to each GT–ASP rule has been opened.

Figure 8 shows the scree plot for the amount of variance accounted by each extracted factor. Further investigation of the factor loadings (Table 3) suggested that six of the variables were highly correlated with the first extracted factor.

The results of the factor analysis indicate that identification tasks (assign primary ID and assign designation) are highly correlated with the total number of errors as well as the mean reaction time after windows opened (i.e., latency). Therefore, the findings suggest that correct identification of unknown aircraft is key to overall good performance.

To demonstrate the use of SDT to analyze participant performance in GT–ASP, the first transfer scenario was again used. All time window outcomes were first converted to SDT outcomes. The number of false-alarm intervals for each time window was then calculated using Equation 7. The aggregate number of false-alarm intervals per participant was, therefore, the summation of intervals for all time windows in the scenario. The measure, A_G, was finally calculated to determine the sensitivity of each participant's actions to required situations in GT–ASP. Figures 9 through 12 show the bivariate correlations between A_G and each of the four possible stimulus–response outcomes.

The bivariate plots suggest a high correlation between A_G and the number of hits (r = .789) as well as a negative correlation between A_G and the number of false alarms (r = –.921). To better evaluate the significance of each stimulus–response outcome as a predictor of sensitivity, a multiple regression model was used. The form of the regression model was as follows:

$$A'_G = a + b_1\,(misses) + b_2\,(false\,alarms) + b_3\,(false\,alarm\,intervals) + b_4\,(hits)$$

Results of the regression analysis are shown in Table 4.

It is seen from Table 4 that misses, false alarms, and hits were significant predictor variables for participant sensitivity in GT–ASP. It is perhaps equally significant that the variable that is least diagnostic (i.e., the number of times that nothing was done when no

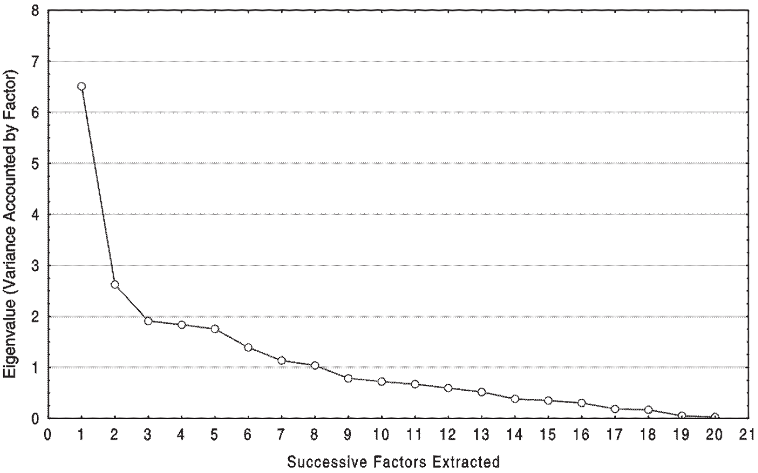

FIGURE 8 Scree plot for successive factors extracted.

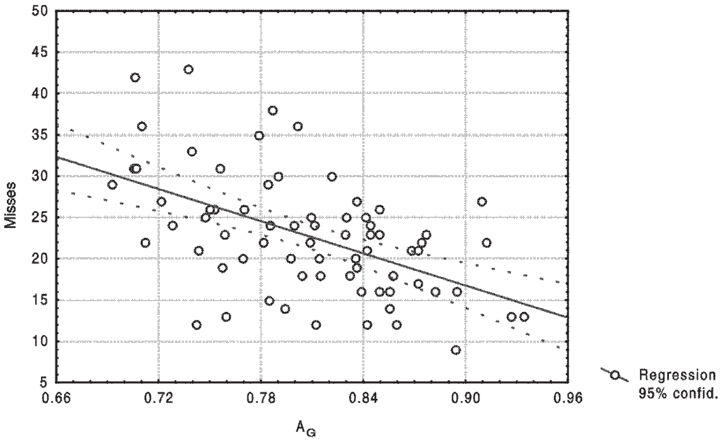

FIGURE 9 Plot of A_G versus misses. Correlations: $r = -.5245$.

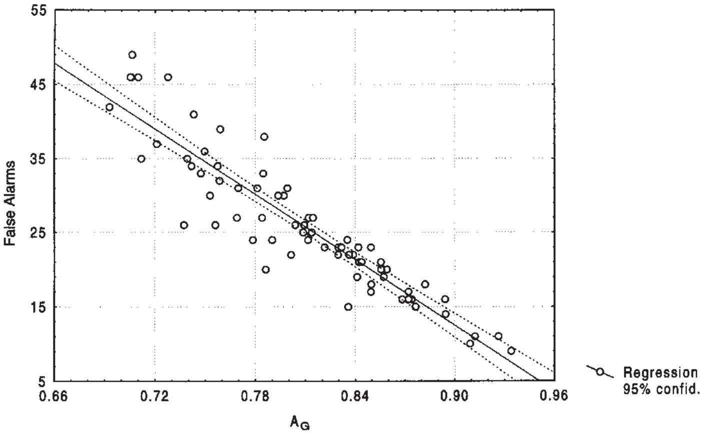

FIGURE 10 Plot of A_G versus false alarms. Correlation: $r = -.9211$.

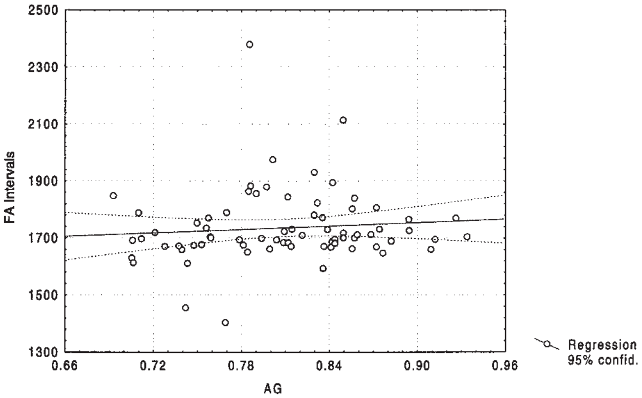

FIGURE 11 Plot of A_G versus false alarm intervals. Correlation: $r = .09225$.

TABLE 3
Significant Factor Loadings

	Factor 1	*Factor 2*
R8ERR	.777529	*ns*
R9ERR	.838927	*ns*
TOTERR	.953719	*ns*
MTACT	.759150	*ns*
R8MT	.743501	*ns*
R9MT	.719211	*ns*
Explained variance	6.508721	2.626776
Proportion of total	.282988	.114208

Note. Significant loadings are > .700000. *ns* = not significant.

situation was required) was not a significant predictor of sensitivity. Therefore, A_G stands as a viable indicator of the ability of an operator to correctly meet required situations.

6. SUMMARY AND CONCLUSIONS

A research approach to evaluate operator performance in highly automated and complex systems has been proposed. The key concept within the approach is a notion of time windows. The time window construct provides a computational framework to dynamically evaluate operator actions in the context of heterogeneous task demands. The primary contribution of this research is the provision of a functional link between operator actions and the domain with which he or she interacts.

To implement time windows in a working model, a blackboard paradigm was introduced. The blackboard model is suited to accommodate the time window construct because of its ability to reason opportunistically about the availability of situations and the timeli-

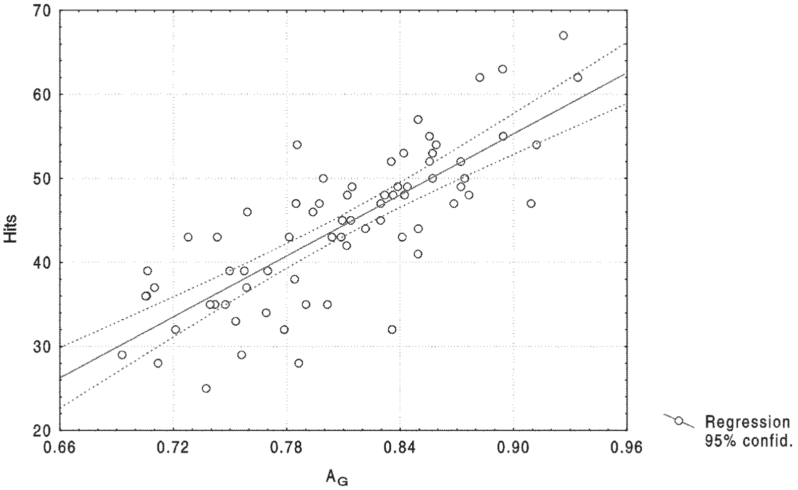

FIGURE 12 Plot of A_G versus hits. Correlation: r = .78881.

TABLE 4
Summary of Regression Analysis for Variables Predicting A_G[a]

Variable	*B*	*SE B*	2
Misses	0.000	0.0002	0.049*
False alarms	–0.004	0.0001	–0.699*
False alarm intervals	0.000	0.0000	–0.035
Hits	0.003	0.0001	0.469*

Note. $R2 = 0.998$ ($p < .05$).
[a] $N = 72$.
*$p < 0.05$.

ness of operator actions. It was argued that human-in-the-loop simulations are ideal tools to investigate dynamic phenomena without concerns of the oversimplified laboratory environment or the unconstrained real world. Therefore, requirements for implementation of the blackboard model were discussed. Moreover, a study that implemented the blackboard model in a human-in-the-loop simulation was used to illustrate the viability of the time window construct to provide a framework for operator performance. Two methods for analysis of time window outcomes were discussed to provide complementary perspectives on operator attunement to the constraints.

There are two major limitations in the current formulation of time windows. First, it does not explicitly account for cognitive activity. Although it is assumed that cognitive activity is necessary to effect action, an action is specifically defined as a detectable act at a defined point in time. Second, the navy domain in which the time window construct was used contains highly defined rules of operation. In less defined environments, however, the construction of time windows will be highly contingent on the reliability of the domain and task analysis results.

Although some evidence of the viability of time windows has been shown here, more work remains. Based on the broad range of topics covered in the proposed research approach, it is hoped that the reader gains an appreciation of the interdisciplinary nature of the research. In fact, considerable efforts are being made across multiple disciplines to develop a general theoretical account of activity in terms of interactions of agents with systems in their environments (Greeno, 1994).

REFERENCES

Adeli, H., & Yu, G. (1995). An integrated computing environment for solution of complex engineering problems using the object-oriented programming paradigm and a blackboard architecture. *Computers and Structures, 54,* 255–265.

Allen, J. F. (1983). Maintaining knowledge about temporal Intervals. *Communication of the ACM, 26*(11), 832–843.

Beach, L. R., & Lipshitz, R. (1993). Why classical decision theory is an inappropriate standard for evaluating and aiding most human decision making. In G. A. Klein, J. Orasanu, R. Calderwood, & C. Zsambok (Eds.), *Decision making in action: Models and methods* (pp. 21–35). Norwood, NJ: Ablex.

Booch, G., Rumbaugh, J., & Jacobson, I. (1999). *The unified modeling language user guide.* Reading, MA: Addison-Wesley.

Brehmer, B., Leplat, J., & Rasmussen, J. (1991). Use of simulation in the study of complex decision making. In J. Rasmussen, B. Brehmer, & J. Leplat (Eds.), *Distributed decision making: Cognitive models for cooperative work.* New York: Wiley.

Caldeira, J. D. (1980). Parametric assumptions of some “nonparametric” measures of sensory efficiency. *Human Factors, 22,* 119–120.

Cattell, R. B. (1966). *Handbook of multivariate experimental psychology*. Chicago: Rand McNally.

Chu, R. W., Jones, P. M., & Mitchell, C. M. (1991). The Georgia Tech Payload Operations Control Center simulation: Design and implementation in C++. *Proceedings of the Object-Oriented Simulation Track of the 1991 Society for Computer Simulation Multiconference.* Anaheim, CA.

Craig, A. (1979). Nonparametric measures of sensory efficiency for sustained monitoring tasks. *Human Factors, 21,* 69–78.

Erman, D. L., Hayes-Roth, F., Lesser, V. R., & Reddy, D. (1980). The HEARSAY–II speech understanding system: Integrating knowledge to resolve uncertainty. *ACM Computing Survey, 12,* 213–253.

Fowler, M., & Kendall, S. (2000). *UML distilled: A brief guide to the Standard Object Modeling Language* (2nd ed.). Reading, MA: Addison-Wesley.

Gabbay, D. M., Hodkinson, I., & Reynolds, M. (1994). *Temporal logic: Mathematical foundations and computational aspects.* Oxford, England: Oxford University Press.

Green, D. M., & Swets, J. A. (1966). *Signal detection theory and psychophysics.* New York: Wiley.

Green, P. E. (1978). *Analyzing multivariate data.* Hinsdale, IL: Dryden.

Greeno, J. G. (1998). The situativity of knowing, learning, and research. *American Psychologist, 53,* 5–26.

Greeno, J. G. (1994). Gibson’s affordances. *Psychological Review, 101,* 336–342.

Greeno, J. G., & Moore, J. L. (1993). Situativity and symbols: Response to Vera and Simon. *Cognitive Science, 17,* 49–60.

Hammond, K. R. (1986). Generalization in operational contexts: What does it mean? Can it be done? *IEEE Transaction on Systems, Man, and Cybernetics, 16,* 428–433.

Hodge, K. A., Rothrock, L., Kirlik, A. C., Walker, N., Fisk, A. D., Phipps, D. A., & Gay, P. E. (1995). *Training for tactical decision making under stress: Towards automatization of component skills* (Human Attention Performance Laboratory Technical Report No. 9501). Georgia Institute of Technology, School of Psychology.

Howie, D. E., & Vicente, K. J. (1998). Measures of operator performance in complex, dynamic microworlds: Advancing the state of the art. *Ergonomics, 41,* 485–500.

Jones, P. M., Chu, R. W., & Mitchell, C. M. (1995). A methodology for human–machine systems research: Knowledge engineering, modeling, and simulation. *IEEE Transactions on Systems, Man, and Cybernetics, 25,* 1025–1038.

Kirlik, A. (1995). Requirements for psychological models to support design: Toward ecological task analysis. In J. M. Flach, P. Hancock, J. Caird, & K. J. Vicente (Eds.), *Global perspectives on the ecology of human–machine systems* (pp. 68–120). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Laudeman, I. V., & Palmer, E. A. (1995). The quantitative measurement of observed workload in the analysis of aircrew performance. *The International Journal of Aviation Psychology, 5,* 187–197.

Militello, L. G., & Hutton, R. J. B. (1998). Applied cognitive task analysis (ACTA): A practitioner’s toolkit for understanding task demands. *Ergonomics, 41,* 1618–1641.

Moray, N., Dessouky, M. I., Kijowski, B. A., & Adapathya, R. (1991). Strategic behavior, workload, and performance in task scheduling. *Human Factors, 33,* 607–629.

Nii, H. P. (1986). Blackboard systems: The blackboard model of problem solving and the evolution of blackboard architectures. Part I. *AI Magazine, 7,* 38–53.

Raby, M., & Wickens, C. D. (1994). Strategic workload management and decision biases in aviation. *The International Journal of Aviation Psychology, 4,* 211–240.

Rubin, K. S., Jones, P. M., & Mitchell, C. M. (1988). OFMspert: Inference of operator intentions in supervisory control using a blackboard architecture. *IEEE Transactions on Systems, Man, and Cybernetics, 18,* 618–637.

Sanderson, P. M., Verhage, A. G., & Fuld, R. B. (1989). State-space and verbal protocol methods for studying the human operator in process control, *Ergonomics, 32,* 1343–1372.

Suchman, L. A. (1987). *Plans and situated actions: The problem of human machine communication.* Cambridge, England: Cambridge University Press.

Swets, J. A. (1996). *Signal detection theory and ROC analysis in psychology and diagnostics: Collected papers.* Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Vicente, K. (1997). Heeding the legacy of Meister, Brunswik, & Gibson: Toward a broader view of human factors research. *Human Factors, 39,* 323–328.

Vicente, K. J. (1999). *Cognitive work analysis: Toward safe, productive, and healthy computer-based work.* Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Vicente, K. J., & Kirlik, A. (1992). On putting the cart before the horse: Taking perception seriously in unified theories of cognition. *Behavioral & Brain Sciences, 15,* 461–462.
Vranes, S., Lucin, M., Stanojevic, M., Stevanovic, V., & Subasic, P. (1991). Blackboard metaphor in tactical decision making. *European Journal of Operational Research, 61,* 86–97.
Wickens, C. D., & Hollands, J. G. (2000). *Engineering psychology and human performance* (3rd ed.). Upper Saddle River, NJ: Prentice Hall.
Wickens, C. D., & Kessel, C. (1979). The effects of participatory mode and task workload on the detection of dynamic system failures. *IEEE Transactions on Systems, Man, and Cybernetics, 9,* 24–34.