In the Proceedings of the 1995 International Symposium on Command and Control Research and Technology, National Defense University, Washington D.C., June, 1995.

Predicting Human Performance in Dual-Task Tracking and Decision Making with Computational Models using the EPIC Architecture

David E. Kieras
Artificial Intelligence Laboratory
Electrical Engineering & Computer Science Department
University of Michigan
1101 Beal Avenue
Ann Arbor, Michigan 48109-2110
(313) 763-6739
kieras@eecs.umich.edu

David E. Meyer
Department of Psychology
University of Michigan
525 East University
Ann Arbor, Michigan 48109-1109
(313) 763-1477
demeyer@umich.edu

Abstract

EPIC (Executive Process-Interactive Control) is a human information-processing architecture especially suited for modeling human multiple-task performance. The EPIC architecture includes peripheral sensory-motor processors surrounding a production-rule cognitive processor, and is being used to construct precise computational models for basic multiple-task situations. Some of these models are briefly illustrated here to demonstrate how EPIC applies to multiple-task situations and clarifies some basic properties of human performance.

1 Introduction

This paper is a brief report on the current status of our work with the EPIC architecture for human information processing, which is being developed under ONR sponsorship.[1] EPIC is a general framework, represented as a simulation modeling environment, in which models of human performance in specific tasks may be constructed. The goal of the EPIC project is to develop a comprehensive computational theory of multiple-task performance that (a) is based on current theory and results in cognitive psychology and human performance; (b) will support rigorous characterization and quantitative prediction of mental workload and performance, especially in multiple-task situations; and (c) is useful in the practical design of systems, training, and personnel selection.

EPIC is similar in some ways to previous approaches to human performance modeling such as HOS (Lane, Strieb, Glenn, & Wherry, 1981), SAINT (Chubb, 1981) and others (e.g. see MacMillan, Beevis, Salas, Strub, Sutton, and Van Breda, 1989, for a survey). It extends previous theoretical proposals as well, such as those by Card, Moran, and Newell (1983), Schneider and Detweiler (1987), and Norman and Shallice (1986). There are several key differences with previous approaches. First, our models are based on a cognitive architecture, a structure of processing mechanisms whose properties adhere to recent and detailed empirical evidence, especially concerning multiple-task performance, and modern computational theories of cognition. Second, we have adopted a rigorous theoretical approach consisting of constructing and testing computational models and subjecting them to detailed quantitative comparison with data. Constructing such models involves a detailed analysis of the experimental task, which is usually overlooked in conventional psychological theorizing, but is a key advantage of the computational model approach (see Kieras, 1990). Third, our computational models are generative; that is, in an EPIC model, a simulated human with general procedural knowledge of the task interacts with a simulated task environment, and the model then generates the specific pattern of activities necessary to perform specific tasks. Thus the task analysis reflected in the model is general to a class of tasks. In summary, our EPIC project is both more theoretically advanced and more empirically accurate than most prior modeling approaches.

Our primary focus is on multiple-task performance, in which the human concurrently performs a set of tasks; the tasks are independent, in that each could be meaningfully described and conducted in isolation. A good example of a multiple-task situation is an airplane cockpit; for example, a pilot may need to simultaneously pilot the aircraft and track an enemy target. In a multiple-task situation, the main problem confronting the human is to execute the independent tasks in a coordinated fashion that meets some constraints on overall performance, such as giving one task priority over the other. We have focused on multiple-task performance for two reasons: First, it is of great practical importance, but is theoretically underdeveloped. Second,

[1] This work was supported by the Office of Naval Research Cognitive Sciences Program under grant N00014-92-J-1173 to the authors.

the multiple-task situation stresses human capabilities very seriously, and so the observed patterns of behavior set very strong constraints on the human information-processing system architecture. Thus our analyses of even simple multiple-task situations have resulted in detailed hypotheses about the information-processing mechanisms that are represented in the EPIC architecture.

We have applied certain principles for computational modeling of human multiple-task performance:

- Our computational models are built in terms of a detailed general architecture that characterizes human perceptual, cognitive, and motor mechanisms, and which is required to be accurate and applicable across task domains.
- A central role is given to cognitive strategies for task execution, which we represent using production systems.
- Executive processes for coordinating multiple tasks are treated simply as additional strategies.
- EPIC does not assume an inherent central-processing bottleneck. We attempt to explain performance decrements in multiple task situations in terms of the strategic effects of the task instructions and perceptual-motor constraints.

This paper provides a brief description of the EPIC architecture, our approach to modeling multiple task situations, a summary of how we have applied EPIC in previous work, and a description of the results of applying EPIC to some dual-task situations involving simultaneous tracking and decision-making tasks.

2 The EPIC Architecture

EPIC was designed to explicitly couple perceptual-motor mechanisms with a cognitive analysis of procedural skill represented by production-system models such as CCT (Bovair, Kieras, & Polson, 1990), ACT (Anderson, 1976), and SOAR (Laird, Rosenbloom, & Newell, 1986). Thus, EPIC has a production-rule cognitive processor surrounded by perceptual-motor peripherals; applying EPIC to a task situation requires specifying both the production-rule programming for the cognitive processor, and also the relevant perceptual and motor processing parameters. EPIC computational models are generative in that the production rules supply general procedural knowledge of the task, and thus when EPIC interacts with a simulated task environment, the model generates the sequence of serial and parallel activities required to perform a particular task. The model is driven by the sequence of task events external to the human operator, such as which characters appear at what location over time on a display screen at what time, possibly in response to actions performed by the operator.

Figure 1 shows the overall structure of processors and memories in the EPIC architecture. At this level, EPIC is conventional in some respects. However, there are many important new concepts in the EPIC architecture which this brief presentation will highlight.

As shown in Figure 1, EPIC has a conventional flow of information from sense organs, through perceptual processors, to a cognitive processor (consisting of a production rule interpreter and a working memory), and finally to motor processors that control effector organs. There are separate perceptual processors with distinct processing time characteristics, and separate motor processors for vocal, manual, and oculomotor (eye) movements. Also included are feedback pathways from the motor processors, as well as tactile feedback from the effectors, which are important in coordinating multiple tasks. The declarative/procedural knowledge distinction of the "ACT-class" cognitive architectures (see Anderson, 1976) is represented in the form of separate permanent memories for production rules and declarative information. At this time, we do not completely specify the properties of working memory (WM), because clarifying what types of working memory systems are used in multiple-task performance is one of our research goals. For now, WM is assumed to contain all of the temporary information tested and manipulated by the cognitive processor's production rules, including task goals, sequencing information, and representations of sensory inputs.

A single stimulus input to a perceptual processor can produce multiple outputs to be deposited in WM at different times. The first output is a representation that a perceptual event has been detected, followed later by a representation that describes the recognized event. For present purposes, we assume that the mean detection time is fixed and fairly short (e.g. 50 ms), while the recognition process takes additional time after the detection process, and depends on the properties of the stimulus. For example, recognizing letters on screen in a typical experiment might take on the order of 150 ms after the detection time. At present, we have estimated these parametric recognition times from the empirical data being modeled.

EPIC's cognitive processor is programmed with production rules, and so in order to model a task, we must supply a set of production rules that specify what actions in what situations must be performed to do the task. We are using the interpreter from the Parsimonious Production System (PPS) which is especially suited to task modeling work (Bovair, Kieras, & Polson, 1990). One important feature of PPS is that control information such as current task goals is simply another type of WM item, and so can be manipulated by rule actions. A critical difference from many other production-system architectures is that on each cognitive processor cycle, any number of rules can fire and execute their actions; this parallelism is a fundamental

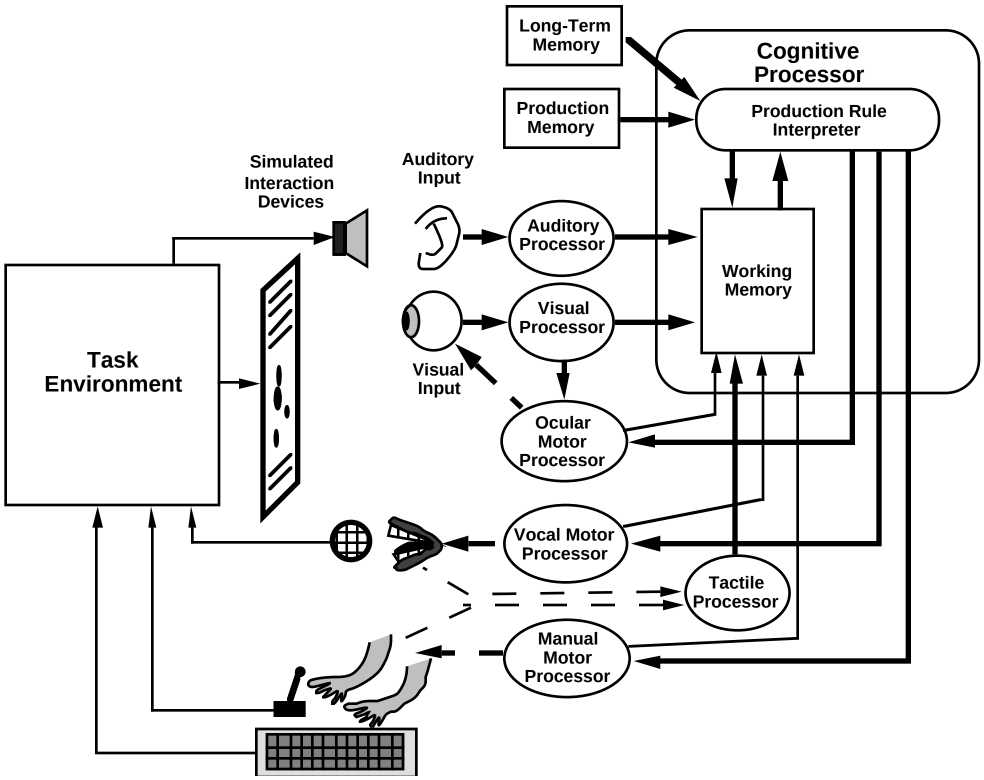

Figure 1. Overall structure of the EPIC architecture showing information flow paths as solid lines, mechanical control or connections as dotted lines. The processors run independently and in parallel; task performance is simulated by having the EPIC model interact with a simulated task environment.

feature of PPS. The cognitive processor accepts input only at the beginning of each cycle, and produces output at the end of the cycle, whose mean duration is estimated to be 50 ms. Thus, unlike in some other information-processing architectures, the EPIC cognitive processor is not constrained to be doing only one thing at time. Rather, multiple processing threads can be represented simply as sets of rules that happen to run simultaneously.

Another important of EPIC involves the basic temporal relationships of its processors. The perceptual processors in EPIC are "pipelines," in that an input produces an output at a certain later time, independently of what particular time it arrives. However, the cognitive processor accepts input only every 50 ms, and is constantly running, not synchronized with external events. This means that perceptual processor output to the cognitive processor must wait an average of 25 ms until it is accepted. Thus the temporal resolution on sensory events is limited centrally, rather than reflecting temporal integration by the perceptual processors. Our proposal, along with the 50 ms cognitive cycle time, is supported nicely by work on human simultaneity judgments (Kristofferson, 1967).

The EPIC motor processors are much more elaborate than in previous models. Certain results (e.g., McLeod, 1977) motivate our assumptions that the motor processors operate independently, but the hands are bottlenecked through a single manual processor. Thus, according to EPIC, the hands normally cannot be controlled independently; rather they can be operated either one at a time, or synchronized with each other. Past research on movement control (e.g., Rosenbaum, 1980) suggests that movements are specified in terms of features, and the time to produce a movement depends on its feature structure as well as its mechanical properties. We have represented this property in highly simplified models for the motor processors. The input to the motor processors consist of a symbolic name for the desired movement, or movement feature. The processor recodes the symbol into a set of movement features, and then initiates the movement. The external device will then detect the movement after some additional mechanical delay. For example, using our estimates, if the desired movement is to press a button with the right-hand index finger, the symbolic name would be recoded into the movement features <RIGHT, INDEX>, taking an average of 50 ms each, followed by 50 ms for the movement initiation, and a final 10 ms for the mechanical motion of pressing the button. The manual motor processor contains mechanisms for a variety of different movement styles, each with its own feature set and execution time. Examples are pressing a button under one of the fingers, pecking at a key some distance away, pointing at an object, or plying a joystick to position a cursor.

An important empirical result is that effectors can be preprogrammed if the movement can be anticipated (Rosenbaum, 1980). In our model, this takes the form of instructing the motor processor to generate the features, and then at a later time instructing the motor processor to initiate the movement. As a result of the pre-generation of the features, the resulting movement will be made sooner. Finally, we assume that a motor processor can prepare only one movement at a time, but this preparation can be done in parallel with the physical execution of a previously commanded movement.

3 Modeling Task Performance with EPIC

3.1 Constraints in Model Construction

How we use EPIC to model task performance can be summarized in terms of what we have to supply to construct a model, the results produced by the model, and which parameters are free to vary and which are fixed. The inputs to the EPIC modeling process are:

- A production-rule representation of the procedures for performing the task;
- The physical characteristics of objects in the environment, such as their color or location;
- A set of specific instances of the task situation, such as the specific stimulus events and their timing;
- Values of certain time parameters in the EPIC architecture.

The output of the EPIC model is the predicted times and sequences for the actions in the selected task instances.

The parameters and model properties that are fixed are:

- The basic structure of the EPIC architecture, such as the internal mechanisms and connections of the processors;
- Most processing time parameters;
- The feature structure and time parameters of the motor processors.

The parameters and properties that are free to vary are:

- The production rule programming for the task, since it must represent the task procedures;
- Certain task-specific perceptual encodings and their time requirements, such as the time to recognize the stimulus shapes appearing on a screen in a particular system. These must be estimated in some way, but are constrained to be constant over similar stimuli.
- The specific styles of movements made by a person. These are often not adequately constrained by the task, and so must be chosen on the basis of observation or left free to vary.

Once the EPIC model is constructed, we generate predictions of task performance by simulating the human interacting with the task environment in simulated real time, in which the processors run independently and in parallel. We include a process that represents the task environment, and which generates stimuli and collects the responses and their simulated times over a large number of trials. To represent human variability, the processor time parameters are varied stochastically about their mean values with a regime that produces a coefficient of variation for simple reaction time of about 20%, a typical empirical value.

3.2 Modeling Multiple-Task Performance

3.2.1 Rationale for EPIC's basic assumptions

The literature on multiple-task performance is extensive, and will not be summarized here; for a review, see Gopher and Donchin (1986). Of course human information processing is limited in capacity, and it has been traditionally assumed that there is a single-channel bottleneck (Welford, 1952). But humans can do multiple tasks, sometimes impressively well, and their ability to do so depends strongly on the specific combinations of tasks involved. The multiple-resource theory (Wickens, 1984) is an attempt to summarize these relationships. They pose a fundamental theoretical dilemma about how to reconcile the complex patterns of people's multitasking abilities with some notion that the overall capacity of the human system is limited.

In developing EPIC, our theoretical strategy is to make some radical simplifying assumptions and then explore their consequences through modeling. We assume that all capacity limitations are a result of limited structural resources, rather than a limited cognitive processor. Thus, the EPIC cognitive processor can fire any number of rules simultaneously, but since the peripheral sense organs and effectors are structurally limited, the overall system is sharply limited in capacity. For example, the eyes can only fixate on one place at a time, and the two hands are bottlenecked through a single processor. We also assume that certain apparent limitations in central capacity arise when modality-specific working memories must be used to maintain task information, but we have not yet tested this assumption in the EPIC framework. Thus far, this simple and radical set of assumptions about the nature of multiple-task processing limitations has held up well.

3.2.2 Multiple tasks and executive processes

Some theories of multiple-task performance postulate an executive control process that coordinates the separate multiple tasks (e.g. Norman & Shallice, 1986). We do likewise, but a key feature of our approach is that the executive control process is just another set of production rules. These rules can control other task processes by manipulating information in WM. For example, we assume that each task is represented by a set of production rules that have the task goal appearing in their conditions, and so an executive process rule can suspend a task by removing its governing goal from WM, and then cause it to resume operation by reinserting the goal in WM. Also, the executive process can cause a task to follow a different strategy by placing in WM an item which task rules test for, thus enabling one set of rules, and disabling another. In addition, the executive process may control sensory and motor peripherals directly, such as moving the eye fixation from one point to another, in order to allocate these resources between two tasks. Thus, rather than postulating an executive control mechanism that is somehow different in kind than other cognitive mechanisms, EPIC has a uniform mechanism for the control of behavior, both at the executive level and at the detailed level of individual task actions. As a corollary, learning how to coordinate multiple tasks is simply learning another (possibly difficult) skill, as has been proposed by some recent investigators (Gopher, 1993).

3.3 Previous Work with EPIC

3.3.1 A basic dual-task paradigm

Our first work with EPIC focused on the simplest and most heavily studied dual-task procedure in the research literature, the so-called Psychological Refractory Period (PRP) procedure. The PRP procedure consists of two temporally overlapping choice reaction-time tasks; the subject is instructed to make the response to the first stimulus before making the response to the second stimulus. The primary measure of interest is the reaction time to the second stimulus, which may be affected by the temporal spacing between the two stimuli. The basic empirical result is that the second response is substantially delayed as the spacing between the two stimuli decreases. The conventional interpretation of this effect (the PRP effect) is that the human has a central bottleneck, and so the second response cannot be selected or initiated until the first response has been made. However, the details of the effect and how it depends on other factors such as the stimulus and response modalities of the two tasks, make up a complex pattern that has never been satisfactorily explained in any detail.

Meyer and Kieras (1995) provide an exhaustive treatment of the PRP effect using EPIC simulations, and mathematical analyses based on them, to account quantitatively for the results in many published experiments. The correct interpretation of the PRP effect is that in order to conform to the task instructions, subjects must adopt a strategy that delays initiating the second response until they can ensure that it does not occur before the first response; the magnitude of the delay in the second response depends on how much of the second-task processing can be overlapped with the first task, which in turn depends on the details of the task structure (e.g. whether eye movements are required), the task difficulty, and the task modalities. The EPIC architecture captures the relevant constraints very well; Meyer and Kieras were able to construct models that accounted for the specific patterns of effects in quantitative detail, and revealed the underlying structure of the phenomena. More details are available in Meyer and Kieras (1995).

3.3.2 Multimodal interface performance prediction

A second line of work consists of using EPIC to predict performance with systems involving high-performance multimodal tasks (Kieras, Wood, & Meyer, 1995). The task domain is that of telephone operators, who collect billing numbers spoken by customers, enter them into a computer workstation, and verify the number before allowing the call to proceed. The volume of this work is such that saving a few seconds of work time per call is worth millions of dollars annually in labor costs. Human operators normally overlap speaking with and listening to the customer with pressing keys and watching for information to appear on the screen. The time taken to handle the call is not simply the sum of the individual activity times, but is a complex function of which activities can be overlapped and to what extent.

Prior work in this domain (Gray, John, & Atwood, 1993) demonstrated that changes in the workstation design could be analyzed and predicted in terms of the patterns of perceptual, cognitive, and motor activities required to perform specific instances of the task. This prior work used hand-constructed schedule charts (PERT charts) to analyze the work flow in a fine-grain analysis of a set of videotaped benchmark task performances. The schedule charts were then modified by hand to predict performance with a somewhat different workstation design that would cause certain activities to be performed in a different order. Although the Gray, John, and Atwood (1993) work was successful, and is attributed with millions of dollars in savings, the process of constructing the models was extremely slow and difficult.

Kieras, Wood, and Meyer (1995) demonstrated how EPIC models for such tasks could be constructed on an a-priori basis, starting with a simple procedural task analysis which was then translated in a standardized format into a set of EPIC production rules. Using additional videotaped performances, the predictions of the EPIC model were compared to actual data at the same fine-grained level. The predictions of task execution time were accurate within 10%. The EPIC architecture accurately represented the perceptual and motor constraints in the task, making it possible to easily construct a model on an a-priori basis that predicts the task time accurately enough to aid in choosing between alternative designs.

4 Models for Tracking/Choice Dual Tasks

The main focus of this paper is on a classic dual-task paradigm involving simultaneous tracking and choice reaction. The remainder of the paper will describe two studies, one involving a simple form of this paradigm, the other, still in progress, involving a complex form. Developing an EPIC model for tracking brings out many key issues about the nature of tracking; our approach has been to represent tracking movements as a distinct motor processor "style" in which the human has a well-learned skill for operating the tracking control; the cognitive processor need only supervise the process.

4.1 A Simple Tracking/Choice Task

We have modeled some results obtained by Martin-Emerson and Wickens (1992) which are especially revealing about the role of eye movements and the use of visual information during such a dual task. Figure 2 shows the EPIC model display of the task. The display is not exactly like the actual experimental display; rather we use such displays as a debugging aid to show what the model is doing during development. The display shows the visual environment of EPIC with the objects in their correct sizes and positions; the small gray circle shows the location and size of EPIC's fovea, currently on the choice stimulus, and the larger gray circle marks the boundary of the parafovea, a region of intermediate discriminative ability.

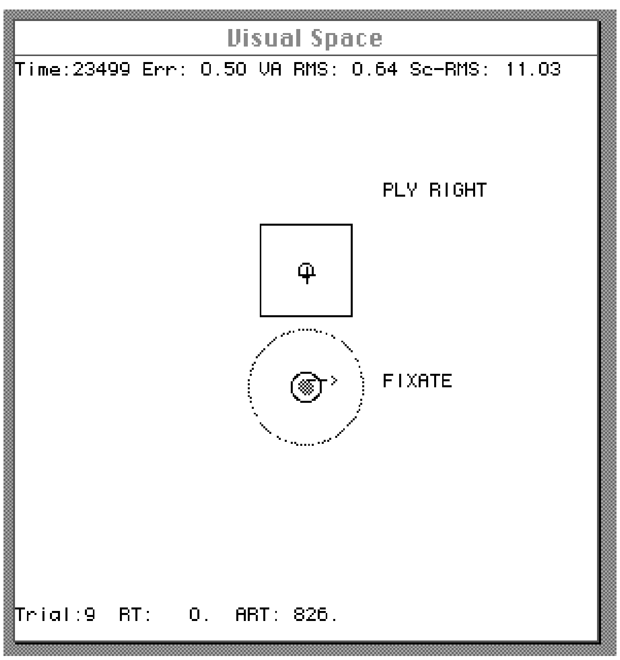

Figure 2. EPIC model display for the Martin-Emerson and Wickens task.

The compensatory tracking task uses the portion of the display in the upper box; a cursor (the cross) must be kept centered on a target (the small circle), using a joystick manipulated by the right hand. Occasionally a stimulus appears in the choice stimulus area (the solid circle below the tracking box), which is either a left- or right-pointing arrow (not shown to scale in the display). The subject must respond by pressing one of two buttons with the left hand as soon as possible, while attempting to maintain the cursor on the target.

The major independent variable is the distance (in visual angle) between the tracking target and the choice stimulus, and a second variable is the difficulty of the tracking task. The two dependent variables are the reaction time to the choice task and the RMS error in the tracking task, collected for a two-second period following the onset of the choice stimulus. As shown by the Observed curves in Figure 3, choice reaction time increases with the angular distance between the target and the choice stimulus, but is unaffected by tracking difficulty. The RMS error increases somewhat with the angular separation, and to an equal extent for both levels of tracking difficulty.

Our models for this task assume that successful tracking requires that the eye be kept on the tracking cursor, but that in order to discriminate the choice stimulus, the eye must be moved to the choice stimulus. However, if the choice stimulus is close enough to the eye position, parafoveal vision will be adequate to discriminate the stimulus without moving the eye. Hence the two tasks often, but not always, compete for use of the eye. Finally, because both tasks involve manual responses, they compete for access to the manual motor processor. We will illustrate how EPIC can be applied to this task with two models.

4.1.1 A simple lockout model

The Lockout Model uses a simple strategy, shown in flowchart form in Figure 4, that is consistent with traditional thinking about dual task situations, namely the lower priority task is locked out (suspended) while the higher priority task is executed. The tracking task rules simply make a motor movement whenever the cursor is adequately far off the target. The executive process normally allocates control of the eye to the tracking task, where a production rule ensures that the eye makes a movement to the cursor anytime it is too far off. The oculomotor processor also can autonomously make small adjustments using perceptual information about object movements. When the choice stimulus appears, the executive process suspends the tracking task, activates the choice task, and then allocates control of the eye to the choice task, moving it to the stimulus if it is too far away to be discriminated. When the choice response has been initiated, the executive resumes the tracking task and returns control of the eye to the tracking task. In this way,

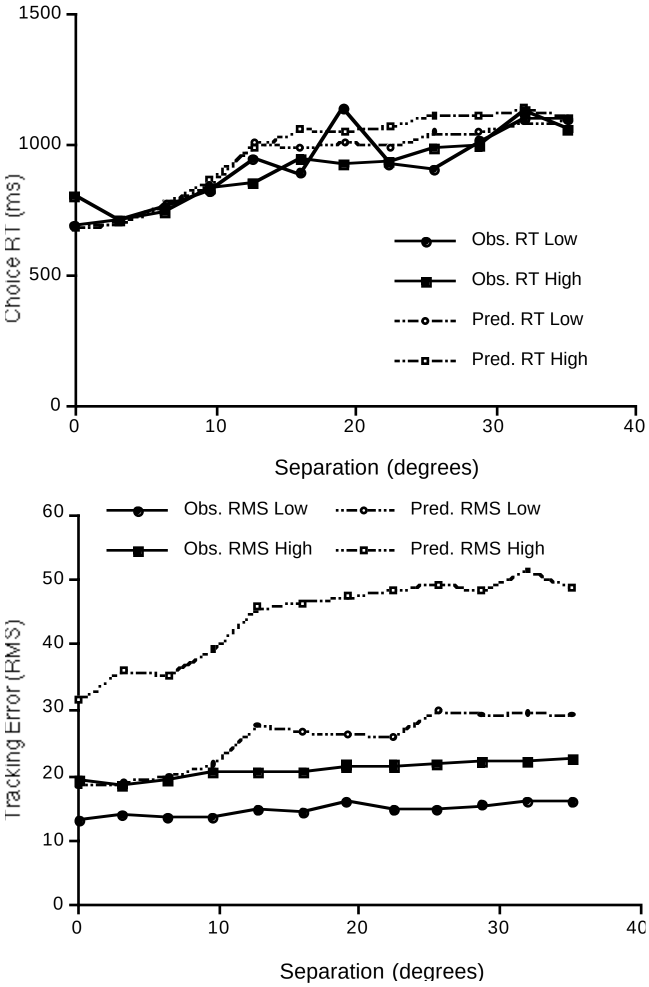

Figure 3. Predicted and observed effects of stimulus separation and tracking difficulty for the Lockout Model. Observed are large solid points and lines; predicted are small open points and dotted lines.

using what we term lockout scheduling, the executive process allows only one task to be done at a time, ensuring that the choice task has priority over the tracking task, and that the eye and the manual motor processor are only used for one task at a time.

Unfortunately, this simple strategy does not fit all aspects of the data. Figure 3 shows the predicted and observed values for the choice reaction time and the tracking error. Using a best estimate from the data of the perceptual recognition time for the choice stimulus, we can fit the choice reaction time data fairly well. There is no effect of tracking task difficulty since the choice task is given priority over tracking. The first few points are fairly flat, due to the parafoveal recognition of the choice stimulus. The upward slope of the curves at larger separations reflects the time required to move the eye. The fit of the simulated tracking data is extremely poor, however. The overall magnitude of tracking error is seriously over predicted, as is the effect of tracking difficulty and the effect of visual

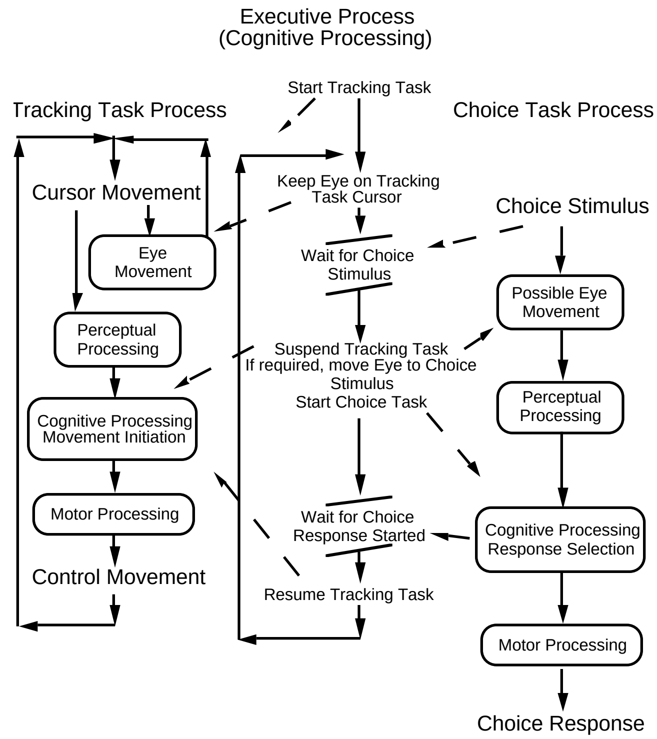

Figure 4. Flowchart of Lockout Model strategy for the Martin-Emerson and Wickens task.

separation.

The Lockout Model cannot be made to fit the tracking data better by adjusting the relevant parameter values; it is already using the minimum plausible time estimates for all of the perceptual and motor parameters involved. Rather, note that the RMS error is measured for a brief (2 s) period of time starting with the onset of the choice stimulus, meaning that if tracking is suspended for too long during this period, the effect will be substantial. The Lockout Model suspends the tracking task for such a long time that considerable tracking error accumulates; it is simply too inefficient.

4.1.2 An interleaved model

We constructed a second model, the Interleaved Model, in which the executive overlaps the processing on the two tasks as much as possible; this strategy is shown in Figure 5. When the choice stimulus appears, the executive moves the eye to the choice stimulus, and then immediately begins to move it back, relying on the "pipeline" property of the visual system to acquire the stimulus and continue to process it, even after the eye has returned to the tracking cursor. The tracking task is suspended only while the eye is away looking at the stimulus arrow for the choice task. The model uses the same approach as in our PRP models for allocating control of the manual motor processor. When the choice task rules have chosen the response, it signals the executive, which again suspends the tracking task, and allows the choice task to command the manual motor processor, and then resumes the tracking task right away.

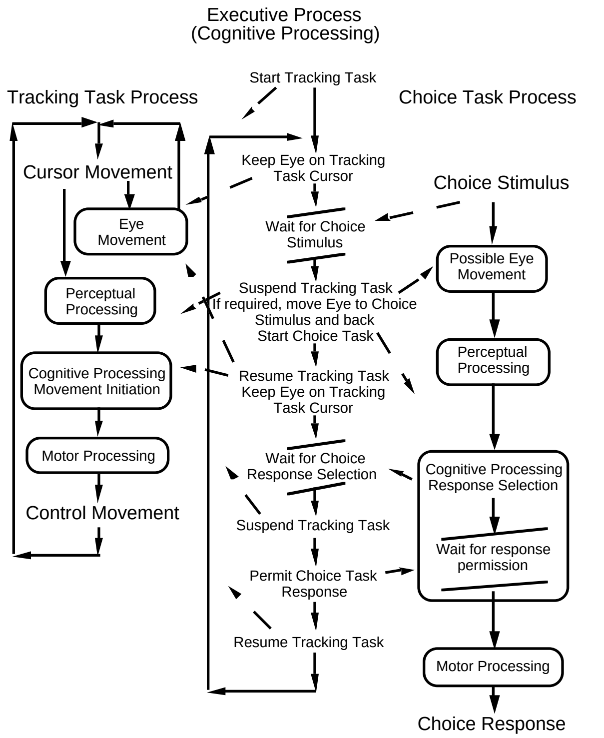

Figure 5. Flowchart of the Interleaved Model strategy for the Martin-Emerson and Wickens task.

Thus, the same task priorities are honored, but the tracking task is interrupted as little as possible. The predictions from this model are shown in Figure 6. The choice reaction times are again well fit, but now the tracking task predictions are extremely close as well. Since the interleaved executive allocates the eye and the manual motor processor to the tracking task for the maximum amount of time, the tracking task rules can squeeze in a few movements while the choice task is underway, resulting in substantially less tracking error than the Lockout Model.

4.1.3 Conclusions

Two important substantive conclusions stem from this modeling work. First, the control of the eye is critical in dual task paradigms. Second, subjects can and apparently do use subtle strategies for coordinating dual tasks in surprisingly efficient ways.

Another general conclusion concerns a common misunderstanding about computational models. They do not in fact have so many "degrees of freedom" that they can be made to fit any data at any time. As in other efforts such as ACT and SOAR, working within the fixed EPIC architecture sets powerful constraints. Given the basic lockout strategy, there were no parameter values or specific strategy details that would allow us to fit the data as a whole. The only way EPIC could be applied to fit the data was by assuming a fundamentally different strategy. Of course, further empirical work could be done (e.g. eye tracking) to determine whether the Interleaved Model strategy in fact describes the use of the eye correctly. Thus,

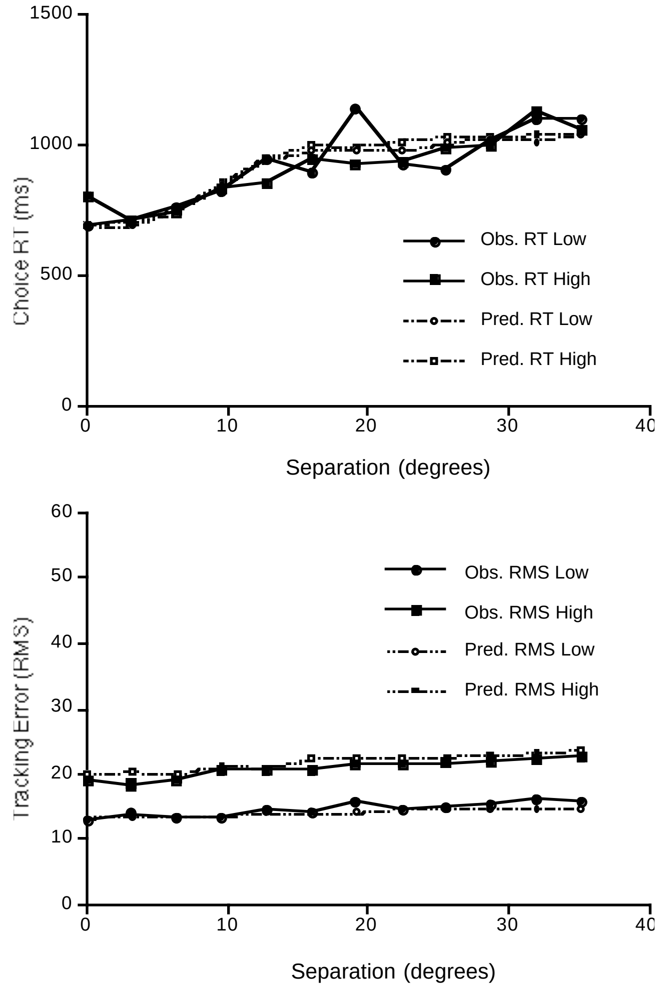

Figure 6. Predicted and observed effects of stimulus separation and tracking difficulty for the Interleaved Model. Observed are large solid points and lines; predicted are small open points and dotted lines.

a general conclusion (also observed in our earlier work) is that the exercise of seeking quantitatively accurate accounts of data within a fixed architecture is extremely informative, both about the accuracy of the architecture theory and also the structure and requirements of the task.

4.2 A Complex Tracking/Decision-Making Task

The modeling work on the Martin-Emerson and Wickens task laid the foundations for our current work on a more complex dual tracking/choice task. This task was developed by Ballas, Heitmeyer, and Perez (1992a, 1992b) to resemble a class of tasks performed in combat aircraft in which analyzing the tactical situation is partially automated by an on-board computer. To help the explanation, Figure 7 shows our EPIC model display for this task. The right hand box contains a pursuit tracking task in which the cursor (cross) must be kept on the target (small box); the eye is shown currently on the cursor. Average tracking error data were collected during various phases of the experiment. The left-hand box contains the choice task, a tactical decision task in which objects (or "tracks") must be

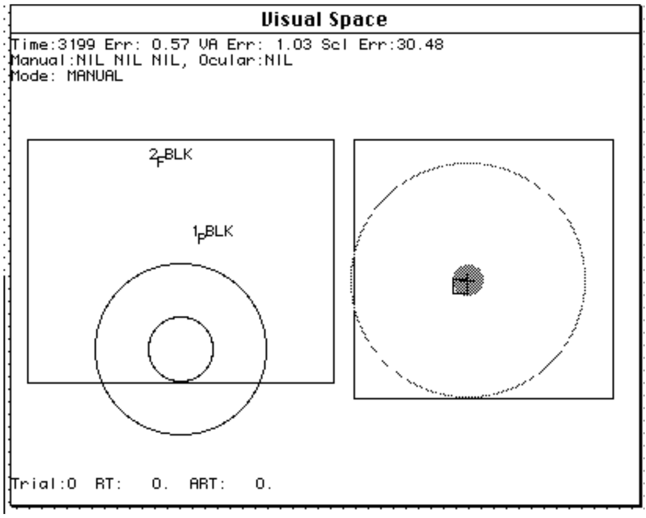

Figure 7. EPIC model display for the Ballas et al. task.

classified as hostile or neutral based on their behavior. These objects represent fighter aircraft, cargo airplanes, and SAM sites that move down the display as the subject's aircraft travels.

In the actual Ballas et al. (1992a) display, each type of object was coded by an icon; for simplicity, in the EPIC display they are represented instead by a code letter. A track number identifies each object. Objects appear near the top of the display, and then move down the display. After some time, the on-board computer would attempt to designate the objects, indicating the outcome by changing the object color from black to red, blue, or amber, which the EPIC display shows with a three-character abbreviation. If the object became red (hostile) or blue (neutral) the subject had to simply confirm the computer's classification; the response was typing on a keypad a key for the hostile/neutral designation followed by the key for the track number. If the object became amber, the subject had to classify the object based on a set of rules for the object's behavior, and then type the hostility designation and track number. After the response, the object changed color to white, and then disappeared from the display some time later. The basic dependent variable is the reaction time to the objects, measured from when the object changed color to when the first of the two response keystrokes were made.

Ballas et al. varied two major aspects of this task. One set of manipulations concerned the format of the tactical display and the response. The above description is for one of the four combinations; the other combinations consisted of using a tabular display instead of the graphical radar-like display, and a touchscreen response procedure instead of the keypad. At this time we have modeled only the one interface combination described above.

The second manipulation concerns the effects of adaptive automation. From time to time during the task, the tracking task would become difficult, and the on-board computer would take over the tactical task, signaling when it did so. The computer would then generate the correct responses to each object at the appropriate time, with the color changes showing on the display as in the manual version of the task. Later, the tracking would become easy again, and the computer would signal and then return the tactical task to the subject. Ballas et al. observed an automation deficit effect, in which for a time after resuming the tactical task, subjects produced longer response times in the tactical task compared to their normal steady-state manual performance. This effect represents some of the serious concerns about possible negative effects of automation in combat situations; if the automation fails, the operator can lack situation awareness, and it might take a long time to "catch up."

4.2.1 A preliminary model

At the time of this writing, our EPIC models for the Ballas task are in a preliminary stage; we have not yet begun to produce quantitatively accurate fits, but have begun to capture the qualitative phenomena. This description of our results is subject to revision in later presentations of this work.

We have estimated the parameters for the basic perceptual encoding operations required in the tactical task, namely recoding the blue and red colors to the appropriate key, and recognizing the hostility of behaviors of different kinds of objects, which takes considerably longer (more than a second). We have assumed that assessing the hostility of an object requires that it be fixated, but that an object's color would be available parafoveally, and color changes, like object onsets and offsets, would be visible in peripheral vision.

Our current preliminary model is an initial simple one, structured much like the lockout model described above for the Martin-Emerson and Wickens task; we may need a more complex interleaved model to fully fit the data. When the tactical task is being done by the subject (as opposed to the on-board computer), the executive process allows the tracking task to run until it is time to work on the tactical decision task. Thus it ignores the simple appearance of an object and instead waits for a detection of a color change in peripheral vision. The executive then suspends the tracking task, and allocates the eye to the tactical task. The eye is moved to the changed object, and the appropriate response made when the perceptual information (color coding or hostility behavior) becomes available. If additional objects have changed color in the meantime, the tactical task rules choose one at random, move the eye to it, and process it. When no more objects remain to be processed, the tracking task terminates, and the executive then returns control of the eye to the tracking task and restarts it. We expect to be able to fit the tactical reaction time data and the tracking task performance well, since this task is essentially just a complicated version of the Martin-Emerson and Wickens task. At this point, our preliminary model reproduces the qualitative relationships associated with single and dual-task reaction times and average tracking errors.

4.2.2 A preliminary explanation of the automation deficit effect

More interesting though, is our current hypothesis for the source of the automation deficit effect. We assume that when the tactical task is automated, the subject simply ignores the color changes appearing in the tactical display and does not bother to store any information about the state of the display in working memory. When it is time to resume the tactical task, multiple objects typically need to be processed, and since there is no record of which have changed colors or the order in which they appeared, the task strategy simply picks the first one to inspect at random. After moving the eye to it and waiting for the color to become available, the strategy processes the object as usual if it is red, blue, or amber. However, if it is white or black, it cannot be processed, and so another object is picked at random. When all candidate objects have been dealt with, tracking is resumed, and future object changes are processed as they appear.

The automation deficit results from the fact that when the tactical task is being performed normally, objects are usually processed in the order that they change color, keeping the average reaction time to a minimum. In contrast, when the tactical task is resumed, multiple objects must be inspected, and no information has been kept on the order in which they have appeared or changed color (otherwise, the automation is of little value!). Thus the objects are inspected in random order, meaning that objects that changed first will have to wait longer on the average to be inspected than if they were processed in order.

Figure 8 illustrates what can happen during tactical task resumption. Although Track 1 (a blue plane) was the earliest changing object, the strategy happened to pick Track 2 (a red fighter) to process first instead. After Track 2 is processed, the strategy will choose one of the remaining three objects to inspect next. If Track 4 is chosen, time would be wasted since a black target should not be processed yet; if Track 3 (an amber plane) is picked, the other tracks would go unprocessed for a long time while the model waits for the hostility status of Track 3 to become apparent. Thus it may be a long time before Track 1 gets processed. As the model performs the task, it will catch up after some time, and objects will again be processed mostly in the order that they change. Thus, relative to steady-state performance, performance on the tactical task is depressed for some time following its resumption; temporarily, objects may take longer than normal to get processed.

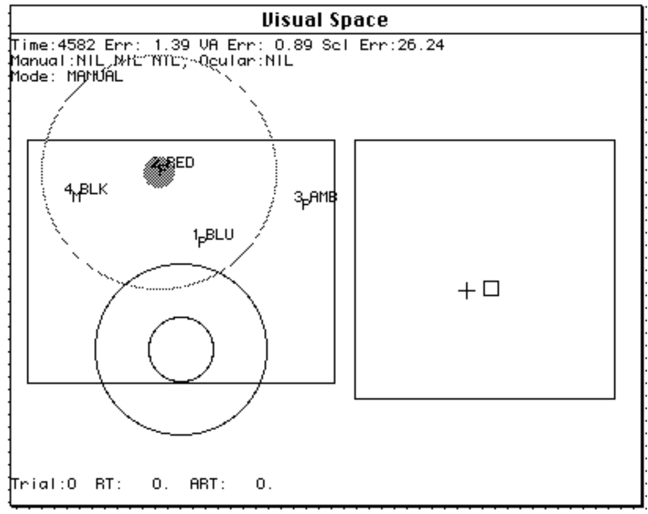

Figure 8. Automation deficit effect: After resuming the tactical task, the model inspects the objects out of order.

4.2.3 Conclusions and current plans

At the time of this writing, our preliminary model for Ballas et al. does not yet produce the quantitatively accurate times for the single-task processing, and so quantitative predictions of the automation deficit effect are not yet definite. However, using the Ballas et al. measure, our preliminary model produces an automation deficit of about 500 ms with two of the actual task scenarios, which compares favorably to the value of about 800 ms observed in the same interface condition over all of the task scenarios. Thus our current hypothesis about the cause of the automation deficit appears plausible and worth pursuing further.

Our current plan is to develop a model that accounts quantitatively for performance in all of the Ballas et al. (1992a) experiment conditions, including the effects of different display and response formats, as well as the automation deficit, and the interactions of all of the factors.

If our current explanation for the automation deficit is borne out by our more complete and accurate models, there may be some important implications for display and task design. For example, according to this hypothesis, resuming the tactical task could be done more efficiently if it is possible to easily detect the highest-priority object on the display. That is, suppose the first-changed object currently on the display was coded by making it blink, which would be salient in peripheral vision. The subject could simply look at the blinking object in order to ensure that the objects were processed in priority order. Alternatively, the automated version of the task could use a different, less salient, way of representing its activity, so that the subject could still profitably monitor for the same perceptual events that are important in the manual version.

Of course, how these issues show up in actual cockpit displays must be considered, but we are optimistic that EPIC will provide a framework for analyzing interface designs for these high-performance multiple-task situations.

5 General Conclusions

At this point, EPIC is definitely a research system, and certainly is not ready for routine use by system or interface designers. However, there is a technology transfer precedent: earlier work with the CCT production rule models for HCI (Bovair et al., 1990), led to a practical interface design technique (see John & Kieras, 1994). Likewise, as the EPIC architecture stabilizes and experience is gained in applying it to human-system analysis problems, we should be able to devise a simplified approach that will enable designers to apply EPIC to develop improved human-system interfaces.

Our experience with the EPIC architecture also suggests some meta-level conclusions about the role of cognitive modeling in the science and engineering fields of human performance and human-system interaction:

- As mentioned above, powerful constraints are imposed by quantitatively fitting fixed-architecture models to detailed performance data. Thus, working with computational models is not necessarily an arbitrary exercise.
- Computational models that are based on human information-processing can usefully predict details of human performance in system design and evaluation situations.
- Developing and applying a cognitive model to task situations relevant to real design problems is a demanding test of cognitive theory; if the theory successfully represents important properties of human abilities, it should in fact be useful in practical settings.

In short, a comprehensive, detailed, and quantitative theory of human cognition and performance is the best basis for applied cognitive psychology. Rather than relying only on general psychological principles, or brute-force application of experimental methodology, system design can be best informed by using a theory that can address phenomena at the same level of detail that design decisions require. It really is true that "Nothing is more useful than a good theory!"

Acknowledgement

Thanks are due to James Ballas of NRL for his generous assistance in making his task software available and supplying new analyses of his data.

References

Anderson, J. R. (1976). *Language, memory, and thought.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Bovair, S., Kieras, D. E., & Polson, P. G. (1990). The acquisition and performance of text editing skill: A cognitive complexity analysis. *Human-Computer Interaction,* **5**, 1-48.

Ballas, J. A., Heitmeyer, C. L., & Perez, M. A. (1992a). *Direct manipulation and intermittent automation in advanced cockpits.* Technical Report NRL/FR/5534--92-9375. Naval Research Laboratory, Washington, D. C.

Ballas, J. A., Heitmeyer, C. L., & Perez, M. A. (1992b). *Evaluating two aspects of direct manipulation in advanced cockpits.* In Bauersfeld, P., Bennett, J., and Lynch, G., *CHI'92 Conference Proceedings: ACM Conference on Human Factors in Computing Systems*, Monterey, May 3-7, 1992.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Chubb, G. P. (1981). SAINT, a digital simulation language for the study of manned systems. In J. Moraal & K. F. Kraas (Eds.), *Manned system design.* (pp. 153-179). New York: Plenum.

Gopher, D., & Donchin, E. (1986). Workload: An examination of the concept. In K.R. Boff et al. (Eds.), *Handbook of perception and performance,* Vol. II (pp. 41.1-41.49). New York: Wiley.

Gopher, D. (1993). Attentional control: Acquisition and execution of attentional strategies. In D. E. Meyer & S. Kornblum (Eds.), *Attention and performance XIV.* Cambridge, MA: M. I. T. Press.

Gray, W. D., John, B. E., & Atwood, M. E. (1993) Project Ernestine: A validation of GOMS for prediction and explanation of real-world task performance. *Human-Computer Interaction*, **8**, 3, pp. 237-209.

John, B. E. & Kieras, D. E. (1994). *The GOMS family of analysis techniques: Tools for design and evaluation.* Carnegie Mellon University School of Computer Science Technical Report No. CMU-CS-94-181. Also appears as the Human-Computer Interaction Institute Technical Report No. CMU-HCII-94-106.

Kieras, D. E. (1990). The role of cognitive simulation models in the development of advanced training and testing systems. In N. Frederiksen, R. Glaser, A. Lesgold, & M. Shafto (Eds.), *Diagnostic monitoring of skill and knowledge acquisition.* Hillsdale, NJ: Erlbaum.

Kieras, D., Wood, S., & Meyer, D. (1995). Predictive Engineering Models Using the EPIC Architecture for a High-Performance Task. To appear in *CHI'95 Proceedings of the ACM Conference on Human Factors in Computing*, Denver, May 7-11, 1995.

Kristofferson, A. B. (1967). Attention and psychophysical time. In A. F. Sanders (Ed.), *Attention and performance* (pp. 93-100). Amsterdam: North-Holland Publishing Co.

Laird, J., Rosenbloom, P, & Newell, A. (1986) *Universal subgoaling and chunking*. Kluwer Academic Publishers: Boston.

Lane, N. E., Strieb, M. I., Glenn, F. A., & Wherry, R. J. (1981). The human operator simulator: An overview. In J. Moraal & K. F. Kraas (Eds.), *Manned system design* (pp. 121-152). New York: Plenum.

Martin-Emerson, R., & Wickens, C. D. (1992). The vertical visual field and implications for the head-up display. *Proceedings of the 36th Annual Symposium of the Human Factors Society*. Santa Monica, CA: Human Factors Society.

McLeod, P. D. (1977). A dual task response modality effect: Support for multiprocessor models of attention. *Quarterly Journal of Experimental Psychology*, **29**, 651-668.

McMillan, G. R., Beevis, D., Salas, E., Strub, M. H., Sutton, R., & Van Breda, L. *Applications of human performance models to system design*. New York: Plenum Press, 1989.

Meyer, D. E., & Kieras, D. E. (1995). A Computational Theory of Human Multiple-Task Performance: The EPIC Information-Processing Architecture and Strategic Response-Deferment Model. *Psychological Review*, accepted for publication.

Norman, D. A., & Shallice, T. (1986). Attention to action: Willed and automatic control of behavior. In R. J. Davidson, G. E. Schwartz & Shapiro (Eds.), *Consciousness and self-regulation, Vol. 4*. New York: Plenum Press.

Rosenbaum, D. A. (1980). Human movement initiation: Specification of arm, direction, and extent. *Journal of Experimental Psychology: General*, **109**, 475-495.

Schneider, W., & Detweiler, M. (1987). A connectionist/control architecture for working memory. In G. H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 21, pp. 53-119). Orlando, FL: Academic Press.

Welford, A. T. (1952). The "psychological refractory period" and the timing of high speed performance - A review and a theory. *British Journal of Psychology*, **43**, 2-19.

Wickens, C. D. (1984). *Engineering psychology and human performance*. Columbus, OH: Charles F. Merrill.