Computational Models for the Effects of Localized Sound Cuing in a Complex Dual Task

David E. Kieras
University of Michigan

James Ballas
Naval Research Laboratory

and

David E. Meyer
University of Michigan

EPIC Report No. 13 (TR-01/ONR-EPIC-13)

January 31, 2001

This research was supported by the Office of Naval Research, under Grant Number N00014-96-1-0467.
 Requests for reprints should be sent to: David E. Kieras, Artificial Intelligence LaboratoryElectrical Engineering & Computer Science Department, University of Michigan, 1101 Beal Avenue, Ann Arbor, MI 48109-2110, kieras@eecs.umich.edu.

Computational Models for the Effects of Localized Sound Cuing in a Complex Dual Task

David E. Kieras
Electrical Engineering and Computer Science Department
University of Michigan

James Ballas
Naval Research Laboratory

David E. Meyer
Psychology Department
University of Michigan

Abstract

This report covers the modeling work performed in conjunction with a project at the Naval Research Laboratory, carried out by James Ballas, exploring the effects of using spatialized sound as a task cue in a visually-presented dual task setting that demonstrated an "automation deficit" effect. This effect is temporarily impaired performance when an automated task must be resumed by the human operator. Previous work on modeling this task and automation deficit effect led to the conclusion that the deficit effect was due to ambiguity in the visual display concerning which task event had priority for the operator's attention.

An hypothesis was that providing an auditory cue to supplement the visual display, in the form of synthetically localized sound, should allow the operator to attend to the highest-priority event, and thus mitigate the automation deficit effect. The observed results weakly supported this hypothesis, but a better characterization is that the use of localized sound led to a consistent, but small, effect of faster responding across the board. Additional modeling work suggests that the only simple explanation for this pattern of effects is that the onset of localized sound can produce a reflexive eye movement to the sound source. The magnitude of the effect is consistent with psychophysical tasks on the facilitation of visual choice reaction tasks by localized sound stimuli.

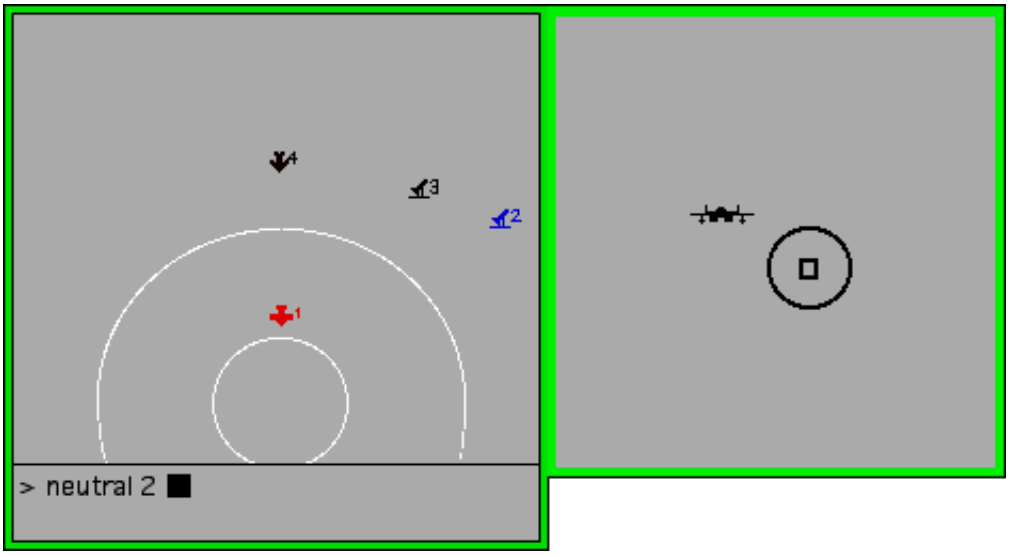

Figure 1. Screen shot of display, showing feedback from input keystrokes.

Introduction

This report presents the modeling work performed in conjunction with a project at the Naval Research Laboratory, carried out by James Ballas, exploring the effects of using spatialized sound as a task cue in a visually-presented dual task setting that demonstrated an “automation deficit” effect. This effect is temporarily impaired performance when an automated task must be resumed by the human operator. Previous work on modeling this task and automation deficit effect led to the conclusion that the deficit effect was due to ambiguity in the visual display concerning which task event had priority for the operator’s attention. A preliminary study and its modeling supports this conclusion, in that a lower workload that removes the ambiguity from the task situation eliminates the automation deficit effect. However, another way to remove the ambiguity at high workload would be to use a spatialized sound cue to designate the highest-priority visual event. This hypothesis led to a series of experimental studies, whose results were explained with computational modeling presented in this reports.

The Task and Automation Deficit

A cockpit-like dual task. The task was developed by Ballas, Heitmeyer, & Perez (1992a, b) to resemble a class of multiple tasks performed in combat aircraft in which the subject must both perform a task such as tracking a target, and at the same time keep up with the tactical situation using sensors such as radar, with partial automation support by an on-board computer. Figure 1 shows a sketch of the display. The right hand box contains a pursuit tracking task in which the circle cursor must be kept on the target with a joystick operated with the right hand. The left-hand box is a radar-like display that contains a tactical decision task in which objects (“tracks”) must be classified as hostile or neutral based on their behavior, and the results entered by means of a keypad under the left hand. These objects appear as icons that represent fighter aircraft, cargo airplanes, and SAM sites. A number identifies each object on the display. To avoid the overloaded term “object” or the military jargon of “track”, the term *blip* will be used to refer to the objects on the radar display. The center of the concentric circles at the bottom of the display represent the position of *ownship*, the position of one’s own aircraft.

The blips appear near the top of the display, and then move down. The fictitious on-board computer attempts to classify each blip, indicating the outcome after some time by changing the blip color from black to red, blue, or amber. These color changes are termed *events* because these color changes are the stimuli to which the subject must respond. If the blip changes to red (hostile) or blue (neutral), the subject must simply confirm the computer's classification by typing a code key for the hostile/neutral designation followed by the key for the blip number. If the blip changes to amber, the subject must observe the behavior of the blip and classify it based on a set of rules, and then type the hostility designation and blip number. After the response, the blip changes color to white, and then disappears from the display 10 sec later. The basic dependent variable is the reaction time to the events, measured from when a blip changes color to when each of the two keystrokes are made in response.

The rules for classifying a blip depend on the type of the blip. A fighter blip is hostile if its trajectory will intersect the ownship position (indicated by a constant bearing from ownship). A cargo plane is hostile if it is traveling at a high speed down the display. A missile site is hostile if its position will lie within the outer range ring (the larger concentric circle) at some point. Note that the blips for cargo planes move vertically down the display, as do the blips for missile sites, corresponding to the ownship movement, but blips for fighters usually move diagonally down the display.

Ballas et al. varied the format of the tactical display and the response. The above description is for the graphical keypad interface; the other combinations consisted of using a tabular display

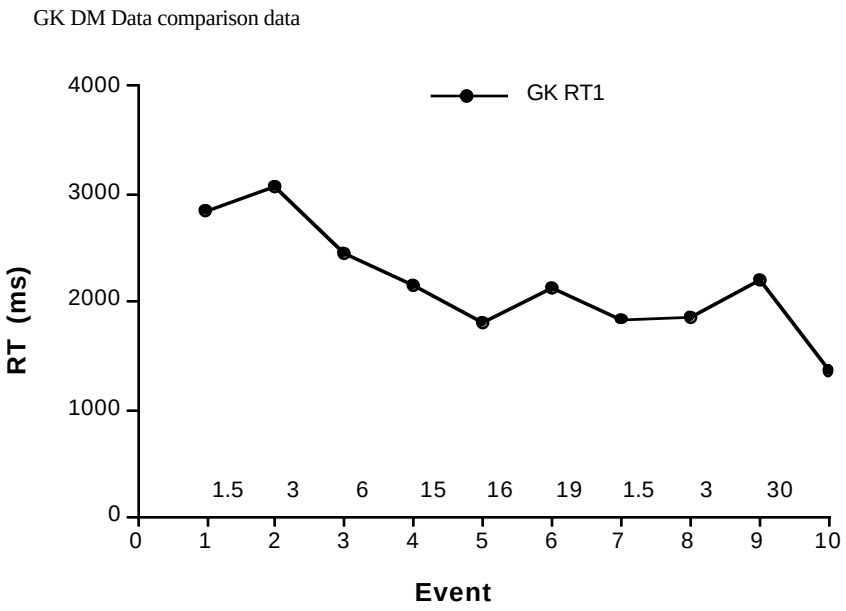

Figure 2. Illustration of the automation deficit effect, using data from Ballas et al. (1992a, b). First response times (RT1) are shown for each event (blip color change) after the tactical task is resumed. The numbers above the x-axis are the interval in seconds between each pair of events. The event sequence 1, 2, and 3 have the same inter-event spacing as event sequence 7, 8 and 9. The automation deficit effect appears as elevated RTs for the first events (e.g. 1) relative to the comparison events (e.g. 7) after the task has been underway for some time.

instead of the graphical radar-like display, and a touchscreen response procedure instead of the keypad. This work concerns only the graphical display with the keypad responses.

Ballas et al. also studied 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 blip 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 to perform. The signal was a loud buzzer sound and a change in the border of the tactical task window - a thick bright border meant that the task was in manual mode. How subjects dealt with the transition was measured by recording the time required to respond to the individual events, counting from when they had to resume the tactical task.

The experiments involved performing multiple scenarios, which are specified sequences of blip types, positions, trajectories, and color-change events. The scenarios relevant to the measurement of the automation deficit effect had *epochs* of manual tactical task performance. In what follows, the term *event* refers the the event of a blip changing color from the initial black color to red, blue, or amber. Each epoch consisted of a series of events whose timing and type were controlled. The major dependent variable is the response time for each event as a function of its position in the epoch.

In the scenarios resulting in automation deficit effects, the blips and color-change events within an epoch were not uniformly spaced in time; rather they occurred in two waves, the first when the task had to be resumed in manual mode at the beginning of each epoch, and the second about two-thirds of the way through the epoch. Some of the event time structure was fixed, with the remaining allowed to vary stochastically within the overall two-wave structure.

To make this presentation concrete, Figure 2 shows the mean reaction time for the first response (RT1) from the experiment reported in Ballas et al. (1992a, b). The horizontal axis corresponds to each event following resumption of the manual tactical task; i.e., Event 1 is the first color-change event, Event 2 is the second, and so forth. Above the x-axis is shown the time interval between events in seconds. Events 1, 2, 3, and 4 were set to appear at closely spaced increasing fixed intervals, as are Events 7, 8 and 9. The other events are widely spaced at randomly chosen intervals whose mean values are shown. Thus there is a high workload at the beginning of task resumption, a low-workload period, followed by another high-workload peak with the same event types and spacing, and a final low-workload period.

Thus the two waves have similar high workloads in terms of the event spacing and number of blips on the screen, and are both followed by fairly inactive times. Performance on Events 1-3 can be compared with performance on Events 7-9 to compare initial resumption performance with steady-state performance. Thus although there is variation between the epochs in each scenario, the epochs have a definite structure which subjects were apparently were aware of. For ease in discussion in the rest of this paper, this structure can be described in terms of phases: an initial "resumption panic" phase in which the tactical task is resumed at the peak of the first wave of blips, followed by a post-panic catching-up phase at the tail of the first wave, followed by widely-spaced events in the first "doldrums" phase. The next phase is a “clump panic” for the second wave of closely spaced events, then a second post-panic phase, and then a second, very short, doldrums phase.

The automation deficit effect. Ballas et al. (1992a,b) observed an automation deficit effect, in which during the resumption panic phase, the period after resuming the tactical task, subjects produced longer response times for matched events compared to their normal steady-state manual performance, that is, the events at the clump panic at the second wave. Thus, as shown in Figure 2, the times for Events 1, 2, and 3 are longer than the matched Events 7, 8, and 9, producing an overall descending shape to the RT profile. The reaction times for the first few events during the resumption panic and catching-up phase are substantially longer than those for later events of similar structure during the clump panic and its catching-up phase. Since Ballas et al. had arranged for Events 1 and 7 to be exactly matched in terms of the type of blip, they reported the automation deficit effect in terms of simply the difference between the RT for Event 1 and Event 7, which is 1312 ms for the first response keystroke. 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.”

Not shown in this graph are other effects. For example, the different event types had very different reaction times across the board. The *confirm* events, in which a red or blue blip had to be responded to as hostile or neutral, were considerably faster than the *classify* events, in which an amber blip had to be studied to determine its hostility. The time for this depends on the type of the track (airplane, fighter, or missile site). Since the distribution of these event types was not uniform across the scenario epochs, one could wonder whether differences in these response time could account for the observed fluctuations in the RTs shown in these graphs. Some preliminary analysis shows that some minor fluctuations in the reaction time could be accounted for by the average RT for each event type weighted by the frequency of appearance of each event type at each event number. However, the features most clearly present in Figure 2 could not be accounted for so simply - rather the large fluctuations in reaction time are a result of a combination of the event spacing and whether the subject has just resumed the tactical task.

A preliminary explanation of the automation deficit effect. Before getting into the details of the models for this task, a simplified explanation for the automation deficit effect can be presented. Such effects seem subtle, and would seem to require complex explanations in terms of ill-defined constructs such as situation awareness. However, the relatively simple explanation

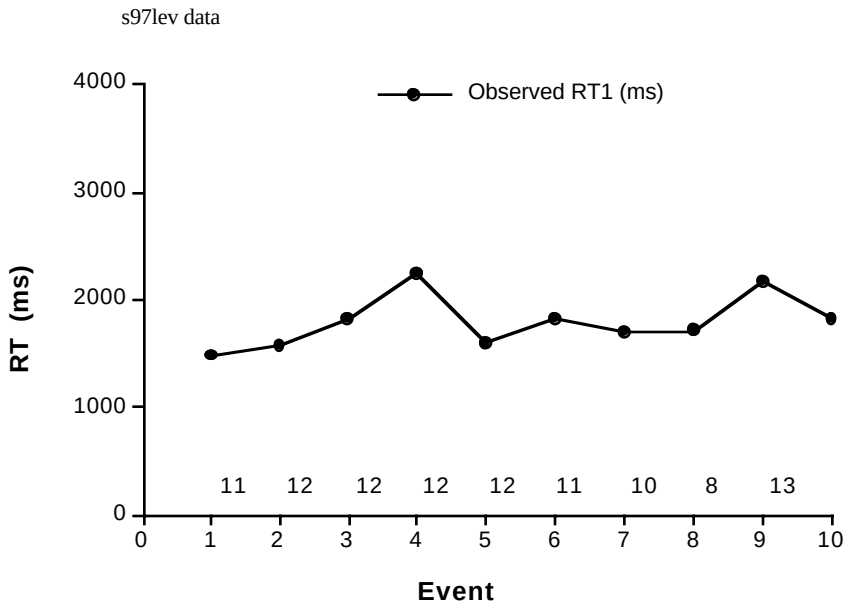

Figure 3. Reaction times (RT1) for events following task resumption in a level-load scenario in which only one blip at a time changes color. The inter-event times are shown above the x-axis. While there is some variation in RT, there is no automation deficit effect due to the lack of ambiguity immediately following task resumption.

advanced in this paper is based on the hypothesis that the subject monitors the tactical display (keeps track of tactical task events) when the tactical task is being done, but not when it is automated. 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 tactical display in working memory (otherwise, the automation is of little value!).

When it is time to resume the tactical task, there are many blips on the screen that need to be inspected and possibly responded to. Since there is no record of which have changed colors prior to task resumption, or the order in which they did so, the subject simply picks a blip to inspect at random. After moving the eyes to it and waiting for its color to become cognitively available, the subject processes the blip as usual if it is red, blue, or amber. However, if it is white or black, it cannot be processed, and so another blip is picked at random. But once the eyes have been moved to the tactical display, the color of many of the blips can be seen, so that a colored blip can be chosen as the target for the second eye movement. This blip can then be responded to, and then any other colored blips readily selected for processing.

When all candidate blips have been dealt with, the eyes are moved back to the tracking window, and tracking is resumed. When a blip changes color, the color change event will be detected (even in peripheral vision, as a change in luminance), and the eyes moved directly to the changed blip. This means that subsequent events are processed as they appear and in the same order.

The automation deficit results from the fact that when the tactical task is being performed in steady state, the blips 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 blips must be inspected, and no information has been kept on the order in which they have appeared or changed color. Thus the blips are inspected in random order, meaning that blips that changed first will have to wait longer on the average to be inspected than if they were processed in order. As the events continue to occur, the subject will begin to catch up, and blips will again be processed in the order

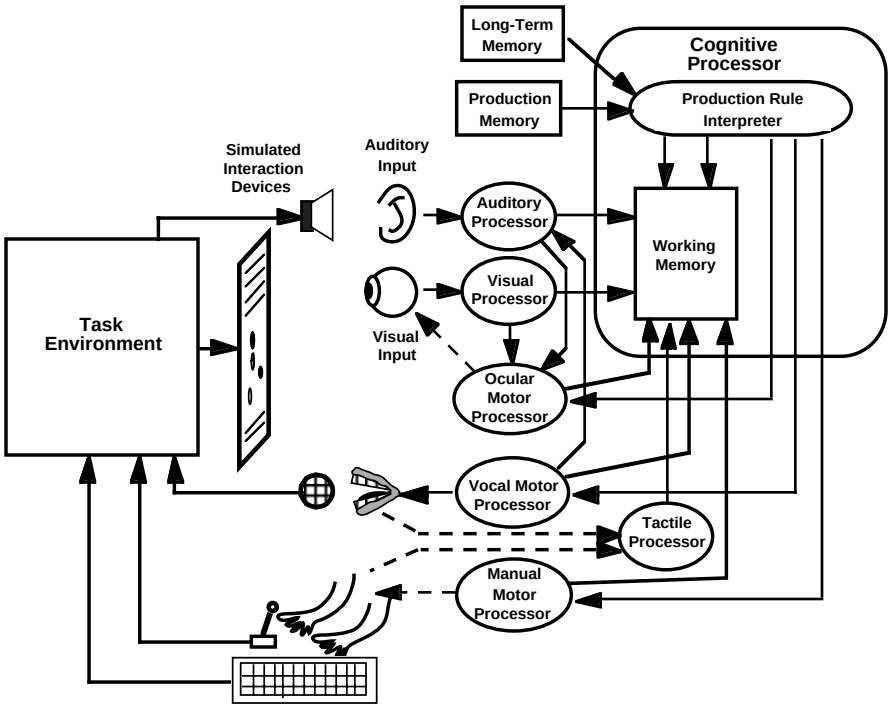

Figure 4. The overall organization of the EPIC architecture. The simulated task is on the left, the components of the simulated human are on the right.

that they change. Thus, relative to steady-state performance, performance on the tactical task is depressed for some time following its resumption; temporarily, events may take longer than normal to get processed.

One implication of this explanation is that the size of the automation deficit effect depends on the event density at the time of task resumption - e.g., if there is only one blip to inspect at the time of resumption, then there will be no out-of-order processing to delay the response. Additional data collected by in Ballas's laboratory shows just this effect, using a scenario in which the events after resumption are fairly widely spaced through the epoch. These results are shown in Figure 3, which shows the first reaction times for events that are spaced on the order of 10 - 13 seconds apart on the average. Thus, there is no wave of events at the time of resumption, and so there is no elevation of response times. Not only is the difference between Event 1 and Event 7 greatly reduced, the difference is actually about 200 ms in the reverse direction.

Modeling the Task

Overview of the EPIC Architecture

The models for the task were constructed using the EPIC architecture for human cognition and performance, which provides a general framework for simulating a human interacting with an environment to accomplish a task. EPIC will not be described in full detail here. A more thorough description is presented in Kieras & Meyer (1997). In brief, EPIC resembles the Model Human Processor (Card, Moran, & Newell, 1983), but differs in that EPIC is an implemented

computational modeling system and incorporates more specific constraints synthesized from human performance literature. Figure 4 provides an overview of the architecture, showing perceptual and motor processor peripherals surrounding a cognitive processor; all of the processors run in parallel with each other. To model human performance of a task, the cognitive processor is programmed with production rules that implement a strategy for performing the task. When the simulation is run, the architecture generates the specific sequence of perceptual, cognitive, and motor events required to perform the task, within the constraints determined by the architecture and the interface. For example, the current orientation of the eye determines how visual events are detected and recognized, and movement times are governed by relationships such as Fitts' Law.

EPIC consists of a production-rule cognitive processor and perceptual-motor peripherals. To model human performance aspects of accomplishing a task, a cognitive strategy and perceptual-motor processing parameters must be specified. A cognitive strategy is represented as a set of production rules, much the same way that the ACT-R (Anderson & Lebiere, 1998), and Soar (Laird, Newell, & Rosenbaum, 1987; Newell, 1990) represent procedural knowledge. The simulation is driven by a description of the task environment that specifies aspects of the environment that would be directly observable to a human, such as what objects appear at what times, and how the environment changes based on EPIC's motor movements. EPIC computational models are generative in that the production rules only represent general procedural knowledge of the task, and when EPIC interacts with the task environment, EPIC generates a specific sequence of perceptual, cognitive, and motor activities required to perform each specific instance of the task.

As shown in Figure 4, information flows 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. All processors run independently and in parallel.

In short, EPIC is applied to a task as follows: The production-rule strategy directs the eyes to objects in the environment. The eyes have a resolving power which determines the processing time required for different object features, such as location and text. When information needed to determine the next motor movement arrives in working memory, the strategy instructs the ocular motor and manual motor processors to move the eyes and hands as required to complete that portion of the task.

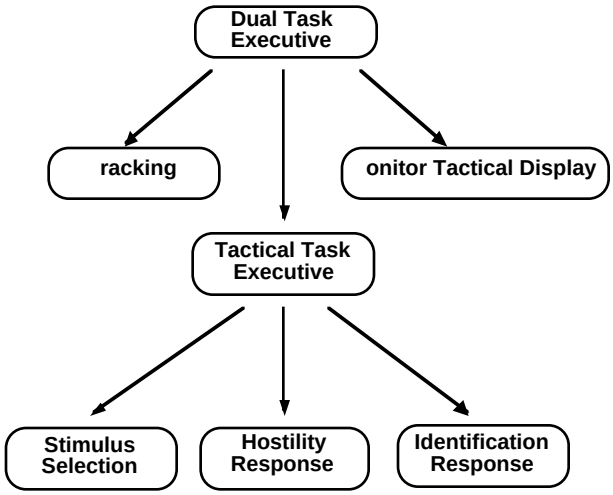

Figure 5. The hierarchy of strategy components used in the task model.

A single stimulus in the task environment can produce multiple outputs from a perceptual processor to be deposited in working memory at different times. First the detection of a perceptual event is sent, followed later by features that describe the event. The perceptual processors are "pipelined." If an object's features begin moving to working memory, the arrival of those features will not be delayed by any other processing. Working memory contains these items deposited by perceptual processors, as well as control information such as the current task goal. At the end of each simulated 50 ms cycle, EPIC fires all of the production rules whose conditions match the current contents of working memory. EPIC allows for parallel execution of production rules in the cognitive processor, and some parallelism in each motor processor.

Information processing and motor movement times are held constant across modeling efforts, and are based on human performance literature. The times for aimed manual movements, for example, are determined by Fitts' law.

A few additional key properties of EPIC can be mentioned; additional ones will be introduced as needed. The perceptual and cognitive processors are always operating in parallel, but it is up to the task strategy (the production rule programming) to take advantage of the parallel capabilities of the motor processors. Cognitive processing is multi-threaded, in that multiple sequences of production-rule execution can be underway simultaneously, which enables sophisticated models of complex multiple-task situations (Meyer & Kieras, 1997a,b; Kieras, Meyer, Ballas, & Lauber, in press).

Task Model Description

Task Strategy. To model the present task, a set of production rules were written to implement a task strategy; these rules are organized in a hierarchy shown in Figure 5 reflecting the overall structure of the task. Since EPIC allows fully multi-threaded cognitive processing, many of the tasks shown in the figure execute in parallel.

Dual-task executive. The top level is a dual-task executive process that controls execution of the tracking task, the tactical task, and a task that monitors the tactical display for color changes and other relevant events. When the tactical task is automated, the dual-task executive runs only the tracking task, effectively ignoring the tactical display entirely. When the auditory signal to resume the tactical task is recognized, the executive shuts down the tracking task, starts the monitoring process and initiates the tactical task process to handle the first event. Notice that the tactical task is controlled by its own sub-executive process. When the tactical task no longer has events to process, it terminates, and the dual-task executive restarts the tracking task. If the monitoring task then detects events such as a color change in peripheral vision, the dual-task executive will shut down the tracking task and restart the tactical task to handle the event. When the signal is made to return to automated mode, the dual-task executive will shut down the tactical task process if it is still underway, and then allow the tracking task to execute continuously and then waits for the next resumption signal.

Tracking task. The tracking task, when active, simply keeps the eye on the tracking target with one production rule and keeps the cursor on the target with another rule. This tracking rule commands the manual motor processor to execute an aimed joystick movement whenever the cursor is too far from the target. Further details of how this task is represented is not necessary for the present description. However, because the tracking task is shut down while tactical task processing is underway, the model predicts that no tracking movements will be made during tactical task processing. Evidence supporting this assumption is presented in Ballas, Kieras, Meyer, Brock, and Stroup (1999).

Monitoring process. The monitoring process is responsible for monitoring the state of the

tactical display, and labeling newly appeared objects in working memory as blips - candidate stimuli to be processed, and deciding when a tactical stimulus is present. These functions are only performed if the monitoring task is active, which is normally only when the tactical task is in manual mode. The dual-task executive is responsible for activating and deactivating the monitoring process.

The monitoring process posts a notice in working memory that a tactical stimulus is present if a blip has changed color, or has become "too close" to the ownship circle. The dual-task executive will respond to this notice and switch tasks. The model has three levels of closeness defined for when a blip is considered too close, whose specific values are free parameters set for a particular run of the simulation. The criterion level currently in effect can be changed by the dual-task executive.

It is assumed that the relevant visual properties are available in peripheral vision: a color change event will show as a change in luminance even if the actual colors can not be seen, and the location of blips relative to ownship is available for closeness judgments.

Tactical task executive. The tactical task sub-executive process coordinates three subprocesses: Stimulus selection first selects a blip for processing, then the hostility response process selects and produce the hostility designation response keystroke for the selected blip, and then the identification response process selects and executes the response that identifies the track, namely a keystroke for the target ID (track number). During the tactical task, the eyes must be moved around the display; each subprocess is responsible for moving the eye, under supervision of the tactical task executive. The tactical task executive allocates the eyes to each subprocess in turn, and waits for each subprocess to release the eyes before allocating them to the next subprocess (see Kieras, Meyer, Ballas, and Lauber, in press, for more discussion of executive strategies for resource allocation). The stimulus selection process moves the eyes from one blip to the next until an appropriate blip has been located. The hostility designation process leaves the eyes on the chosen stimulus until the hostility characteristics can be determined. Finally, the identification response process moves the eyes to the track number in order to acquire it for the target ID response. The stimulus selection process gets control of the eyes next to search for the next blip to process.

While logically the blip must first be selected and then the two responses made in order, it is possible for these three steps to be overlapped considerably as long as these basic response order constraints are satisfied for each blip. If the task strategy overlaps the processing heavily, then performance will be very fast; if not, performance will be substantially slower. Generally speaking, once basic perceptual and motor delays have had their effect, the performance speed depends primarily on the extent to which the task processes are overlapped by the task strategy. The tactical task executive coordinates the three subprocesses to control the extent to which the subprocesses overlap.

Stimulus selection process. The basic signal to process an event is that a blip changes color from black to red, blue, or amber. Stimulus selection chooses a blip to process, taking into account that if tracking is underway, the eyes have been kept on the tracking target, so most of the tactical display is in peripheral vision. While the color change can be noticed in peripheral vision, the color itself is not available except parafoveally. But once the eyes are on the tactical display, the colors for most of the blips will be available. Thus stimulus selection may have to examine more than one blip before finding one ready to process.

The stimulus selection process selects a stimulus that is the highest priority for processing. The rule is as follows, in descending order of priority: (1) colored blips (red, blue, or amber) because these have already changed and need to be responded to quickly; (2) a blip that has changed color

but whose color is not yet available; (3) blips whose color is unknown; (4) blips that are considered close to ownship. If more than one blip meets the rule for highest priority, one is chosen at random. The process then moves the eyes to the candidate blip and waits until both the eye movement is complete and the color of the blip is available. If the blip is red, blue, or amber, it becomes the selected stimulus and the process terminates. If the blip is black or white, the selection process starts over to choose another candidate blip using the priority rules. But if the candidate blip is both black and tagged as close to ownship, the eyes are kept on it, and the stimulus selection process (and the tactical task as a whole) will continue to run, waiting for some blip to change color. If the watched blip changes color, it becomes the selected stimulus. If some other blip changes color first, the selection process starts over and chooses it.

If the stimulus selection process successfully finds a blip that needs processing, it terminates with the chosen visual object tagged as the new stimulus and with the eyes already on it. If there is no blip suitable for processing, the stimulus selection process places a signal in working memory that no stimulus is available and terminates. The tactical task executive will terminate the tactical task, and the dual-task executive will then reactivate the tracking task.

Once stimulus selection places the eyes on the blip, the visual processing necessary to recognize the hostility behavior of the blip is underway while other processing is going on. The time required to recognize the behavior is on the order of a second, and was independently estimated for each type of blip.

Designation process. When the tactical task executive determines that the stimulus selection process has chosen a stimulus to work on, it places an item in working memory that identifies the stimulus visual object to the designation subprocess.

Since the eyes are already on the selected blip, the designation subprocess simply checks the color of the blip and responds with a series of production rules similar to those in our other models of choice reaction tasks (see Meyer & Kieras, 1997a,b). If the blip is red or blue, then the correct designation response is simply hostile or neutral and the corresponding keystroke is selected. However, if the blip is amber, the process must wait until the behavior of the blip has been visually recognized; the time required (on the order of a second) depends on the type of the blip, and was independently estimated. Note that the blip types and whether they were red, blue, or amber, were

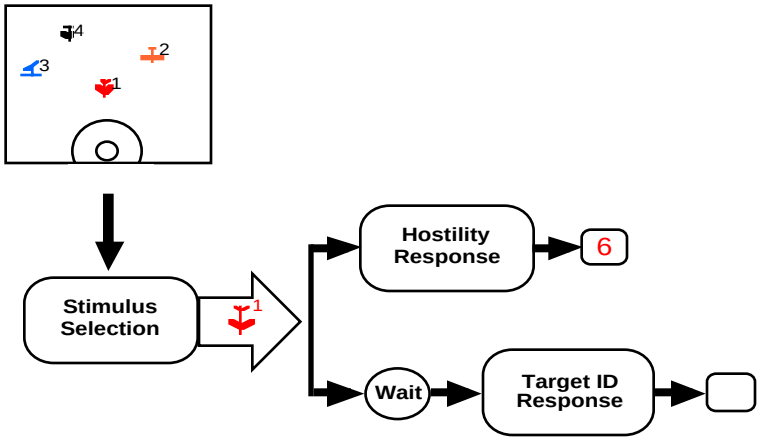

Figure 6. Illustration of the overlap relationship between the subprocesses of the tactical task. The hostility response processing can overlap in time with the identification response processing, but the task requires that the identification response be made second; this can be ensured by delaying the start of the identification response processing.

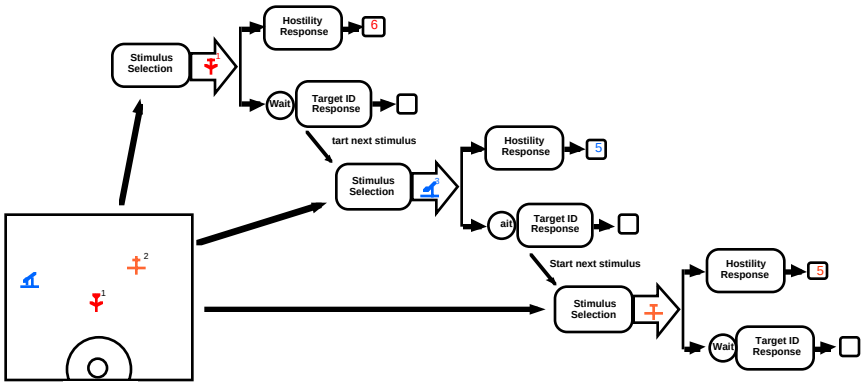

Figure 7. Selection of the next stimulus can overlap processing of the responses to the current stimulus. Thus several stimuli could be processed in succession more rapidly than the sums of the individual processing times.

varied non-uniformly over the events, and so the average designation time for an event depends on the exact mix of blip types and colors. The process then signals that it is done with the eye, selects the proper keystroke, commands the motor processor to execute the selected keystroke response, and finally terminates when the response is deemed complete.

Track identification process. The track identification process first moves the eyes to the track number, and waits until the track number is recognized. It signals that it is finished using the eye as soon as it has commanded the eyes to move to visual object for the track number. Under the EPIC architecture, the visual information for the track number will enter the visual processing "pipeline" and will eventually get recognized even if the eyes get moved shortly after they has foveated the track number. A sequential set of production rules then tests the recognized track number, in increasing order, and selects the corresponding keystroke response.

Motor overlapping. In both the designation and track identification subprocesses, a single production rule commands the manual motor processor to make the selected response keystroke. This rule follows the motor-overlapping rubric of including a test that the manual motor processor has completed the preparation of any previously-commanded response and is now ready to accept a command for a new movement; the preparation of this new movement might overlap the execution of the previously-commanded movement. This rubric permits response movements to be made in sequence at top speed without motor "jamming" or omission of a movement. Thus if the designation and track identification processes select their responses quickly enough, the responses can be produced at the maximum possible speed allowed by the architecture.

Eye allocation. Since this is a visually demanding task, how the eyes are handled is critical to determining task performance. Since the dual-task executive enforces mutual exclusion between the tracking task and the tactical task, the eyes can be controlled by only one of the two tasks at a time. If the tracking task is active, a tracking task production rule ensures that the eyes are kept on the tracking target; this rule is disabled when tracking is not active. When the tactical task is started, the tactical task executive first allocates the eyes to the stimulus selection process, then to designation, then to track identification, and then back to stimulus selection again. Each subprocess signals when it is finished using the eyes, so that the tactical task executive can allocate them to the next process in sequence. When the tactical task terminates, it effectively surrenders control of the eyes to the tracking task.

Overlapping tactical task subprocesses. These three tactical subprocesses can run in parallel, or overlap, to a considerable extent, as long as the task constraints are observed. As

shown in Figure 6, two successive responses must be made for a single track, and the responses must be made in the correct order. Thus the second response process must wait for the first response to be produced, but the duration of the wait need only be long enough to produce the correct ordering. Thus some of the second response processing could be concurrent with the selection and production of the first response. In addition, the stimulus selection process for the next stimulus could overlap with the second response process. Thus if several stimuli are processed rapidly in sequence, the processing could overlap substantially, as illustrated in Figure 7.

The models were built in such a way that the tactical task executive could dynamically control the amount of overlap in the tactical task subprocesses. The issues in such overlapping have a strong family resemblance to how computer Operating Systems are organized to allow prioritized concurrent task execution and manage the allocation of peripheral resources to different tasks so as to maximize system throughput (see Kieras, Meyer, Ballas, and Lauber, in press). In EPIC, the production rule subprocesses can all execute simultaneously, so the allocation management involves the peripheral resources of the eye and the hands. Of course, just as Operating Systems have different algorithms for scheduling processing and allocating resources, there are many possible executive task strategies within the limits imposed by the architecture and task structure. These limits will be described next.

Architectural limits on overlapping. The eyes must be moved from one blip to another, and from a blip to its associated track number, and then to the next blip to be inspected. At each point, the eyes must be kept on its target for at least the minimum time required for the visual system to acquire the information about a blip or track number. Two factors determine the timing of when eye movements can be commanded. First, visual information is pipelined through the visual system, so the eyes need not remain on an object for the full time the information is needed; rather they can be moved after a short dwell time and the visual information will still be recognized and passed on to the cognitive processor. Second, as in the other motor processors, an eye movement takes time to prepare prior to the actual movement. Thus a movement can be commanded somewhat in advance of the actual time when it needs to be made, allowing for very quick eye movements (cf. Hornoff & Kieras, 1997, 1999).

The models assume that the subject gets enough practice so that it is not necessary to look at the keypad, opening the way for considerable overlapping. As soon as the color or behavior of a blip has been recognized, the eyes can be moved to the track number while hostility response is selected and produced. As soon as track number has started into visual processing, control of the eyes can be passed to the stimulus selection process, which could actually be started even earlier to choose the next blip to process. Thus the next stimulus selection can be underway while the track id response is selected and produced.

EPIC assumes that while the hands can only be executing a single (possibly two-handed) movement at a time, the preparation for the next movement can be made during execution of the current movement, so it is possible to “pipeline” a series of movements through the manual motor processor at high speed. But on the other hand, the motor processor can only prepare one movement at a time. Thus if two concurrent processes each want to produce a manual response, the executive must exert control to make sure that the response commands are sent to the manual processor one at a time and in the right order.

Executive control of overlapping. The tactical task executive can control the extent to which these three subprocesses are overlapped, subject to the logical constraints on the overall task. It turns out that while the allocation of the eyes places strict constraints on the order of processing, it does not completely determine when the next process in order can be started. First, the designation process cannot begin executing until the eyes has been placed on a selected blip.

Thus no overlapping is possible between stimulus selection and designation response selection. However, once the designation information has been acquired, the eyes are free to be moved to the track number while the designation response is being produced. Once the process of recognizing the track number is underway, the eyes are then free to be used by the stimulus selection process to choose the next stimulus. The eyes can be allocated to the next subprocess well ahead of when the current subprocess completes, so the two response subprocesses are constrained only by the need to produce the responses in the correct order. The allocation of the eyes and the speed with which they can be moved determines the maximum extent of overlapping of the three processes. Below that maximum, the amount of overlap depends on whether the tactical task executive requires each subprocess to wait until the previous one is complete, or allows them to execute in parallel as much as possible.

The amount of overlapping between the three tactical subprocesses is controlled by *unlocking events*, a concept we had previously developed for the PRP task (Meyer & Kieras, 1997a,b). Unlocking events are the signals for when the next process is allowed to proceed in order to ensure that responses are made in the correct order. For example, an early unlocking event is the firing of the production rule that starts up the designation process. When the executive detects this event, it can start the track identification process. As soon as the track identification process gets control of the eye, it will move the eye to the track number and start its response selection processing as soon as the visual recognition of the track number is complete. These initial processing steps will happen simultaneously with most of the designation response selection and production. Thus this very early unlocking event means that the time required to start the track identification process and acquire the track number will be hidden by the completion of designation responding.

A very late unlocking event would be when the designation process finishes its response keystroke. The tactical task executive would thus delay the start of track identification until all of the designation response process was complete. All of the time associated with starting the track identification process will appear in the second RT, along with all of the time required to select and produce the designation response.

The unlocking delay between the designation process and the track identification process basically determines the inter-response-interval (IRI) of the two keystrokes for an event. Likewise, the unlocking delay between the track identification process and the stimulus selection process determines the IRI between the second keystroke for an event and the first keystroke for the next event. It is possible to change these IRIs over a wide range by choosing different unlocking events. In this work, early versus late unlocking events were always chosen to be events that occurred within the same process. The early unlocking event used in all the models reported here was that the current process was starting; the late events were either the commanding of a response movement, or the completion of a response movement.

The Bracketing Heuristic

The amount of overlapping enforced by the executive strategy has a major impact on task performance (Meyer & Kieras, 1999); “daring” strategies that maximize overlapping can produce substantially faster performance than more “conservative” strategies. This aspect of task strategy is not strongly determined by either the task requirements or the architecture, but rather is a result of factors such as the amount of practice, level of motivation, or long-term fatigue avoidance, or even the subject’s possibly haphazard efforts to formulate a task strategy (Kieras & Meyer 2000).

Kieras & Meyer (2000) pointed out that such optional aspects of task strategy can not be predicted reliably, but could only be identified by careful construction and evaluation of strategies that result in performance patterns that match the observed data in detail. Such models have been constructed for these data, but the task strategies they implement are quite complex: The subjects

can anticipate the workload based on the two-peak pattern repeated throughout the experiment, and dynamically modulate the extent of overlapping and whether close blips are watched in anticipation of color change.

Here we present a simpler, more robust, analysis based on the *bracketing heuristic* (Kieras & Meyer 2000). The concept is to start with a basic strategy for the task, such as the one just outlined, and permute it into two versions: a *fastest-possible* strategy that drives the architecture at the highest speed possible that still meets the basic task requirements, and a *slowest-reasonable* strategy that conforms to the task instructions and requirements with no "bells and whistles" to increase speed. Observed performance should fall somewhere between the performance predictions of models using these two strategies, which thus *bracket* the actual performance.

Bracketing is a way to construct truly predictive models in complex task domains where the optional strategy optimizations subjects would devise cannot be forecast. In addition, bracketing could guide the construction of models that match the data. However, bracketing can also be used to explain phenomena independently of the optional aspects of task strategies. Thus, bracketing models for the sound and no-sound conditions may be able to account for the important effects in a simple and straightforward way without the elaborate detail of models that match the data and subtle hypotheses about how subjects chose to handle the tasks.

The Bracketing Strategies

The bracketing heuristic has to be elaborated in the context of a multitask situation like the Ballas et al. (1992a, b) task — what is the meaning of "slowest-reasonable" and "fastest-possible" when there are two tasks that must compete for processing resources? We resolved this in terms of the relative priorities of the two tasks. In the experimental procedure, the tactical task was designated

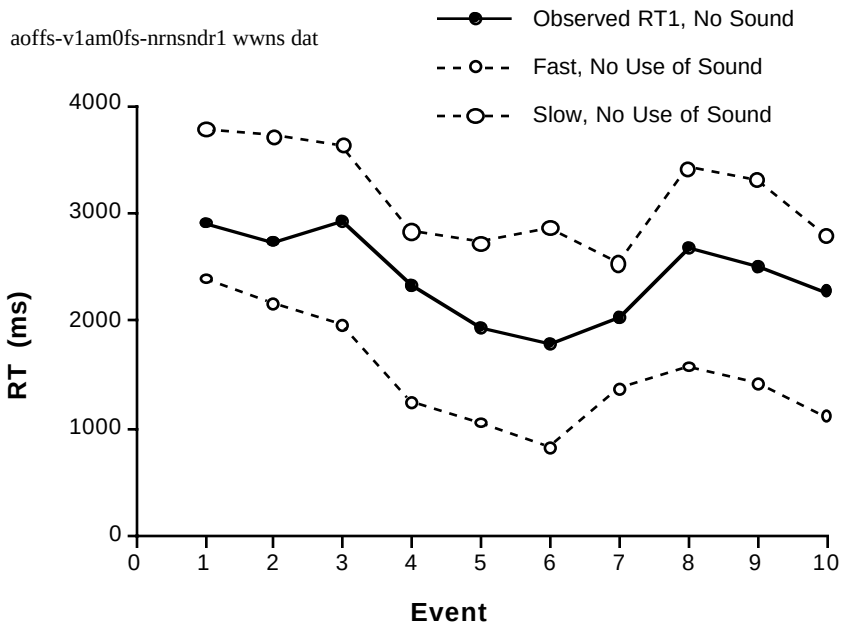

Figure 8. Observed RT1s for each event after task resumption from the no-sound condition described below (solid points and lines) compared to the bracketing models that perform the task without using any of the sound information. The fastest-possible model is the lower curve with small open points and dotted lines; the slowest-reasonable is the upper curve with the large open points and dotted lines. Note the automation deficit effect (Event 1 vs. Event 7) is present in both models and the data.

as the highest priority; fastest-possible thus means that the tactical task is executed as fast as possible regardless of the effect on the lower-priority tracking task. Slowest-reasonable means that the higher priority of the tactical task should honored, but no more so than the overall task instructions explicitly require.

The fastest possible and slowest reasonable models for the keypad interface had identical stimulus selection processes, and identical criteria for when the eye was deemed free to be used by the next subprocess. The fast model used a set of very early unlocking events, and the slow model used very late unlocking events.

Fastest-possible model. The fastest-possible task strategy corresponds to the most extreme interpretation of the task instructions. Because the tactical task supposedly has higher priority than the tracking task, this strategy ignores the tracking task if there is anything at all to be done on the tactical task. For example, if there is even a single blip on the tactical display, then the eyes are kept on it until it changes color, resulting in faster responding than if the eyes had been moved back to the tracking task display. This was implemented by simply setting the criterion for closeness to an extremely large value, so that all blips were considered close to ownship, and thus triggered the tactical task as soon as they appeared.

In addition, the tactical display is monitored at all times, even while the tactical task is automated, because this will speed up identifying relevant blips when the tactical task is resumed in manual mode. Furthermore, the fastest-possible strategy overlaps the three tactical task subprocesses as much as possible, using the early unlocking events. Thus the fastest possible model was always ready to respond to a tactical stimulus right away, and heavily overlapped the response and stimulus selection processes.

Slowest-reasonable model. The slowest-reasonable task strategy implements a nominal adherence to the task instructions. The instructions imply that the tracking task should be performed until a blip changes color in the tactical task, so under the slowest-reasonable strategy, there is no attempt to anticipate when the tactical task needs attention. Likewise, the instructions imply that when the tactical task is automated, there is no need to monitor the tactical display. Thus this strategy only monitored the display when the task was in manual mode, and only initiated the tactical task when a color change event was noticed. The tactical task terminated as soon as there was no longer a colored blip to process. The tactical task executive coordinated the three subprocesses by using late events during designation and track identification. These late events were when the corresponding keystroke was physically completed, producing a very slow and deliberate style of responding and having little or no overlapping for the three subprocesses of the tactical task — each response movement must be complete before the next step in processing for the tactical task began; no advantage is taken of the ability of the EPIC architecture to overlap motor movements. Thus the only overlapping is that provided by perceptual processing, which always runs in parallel with other processing.

Bracketing the Basic Performance

As an illustration of bracketing results, Figure 8 shows the predicted RT1s from the fast and slow models for data presented below when sound cues are neither present in the task nor used by the model, along with the observed RT1s from the no-sound condition. Throughout this paper, observed times are shown as solid plotting points and lines, and predicted times with open points and dotted lines. Note first that the predicted times indeed bracket the observed times; the slow model times are well above the observed, and the fast model times are substantially below. Also, both fast and slow models show an automation deficit effect in which the first few events take longer than the matching events during the second high-workload period.

Comparing the RTs for Events 7 and 8 relative to those for Events 5, 6, and 10, shows a general result that the second of a pair of closely spaced events has an elevated RT, especially for the slow model. The mechanisms in the model that produce this result are similar to those that produce the psychological refractory period (PRP) effect, a laboratory phenomenon obtained when two simple choice reaction tasks are overlapped in time. Basically, because the two events are closely spaced in time, the second event processing must wait for some part of the processing of the first event to be complete. If enough time intervenes between the events, the second response is not delayed because the first response processing will be complete. Extensively previous modeling work with EPIC (1997a,b) shows that the PRP effect is due either to conservative unnecessarily sequential task strategies, or to peripheral processing bottlenecks, such as the need to move the eye from the first to the second stimulus, or to responses having to queue up for control of the same motor processor. The fast model has less of this PRP-like effect because it overlaps processing very heavily. A similar effect would be expected to appear for the closely-spaced Events 1 and 2, but as discussed previously, the automation deficit effect can result in a substantially elevated time for Event 1, which can also spill over into Events 2 and 3, hiding the PRP-like effect.

Can Auditory Cues Reduce the Automation Deficit?

If the automation deficit is a result of the subject not knowing which blip to process first immediately after task resumption, then supplying a cue to this blip should alleviate the automation deficit. The visual display is ambiguous; once multiple blips are present and have changed colors, there is no information about which changed first. One approach would be to have the first-changed blip displayed in some attention-getting fashion such as flashing. But such approaches have the disadvantage of making a complex display more complex. Rather, using the auditory channel to signal the high priority blip represents a potentially valuable supplement to conventional computer interfaces, which have generally emphasized the visual modality and underutilized the auditory.

Ballas and his coworkers collected data in an version of the task that included localized sound cues. Since the blips appear at spatial locations on the display, the clearest way to single one of them out with an auditory cue would be provide a sound source located at the same apparent position. Additionally, the sound itself could indicate the type of the blip (fighter, missile, plane) which might help as well. As a brute-force attempt to determine whether a sound cue would overcome the automation deficit effect, this experiment used sound to provide both blip location and blip type information.

The basic design of the experiments will be summarized here; details of the experiment can be found in (Ballas, Brock, Stroup, Kieras, & Meyer, 1999). Subjects were first trained on the tactical task and performed extended practice with the two tasks in a *dual task* mode - a pair of scenarios which did not have the epoch structure discussed above, but simply had a series of tactical task events that had to be handled concurrently with tracking. Then the subjects performed the *adaptive automation* version of the task in which they alternated between handling the tactical task themselves versus letting the fictitious on-board computer handle it. One experiment had subjects performing this task as before with a certain scenario without sound cues. A second, later, experiment had subjects performing this task both with and without sound cues, but with two different scenarios. For purposes of comparing the detailed strategy effects of sound cues in the modeling work, it is essential to compare performance under the same scenario; unfortunately, this could only be done by comparing the sound condition from the second experiment with the no-sound condition from the first experiment that used the same scenario. Note that comparing sound and no-sound effects on a within-subject basis would require either exposing subjects to the same scenario twice, which would be suspect because subjects might recognize the event patterns, or

using two different scenarios, which would add considerable noise to the data since the specific event sequences in the scenario are extremely strong determiners of performance. However, the second experiment was planned as a within-subject design with two different scenarios, so getting a between-subject design on the same scenario required using two groups from two separate experiments. There is no compelling reason to consider the subjects in the two groups as being differently sampled, but the experiment is clearly not ideal in design. In the rest of the paper, these two groups of subjects and their data will be referred to as the *sound* condition (N = 13) and the *no-sound* condition (N = 11).

In preview, the experimental results suggest that supplying auditory cues produces a small uniform speed-up in performance, rather than a specific elimination of the automation deficit. This effect has a straightforward explanation in terms of the EPIC architecture and models for the task. However, the final effect sizes are quite small compared to the total variability produced by relatively small numbers of subjects performing a quite complex task under conditions of only moderate practice. Contributing to the smallness of the effect is the problem that no-sound subjects had more practice in the task than than the sound condition subjects, which might have reduced the benefits of sound. The individual differences are quite substantial, and the within-subject variability is quite large as well. In short, the benefits of providing sound cues are modest and similar in size to the corresponding effects produced in psychophysical experiments on auditory localization (e.g., Perrott, Saberi, Brown, & Strybel, 1990; Perrott, Toktam, Saberi, & Strybel, 1991).

But in the present results, these small effects are appearing in a complex decision-making and performance task, and so do not achieve conventional levels of significance with ordinary statistical methods. Using unusually sensitive statistical techniques suggests that the effects are systematic. Moreover, they appear to be consistent, both across the data set, and in comparison to similar auditory location effects reported in the literature. In addition, the effects can be accounted for in a straightforward way in the computational models, which argues that they should be taken seriously as a basis for future exploration on the use of the auditory modality.

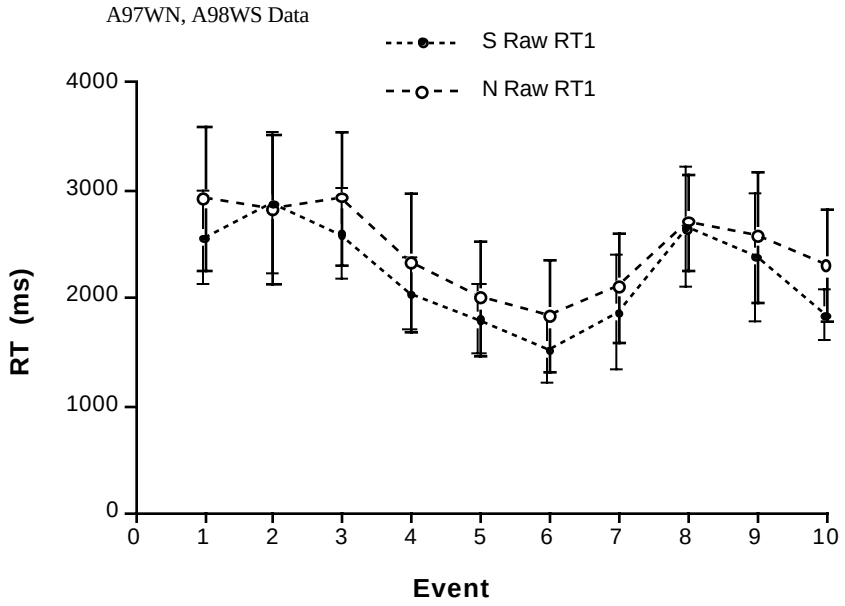

Figure 9. Mean RT1 for the sound (solid points) and no-sound conditions(open points). Approximate 95% confidence intervals are shown.

Empirical Results

Other work with data from this task has concluded that the appropriate way to deal with the two responses made to each event is to treat them in terms of the reaction time for the first response, RT1, and the delay between the first and the second response to the event, the inter-response-interval, or IRI. Generally, in the present results the IRI was not informative, so to save space, they will not be presented; all of the effects will be discussed in terms of the first response time, RT1.

Effects of Sound during Practice Sessions

A coarse look at the effects of sound comes from comparing performance on two of the training scenarios used in the experiments. These scenarios did not involve any automation and resumption of the tactical task; rather the subjects performed both the tracking and tactical tasks continuously for some time. Thus there is no task resumption sequence of events. Rather, the reaction times for each color-change event in the entire scenario are simply averaged together. One group performed the scenarios without sound, and had an average RT1 of 2051 ms; the other group with sound cuing had an average RT1 of 1867 ms. The difference is 184 ms, or a 9% reduction in favor of the sound cuing. However, this difference was not significant at conventional levels ($p > 0.2$) in a between-subjects ANOVA conducted on the mean RT1s for each subject in the two conditions.

Effects of Sound on Automation Deficit Epochs

The more interesting data concerns the automation deficit effect, namely the times to respond to each event following task resumption. Figure 9 shows the mean RT1s for each event for the groups with sound cues and no sound cues, based on the mean RT1 for each subject for each Event number. The error bars shown are approximate 95% confidence intervals computed for the 11 or 13 subject means underlying each plotted mean data point. While the variability is quite large, it is also clear that the mean for the sound condition is consistently lower on each event except for Events 2 and 8. The fact that the times are equal for the two groups at these two points is an argument that the generally lower times for the sound group was not simply a result of some overall gross difference between the two subject groups.

As would be expected from the large confidence intervals, when these data were subjected to an ANOVA on each subject's mean RT1 for each event number (made balanced at 11 subjects/group by dropping the fastest and slowest subject from the larger sound group), the effect of sound, and interaction of sound and event were clearly non-significant ($p > .3$). Of course, the effect of event was quite significant.

Effects of Sound Cuing on Individual Performance

The overall mean RT1s for the no sound and sound conditions, averaged across events, were 2449 and 2180 ms, respectively, a difference of 269 ms, similar to, and somewhat larger than, the effect size observed in the practice sessions. Inspection of the individual subject overall means suggested that the effect of sound was to speed up the slower subjects. That is, the slowest subjects with sound cuing were faster than the slowest subjects without sound, but there was little difference in the fastest subjects. To show this quantitatively, each group was split at its median. The average RT1 for the subjects below their group's median was 1730 ms for no-sound versus 1636 ms for sound, a difference of only 94 ms. In contrast, the average RT1 for subjects above their group's median was 3210 for no-sound versus 2780 for sound, a rather large difference of 430 ms.

Determining the statistical significance of the above-median difference in means is not simple, since under the null hypothesis of sampling subjects' mean RT1s from the same population, the expected difference between the above-median means is not zero, because of the unequal group size and the clearly non-symmetrical distributions. Rather than attempt to derive the distribution of this statistic under assumed population distributions, a powerful, but little-known, technique was used to assess the statistical reliability of the observed difference in above-median mean RT1s. This is Fisher's *method of randomization* (see Bradley, 1968, for a presentation). Basically, under the null hypothesis of sampling from identical populations, each subject's score is equally likely to turn up in one group or the other (the different sample sizes taken into account). Thus the distribution of the test statistic (the difference between the above-median mean RT1s) can be computed by listing all possible assignments of the scores to the two groups, computing the test statistic for each combination, and tabulating the relative frequencies of each value. The significance level of the actual observed result (a difference of 430 ms) is given by the proportion of computed values equal to or greater than the observed result. This technique has the virtues of producing an exact significance value without the possible distortions produced by assuming a normal or other approximation to the population distribution, and uses all of the metric information in the data, unlike the closely related tests based on ranks.

A simple computer program was used to generate all of the 2,496,144 possible data combinations and count the number whose test statistic exceeded the observed value. This yielded a significance level of 0.130952. As a check, another computation showed that the difference between the overall means of the two groups was significant at 0.200318, somewhat more significant than the routine ANOVA result. Thus, if one is willing to accept larger-than-customary probabilities of Type I error, these results suggest that the sound cuing did improve general performance, and did so primarily by helping the slower subjects perform faster.

Effects of Sound Cuing after Removing General Individual Differences

A final look at the data had two goals: one was to increase the sensitivity of the analysis of statistical reliability; the second was to put the data into a form more suitable for comparing to the computational models by removing general individual differences. The rationale for this transformation requires some discussion.

Testing strategy versus general individual differences. Most computational models of cognition, or cognitive architectures used in building such models, make strong statements about the strategy subjects use to perform the task, because this determines the type and sequence of perceptual, cognitive, and motor operations involved in performing the task. Often it is impractical to model each individual subject's strategy meaningfully, so the modeler simply devises some single strategy (or small number of strategies) that are believed to represent what the subjects generally do in the task. The extent to which the observed performance of the subjects departs from this assumed strategy is the critical measure of the accuracy of the model. This information is conveyed by the consistency with which each individual subject's data follows the pattern of performance times predicted by the strategy-following operation of the model.

In contrast to this strong commitment to the architecture and task strategy, most models and architectures make no statements about the range of performance produced by general individual differences. For example, even if subjects all followed the same strategy, some subjects might be basically faster than others, or more strongly influenced by task factors than others. To use a computer analogy, all subjects might be executing the same program, but individual subjects might have their own CPU clock speed, their own LTM access time, and so forth. Such differences would produce a wide range not just of average baseline performance times, but also a wide range of average effects produced by factors incorporated in the strategy - essentially the scale of the effects in the data.

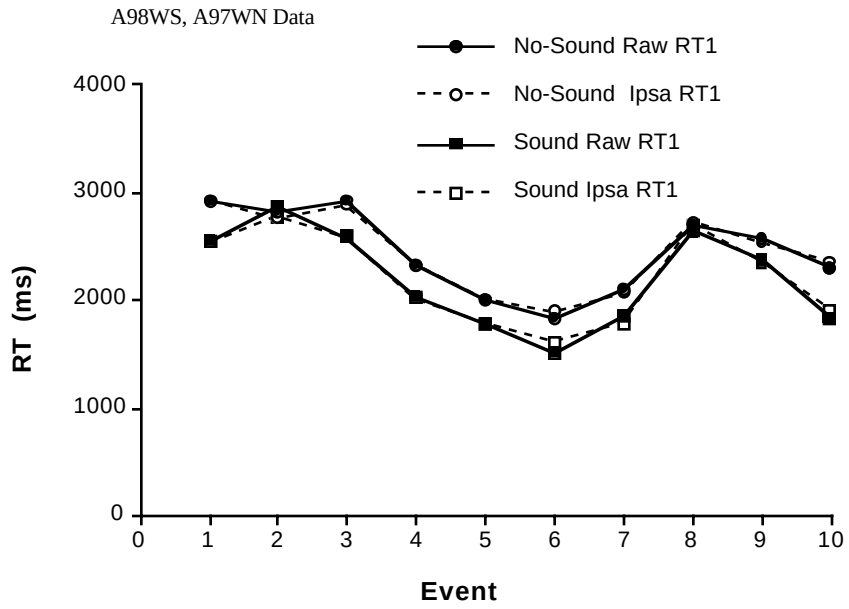

Figure 10. Comparison of Raw mean RT1s for the sound and no-sound conditions with the the mean of the transformed ipsatized values. The means are essentially unaffected by the transformation.

The architecture and models do not make any claims at all about the distribution of these differences in baseline or scale of performance times. Thus random variation between individual subjects' overall mean times and scale of times is simply not relevant to assessing the validity of the model. However, this irrelevant variation increases the total noise level in the data, making significance tests less diagnostic of the effects of interest, and comparisons to models less valid. Thus a clearer picture can be obtained for how well performance data from a group of subjects conforms to a strategy-following model by removing the irrelevant variability in the data (between-subject differences in baseline and scale) and leaving only the relevant variability (within-subject differences in due to noise and strategy variation). *Ipsatization* of the data, a technique used in individual-differences research, can perform this task in a straightforward way.

Ipsatization method. The ipsatization process puts all of the subjects on the same baseline and scale, thereby removing between-subjects variability, but leaves within-subject variability. The raw data, consisting of the mean RT1 for each subject on each event, were first replaced by within-subject z-scores calculated as follows: For subject *i*, the mean m_i and standard deviation s_i of his or her set of 10 $x_{i,j}$ RT1 values were calculated. Then for each score *j* for subject *i*, the RT1 value $x_{i,j}$ was replaced by the z-score having the value:

$$z_{i,j} = \frac{x_{i,j} - m_i}{s_i}$$

The resulting set of z-scores will have a mean of zero and standard deviation of one, meaning that all of the subjects now have the same baseline (mean) and scale (standard deviation). The remaining variability in the data directly reflects the extent to which the pattern of changes in RT1 is the same across subjects. For example, if all subjects have a large z-score for Event 1, and a small z-score for Event 7, then they all demonstrate an automation deficit effect that is independent of their individual differences in baseline and scale.

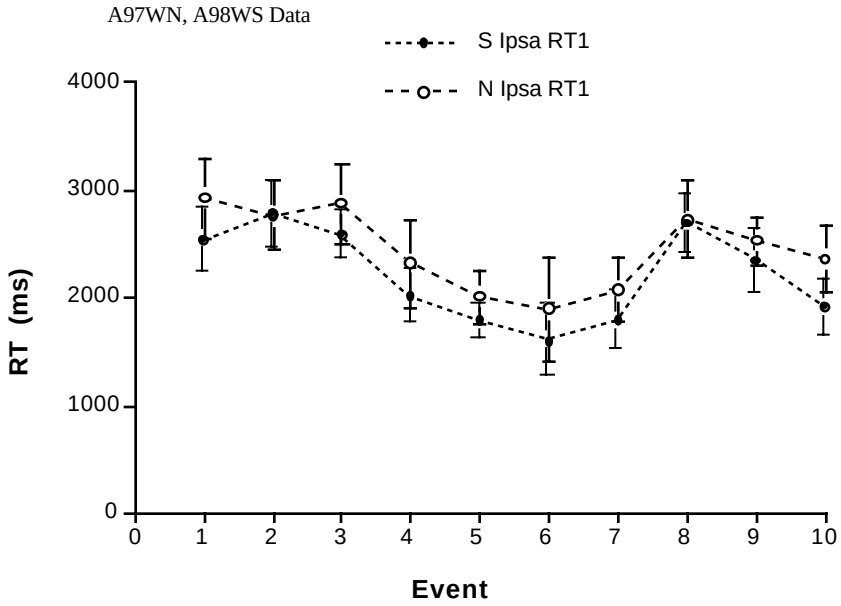

Figure 11. Mean RTs for the sound (solid circles) and no-sound (open circles conditions) after ipsatization transformation. Approximate 95% confidence intervals are shown which are based on the residual variation.

It is difficult to reason about effect sizes with this z-score form of the data. Thus, for convenience in plotting and working with the data, it was transformed back to the original domain of baseline and scale. The overall mean *M* and standard deviation *S* of the raw mean scores (mean RT1s for each subject on each event) was calculated separately for the no-sound and sound groups. This single pair of values for each group was then used to transform the individual z-scores back to same overall mean and variability shown in the previous plots for each group:

$$T_{i,j} = S \bullet z_{i,j} + M$$

The final result is a set of transformed scores in which the individual differences in between-subject baseline and scale have been removed, but the within-subject variability remains and has been normalized to the same value, and the mean difference between the two groups has been preserved.

Ipsatization results. Figure 10 compares the raw and ipsatized mean RT1s to show how the overall means are unaffected, thanks to the back-transformation of the z-scores to the original baseline and scale. Figure 11 compares the ipsatized data for the two groups with approximate 95% confidence intervals based on the variability of the scores underlying each plotted mean. Since these intervals are based only on within-subject variability, they are rather smaller than in the raw data (Figure 9), but they still reflect important variability in task performance. Overall, this plot bolsters the suggestion that the sound group generally performs faster than the no-sound group, with definite exceptions on Events 2 and 8, as mentioned above. The transformed data plotted in Figure 11 will be used in the modeling results reported below.

The transformed ipsatized scores can also be subjected to an ANOVA, but the results require careful interpretation. Dropping the same fastest and slowest subjects from the sound group as before to get equal sample sizes, the main effect of both event and sound was strongly significant ($p < .00$), but the interaction of event and sound was grossly non-significant. The no-sound/sound

difference appears significant in this analysis because the difference between the groups was first removed along with the other between-subjects variability, and then replaced as a constant difference between the groups in the back-transforming process. Since the within-subjects variability is not involved in testing this effect, the error term has thus been reduced to practically zero. (Note that an ANOVA on the z-score data would by definition produce no effect at all of the group factor, since all subjects would have a mean of zero.) However, it is useful to note that the lack of a significant interaction suggests that the effect of the sound difference is indeed uniform across the events; there is no evidence that the sound cue resulted in a fundamentally different task strategy, even when the sensitivity of the analysis to the sound effect was greatly enhanced by the ipsatization.

Despite the lack of a meaningful main effect in this last ANOVA, the difference between the two groups in Figure 11 relative to the confidence intervals based on within-subject variability, suggests that the sound group was consistently somewhat faster than the no-sound group, by a constant amount, with the exception of Event 2 and 8, as noted before. There is a hint that Event 1 is somewhat faster with sound present than without sound, relative to the time for subsequent events.Taking the difference between Event 1 and Event 7 as a basic measure of automation deficit, gives a difference of 824 ms for the no-sound group, and a difference of 698 ms for the sound group. This difference of only 124 ms does not approach significance. Thus there is no evidence for a difference in strategy produced by the availability of sound; rather there is simply a somewhat faster responding across the board. Although this effect is not significant at the normal levels of significance, it is still a useful exercise to see how an effect like this could be accounted for within a computational cognitive architecture.

Modeling the Effects of Localized Sound

Hypothesis 1: Localized Sound Guides Stimulus Selection

If sound helps to identify the relevant blip, then it should play a role in stimulus selection. More specifically, the stimulus selection process could give priority to a blip that is making a sound (an unambiguous indicator of the high-priority blip) rather than choosing only on the basis of the possibly ambiguous visual information. Following through on this hypothesis required that the EPIC architecture be able to associate a sound source at a location with a visual object at that same location. EPIC as originally developed (Kieras & Meyer, 1997) did not have an spatial representation in the auditory modality, so a significant change to the architecture had to be made. EPIC originally had an internal representation of visual objects arranged in space - in essence a computational representation of the notion of *object files* (cf. Treisman, 1988). When a production rule commanded the oculomotor processor to move the eyes to fixate one of these objects, the oculomotor processor would retrieve the location of the object from this internal spatial representation, and use it to program the eye movement to the object. A similar approach is used for aimed manual movements such as pointing, or the movement used to operate the joystick in the tracking task.

To accommodate auditory localization, the architecture was generalized: The objects in the object file representation were generalized to have not just visual properties, but auditory properties as well. If an object is emitting localized sound but is not visible, it will still be present in the object file spatial representation, with its location in space specified by the location of the sound source. The visual location, if available, is assumed to be more accurate and so dominates the auditory location information. Thus an object can now be selected based on its sound properties, and the

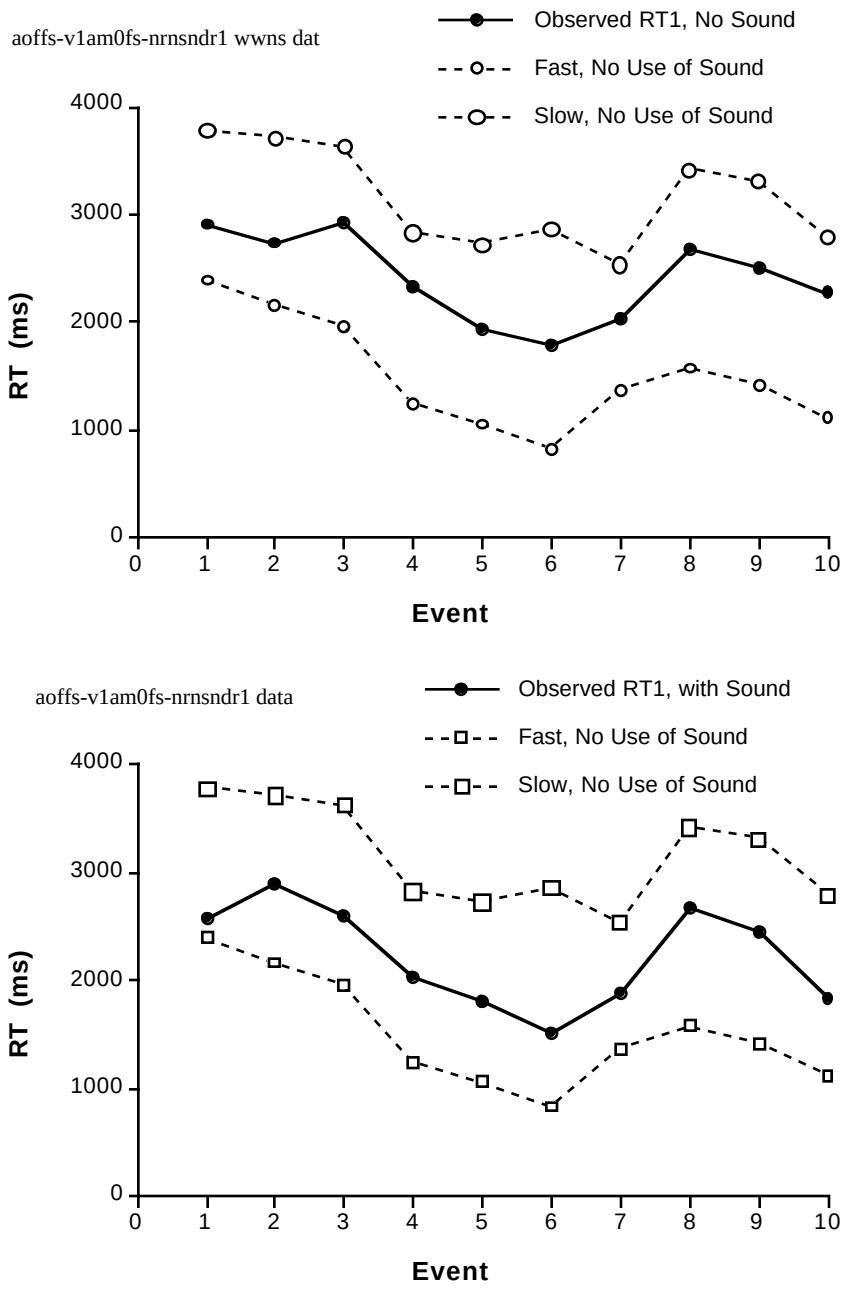

Figure 12. Comparison of the bracketing models that do not use sound with observed RT1s from both the no-sound condition (upper panel) and the sound condition (lower panel).

eyes moved to it using the sound location. This model will be called the *sound-selection* model.

The corresponding modification to the model strategy was limited to the stimulus selection rules. First priority was given to an object designated by sound over objects with a color, and the eyes immediately moved to it. Since the sounding object was always the highest priority blip, this approach guarantees that the blips would be inspected in correct priority order. This was the only change made to the model strategy.

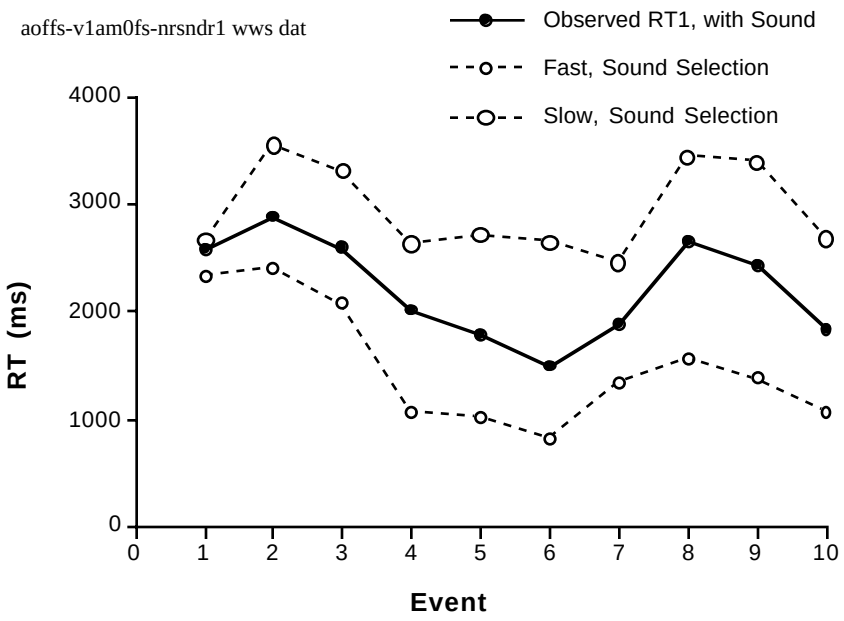

Figure 13. Observed RT1s from the sound condition compared to the sound-selection model in which the localized sound guides the search.

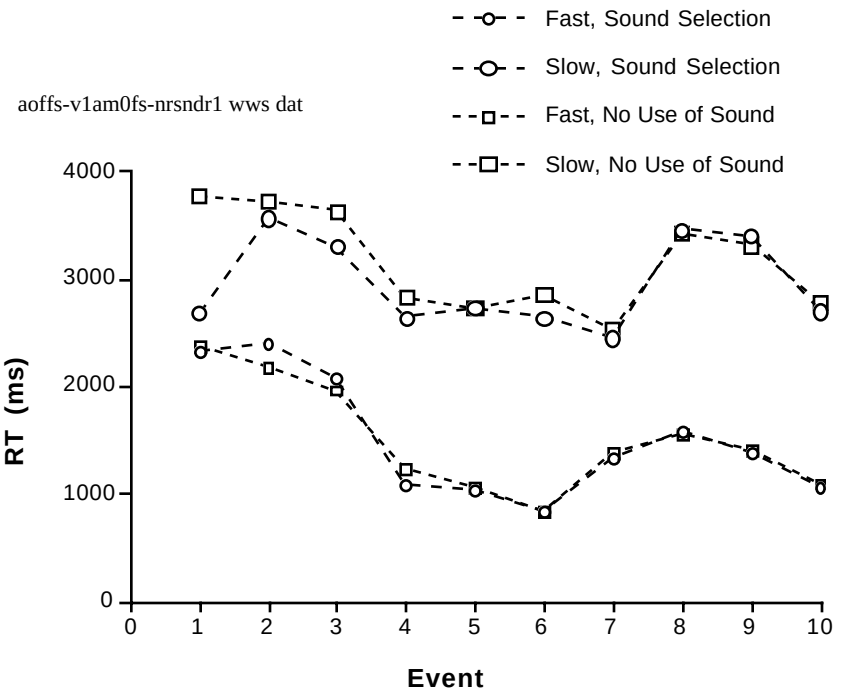

Figure 14. Comparison of the sound-selection bracketing models with the models that do not use sound. The only substantial difference in predicted RT1 is on the first event.

Results. The first question is how the sound and no-sound data compare to a model that does

not use sound. Accordingly, both panels of Figure 12 show predictions from the fast and slow models that do not use sound. The upper panel shows the data from the no-sound condition (and is identical to Figure 8), and the lower panel shows the data from the sound condition. The no-sound model still brackets the sound data, but since the sound data is somewhat faster than the no-sound data, it lies closer to the fast no-sound model. Relative to the model predictions, the sound and no-sound data follows the same pattern, but the sound data is closer to the fast model overall. The hinted-at smaller automation deficit for the sound condition shows up as an apparent decrease in the Event 1 time relative to the models in these plots. But the basic result is that both sound conditions are consistent (inside the brackets) with a model that does not use sound. This is a simple consequence of the fact that this model has a very wide bracket since it does not constrain the predicted data very much, and so a wide range of observed data would be consistent with it.

However, the sound-selection model does constrain the predicted data to a greater extent. The key result appears in Figure 13, which shows the sound data with the bracketing sound-selection models that use sound to guide stimulus selection. Both the slow and fast model are quite close to the observed value on Event 1. The use of the sound cue immediately directs the model to work on the correct first blip; the slow model performs quite rapidly at this point because most of the slow model strategy options affect what the model does while underway, and not so much how quickly it resumes the tactical task. Compared to the no-sound slow model, the sound-selection slow model demonstrates a huge reduction of automation deficit as represented by the Event 1 versus Event 7 difference. In contrast, the fast model does not change much between selection and no-sound versions, as is made clear in Figure 14.

Figure 14 compares just the model predictions. The fast model benefits not at all from the use of sound. The reason is that the fast model is monitoring the tactical display at all times, and when the tactical task is in manual mode, the fast model keeps the eyes on a blip in the tactical display as long as there is an unprocessed blip present. Thus a sound cue to the highest priority target is no benefit, because the eyes are probably already on the relevant blip. The slow model only benefits substantially from the sound cue on the Event 1, because there can be considerable uncertainty in the absence of sound about which blip is highest priority. Once the eyes are on the tactical display, as for Events 2, 3 and 4, again the sound cue is only slightly useful in distinguishing the correct blip. Thereafter, again the sound cue does not help distinguish the blips to any great extent because the tactical display is being monitored.

Conclusion. Thus the benefit of sound for stimulus selection appears to be quite limited in terms of the shape of the automation deficit profile - the very first blip can be selected more quickly if the subject is following something like the slow strategy, but that is the only substantial effect. This comparison does lend some credence to apparent but non-significant result in the data that sound produces a lower RT1 on Event 1 compared to the no-sound condition. If some of the subjects were following something like the slow strategy, they would benefit somewhat on Event 1, but as both the data and models suggest, sound really doesn't change the task strategy for subsequent events - both the fast and slow models are almost indistinguishable. While this explains the lack of a different pattern produced by sound cues in the subjects' performance, it does not explain the general, across-the-board, improvement in performance produced by the use of sound.

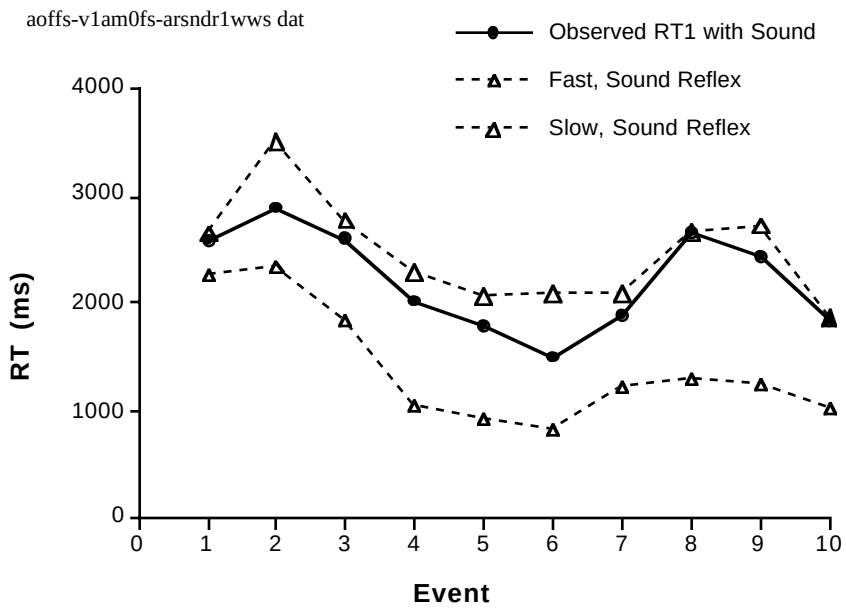

Figure 15. Observed RT1s from the sound condition compared to the bracketing models for the sound-reflex models. The slowest-reasonable model almost matches the data.

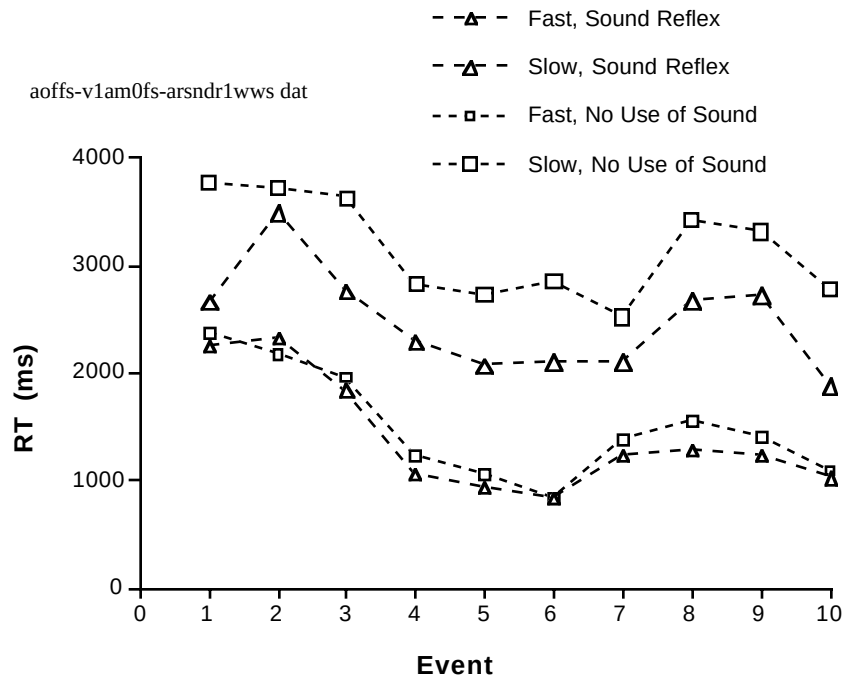

Figure 16. Comparison of predicted RT1s from the bracketing models that do not use sound with the sound-reflex models. The fast models perform identically, but the sound-reflex slow model is faster than the no-sound model.

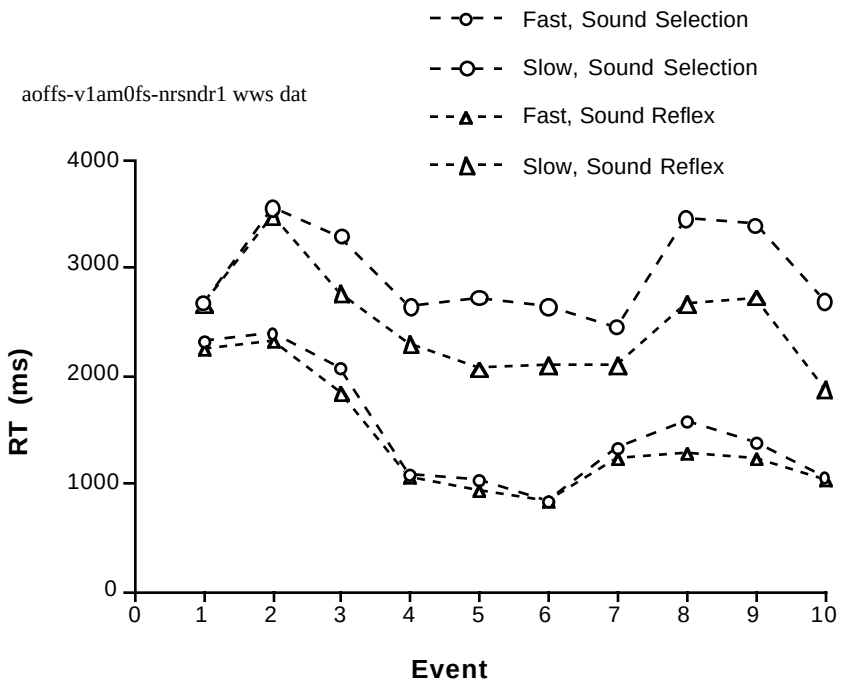

Figure 17. Comparison of the sound-selection and sound-reflex models. The fast models perform the same; after the first few events the slow sound-reflex model is faster than the slow sound-selection model.

Hypothesis 2: Localized Sound Triggers Eye Movements

The failure of the sound selection model to account for the general benefit of sound leads to a second hypothesis on the effect of localized sound in this task. Various experimental results show that providing localized sound improves reaction time in a visual choice reaction task. In typical such studies (e.g. Perrott, Saberi, Brown, & Strybel, 1990; Perrott, Toktam, Saberi, & Strybel, 1991), the subject is seated in the center of a spherical arrangement of small visual displays on which a choice reaction stimulus can be presented. Each display is mounted on a sound transducer, which produces a sound located at the same position as the display. The subject begins each trial looking at the display at the straight-ahead position. The sound and the visual stimulus are presented simultaneously. In the control condition, the sound always comes from the straight-ahead position, while the visual choice stimulus appears in various positions. In the experimental conditions, the sound comes from the same position as the visual stimulus. The results show a substantial benefit of the sound coming from the stimulus location, especially when the stimulus position is at a large angle from the straight-ahead position. But over ranges similar to the display in this study, Perrott et. al. (1990) observed effects of about the size observed in these data. For example, they reported benefit of localized sound of about 175 ms for targets within 10 degrees of fixation, and 200-500 ms for targets within 10 to 80 degrees.

Given the extreme simplicity of tasks such as Perrott et al., it seems unlikely that the sound cues are being used in some subtle way in the task strategy. Rather, it seems more likely that this facilitation is due to a basic orienting reflex - perhaps localized sound simply triggers an eye movement to the location of the sound source. Without the localized sound cue to the stimulus location, the task requires an extensive and slow visual search, involving head movements at large angles. With the cue, there is an immediate eye and head movement to the source of the sound, which then "automatically" places the eyes on the visual stimulus.

Investigating this hypothesis required an additional architecture change. The EPIC architecture already included reflexive eye movements made in response to sudden visual onsets or movements of visual objects. Only a minor change was required to have a sound onset trigger a reflexive eye movement as well. The same dynamic timing characteristics were used as in the visually-triggered reflex, and also similarly, the reflex response can be enabled or disabled by production rule actions. For simplicity, the sound-selection model strategy of giving priority to a sounding blip was still used; the oculomotor reflex will already have started the eye on its way to the sounding blip when the stimulus selection strategy redundantly (but with no significant loss of time) decides to move the eye there. Thus the cuing benefit of the sound can be combined with its orienting reflex properties, and the same model strategy will perform correctly even in the absence of sound. The resulting model is called the *sound-reflex* model.

Results. Figure 15 shows the observed RT1s in the sound condition compared to the sound-reflex bracketing models. In marked contrast to the sound-selection and no-sound models, these models bracket the data much more tightly; the slow model is facilitated by the sound reflex so much as to essentially match the observed mean RTs at several points in the data. Thus, even if subjects were following a strategy much like the slowest-reasonable, their performance would be similar to that predicted by the corresponding model.

Figure 16 compares the fast and slow predictions from the sound-reflex model with the no-sound model. Again the fast models are not affected by the use of sound, but the slow sound-reflex model not only has a substantially faster Event 1 time, but is also generally faster than the no-sound model. This is a remarkable performance facilitation to be produced by such a simple architectural mechanism.

Figure 17 shows the comparison between the sound-selection and sound-reflex models. Again the fast models are basically indistinguishable. Because the task resumption delays and PRP-like effects render the faster reflex eye movements irrelevant at the beginning, the slow sound-selection model performs the first events as quickly as the slow sound-reflex models. But thereafter, the eye movement reflex allows the the sound-reflex model to respond generally faster than the sound-selection model, which must "decide" where to move the eyes.

Conclusions. The overall implications of these results lies in where subjects' strategies stand in relation to the fastest-possible and slowest-reasonable models. In general, the group of subjects is assumed to produce performance somewhere between the fastest-possible and slowest-reasonable models, either in choosing their own individual trial-by-trial strategy, or their own individual whole-experiment strategy. In no case is their aggregate performance as fast as the fastest-possible model, meaning that on the whole they are definitely following some slowest-possible strategy options, such as attempting the tracking task between tactical task events. A single-strategy model could be devised that contains a judicious mixture of slowest-reasonable and fastest-possible features. Such a model could be easily constructed simply by minor performance enhancements to the slowest-reasonable sound-reflex model to minimize the PRP-like effect in the early events and anticipate the color-change somewhat for the middle events.

But for present purposes, the question is how the localized sound produces a benefit of a relative small but consistent size. Only the sound-reflex model both gets close to the observed performance level and produces a consistent benefit of sound. Thus, the sound-reflex hypothesis appears to be the correct explanation for the benefits of sound in these data.

General Conclusions

Based on the modeling results, localized sound can be of value in a complex task in at least two ways: one is by removing ambiguity in which item on a visual display should be selected for processing; the other is by triggering a reflexive eye movement to a location of interest in the environment. In the modeled task, the reflex function appears to be more important and pervasive than the selection function.

The substantive force of these results are limited due to the fundamental problems in this data set, as previously discussed. However, the basic conclusion about the role of sound cuing as a trigger for eye movements is justified both from related empirical work, the consistency of its appearance in these data, and the simplicity of its explanation in terms of the EPIC architecture.

There are some key methodological lessons to be learned from this work as well. The basic one is that attempting to demonstrate effects of low-level processing mechanisms in a complex task is fraught with empirical difficulties that are easy to underestimate. Traditional notions of good experimental design that increase the sensitivity of the data are hard to apply in complex tasks in which we seek to characterize human performance in detail. But in defense of the present work, it should be noted that first, multi-modal interfaces have so much intuitive appeal that taking risks on exploring them empirically is justified, and second, the present sound-cuing experiment was deliberately designed as a "kitchen sink" study that used the sound cuing as fully as it seemed possible to do so. If we had predictions that the effects of sound would be so small, they would simply have been unbelievable given the intuitive expectations, and the experimental results would have had to be collected anyway.

The work makes two positive methodological contributions. The first is a demonstration that model construction can help pull interesting and useful results out of data in which the task complexity threatens to overwhelm a small effect. The model essentially accounts for much of the complex variation produced by the task and its complicated stimuli; the residuals remaining after this account indicate the presence of small effects that in turn can be accounted for by carefully elaborated additions to the model. This approach of using models to isolate small effects in complex tasks might be a valuable strategy in future research.

A second positive methodological result is the value of bracketing models in reasoning about the theoretical implications of a set of data. As argued in Kieras & Meyer (2000), constructing a single model that closely fits the data is both difficult and relatively under-determined, whereas bracketing models can be constructed fairly rapidly and consistently. This work, along with the similar work on other data from this task (Kieras, Ballas, & Meyer, 2001) makes a valuable contribution toward the successful and efficient use of cognitive modeling by showing how conclusions can be drawn about possible mechanisms just by comparing bracketing models with each other as well as with the data.

Acknowledgement

This research was supported by the Office of Naval Research under Grant No. N00014-96-1-0467 to the first author

References

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

Ballas, J., Brock, D. Stroup, J. Kieras, D. and Meyer, D. (1999) Cueing of Display Objects by 3-D Audio to Reduce Automation Deficit. In *Proceedings of the Fourth Annual Symposium and Exhibition on Situational Awareness in the Tactical Air Environment*. The Naval Air Warfare Center, Patuxent River, MD, June 8-9, 1999, pp 100-110.

Ballas, J. A., Heitmeyer, C. L., & Perez, M. A. (1992). *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. (1992). *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.

Ballas, J., Kieras, D., Meyer, D., Brock, D. and Stroup, J., (1999) How is Tracking Affected by Actions on Another Task? In *Proceedings of the 10th International Symposium on Aviation Psychology,* May 3-6, 1999, Columbus, Ohio.

Bradley, J.V. (1968). *Distribution-free statistical tests.* Englewood Cliffs: New Jersey: Prentice-Hall.

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

Hornof, A. J., & Kieras, D. E. (1997). Cognitive modeling reveals menu search is both random and systematic. *Proceedings of ACM CHI 97: Conference on Human Factors in Computing Systems,* New York: ACM, 107-114.

Hornof, A. J., & Kieras, D. E. (1999). Cognitive modeling demonstrates how people use anticipated location knowledge of menu items. *Proceedings of ACM CHI 99: Conference on Human Factors in Computing Systems,* New York: ACM, 410-417.

Kieras, D.E., Ballas, J. A., & Meyer, D.E. (2001). Towards demystification of direct manipulation: Cognitive modeling charts the gulf of execution. *Proceedings of ACM CHI 2001: Conference on Human Factors in Computing Systems,* New York: ACM.

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

Kieras, D. E., & Meyer, D. E. (2000). The role of cognitive task analysis in the application of predictive models of human performance. In J. M. C. Schraagen, S. E. Chipman, & V. L. Shalin (Eds.), *Cognitive task analysis.* Mahwah, NJ: Lawrence Erlbaum, 2000.

Kieras, D.E., Meyer, D.E., Ballas, J.A., Lauber, E.J. (in press) Modern computational perspectives on executive mental processes and cognitive control. Where to from here? In S. Monsell and J. Driver (Eds.), *Control of cognitive processes: Attention and Performance XVIII*. Cambridge, MA: MIT Press.

Laird, J. E., Newell, A., and Rosenbloom, P.S. (1987) Soar: An architecture for general intelligence. *Artificial Intelligence*, **33**, 1-64.

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

Meyer, D. E., & Kieras, D. E. (1997b). A computational theory of executive control processes and human multiple-task performance: Part 2. Accounts of Psychological Refractory-Period Phenomena. *Psychological Review*. **104**, 749-791.

Meyer, D. E., & Kieras, D. E. (1999). Precis to a practical unified theory of cognition and action: Some lessons from computational modeling of human multiple-task performance. In D. Gopher & A. Koriat (Eds.), *Attention and Performance XVII*.(pp. 15-88) Cambridge, MA: M.I.T. Press.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Perrott, D.R., Saberi, K., Brown, K., & Strybel, T. (1990). Auditory psychomotor coordination and visual search behavior. Perception and Psychophysics, 48, 214-226.

Perrott, D.R., Toktam, S., Saberi, K, & Strybel, T. (1991). Aurally aided visual search in the central visual field: Effects of visual load and visual enhancement of the target. Human Factors, 33(4), 389-400.

Treisman, A. M. (1988). Features and objects: The fourteenth Bartlett memorial lecture. *Quarterly Journal of Experimental Psychology*, *40A*, 201-237.