

AN ARCHITECTURE AND MODEL FOR COGNITIVE ENGINEERING SIMULATION ANALYSIS: APPLICATION TO ADVANCED AVIATION AUTOMATION

Kevin M. Corker
NASA-Ames Research Center
Moffett Field, CA 94035-1000

Barry R. Smith
US Army Aeroflightdynamics Directorate
Moffett Field, CA 94035-1000

Abstract

The process of designing crew stations for large-scale, complex automated systems is made difficult because of the flexibility of roles that the crew can assume, and by the rapid rate at which system designs become fixed. Modern cockpit automation frequently involves multiple layers of control and display technology in which human operators must exercise equipment in augmented, supervisory, and fully automated control modes. In this context, we maintain that effective human-centered design is dependent on adequate models of human/system performance in which representations of the equipment, the human operator(s), and the mission tasks are available to designers for manipulation and modification. The joint Army-NASA Aircrew/Aircraft Integration (A³I) Program, with its attendant Man-machine Integration Design and Analysis System (MIDAS), was initiated to meet this challenge. MIDAS provides designers with a test bed for analyzing human-system integration in an environment in which both cognitive human function and "intelligent" machine function are described in similar terms. This distributed object-oriented simulation system, its architecture and assumptions, and our experiences from its application in advanced aviation crew stations are described.

Introduction

The A³I Program was initiated in 1985 to support exploration of computational representations of human-machine performance to aid designer's of crew systems. The major product of this effort is a human factors computer-aided-engineering system called MIDAS (Man-machine Integration Design and Analysis System). MIDAS is intended to revise the system design process in order to place more accurate information into the hands of the designers early in the process of human engineering design so that the impact and cost of changes are minimal. It is also intended to identify and model human/automation interactions with flexible representations of human-machine function. The crew station development process, as it is currently undertaken, is illustrated in Figure 1. The design proceeds from requirements and capabilities in conceptual

design, through increasing specification to hardware and software prototypes and simulation tests. Human performance evaluation occurs after prototype design and development. Results from testing the prototype are then used to guide prototype redesign.

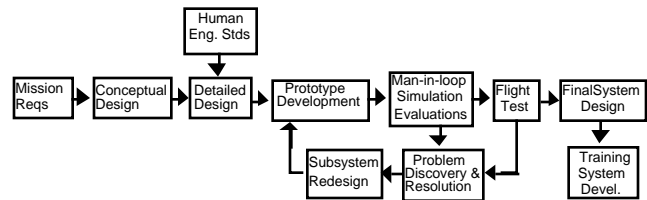


Figure 1. Current Crew Station Development Process

While later, detailed-design phases in the traditional process have "hard" data to guide development, the most advantageous time for making improvements to a system's design is early in the development process. In fact, studies have shown that while less than 15% of a system's cost is incurred during the concept exploration, demonstration and validation phases, 70-85% of the actual life-cycle cost is dictated by the decisions made in these early stages of development¹.

The traditional design process often considers ergonomics late; this often results in costly revisions required to solve human factors problems. The time-consuming and expensive process of hardware simulation development results in the technology of system development that does not take advantage of the human factors empirical data and knowledge bases. Finally, the training system requirements are considered last, completely decoupling them from the design process. MIDAS integrates the design process by using human performance models in the conceptual design phases of system development. Crew station development enabled by the A³I methodology is illustrated in Figure 2.

In this process human performance considerations are accounted for early in the designs and played out for evaluation in the simulation mode. Iteration in this mode is flexible and timely. The flow then proceeds with a refined design into the standard design and prototype development phases.

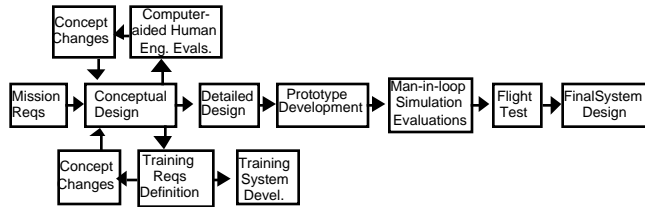


Figure 2. Revised Crew Station Development Process

MIDAS provides a prototyping test-bench, based on human performance models. Designers can work with computational representations of the crew station and human operators, rather than relying solely on hardware simulators and man-in-the-loop studies, to discover problems and ask "what-if" questions regarding the projected mission, equipment, and environment. The advantages of this approach are found in reduced development time, reduced development costs, early identification of human performance limits, and support for the integration of training system requirements and development.

In addition to its use in development and design, MIDAS offers a structure or framework in which to test and implement models of human cognition. The MIDAS framework systematizes and unifies the interaction of human performance representations in a common structure and with a common language for interaction. The representation is a tightly linked set of computational descriptions of the elemental aspects of human performance. Models of human performance from perception through cognition and action are implemented within this framework. The interplay of the models produces simulations of behavior.

While the MIDAS framework supports models that produce simulations of human behavior, the structure is not an embodiment of a "unified theory of cognition", perception, or action in the terms specified by Newell². Newell's definition of a unified theory specified a "single set of mechanisms for all of cognitive behavior" (pg. 15). The range of behavior that the single set of mechanisms is required to cover are shared, in part by the MIDAS system, i.e., decision making, memory, perception and motor behavior, routine action and skill. Other behaviors of concern to Newell, e.g., problem solving, learning, language, motivation, emotion, an imagining, are beyond the scope of the MIDAS effort. MIDAS exhibits some of the characteristics of good theory. It provides systematic conceptual coherence for examining human performance. It supports (but does not generate) models that are systematically refinable, and that produce behavior that is verifiable relative to human performance of similar tasks in similar environments. The MIDAS system does not make

"structural" assumptions about human performance representation in the sense that it can be considered "unified". The system does make "functional" requirements for the generation of human performance explicit and supports multiple model formalisms in meeting those functional requirements. In this sense the MIDAS system is more aligned with Newell's definition of cognitive "framework" than it is with theory.

Changing the design process in this way is not a simple task. Our dependency on human performance models is subject to the limitations described by Wickens:³

"Unfortunately, it appears that those models which have the most precise quantitative formulation and have received the strongest empirical validation, have been derived in domains that may be furthest removed from [complex, heterogeneous task environments]...." (page 180)

MIDAS Approach

The MIDAS system is a set of interactive and integrated software modules, editors, and analytic simulation tools. The system is structured using an object-oriented software paradigm with an architecture based in agent-actors theory.^{4,5,6} The requirements that compel the MIDAS approach are discussed in the next section. Our response to these requirements is detailed in the subsequent two sections in a detailed discussion of the human performance models that animate the simulation. Finally, a discussion of applying the MIDAS system to several operational domains serves as the conclusion of this paper.

Analysis System Requirements

The basic goal of the MIDAS project is to facilitate crew station design for advanced automation by a computational simulation. In order to provide efficient, effective, and accurate representation of the human/system performance, that simulation system entails the following elements:

- a set of tools and direct-manipulation interfaces that allow efficient examination and modification of the elements of the simulation system.
- a common language for representation of: (i) the components of the aviation system (aircraft dynamics, sensors, displays, and control systems), (ii) the human operators who act through and interact with the advanced aircraft system, and (iii) the task environment in which that action takes place.

- a mechanism for propagating the impact of manipulations in the scenario, equipment, or operator profiles to downstream mission performance. Meeting this need requires the simulation to support inferences as to the effect of system or procedural modification.

We have found that the structure of a simulation system which supports the aforementioned elements has the following characteristics:

- **Modifiability and Manipulability.** It is essential that the analytic simulation support modification and manipulation of its fundamental components. The basic mode of operation for the analysis system is to explore the impact of changes to the baseline design. Thus, the capability for systematic change is critical. Of equal importance is system extensibility. To be generally useful, the analytic aid should be applicable to many types of design changes, and to many operational domains. We have designed the MIDAS architecture so that extensions of this type can be implemented with minimal disruption to the existing core system.

- **Examinability.** The analysis system must provide designers with explicit and examinable reference to the rules, decision making strategies, heuristics, and assumptions under which the man/machine system is operating, as well as to operational measures of the effectiveness of a particular design. For example, at any point in the simulation a designer should be able to examine the cognitive state of each of the human operators, the rules that are being used to guide their

behavior, and their nominal workload. The designer should also be able to perform sensitivity analyses on critical parameters of the human-machine system. This includes examining the elemental behavioral models for the human operator. Similarly, the state of equipment or mission progress should be able to be probed in order to relate the system state to the operator's performance.

- **Dynamic Analysis Capability.** The simulation system must produce a stream of behavior in the form of dynamic timelines describing not only its state and structure, but also dynamic sequences of action and contingent responses of the human/system behavior. In order to be an effective tool, the system must support testable hypotheses. Designers must be able to analyze the events occurring in a simulation scenario, and relate this performance to part-task or full-mission simulation data. In MIDAS each action taken, decision made, and communication event is logged by the analysis system.

System Architecture

There are two basic perspectives on the MIDAS system architecture that need to be provided in order to capture the system's utility. These are the functional and structural bases of the system.

Functional Architecture:

The functional view of the full MIDAS system and its components is provided in Figure 3.

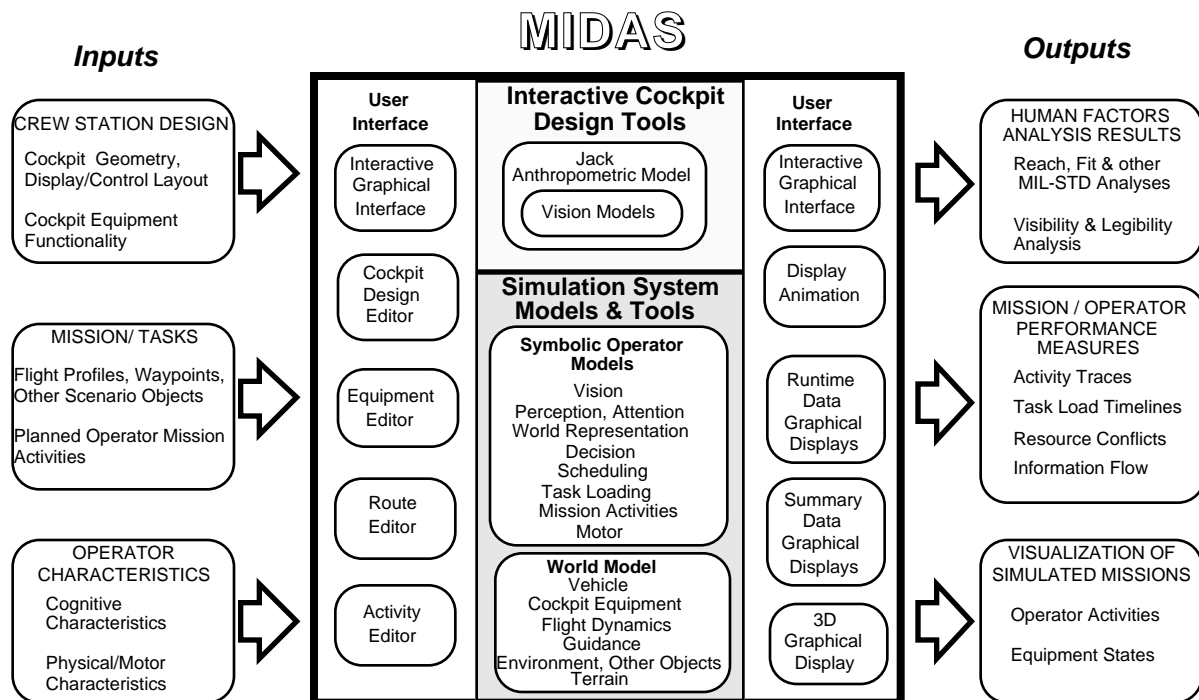


Figure 3. Functional View of the MIDAS System

MIDAS is intended to operate in three modes. First, there is a **Specification Mode** in which users, i.e., designers of prototype systems, are provided with a set of tools to specify operator characteristics, mission characteristics, and characteristics of the physical plant or crew-station. Activity, equipment, and mission editors, as well as rule-base and human performance model editing are also supported. Once the system to be examined is specified, MIDAS supports the designer in two paths for analysis. The first path is termed **Interactive Mode**. This functional mode supports the use of scenario-independent layout and animation of displays and controls for assessments of visibility and legibility, examination of anthropometric characteristics, and analyses of cockpit topology and configuration. The output of these analyses are information of the type that has been traditionally considered human engineering data for comparison to usability standards, e.g. MIL-STD-1472. The other analysis path for MIDAS users to take is that of dynamic simulation.

The **Simulation Mode** provides facilities whereby the specifications of the human operator, the equipment (including vehicle aerodynamics and guidance functions) and the mission goals can be run together. This simulation mode exercises all of the system and the elemental human performance models in an integral process that results in activity traces, task load timelines, information flow analyses, as well as mission performance measures. Operator measures enable identification of significant human performance variables, such as potential resource conflicts, task-loading as a function of configuration/mission, and information requirements. Mission performance measures include operational timelines, routes, contingencies, and mission effectiveness metrics.

Structural Architecture:

The MIDAS system is fundamentally built on object-oriented software structures. Object-oriented programming paradigms were originally created to facilitate the creation of computer simulation systems, and they continue to be the best means for implementing event-focused modular computer simulations. An object-oriented simulation is composed of objects, software entities with local state. An object's state (the description of its parameters and defining values) is stored as values in structures called slots. The slot values of an object comprise the specification of its attributes. For example, in the case of a physical object, shape, size, and position values are typical descriptors. Object-oriented software also provides for procedures or methods that can operate on some, or all, of an object's slot values. Objects can be organized into higher level structures, or groupings, called classes, defined by

common characteristics. Objects interact with each other by exchanging messages. The way an object responds to messages and the way it alters its slots depends on an object's class. In an object-oriented simulation system, it is frequently the case that the description of a class of objects inherits the description of other classes. For example, in the MIDAS system the human-operator class and the flight-maintenance-computer class include the characteristics of the class intelligent-agent. The behavior of an intelligent agent, once described, can then be incorporated every time a class of object embodying intelligent behavior is described.

Agent Architecture:

The simulation is further organized by the imposition of an architecture that provides a framework for the interaction among all of the objects that comprise a MIDAS simulation. The interaction of these elements of MIDAS is provided through an "agent" architecture, designed to provide a basis for multiple, concurrent, and independent intelligent agents.

We have cited system requirements of modularity, extensibility, perspicuity, and analytic tractability. The agent architecture allows us to meet these requirements. Further, the agent architecture provides a rigorous syntax for communication among system components, and a uniform description of the objects that comprise the simulation system. Agents in MIDAS all share a common structure, illustrated in Figure 4.

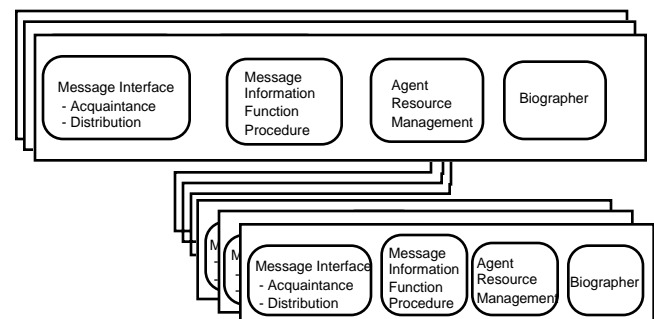


Figure 4. MIDAS Agent Structure

Each agent is composed of a message interface that determines from which and to which other agents messages can be received. The agent has a function, method, or procedure it performs upon receipt of appropriate messages from other agents or at specified simulation times. Agents can also have the capability to manage resources associated with their own action (e.g., the sufficiency or availability of information necessary to support their computation or function). Finally, each

agent has a biographer that tracks all of the messages received, operations performed and messages sent for any given agent. The biographer function is critical to the examinability of MIDAS. The action of the biographer can be controlled (turned on or off) by simulation time-tags, by events, or by the user of the MIDAS system.

MIDAS Agent Content

The MIDAS system is composed of the agents described below. This description is the current set implemented to support the simulation and analyses conducted to date. The agent structure encourages incremental development of new capabilities and extension of base capabilities into other application domains, as will be illustrated in the concluding discussion.

Physical Component Agents:

Equipment: MIDAS makes use of current technologies and capabilities by providing Computer-Aided Design (CAD) functions that are able to be integrated with several commercially available and commonly used CAD databases. The MIDAS graphical representation for the physical entities in an environment are created and held in a system called the Cockpit Design Editor (CDE). The CDE is largely domain independent and allows rapid prototyping and animation of virtually any 3-D physical entity. The CDE capabilities have been demonstrated for spacecraft, ground vehicles and commercial aircraft components. The MIDAS system represents not only the graphical or physical aspects of projected equipment, but also supports definition of the functional elements for complex controls and displays. The Equipment Agent fulfills this role and defines the way equipment components in a MIDAS simulation operate. It associates standard operating procedures and functional activities with each component. The Equipment Agent is designed to let the MIDAS user easily develop specific components from more generic ones and build-up complex components from simpler ones. MIDAS includes an equipment editor to create, edit, and delete the functional equipment components required by simulation scenarios. Designers can draw on existing equipment models to rapidly create new, related functional equipment models. In designing equipment models, the equipment editor complements the CDE by representing the functionality of equipment used in a MIDAS scenario and supplying states for graphical equipment animation. The operation of an equipment component may be expressed in four different formats. These four formats are not mutually exclusive -- a single component can use

a mixture of any or all of them. The different formats are: a Finite State Machine (FSM), a time script, a stimulus-response script, or a Lisp method and associated functions. In summary, the equipment editor supports modification of the functional aspects of equipment. The CDE supports modification of the graphical aspects. The two can be used together to edit both aspects of the models simultaneously.

Physical World Agents: In the application that is currently the focus of MIDAS, the physical world is represented by terrain and by aeronautical equipment. The MIDAS system provides the designer with tools to specify the area in which the aircraft operations take place. For helicopter operations the terrain used is selected from a Defense Mapping Agency (DMA) database. In the commercial aircraft focus of MIDAS, terrain is designed to simulate airspace and airport environments. Other physical components or environments can also be represented within MIDAS to support a system or simulated operator's reasoning about physical properties such as location, distance, routes, and visibility. For example, the MIDAS system uses a simple aerodynamics and guidance model to fly a prototype helicopter. Given a predicted reference point in space, a control component computes the collective, cyclic, and pedal control movement required to effect the desired trajectory to the reference point.

The interaction of any physical system that can be described to some level of detail can be represented in the MIDAS agent architecture for integration with displays and controls that serve the human operator in response to the on-going physical world. Although the current implementation of MIDAS concentrates on modeling discrete control functions of the human operator, fully continuous models of human manual control and tracking are an anticipated development in the future.

Human Operator Agents:

The next, and more challenging task, is to provide system designers with an examinable, consistent, and valid representation of the human operator(s) who will be interacting with the equipment and responding to a scenario. The particular interest of this modeling is characterization of the process of perception, decision-making, activity selection, timing, and task-loading incurred by the operator as he or she interacts with the system under study. In addition to the behavioral components of an operator's interaction, we are interested in identifying and tracking the various information requirements and transformations that occur as the scenario behaviors unfold.

Human Performance Representations: In order to populate our evaluation methodology with simulated operators which carry out and supervise automated processes and vehicular control, we have developed, and are elaborating, representations of human cognitive, perceptual and motor operations. These models describe (within their limits of accuracy) the responses that can be expected of human operators in several areas that are critical to safe and reliable operation of advanced automated systems. The fundamental human performance elements of these representations can be applied to any human-machine interaction environment. Tailoring for the particular requirements of a given domain, largely in terms of the human operator's knowledge and rule-base is, of course, a necessary step as the model is moved among domains. Each of the human operators modeled by MIDAS contains the following models and structures, the interaction of which will produce a stream of activities in response to mission requirements, equipment requirements, and models of human performance capabilities and limits.

Physical Representation: An anthropometric model of human figure dimension and dynamics has been developed in conjunction with the Graphics Laboratory of the University of Pennsylvania.⁷ The model used is called Jack™, and is an agent in the overall MIDAS system. The Jack™ agent's purpose is to represent human figure data (e.g., size and joint limits) in the form of a mannequin which dynamically moves through various postures to represent the physical activities of a simulated human operator. The graphic representation of the Jack agent also assists designers in questions of cockpit geometry, reach accommodation, restraint, egress, and occlusion.

Perception and Attention: The simulated human operator is situated in an environment where data constantly streams into the operator's physical sensors. While auditory, haptic, and proprioceptive systems serve an important role in the perception of information relevant to the operator of vehicles, within MIDAS we have presently focused on modeling visual perception.

In brief, during each simulation cycle, the perception agent computes what environment or cockpit objects are imaged on the operator's retina, tagging them as in/out of the peripheral and foveal fields of view (90 and 5 degrees, respectively), in/out of the attention field of view (variable depending on the task), and in/out of focus, relative to the fixation plane. An environmental object can be in one of several states of perceptual attention. Objects in peripheral visual fields are perceived and attentionally salient changes in their state are passed to the updatable world representation. In order

for detailed information to be fully perceived, e.g., reading of textual messages, the data of interest must be in focus, attended, and within the foveal field of view for 200 ms. The perception agent also controls the simulation of commanded eye movements via defined scan, search, fixate, and track modes. Differing stimuli salience and pertinence are also accommodated through a model of pre-attention in which specific attributes, e.g. color or flashing, are monitored to signal an attentional shift. Our models of attention and pre-attention are patterned after the work of Remington and Johnston.⁸ Further detail concerning the implementation of perception and attention within MIDAS can also be found in Banda, Bushnell, et al..⁹

Updatable World Representation (UWR): In MIDAS, the UWR provides a structure whereby each of the multiple, independent human agents, representing individuals and cooperating teams of pilots and flight crews, accesses its own tailored or personalized information about the operational world. The contents of an UWR are determined, first, by pre-simulation loading of required mission, procedural, and equipment information. Then data is updated in each operator's UWR as a function of the perceptual mechanisms previously described. The data of each operator's UWR is operated on by daemons and rules to guide behavior and are the sole basis for a given operator's activity. Providing each operator with his/her own UWR accounts for the significant operational reality that different members of a cooperating control team have different information about the world in which they operate. Further, the individual operator may, or may not, receive a piece of information available to the sensory apparatus as a function of perceptual focus at the relevant point in the mission. It is of some significance that, while ideally the human operators' representation of the world would be consonant with the state of the world, in fact, this is rarely the case. The capability for both systematic and random deviation from the ground truth of the simulation world is a critically necessary component of any system that intends to represent and analyze non-trivial human performance.

As described above, input to the UWR is mediated by attention and perception. These functions are activation filters that allow more or less of the stimuli in the environment to enter the memory structure.

The organization of perceptual data and knowledge about the world in an UWR is accomplished with a structure called a *semantic net*. A semantic net is a structure containing objects called nodes that represent concepts. Relationships among nodes in a semantic network are expressed as links. The types of links can

be described by an analyst or designer to represent relationships that the analyst or designer considers critical. Hierarchies in which system/subsystem relationships are known have also been implemented. For example, “part-of” links can be used to relate a physical subsystem to the system it is part of among the physical elements of the cockpit. An altimeter node would be linked to a pilot-primary-flight-instruments node using a part-of link. This link would support critical reasoning in an emergency, if it were important to simulate an operator’s reasoning that a malfunction in the primary flight instruments could be causing an anomalous altimeter reading. Other uses of part-of links include “team-base” linking among the human elements of a simulation (e.g., a pilot is part-of a flight crew) and mission-based temporal relationships (e.g., an ingress-phase is part-of a mission or a way-point is part-of a flight route.)

The relationship among these nodes is expressed as a “strength” of relatedness, where the strength of such relationships has been investigated to guide models of memory dynamics, i.e. interference, decay, and rehearsal. The “strength” attribute is usually described as a “semantic distance” and is determined by accumulation of subject matter expert ratings using a multi-attribute ranking method. In our current implementation of interference methods, we assume a uniform link-length from any concept to any other association described in the above relations.

Information in the UWR is subject to decay representing a forgetting function. The decay function for working memory is encoded as an exponential that is calculated on the time since an activity or object entered the store. The exponential constant can be applied to classes, types and instances of objects and can have a different rate applied to each object type.

Activity Representation: Activities are MIDAS objects that simulate actions performable by agents in the system. Representations of activities available to an operator are contained in that operator’s UWR. Activities are characterized by:

- preconditions, that define the allowable conditions for their spawning and decomposition;
- spawning specifications, which detail the temporal and logical constraints on any “child activities” that might be needed for activity performance;
- decomposition methods, that describe in a context-sensitive way what children should be spawned to accomplish a higher-level activity’s goals;

- satisfaction conditions, which define their successful completion;
- interruption status and interruption specifications, which detail the interruption and resumption methods for that activity;
- loads, which indicate task performance requirements in terms of visual, auditory, cognitive, and psychomotor resource requirements;
- duration, either estimated or calculated by an activity-specific function; and
- priority, which in this implementation is provided as a fixed table of relative priorities assigned to action and constrained by operator type.

Activities are performed by all of the dynamic entities (agents) in the simulation. However, activities performed by human operators and other intelligent agents in the simulation are of particular interest to the analysts of human-machine systems. An aggregation of MIDAS’ many separate human performance elements, which we term a Symbolic Operator Model (SOM) agent, organizes and mediates the performance of activities through the operations of scheduling, task loading and prioritization.

Rule-based and Decision Activities: This method of introducing activities into the simulation world provides a SOM with the ability to respond to contingencies in the simulation world. Contingent activities involve the application of daemon, rule, and decision theoretic models that act on data incoming to the SOM.

•Daemons. Information about the simulation world is available to a simulated operator’s UWR through the action of objects that model perceptual processes. Changes to the value of any information currently held in the UWR (or the arrival of new information not represented in the UWR) are monitored by “daemons” that may notify other objects in the SOM when a significant change in state has occurred.

•Rules. If conditions are appropriate, a rule may spawn activities in response to changes in the simulation world. If rules are not available to guide behavior then a more computationally intensive decision process is invoked.

•Decisions. In general, decisions are made when the SOM notices anomalous situations triggered by deviations from the expected state. Human decision making has been shown to be context sensitive and variable.¹⁰ MIDAS provides for prescriptive decision methods to be applied according to the amount of time available to make the decision (the “decision horizon”).

The decision horizon is calculated as a function of vehicle and operator state and is used in the selection and application of particular decision strategies. Decision making as currently simulated in MIDAS is a computationally deterministic process. The prescriptive decision making agent includes six algorithms, each of which can be used for selecting among alternative options. As adapted from Payne et al.¹¹, these include:

1. weighted additive
2. equal weighted additive
3. lexicographic
4. elimination by aspect
5. satisficing conjunctive
6. majority of confirming dimensions

Each of these algorithms uses a different combination of attribute values, attribute weights, and attribute cut-off values for calculating the “goodness” value of the options.

Scheduler: Activities decomposed from the goals of the mission by the SOM and its sub-agents are queued and passed to the scheduler. The operation of the scheduler module has been previously reported,¹² and will not be fully detailed here. The scheduler then interacts with the Task Loading Model, through which task loads on the human operators of the system are calculated, to determine an estimated load for the to-be-scheduled activities and to determine an order of activity performance based on a set of operator strategies for scheduling around the available resources. Strategies such as “balanced loading over all resource dimensions” and “task time minimization” have been implemented. The selection of a loading strategy is an option left to the analyst in order to allow exploration of a range of reasonable responses in a given scenario.

Task Loading Model: The Task Loading Model (TLM) in MIDAS has been detailed in a recent publication,¹³ and so only its principal components will be described here. The TLM is based on a body of empirical research and model development by Wickens.¹⁴ The TLM assumes that execution of an activity may utilize resources in any of four categories: visual, auditory, cognitive and motor. The model further assumes that the operator has fixed amounts of these resources available, and that the loads imposed in these four categories vary as a function of the task. The TLM is used to compute task loads on a human operator as a function of the individual loads associated with a given task, and as a function of the interaction of loads on tasks that are performed simultaneously. Task loads are associated with activities during a pre-simulation initialization. A prediction of the loading imposed by the performance of those activities in conjunction with

each other is then provided. The task loading methods applied as the simulation runs are responsive to the task ensemble and to the context in which the tasks are to be performed.

Given a set of partially ordered activities sent to the TLM by the scheduler, the TLM computes the attributes of each action in the set and then calculates a derived load for each activity in the set, taking into account individual loads and modifications based on interactions of the activities in the set. The resulting loads are returned to the scheduler, which uses them as human resource constraints in scheduling the activities.

To produce a stream of human-system behavior, the MIDAS agents described above execute collectively during a mission simulation as depicted on the next page in Figure 4. As previously mentioned, declarative and procedural information about the designated mission and vehicle equipment is held in the simulated operator's Updateable World Representation (UWR). Then, during each simulation time cycle (presently 100 ms), information from the world is filtered by perception and attention models and passed to the UWR. The operator uses this sensed information as required by the levied mission goals to select appropriate lower level activities. These activities are then scheduled and passed to Jack for execution, where they generally affect the cockpit equipment models, allowing the cycle to then repeat.

Application of MIDAS

Although an end-to-end verification of MIDAS' behavioral simulation has not yet been attempted, individual component testing has occurred. Developed incrementally, individual models for visual, cognitive, and psychomotor performance have both strong theoretical underpinnings and part-task laboratory investigations as their basis.

We have also applied MIDAS to current and emerging helicopter designs. These applications have served as a means to ensure the system's ability to accommodate typical human engineering analyses. Through collaborative arrangements with McDonnell Douglas and Boeing Helicopter, we have employed components of MIDAS on the AH-64 Apache Longbow and the MH-47E Chinook variant. In both of these applications, the thrust of our effort was not primarily to critique a particular design, but rather to exercise the embedded models. Within both of these applications, we had great success replicating known crew station geometry and specific concerns about reach and control actuation.

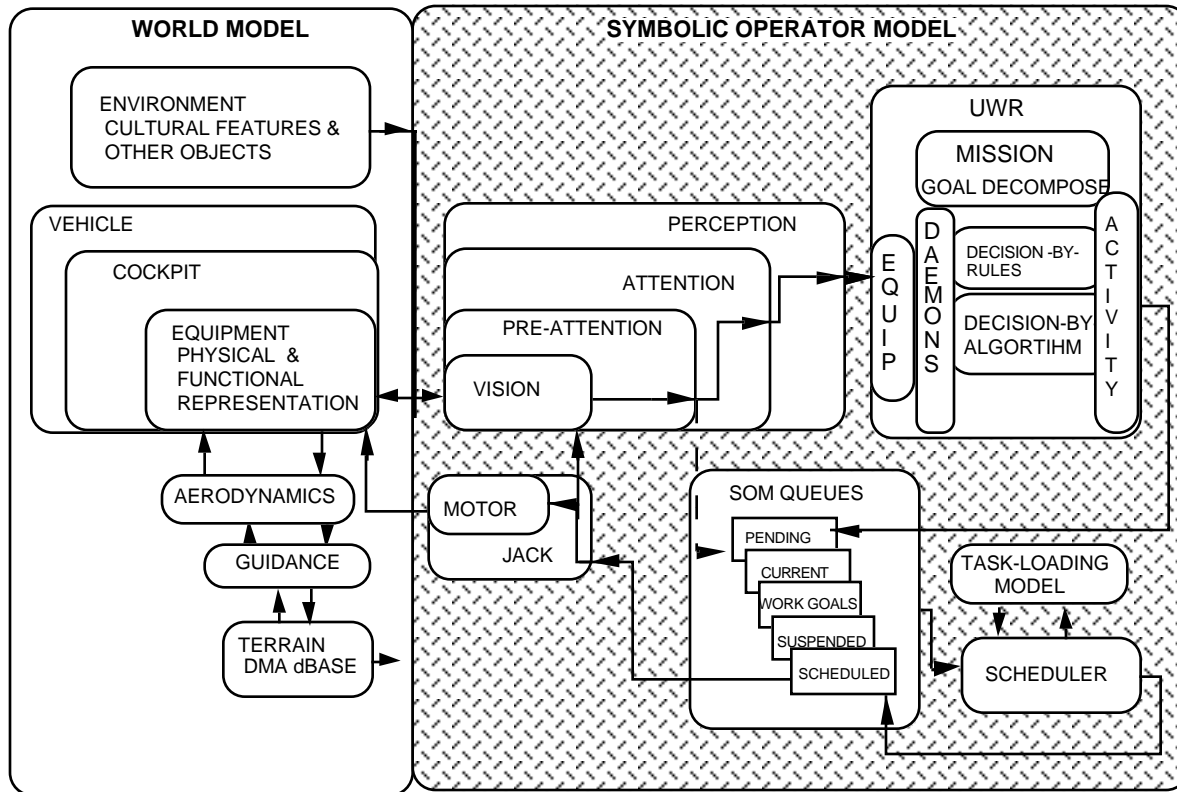


Figure 3. Flow of Information During a MIDAS Simulation

MIDAS' vision models were also used to study legibility and visibility issues, given the cockpit geometry, specified display characteristics, and projected ambient illumination. Predictions from this legibility model proved nearly identical to performance from subsequent empirical tests.¹⁵

Several MIDAS mission scenarios were also created to examine the behavior of the simulation agents previously described. Task timeline and workload estimates were produced and compared to results from similar analysis tools. The resultant behavior and output from the MIDAS simulation system had good correlation with this previous data and also revealed several characteristics about the projected tasks which should be of interest to designers. For example, in one scenario, an equipment malfunction and subsequent display reconfiguration revealed a requirement for the operator to hold in memory a significant engine parameter value, as the equipment design automatically changed modes without any form of prompting. We believe MIDAS is uniquely suited to performing this type of detailed, dynamic procedural simulation. Other simpler static forms of task analysis do not model the equipment nor explicitly represent the context sensitive demands placed on an operator's perceptual, cognitive, or motor resources. In addition, the kind of analyses available

from MIDAS relative to the source and requirements for information to enable human action suggests a fruitful use of MIDAS in the conceptual design for automation and control systems.

In a demonstration of its generalizability, portions of MIDAS are presently being tailored to the design of 911 dispatch console equipment under a cooperative research and development agreement. MIDAS' model and principle basis has made its application to this ground-based domain relatively straightforward, as it has required coding only the domain-specific equipment knowledge and operator activity structures.

While these applications to existing vehicle and console designs have served to hone MIDAS components and point to required functionality, we believe NASA's emerging High Speed Civil Transport will prove to be our ultimate test of relevance. Representing one of the few "new" cockpits to be designed entirely from scratch during this decade, the MIDAS group at Ames Research Center has been tasked to apply our full range of computational design assessment methods to this vehicle. Expected to include extremely sophisticated automation and displays, including enhanced vision and decision aiding systems for the pilots, this application represents the first opportunity for MIDAS to be used

during a vehicle's conceptual design. We are confident MIDAS will be ready for the challenge.

Acknowledgement

The Army-NASA Aircrew/Aircraft Integration (A³I) Program is a joint US Army and NASA exploratory development effort to advance the capabilities and use of computational representations of human performance and behavior in the design, synthesis and analysis of complex, dynamic human/machine systems. The A³I Program is managed jointly through the US Army Aeroflightdynamics Directorate and the NASA Aerospace Human Factors Research Division, both at Ames Research Center, Moffett Field, CA.

Jack is a trademark of the University of Pennsylvania Computer Graphics Lab.

References

1. J.S. Gansler, "Time and Cost Reductions in Flight Vehicle Acquisitions," The Analytical Sciences Corporation (TASC) Briefing to NATO Advisory Group for Aeronautical Research and Development (AGARD), 11 May 1987.
2. A. Newell, *Unified Theories of Cognition*, Harvard University Press, Cambridge, MA, 1990.
3. C.D. Wickens, "Resource Management and Time-sharing," *Human Performance Models for Computer-aided Engineering*, J. Elkind et al. (eds.), Academic Press, New York, NY, pg. 180.
4. G. Agha, "Actors: A Model of Concurrent Computation in Distributed Systems," *Technical Report 844*, MIT Artificial Intelligence Laboratory, Cambridge, MA, 1985.
5. A. Yonezawa and M. Tokoro, *Object-Oriented Concurrent Programming*, MIT Press, Cambridge MA, 1987.
6. R. Milner, "Elements of Interaction: Turing Award Lecture," *Communications of the Association for Computing*, 36, 78-89 (1993).
7. N. Badler, "Computer Graphics Research Laboratory Quarterly Progress Report No. 47," Department of Computer and Information Sciences, University of Pennsylvania, Philadelphia, PA, June 1993.
8. R.W. Remington, J.C. Johnston, and S. Yantis, "Involuntary Attentional Capture by Abrupt Onsets," *Perception & Psychophysics*, 51 (3), 279-290 (1992).
9. C.P. Banda, D. Bushnell, S. Chen, et al., "Army-NASA Aircrew/Aircraft Integration Program: Phase V Man-machine Integration Design and Analysis System (MIDAS) Software Detailed Design Document," *Technical Note TN-93-9216-000-3*, Sterling Software Inc., Palo Alto, CA, 1992.
10. A. Tversky and D. Kahneman, "Judgement under uncertainty: Heuristics and biases." *Science*, 185, 1974, 1124-1131.
11. J.W. Payne, J.R. Bettman, and E.J. Johnson, "Adaptive Strategy in Decision Making," *Journal of Experimental Psychology: Learning, Memory and Cognition*, 14, No. 3 (1988), pp. 534-552
12. R. Shankar, "Z-Scheduler: Integrating Theories of Scheduling Behavior into a Computational Model," *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, 1991, pp. 1219-1223.
13. L. Staveland, "MIDAS TLM: Man-Machine Integrated Design and Analysis System Task Loading Model," *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, 1991, pp. 1367- 1371.
14. C.D. Wickens, *Engineering Psychology and Human Performance*, E. Merrill Publishing Co., Columbus, OH, 1984.
15. A. Arditi, S. Azueta, J. Larimer, M. Prevost, J. Lubin, J. Bergen, "Visualization and Modeling of Factors Influencing Visibility in Computer-Aided Crewstation Design." *Society of Automotive Engineers, SAE Technical Paper 921135*, July 1992.