N91-20707

Cockpit Task Management:
A Preliminary, Normative Theory

Ken Funk

Department of Industrial and Manufacturing Engineering
Oregon State University
Corvallis, OR 97331-2407
(503) 737-2357
funk@guille.ece.orst.edu

Abstract

Cockpit Task Management involves the initiation, monitoring, prioritization, allocation of resources to, and termination of multiple, concurrent tasks. As aircrews have more tasks to attend to, due to reduced crew sizes and the increased complexity of aircraft and of the air transportation system, CTM will become a more critical factor in aviation safety. It is clear that many aviation accidents and incidents can be satisfactorily explained in terms of CTM errors, and it is likely that more accidents induced by poor CTM practivce will occur in the future unless the issue is properly addressed.

Our first step in understanding and facilitating CTM behavior has been the development of a preliminary, normative theory of CTM which identifies several important CTM functions. From this theory some requirements for pilot-vehicle interfaces have been developed which we believe will facilitate CTM. We have developed one prototype PVI which improves CTM performance and are currently engaged in a research program aimed at developing a better understanding of CTM and facilitating CTM performance through better equipment and procedures.

Introduction

Air travel is one of the safest forms of transportation, yet each year hundreds of lives and millions of dollars are lost due to air crashes. Accident investigations reveal that over half of these accidents are attributable to errors by the cockpit crew [Nagel, 1988].

Since crew-induced accidents are rare, the "remedy" has historically been to provide specific fixes for specific causes of specific accidents. For example, ground proximity warning systems were developed in response to a (small) number of controlled flight into terrain accidents. And yaw dampers were installed in response to incidents of Dutch roll, an instability problem characteristic of swept-wing aircraft.

This may have led to what Wiener [1987] calls the "one-box-at-a time" approach to cockpit automation that ignores the need for information and control integration in the cockpit, leaving that integration entirely to the already overburdened aircrew. Responses to specific incidents and problems do not necessarily decrease the liklihood of other incidents and problems. Unless a more general approach to understanding cockpit operations and problems is adopted, it is likely that the trend will continue, perhaps with catastrophic results.

A systems engineering approach to this problem is more desirable than the ad hoc methods now so commonly used. As Sheridan points out [1988], the systems approach provides more precise methods of problem formulation, a basis for simulation and qualitative understanding of systems, a basis for quantitative prediction of system behavior, an accounting framework for design and evaluation, and a language for archival description.

Our own application of systems engineering methods to cockpit operations, has led us to a concept we call Cockpit Task Management (CTM). CTM involves the formulation of goals, the definition of tasks to achieve those goals, and the management and execution of those tasks in a dynamic environment until the goals are achieved. The remainder of this paper presents some background definitions, a preliminary, informal version of a normative theory of CTM, some guidelines for the design of pilot-vehicle interfaces to facilitate good CTM, and a summary of our continuing efforts to improve CTM.

Definitions

A dynamic system is an entity which may be described in terms of input, output, and state. Input is matter, energy, or information having a net flow into the system. Output is net flow of matter, energy, or information out of the system. State is a compact representation of the history of the system which makes possible the prediction of future outputs and of state itself [Padulo and Arbib, 1974]. Input, output, and state may each be decomposed into multiple components. For example, an aircraft is a system whose input components include fuel flow, control yoke movements, and radio clearances from air traffic control (ATC). Aircraft outputs include fuel combustion products, heat and noise, and requests for and acknowledgments of ATC clearances. Aircraft state components include position and altitude, flap angles, and radar mode.

Two systems which are connected by inputs and outputs form a more complex system called a supersystem. The supersystem's inputs are the unconnected inputs of the simpler systems. Its outputs are the unconnected outputs. The state of the supersystem is defined by the combined states of the original systems. Through successive system connections, systems of arbitrary scope and complexity may be defined. If a system is formed from simpler systems through input-output connections, the simpler systems are called subsystems. For example, an aircraft system can be defined as a collection of powerplant, electrical, hydraulic, and avionic systems. With respect to the powerplant system, the aircraft may be considered a supersystem. From the perspective of the aircraft system, the powerplant may be considered a subsystem.

The use of the generic terms system, subsystem, and supersystem, rather than terms like equipment and components, permits the examination of domains from many levels of abstraction. Along with this flexibility, though, comes the potential for ambiguity and confusion. For example, a discussion in which the term "system" was applied to that combination of people, machines, policies and procedures called the air traffic control system as well as to a light emitting diode on an aircraft instrument panel would be problematic without further clarification. For any frame of reference, the analyst must clearly identify the levels of abstraction to which the terms "system," "subsystem," and "supersystem" apply.

A system behavior is a (perhaps continuous) series of system input, state, and output values over a time interval. For example, as an airliner flies from Eugene, Oregon to San Francisco, successive values of input components (including the pilot's movement of the control yoke), state components (including position) and output components (including radio transmissions) over the time interval of the flight constitute a behavior of that system. A system exhibits a behavior if observations of the system yield input, state, and output values exactly matching those of the behavior.

An event is a set of system behaviors in which some state component changes in a significant way at the very end of the time interval. For example, **reach 10,000 feet** is an event consisting of a set of aircraft behaviors. In each behavior of this event the aircraft's altitude increases, reaching a value of 10,000 feet at the end of the behavior's time interval. An event occurs if the system exhibits a behavior which is contained in the event set.

A goal for a system is defined by a set of desired behaviors and a state. Each behavior begins with an initiating event and ends with a terminating event. In any behavior of the system, if the initiating event has not occurred, the state of the goal is **latent**. If the initiating event is imminent, the goal is **pending**. If the initiating event has occurred but the terminating event has not occurred and the actual behavior matches the initial portion of some desired behavior, the state of the goal is **active**. If the initiating event has occurred, the terminating event has occurred, and the actual behavior through the time of the terminating event matches one of the goal behaviors, the state of the goal is **achieved**. If the initating event has occurred but the actual behavior does not match any of the goal behaviors, the state of the goal is **violated**.

A goal's initiating event defines the conditions under which the goal is relevant. A typical flight path consists of a series of waypoints which are geographical points along the route that serve as intermediate destinations. So a goal to arrive **at waypoint 8** is relevant after an **arrive at waypoint 7** event has occurred. On the other hand, a terminating event may take on more than one meaning, as discussed below.

Formally, only one type of goal is necessary. As a practical matter however, goals may be classified as to intent and interpretation. In an **attainment goal**, the terminating event results in some desired state of the system and the intervening input, state, and output values are unimportant. For example a **gear down and locked** goal could be defined by all possible behaviors terminated with an event resulting in the landing gear being in the down and locked state. In this case, it does not matter how the landing gear is lowered (by motor, gravity, or manual operation). Only the final state is important.

In a **maintenance** goal, it is the portion of the behavior between the initiating and terminating events that is important. For example, a goal to maintain **approach speed until touchdown** might be defined by the collection of all behaviors in which the aircraft's speed was within five knots of the approach speed specified in the aircraft operations manual, until touchdown occurred. Here, the immediate objective of this specific goal is not touchdown on the runway, it is maintaining the proper airspeed until touchdown occurs. Put another way, a maintenance goal reflects a set of constraints on system behavior which are active until some event occurs.

A **constrained attainment** goal is an attainment goal in which the intervening behaviors are important. For example, a goal to arrive **at destination area (via waypoints 1, 2, and 3)** might be defined by a set of behaviors in which the aircraft flies from its origin to waypoint 1, to waypoint 2, to waypoint 3, and ends in the area of the destination airport.

A <u>subgoal</u> of a goal is a set of behaviors consistent with those of the goal, but restricted in time and/or in scope. A goal may be decomposed into a set of subgoals, which are goals consistent with the original goal but restricted in some way. Serial subgoals are defined for the original system, but over distinct time subintervals. Parallel subgoals are defined over the entire time interval, yet are defined for subsystems of the original system. A goal may also be decomposed into a combination of serial and parallel subgoals. For example, a goal to approach the destination airport and arrive **at landing position** (prior to final approach) could be decomposed into serial **cleared to approach waypoint** and **at approach waypoint** subgoals and parallel **approach flaps**, **approach power**, and **approach speed** subgoals.

A goal and all of its subgoals form a hierarchy with the goal at the apex. The topmost goal for a flight mission will be referred to as the mission goal. Part of a simplified goal hierarchy for a flight mission is shown in Figure 1.

Goal <u>priority</u> reflects an ordering of a set of goals and/or subgoals, as determined by the relative importance assigned to them by the aircrew. More important goals have higher priorities. For example, a goal to remain **clear of terrain and other aircraft** established to maintain the safety of the aircraft and its passengers is clearly more important than a goal to maintain **$\pm$ 20 degrees roll**, established for passenger comfort. The first goal should then have a higher priority than the second.

<u>Performance</u> is how well a system achieves a specific goal. A <u>performance measure</u> is a function that maps a goal and a system behavior to a value set. The simplest performance measure may take on just two values: "satisfactory" if the goal is achieved (or at least not yet violated), and "unsatisfactory" otherwise. More complex performance measures may map to amore complex, ordered set. For example, a goal to maintain **10,000 feet** may be achieved if the aircraft's altitude stays between 9,900 and 10,100 feet. But a behavior in which the maximum deviation was no more than 25 feet might be preferred to a behavior in which the maximum deviation was 75 feet. In this case, we could say that the system performed better when exhibiting the $\pm$ 25 foot behavior than when exhibiting the $\pm$ 75 foot behavior.

A <u>task</u> is a process completed to cause a system to achieve a goal. A task involves the behaviors of one or more secondary systems or subsystems in order to produce inputs to the primary system to achieve the goal. For example, for the goal to arrive **at waypoint** 7, there must be a **fly to waypoint** 7 task. The pilot, the primary flight controls, the cockpit displays, the electrical system, and the engines are just a few of the secondary systems required to complete the **fly to waypoint** 7 task to achieve the goal for the primary system (the aircraft) to arrive **at waypoint** 7.

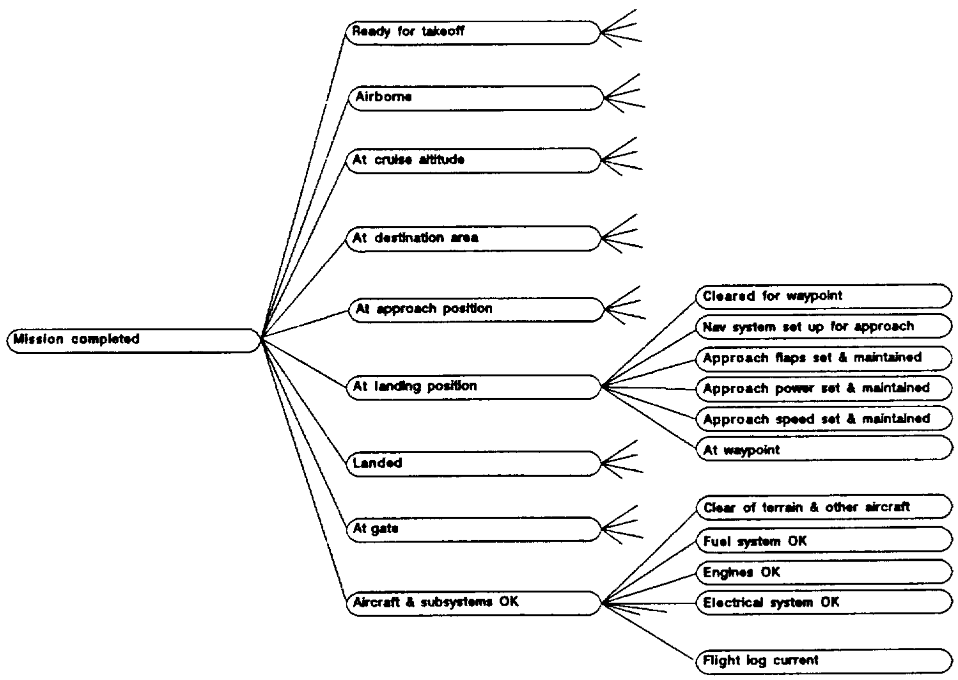

Figure 1: Part of a Simple Goal Hierarchy

Like a goal, a task has state. A task is latent if its goal is latent, pending if its goal is pending, and active if its goal is active. A task is in progress if inputs to the primary system are being applied to achieve the goal. If the task has been in progress but inputs to the primary system to achieve the goal have been suspended, the task is interrupted. A task may be terminated if its goal is achieved, if the goal is not achievable, or if the goal becomes irrelevant. In the case of an unsuccessful termination, the task is considered to be aborted.

The performance of a task is simply the performance of the system with respect to the task's goal while the task is being completed. A pilot keeping the aircraft within 25 feet of a selected altitude is performing a **maintain altitude** task better than one only keeping within 75 feet of the selected altitude.

As we can decompose the goal to approach the airport and arrive **at landing position** into **cleared to approach waypoint** and **at approach waypoint** subgoals, an **approach** task could be decomposed into **get approach clearance** and **fly to approach waypoint** subtasks.

An agenda defines an ordered set of tasks to be completed during a mission. Each task is defined to achieve a specific goal and becomes active when the goal's initiating event occurs. The structure of an agenda follows that of a goal hierarchy but carries additional task information. Figure 2 shows part of an agenda corresponding to the goal hierarchy shown in Figure 1.

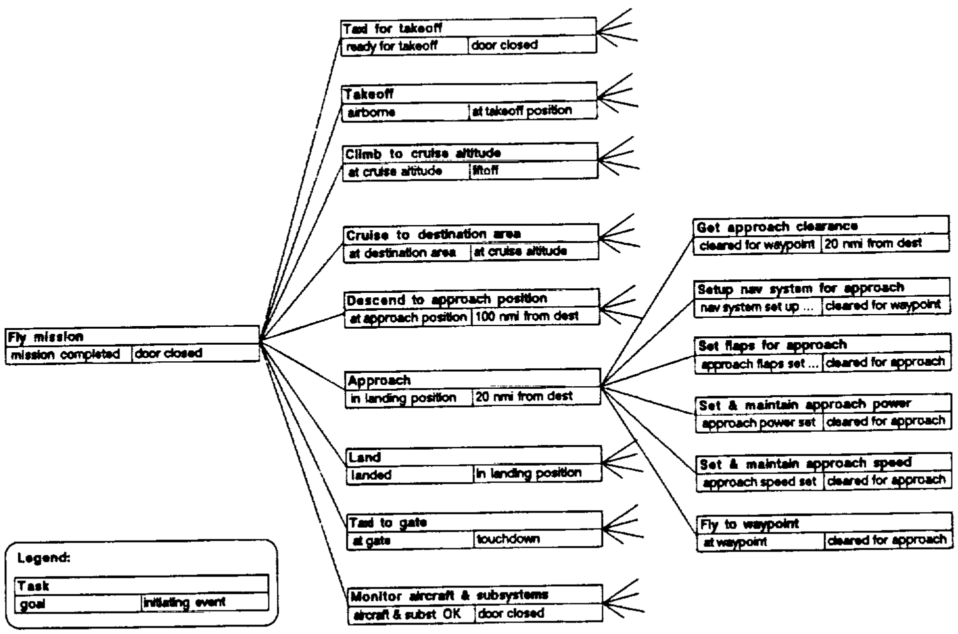

Figure 2: Part of a Simple Agenda

When an initiating event occurs, the corresponding task becomes active. Since two or more tasks may share a common initiating event and since one task may not reach completion before another task becomes active, several tasks may be active at one time. Two or more tasks that are simultaneously active are called concurrent tasks.

Resource-Limited Performance

Executing a task to achieve a system goal, such as to fly an aircraft to a destination, requires that certain inputs be provided to that system over a time interval. These inputs must come from other systems or subsystems, such as pilots, autopilots, and other cockpit equipment. These systems or subsystems are called resources, and resources are required to complete a task. If the resources are not available, that is their outputs cannot be directed to the primary system, the task cannot be completed satisfactorily and the goal cannot be achieved.

A variety of resources are required for cockpit tasks. Equipment resources include autopilots, radios, displays and controls. Human resources include the pilot, first officer, and flight engineer. Some resources are specialized and can only be used for a limited set of tasks. Examples of specialized resources include the landing gear control lever and the altimeter. Other resources are multi-function and can be used for a variety of tasks. Examples of these include multi-function CRT displays and humans.

Since resources are systems, they can be decomposed into simpler subsystems. Of particular interest are the human resources, which can be decomposed into personal sensory, motor, and cognitive resources. Sensory resources include visual, auditory, and other sensory systems which can be used to obtain external system state information necessary for completing a task. Motor resources include hands, feet, voice, and other body systems that can produce inputs to external systems. Cognitive resources are mental subsystems required to perform cognitive tasks, such as those involving pattern recognition, problem solving, and decision making. The resources include the verbal and spatial resources identified and studied by Wickens and his colleagues at the University of Illinois [Wickens 1984; Wickens and Liu, 1988].

Since two concurrent tasks may require the same resources, this poses a potential problem, since resource behavior compatible with achieving one goal may be incompatible with achieving the other goal and the performance of one or more of the tasks may be degraded. That is, task performance is limited by resource availability. With resources like displays or hands and feet, this is obvious. But it is also true for cognitive resources [Navon and Gopher, 1979; Wickens, 1984]. A situation in which task resource requirements exceed resource availability is called a task conflict.

For example, given the agenda in Figure 2, if ATC clearance to an approach waypoint is obtained the **set and maintain approach power** task would become active. Assume that this task requires a multifunction CRT resource on which an engine display format must be shown. Suppose that now a primary electrical system failure event occurs and a subtask to **diagnose and correct electrical system** becomes active. Assume that this subtask requires an electrical system display format on the same CRT resource. If the two display formats cannot be displayed simultaneouly a resource shortage and therefore a task conflict exists.

Even if two CRTs are available to complete both of these tasks simultaneously, there still might be a task conflict due to cognitive resource limitations. Assuming for the purpose of this illustration that no other crewmember is available to assist the pilot in completing these two tasks, he or she may lack sufficient cognitive resources to simultaneously attend to both of them. This might result in errors in completing one or both of the tasks.

Task conflicts like these can be partially resolved through a process of prioritization and resource allocation. In the case outlined above, the pilot can decide that the immediate correction of equipment failure is absolutely essential to the safe continuation of the flight, and temporarily suspend the **set and maintain approach power** task, using all necessary resources to complete the diagnosis and correction subtask.

But if this subtask takes longer than anticipated, flight safety could be endangered, for if the aircraft proceeds at the current altitude longer than the air traffic controller anticipated, the potential exists for collisions with other aircraft travelling at the same altitude. Focussing on one task to the exclusion of others can lead to poor task performance at minimum and disaster at worst.

Given the complex nature of modern aircraft, the speed at which they travel, and the increasing density of air traffic in airspaces, the existence of multiple, concurrent tasks in the cockpit is the norm rather than the exception. Clearly, concurrent tasks must be systematically managed by the aircrew to achieve acceptable levels of system performance.

A Preliminary, Normative Theory of Cockpit Task Management

The process by which the aircrew manages an agenda of cockpit tasks may be called Cockpit Task Management (CTM). Given the requirement to allocate limited resources to tasks in a dynamic environment, some essential functions of CTM are readily apparent. A brief outline of these functions are presented below and a generalized procedure for CTM is shown in Exhibit 1. Please note that the following theory of CTM is a normative one and presents the functions that should be completed. It does not seek, at this point, to explain how they are performed, nor does it explicitly account for errors, which will be discussed later.

```
Procedure: CTM
  create and validate initial agenda
  until mission goal is achieved or unachievable
        activate tasks whose initiation events have occurred
        assess status of active tasks
        terminate tasks with achieved or unachievable goals
        assess task resource requirements
        prioritize active tasks
        allocate resources to tasks in order of priority:
              initiate higher priority tasks not yet in progress
              interrupt lower priority tasks currently in progress
              resume higher priority tasks that were interrupted
        update and validate agenda
  endUntil
End: CTM
```

Exhibit 1: Cockpit Task Management Procedure

Before CTM can begin, an initial planning process must be completed. This planning process yields a set of goals, including a primary mission goal, a hierarchy of subgoals to the mission goal, and perhaps a collection of goals to deal with contingency situations, such as an engine fire or a hydraulic system failure.

Agenda Creation is the first step of CTM and involves the selection and specification of a suitable task to achieve each goal and the definition of the initiating event for each task. The specification of each task includes a list of resources necessary to complete the task, both equipment and human. The creation of the agenda also requires the validation of the goals upon which the agenda is based. It is necessary to make sure that all goals are compatible with the mission goal and with each other.

Once the agenda has been created and validated, true task management begins. This iterative process lasts until the mission goal is achieved or it has been determined that the mission goal is not achievable and no further effort need be expended towards achieving it.

Task Activation is the detection of the initiating event for a task and the recognition that the task should be started. For the initial tasks in a mission, such as the **taxi for takeoff** task, this occurs immediately. For other tasks, such as the **fly to approach waypoint** task, the initiating events and task activation may occur much later in the mission. Some tasks for contingent goals, such as an **extinguish engine fire** task (a subtask of a **monitor aircraft and subsystems** subtask), may never be activated.

Task Status Assessment or task monitoring determines the status of each task, which reflects the achievement of the task's goal. Not only must the current status of the task be assessed, but if the task's goal is not yet achieved, the status of the task must be projected into the future to determine the liklihood that the goal will be achieved.

If the goal is achieved or if the goal has not yet been violated but current trends will likely result in the goal's achievement, the status of the task is **satisfactory**. For example, suppose that **fly to waypoint 7** is an active task. If the aircraft is at waypoint 7 the task's status is satisfactory. If the aircraft is not at waypoint 7, but it is flying in that direction and there is sufficient fuel to reach the waypoint, the task's status is also satisfactory.

If the goal is not yet achieved, not yet violated, but current trends will likely result in a violation of the goal, the status of the task is **marginal**. If **fly to waypoint 7** is active but the aircraft's course is 10 degrees to the right of a heading to waypoint 7, the status of the task is marginal.

If the goal is violated, the status of the task is **critical**. If the aircraft has passed waypoint 7, but has not come within some acceptable range of the waypoint, the **fly to waypoint 7** task is critical.

The above is a minimal classification scheme for task status. These status values probably should be treated as general categories to be further subdivided to provide more resolution in status assessment.

Task Termination removes tasks from competition for resources. Normal task termination is a result of the achievement of the task's goal. So when the aircraft is airborne, the **takeoff** task may be terminated.

A critical task, one whose goal cannot be achieve or at least probably cannot be achieved, may be aborted, thereby terminating it. Such might be the case if dangerous wind shear conditions are detected during a **land** task. When the possibility of aborting a task exists, the agenda should contain contingent tasks to replace the aborted tasks. For example, an **execute missed approach** task should be included in an agenda to replace an aborted **land** task.

Another reason for terminating a task is because its goal is no longer relevant. For example, if a landing gear fails to operate properly on the first try, a **diagnose/correct landing gear** task might be initiated. If later, through no direct action of the crew, the gear operates properly, the goal to diagnose and fix the landing gear would no longer be relevant and the task could be terminated.

Task Resource Requirements Assessment must be performed to determine what resources are required to complete the active tasks. Each task has minimum resource requirements, but in some cases, task performance can be improved by providing additional resources. For example, the performance of a **diagnose/correct engine problem** task might be improved by allocating two rather than one display resources to it, allowing the simultaneous display of engine parameters on one display surface and an engine diagnosis checklist on the other.

Recognizing the improved performance that additional resources can bring may be especially important in correcting marginal or critical tasks. On the other hand, over-allocating resources to one task may interfere with the performance of another, if those resources are limited.

Task Prioritization is an ordering of tasks by priority. Factors which can influence task priority include the following:

1. the priority of the task's goal.
2. the priorities of the goals of other active tasks.
3. the current and projected status of the task.
4. the current and projected statuses of other active tasks.

Task prioritization can ultimately be defined in terms of a pairwise comparison of tasks based on the above as well as other factors, which results in an ordering of active tasks. For example, suppose that both a **maintain $\pm$ 20 degrees roll** task and a **remain clear of terrain and other aircraft** task are active. If the aircrew detects another aircraft on a collision course, they should assign a higher priority to the second task than to the first because the goal to remain **clear of terrain and other aircraft** has greater importance and a higher priority than the goal to maintain **$\pm$ 20 degrees roll**.

Resource Allocation is the assignment of resources to tasks, with preference given to high priority tasks, so that the tasks may be executed. Resource allocation depends directly on task prioritization, and since that is a dynamic process, resource allocation must be dynamic also.

When a newly activated task has a high enough priority, resources are allocated to it and task initiation occurs. This means that the required resources begin exhibiting behaviors consistent with the achievement of the task's goal. In many cases, task initiation requires a communication of the goal to some of the resources so that they can behave accordingly. For example, if one of the resources required for a task is a human crew member, that crew member must be aware of the goal in order to behave in such a way as to bring about its achievement. This is also true for some equipment resources. For example, to fly automatically to a certain location, the aircraft's navigation computer must be "informed" of the goal by the input of the destination's geographical coordinates.

If a lower priority task is in progress and a higher priority task is initiated which requires those resources, then the resources are allocated to the higher priority task. This is called task interruption, and the lower priority task, while still active, is no longer in progress, or it may said to be suspended.

When a high priority task in progress is terminated, for whatever reason, task resumption of a lower priority, suspended task can occur, in which case resources are reallocated back to the lower priority task and it can continue.

Actually, resource allocation based merely on task priority may be insufficient. In some cases at least, resource reallocation may occur due to the specific status of a task. For example, the autopilot may be allocated to a **fly to approach waypoint** task, but the autopilot, due to existing conditions, may not be able to adequately control the descent. It may then be necessary to deallocate the autopilot from the task and allocate a human crewmember to it to achieve the goal.

Agenda Updating is necessary since some cockpit tasks may alter the agenda. If bad weather or other contingencies make a planned route infeasible or undesirable, a planning task may be initiated to change the original route. This will, by necessity, change the agenda. The goals and tasks created by this planning task must be integrated into the agenda, perhaps replacing earlier components. Of course, validation of the candidate changes to the agenda must take place to assure that the mission goal is achieved and that no goal conflicts occur.

Cockpit Task Management Failures

The significance of CTM can best be appreciated by using the framework presented above to examine several aviation accidents and incidents which have occurred in the last two decades. The following accounts are summaries from National Transportation Safety Board Aviation Accident Reports.

> On March 21, 1980, at 1949, Eagle Commuter Airlines, Inc. Flight 108, a Piper PA-31-350, with a pilot, a pilot-in-command trainee, and eight passengers on board, crashed on takeoff from runway 22 at William P. Hobby Airport, Houston, Texas. The pilot, the pilot-in-command trainee, and five passengers were killed, and three passengers were injured seriously. The aircraft was destroyed by the crash and the postcrash fire. The National Transportation Safety Board determines that the probable cause of the accident was a power loss in the right engine for undetermined reasons at a critical point in takeoff, the aircraft's marginal single-engine performance capability, and the captain's incorrect emergency response to the engine power loss when he failed to either land immediately on the remaining runway or to configure the aircraft properly for the engine-out condition. [NTSB, 1981]

> An Eastern Airlines Lockheed L-1011 crashed at 2342 eastern standard time, December 29, 1972, 18.7 miles west-northwest of Miami International Airport, Miami, Florida. The aircraft was destroyed. Of the 163 passengers and 13 crewmembers aboard, 94 passengers and 5 crewmembers received fatal injuries. Two survivors died later as a result of their injuries. Following a missed approach because of suspected nose gear malfunction, the aircraft climbed to 2,000 feet mean sea level and proceeded on a westerly heading. The three flight crewmembers and a jumpseat occupant became engrossed in the malfunction. The National Transportation Safety Board determines that the probable cause of the accident was the

failure of the flightcrew to monitor the flight instruments during the final 4 minutes of flight, and to detect an unexpected descent soon enough to prevent impact with the ground. Preoccupation with a malfunction of the nose landing gear position indicating system distracted the crew's attention from the instruments and allowed the descent to go unnoticed. [NTSB, 1973]

On June 13, 1984, USAir, Inc. Flight 183, a McDonnell Douglas DC9-31, N964VJ, with 5 crewmembers and 51 passengers aboard, encountered turbulence, hail, and heavy rain as it was making an instrument landing system approach to runway 21R at the Detroit Metropolitan Airport, Detroit, Michigan. The airplane landed on the runway about 2500 feet beyond the threshold of runway 21R before the landing gear was extended fully. The airplane skidded bout 3,800 feet before sliding into the grass on the left side of the runway. The crew and passengers were evacuated with only minor injuries. The airplane was damaged substantially. The National Transportation Safety Board determines that the probable cause of the accident was inadequate cockpit coordination and management which resulted in the captain's inappropriate decision to continue the instrument approach into known thunderstorm activity where the airplane encountered severe wind shear. The failure of air traffic control personnel at the airport to provide additional available weather information deprived the flightcrew of information which may have enhanced their decisionmaking process. [NTSB, 1985]

Each of these accidents or incidents was thoroughly investigated by the NTSB, probable cause was assigned, and contributing factors were identified. In the Eagle Commuter accident the captain "... failed to ... configure the aircraft properly for the engine-out condition." "Preoccupation with a malfunction ... distracted the [Eastern Airlines] crew's attention ..." The USAir captain made an "... inappropriate decision to continue the instrument approach into known thunderstorm activity."

In each case conclusions can be and no doubt were drawn about how the accidents could have been prevented. It is likely that these fixes, were they implemented, would have prevented similar accidents from occurring. But specific explanations of and fixes to specific problems do not necessarily prevent accidents of other types from occurring.

If, on the other hand, we examine these occurrences from the perspective of CTM, we can develop a more comprehensive understanding of cockpit errors and perhaps suggest effective ways of preventing a wider variety of accidents from occurring. With that in mind, consider the following, supplementary explanations of the accidents and incidents, from the perspective of CTM.

Faced with multiple, possibly conflicting tasks, the Eagle Commuter captain failed to <u>initiate</u> an engine-out recovery task. The Eastern Airlines crew failed to <u>monitor the status</u> of the primary flight task, possibly because they assigned too high a priority to the tasks of dealing with the malfunctions. The Eastern crew also <u>overallocated resources</u> to the landing gear diagnosis task (all three crewmembers plus a jump seat occupant became totally absorbed in the diagnosis). The USAir captain failed to <u>terminate</u> the landing task, even though continuation of the task placed the higher priority goal of passenger, crew, and aircraft safety at extreme risk.

Pilot-Vehicle Interface Requirements to Facilitate
Cockpit Task Management

The concept of Cockpit Task Management has potential implications for aircrew training and cockpit procedures, and these should be addressed. But our efforts in the past have focussed on cockpit automation, especially the design and development of intelligent pilot-vehicle interfaces (PVIs). Based on the preliminary, normative theory of CTM and the CTM-based analysis of a variety of accidents and incidents, we believe a PVI should perform the following functions to facilitate CTM:

1. Maintain an internal representation of the mission agenda. The PVI should possess knowledge of the agenda for each flight. Once the aircrew has planned a mission, they must be able to create a representation of the mission agenda in the PVI. They must also be able to modify the agenda during the mission as plans change.

2. Display agenda information to the aircrew. The PVI must provide a dynamic agenda display that keeps the aircrew informed about the agenda. It should display information about the state of each goal and the state and status of each task, especially those goals and tasks that are pending or active. The aircrew may not choose to have the agenda display visible at all times, but it must be available and easily accessible to them.

3. Monitor and display aircraft and subsystem states. All PVIs display aircraft and subsystem information, but to facilitate CTM these displays should be controlled by the PVI to emphasize information relevant to pending and active tasks.

4. Monitor task states and inform the aircrew. The PVI should monitor aircraft and subsystem state, note events, and update the agenda. Specifically, the PVI should determine when tasks become pending, active, or terminated. This information should be provided to the aircrew through the agenda display and perhaps through other displays as well, especially when the agenda display is not visible.

5. Determine when tasks are being performed. The PVI must be able to determine when tasks are being performed by the aircrew and by avionics systems. In some cases this may be done implicitly by monitoring aircraft and subsystem states as they change under aircrew and avionics control [Hoshstrasser and Geddes, 1989; Rouse and Hammar, 1990]. In other cases, aircrew intent must be determined by explicit communication from the aircrew that the task is or will soon be underway.

6. Assess task status and inform the aircrew. The PVI should assess task status based on the present or projected status of goals and inform the aircrew through appropriate displays. In the case of marginal or critical tasks, the aircrew should be alerted and perhaps advised so that appropriate and timely action can be taken to maximize the chances of goal achievement.

7. Prioritize tasks and inform the aircrew. Tasks should be prioritized by the PVI and the aircrew should be informed through appropriate displays. Priorities of marginal or critical tasks should be emphasized.

8. Help the aircrew perform specific tasks. Although the major concern here is in facilitating CTM, the functions described above virtually necessitate a PVI architecture that could also support specific task aids, such as planning tools, computational aids, and expert systems for diagnosis and control. The level of support provided by these aids should be selectable by the aircrew and the aids should always remain under aircrew authority. Decisions and control actions provided by the aids should be subject to aircrew authorization, either in real time or by "contractual" arrangement prior to the mission. It is likely that such aids could help improve individual task performance and indirectly improve CTM performance by reducing the cognitive resource demands on the aircrew by the individual tasks.

Steps Toward Better Cockpit Task Management

We have made significant progress in our efforts to understand and facilitate CTM. Our primary accomplishment to date is a prototype PVI and we are currently involved in both theoretical and applied research and development efforts.

The Task Support Subsystem (TSS) is a prototype PVI developed at Oregon State University whose function, in part, is to facilitate CTM [Funk, 1990]. It is a subsystem of an experimental avionics system that runs in a simulated aircraft. Prior to a mission, a mission definition is created which defines the tasks to be accomplished during the flight. During the simulated flight, software modules called Task Agents (TAs) perform the CTM function to see that all tasks are completed satisfactorily.

For each task in the mission there is a TA assigned to it. The TA determines when the task should be started and configures the cockpit for the task. It then monitors the pilot and aircraft subsystems to see that the task is completed correctly and on time. If the pilot fails to act on the task, the TA reminds him via a display and the TA alerts the pilot to actual or anticipated deviations from the task's goal. Most TAs also facilitate task execution by providing procedural prompts and recommendations. Some TAs are capable of completely automating their tasks at the pilot's discretion.

Multiple TAs are coordinated by a high level TA that allocates resources based on priority. A mission display serves to remind the pilot of tasks to be completed and shows the status of each active task.

The TSS, as part of the avionics system, was evaluated by a group of 16 professional pilots in a simulator experiment [Lind et al, 1989]. Each pilot flew two equivalent, simulated missions, one in a baseline cockpit and one with the TSS present. Performance measures involved timing, accuracy and number of errors committed. Statistically significant results favoring the TSS-equipped cockpit were obtained from the data analysis and pilots subjectively rated the TSS-equipped cockpit as superior in terms of situational awareness and workload. Subsequent informal evaluation of the TSS by a variety of pilots and non-pilots have been consistent with the positive results of the experiment.

Our ongoing research involves development of theories of CTM and development of further prototype PVIs to facilitate CTM.

The preliminary, normative theory sketched above is being formalized in the framework of mathematical systems theory [Mesarovic and Takahara, 1975; Funk, 1983]. A simulation model will be developed and validated for internal consistency before finalizing a procedural description of CTM.

The normative theory will serve as the basis for a descriptive theory of CTM which identifies human capabilities and limitations in performing CTM functions. From the descriptive theory will come an error taxonomy, a framework for explaining CTM errors, and a model for predicting CTM performance.

Both theories will be further formalized to create analytic and evaluative methodologies which will be applied to the examination of aviation incidents and accidents from the perspective of CTM and the rating of cockpit equipment and procedures for how they facilitate or impair CTM. We believe that the development and application of these methodologies will also lead to countermeasures to poor CTM, perhaps in the form of general principles as well as specific design guidelines along the lines of those presented above.

From these principles and guidelines we will construct and evaluate further prototype PVIs. Our goal is not just to understand CTM, but to improve it through intelligent engineering research and practice.

We are encouraged by our progress and believe that the CTM concept has significant potential for improving the safety and effectiveness of aerospace systems.

References

Funk, K.H., [1983], "Theories, Models, and Human-Machine Systems," Mathematical Modelling, Vol. 4, pp. 567-587.

Funk, K.H., [1990], "Agent-Based Pilot-Vehicle Interfaces: Concept and Prototype," 1990 TIMS/ORSA Joint National Meeting, Las Vegas, NV, 7-9 May 1990.

Hoshstrasser, B.H. and N.D. Geddes, [1989], "OPAL: Operator Intent Inferencing for Intelligent Operator Support Systems," Proceedings of the IJCAI-89 Workshop on Integrated Human-Machine Intelligence in Aerospace Systems, Detroit, MI, 21 August 1989.

Lind, J.H., C.W. Hutchins, and D.E. Neil, [1990], "The Effect of Knowledge-Based System Assistance on Piloting Performance, Workload, and Satisfaction," IEEE National Aerospace and Electronics Conference, Dayton, OH, 21-25 May 1990.

Mesarovic, M.D. and Y. Takahara, [1975], General Systems Theory: Mathematical Foundations, New York: Academic Press.

Nagel, D.C. [1988] "Human Error in Aviation Operations," in E.L. Wiener and D.C. Nagel (eds.), Human Factors in Aviation, San Diego: Academic Press, Inc., pp. 263-303.

Navon, D. and D. Gopher [1979] "On the Economy of the Human Processing System," Psychological Review, Vol., No. 3, pp. 214-255.

NTSB [1973], Aircraft Accident Report, Eastern Airlines, Inc., L-1011, N310EA, Miami, Florida, December 29, 1972, NTSB-AAR-73-14. Washington, DC: National Transportation Safety Board.

NTSB [1981], Aircraft Accident Report - Eagle Commuter Airlines, Inc., Piper PA-31-350, Navajo Chieftan, N59932, William P. Hobby Airport, Houston, Texas, March 21, 1980, NTSB-AAR-81-4. Washington, DC: National Transportation Safety Board.

NTSB [1985], Aircraft Accident Report - USAir, Inc., Flight 183, McDonnell Douglas DC9-31, N964VJ, Detroit Metropolitan Airport, Detroit, Michigan, June 13, 1984, NTSB/AAR-85/01. Washington, DC: National Transportation Safety Board.

Padulo, L. and M.A. Arbib [1974] System Theory. Washington, DC: Hemisphere.

Rouse, W.B. and J.M. Hammer, [1990], "Computer-Aided Fighter Pilots," IEEE Spectrum, March 1990.

Sheridan, T.B., [1988] "The System Perspective," in E.L. Wiener and D.C. Nagel (eds.), Human Factors in Aviation., San Diego: Academic Press, pp. 27-52.

Wickens, C. D. [1984] "Processing Resources in Attention," in R. Parasuraman and D.A. Davies (eds.), Varieties of Attention. Orlando: Academic Press, pp. 63-102.

Wickens, C.D. and Y. Liu, [1988], "Codes and Modalities in Multiple Resources: A Success and a Qualification," Human Factors, Vol. 30, No. 5, pp. 599-616.

Wiener, E.L. [1987] "Fallible Humans and Vulnerable Systems: Lessons Learned From Aviation," in J.A. Wise and A. Debons (eds.), Information Systems: Failure Analysis, NATO ASI Series, Vol. F32. Berlin: Springer-Verlag, pp. 163-181.