Human Factors Division
Institute of Aviation

University of Illinois
at Urbana-Champaign
1 Airport Road
Savoy, Illinois 61874

EN ROUTE CONTROLLER TASK PRIORITIZATION RESEARCH TO SUPPORT CE-6 HUMAN PERFORMANCE MODELING PHASE II: ANALYSIS OF HIGH-FIDELITY SIMULATION DATA

Esa M. Rantanen, Sharon J. Yeakel, and
Kelly S. Steelman

Final Technical Report
HFD-06-03/ MAAD-06-2
July 2006

Prepared for

Micro Analysis Design Inc.
Boulder, Colorado
Purchase Order NASA MAD 7400.005.01

EN ROUTE CONTROLLER TASK PRIORITIZATION RESEARCH TO SUPPORT CE-6 HUMAN PERFORMANCE MODELING

PHASE II: ANALYSIS OF HIGH-FIDELITY SIMULATION DATA

NASA MAD Purchase Order No. 7400.005.01

FINAL REPORT, APRIL 1, 2005—AUGUST 31, 2006

Esa M. Rantanen, Sharon J. Yeakel, and Kelly S. Steelman
Institute of Aviation, Human Factors Division
University of Illinois at Urbana-Champaign
Willard Airport, Savoy, IL

July 5, 2006

EXECUTIVE SUMMARY

This document describes research to follow up on a previous effort (Rantanen, Levinthal, & Yeakel, 2005) and to provide an empirical foundation for estimation of parameters for air traffic controller performance modeling efforts presently pursued by MA&D within the NASA DAG-TM CE-6 model development. The focus of the project was the task prioritization scheme used in these models. The experiment of Rantanen et al. (2005) tested methods for isolating controllers' tasks for analysis, defining windows of opportunity (WOs) for each of them, and determining prioritization of tasks within a given epoch of time. This groundwork enabled investigation of task prioritization in more realistic setting than the experiment, in the context of full-fidelity evaluation simulations of Future En route WorkStation (FEWS) conducted during the summer of 2005. The simulations were extremely realistic and included a complete set of tasks for the participating controllers to perform. Similar data related to task prioritization as in the Rantanen et al. (2005) experiment were derived from the data collected from these simulations, allowing for a more complete picture of controllers' task prioritization behavior and strategies to be painted.

The results from he present research confirmed some of the findings of Rantanen et al. (2005): As controller workload increased, measured both by different traffic loads imposed on participating controllers in the FEWS simulations as well as by the number of simultaneous tasks available to the controllers in a given time period, they responded later to tasks as they arose and were less consistent in timing of their actions The controllers performance also exhibited diminishing margins of error for too late responses by less time remaining in the WO when tasks were completed as well as less consistent timing of actions as taskload increased.

A new finding was controllers' awareness of the temporal characteristics of their tasks. There was a strong trend to perform tasks that had been 'available' longer, that is, whose WO had opened earlier, before tasks whose WO had been opened later, signifying a 'first come, first served' approach to task prioritization. This trend also seemed to persist as taskload increased. Controllers also seemed to have awareness of the urgency of their tasks as tasks with impending closing of WO were performed before tasks that had longer time available before closing of WO

However, it is clear that task prioritization was also influenced by task characteristics that were separate from the temporal attributes of the tasks. Probabilities for a given task to be performed before another one when both are available simultaneously were calculated by pairing tasks within 30 s epochs, counting the task pairs, and calculating proportions a task was performed before another. Some tasks were clearly prioritized; for example, responding to calls from pilots has inherent urgency and descending arriving aircraft might also serve as a conflict resolution maneuver, which might explain why these tasks were performed more often before conflict resolution tasks than after them.

It is important to keep in mind that these results emerged from aggregation of large amounts of data and that there was significant variability between individual tasks and individual controllers. It is quite clear that innumerable factors must have been present in each prioritization decision made by the participating controllers in the FEWS simulations and that the first come, first served scheme as well as the apparent sense of urgency observed in the results were only two possible factors among many. On the other hand, the fact that these patterns indeed so clearly emerged from the data attests to strength of temporal factors and the controllers' awareness of them. With respect to modeling controller's task prioritization behavior, it appears safe to conclude that no feasible model could possibly consider all factors present in each prioritization decision or individual differences among the decision-makers to make accurate case-by-case predictions of task prioritization. However, considering the temporal task characteristics in prioritization algorithms could very well produce realistic aggregate level predictions of controller performance.

TABLE OF CONTENTS

1. BACKGROUND

1.1. Introduction

This document describes research to follow up on a previous effort (Rantanen, Levinthal, & Yeakel, 2005) and to provide an empirical foundation for estimation of parameters for air traffic controller performance modeling efforts presently pursued by MA&D within the NASA DAG-TM CE-6 model development. The focus of the project was the task prioritization scheme used in these models.

In an experiment conducted by Rantanen et al. (2005) a total of 11 retired FAA controllers and supervisors assigned to the FAA Technical Center's Human Factors Research and Development Laboratory (HFRDL) at Atlantic City International airport, NJ, participated in a part task experimental simulation that presented the participating controllers with several simultaneous tasks in four quadrants on a single display. All events unfolding in the experimental scenarios and controllers' actions were recorded and timed as well. From these data, several dependent variables were derived, focusing on the temporal aspects of controllers' performance and their prioritization of simultaneously available tasks.

The experiment of Rantanen et al. (2005) only considered a limited number of tasks but nevertheless revealed complex interrelationships between them. It is clear that these interactions must be part of any task prioritization scheme used in any HPM efforts. It is equally clear that the HPM should be as complete as possible, containing as many tasks in as many situations as controllers experience and perform them in their jobs as possible. The experiment of Rantanen et al. (2005) tested methods for isolating controllers' tasks for analysis, defining windows of opportunity (WOs) for each of them, and determining prioritization of tasks within a given epoch of time. This groundwork enabled investigation of task prioritization in more realistic setting than the experiment, in the context of full-fidelity ATC simulations.

The Federal Aviation Administration (FAA) conducted during the summer of 2005 evaluation simulations of Future En route WorkStation (FEWS) under three air traffic load configurations at the William J. Hughes Technical Center (WJHTC) HFRDL (Willems, in review). The simulations were extremely realistic and included a complete set of tasks for the participating controllers to perform. Similar data related to task prioritization as in the Rantanen et al. (2005) experiment were derived from the data collected from these simulations, allowing for a more complete picture of controllers' task prioritization behavior and strategies to be painted.

1.2. Background

During the NASA NRA RTO-55 effort (Leiden, 2000), the human performance model (HPM) used a prioritization scheme that assigned task priorities as follows:

1. Short-term conflicts (defined as 'time to conflict' less than 4.5 minutes)
2. Handoffs and metering violations were weighted equally with aircraft 'time to sector transition' used as a tie-breaker
3. Long-term conflicts (defined as 'time to conflict' greater than 4.5 minutes)

As time passes, a long-term conflict waiting to be resolved because of a sequence of handoffs or metering violations eventually will be assigned a higher priority when the time to conflict is less than 4.5 minutes. This scheme was derived through discussions with controllers from the Denver Center.

However, although useful in inhibiting operational errors under high task load, there was no empirical data to support this function.

For ATMSDI CTO-8, new tasks were added to the RTO-55 model. In addition, the times required to complete tasks change significantly based on the concept being modeled (e.g., handoff/TOC with and without data link). Due to CTO-8 schedule constraints, there was not sufficient time to refine the prioritization scheme to account for both of these considerations. Instead, the scheme was replaced with a first in, first out (FIFO) prioritization (based on event detection by the controller) because it did not require any development or testing. Of course, the drawback to a FIFO prioritization is that during high task load a higher numbers of operational errors or missed STA slots could occur in the model than one would expect to see in an equivalent operational setting.

It is apparent that the above scheduling schemes—while expeditious from a modeling point-of-view—are probably much too simplistic to represent the true characteristics of controllers' mental models, temporal awareness, and task scheduling and sequencing performance. While the CE-6 modeling currently underway will develop a task prioritization scheme based on subject matter expert input, the goal of this research is to generate empirical data as well. The accuracy of human performance models could be substantially improved if they were informed by a cognitive model of human timing behavior and performance, which in turn were based on empirical data from relevant tasks and operational environments

1.3. Prior Research

Task network models use human/system task sequence as the primary organizing structure. The individual models of human behavior for each task or task element are connected to this task sequencing structure (Leiden et al., 2001). Task network approach appears a particularly suitable approach to modeling air traffic controllers' jobs, which consist of many tasks with varying degrees of dependency (i.e., some tasks may be independent of each other, others are tightly coupled and must be performed in a strict sequence). As all tasks and subtasks unfold in time, their successful management is primarily a temporal task and the controller's performance is predominantly determined by his or her time management skills and the goodness of his or her temporal mental model (or Level 3 SA) of the situation. Time is hence an attractive variable for investigating the interactions of ATC task load and controller performance as well as a congenital parameter in task network models.

An experiment conducted by Rantanen et al. (2005) developed several measures of controller time-management strategies and performance in temporal terms. These metrics were based on a temporal window of opportunity (WO), defined separately for each individual task presented in the experimental simulations and performed by the participating controllers, and the times the controllers took a first action in the task as well as when the task was completed. Two particular metrics were considered: The time elapsed from the 'opening' of the WO to the time of first action on the task (TFA), and the time remaining in before the WO 'closed' at the time the task was completed (TRm). However, only six subtasks were present in the experimental scenarios, and out of these, WOs could be defined with sufficient reliability and accuracy for only three. The results concerning the tasks of conflict resolution, downlink request, and receiving handoffs are somewhat mixed. Analysis of TFAs for conflict resolution showed that as taskload increased, the participants were less likely to act on conflict resolution at an earliest opportunity than under low taskload conditions. On average, in low taskload conditions the participants first acted on an impending conflict 59 s after the WO for that conflict opened, but waited nearly twice as long under high taskload (mean = 108 s). This shift in mean was associated with increase in variance. The increase in variance is important when we consider the theoretical underpinnings of temporal performance. Indeed, it appears that taskload and resultant mental workload had an impact on controller performance.

A second issue with much greater practical significance is that similar patters were not seen in receiving handoffs and datalink request tasks. As a matter of fact, the trends were opposite, with higher means of TFA in low taskload conditions than under high taskload. A possible explanation to these results may lie in the different nature of the tasks. Receiving handoffs was a simple task that could be accomplished with a single mouse-click. Although the controllers complained about traffic entering the scenarios on conflicting trajectories and made comments to imply that in real life they would have refused the handoff, prompt accepting of handoffs actually became more important under high taskload as it was a prerequisite to control the plane to resolve a conflict. As such, timelier actions on receiving handoff under high taskload than under low taskload can be seen as a reasonable and indeed appropriate response to changing conditions. It must be acknowledged that the changes in TFAs of receiving handoffs were relatively small and not consistent within taskload conditions.

The results with respect to task prioritization were not unequivocal. There was little evidence to support the hypothesis that the controllers' task prioritization was driven by their temporal awareness; either they were not aware of the temporal features of their tasks, or other factors dominated their prioritization decisions. Unfortunately, the issue could not be settled on the basis of the experimental results. It may be hypothesized that task prioritization is driven by task characteristics that are categorical rather than continuous and quantifiable. Support for this conclusion is provided by the very different trends in TFA for the three different tasks analyzed, conflict resolution, receiving handoffs, and responding to downlinked requests. Of these, conflict resolution was clearly the most difficult task, as well as the most important. The difficulty of detecting conflicts as well as the time required for construction and implementation of resolutions to them probably made this task more vulnerable to influences of workload and time pressure than simpler tasks. It must also be remembered that accepting handoffs is, in addition to being a quick and easy task to perform, a prerequisite to subsequent control of the flight (e.g., to implement conflict resolution) and hence the average prioritization between conflict resolution and receiving handoffs is inherently biased towards the latter.

Another aspect worth considering is the nature of the analyses and differences between experimental simulations and realistic situation in operational ATC. Statistics (i.e., minimization of probabilities of both Type I and II errors) is dependent on sufficiently large number of observations, which necessitates aggregation of observations across individual participants and experimental blocks. Yet, even in relatively constrained task environments (e.g., experimental settings) these observations exhibit substantial variability. For example, aggregation of conflict resolution tasks and receiving handoffs in Rantanen et al (2005) did not consider the often unique characteristics of each of these instances. Parsing the data according to such characteristics, however, would severely limit the number of observations available for analysis and undermine the reliability of the results. This is a classical 'Catch-22' situation, and the only remedy is to collect much more data over extended periods of time.

Finally, large differences in performance of individual participants should not be overlooked. These differences were statistically highly significant in almost all analyses we performed and bespeak of inherent variability in working techniques, strategies, and performance of individual controllers working on the same tasks. The distribution-fitting analyses performed by Rantanen et al. (2005) should be particularly helpful in appreciating the full extent of such variability.

1.4. Project Description

The Micro Saint Sharp top-level task network for the events and tasks to be modeled for the Amarillo High sector in the ZFW airspace is depicted in Figure 1. Since airspace events are stochastic in a temporal sense, it is quite common that more than one event is detected during a situation monitoring scan. When this occurs, the controller must prioritize the corresponding tasks and subsequently initiate the highest priority task. In a nutshell, it is this task prioritization scheme that is the focus of this research.

Data for the present research was obtained from the FAA's evaluation simulation of Future En route WorkStation (FEWS) under four levels air traffic load. The simulations used the FAA Target Generation Facility (TGF), an emulator for the HOST computer system and the Display System Replacement (DSR), and the Center Tracon Automation System. The controller environment included full DSR emulations with all operational functions, additional automation functions and modified interfaces (Willems, in review). The simulations were hence extremely realistic and included a complete set of tasks for the participating controllers to perform. Similar data related to task prioritization as in the Rantanen et al. (2005) experiment were derived from the data collected from these simulations, allowing for painting a more complete picture of controllers' task prioritization behavior and strategies under truly realistic conditions.

To support the prioritization scheme, the windows of opportunity must be defined for all controllers' tasks. A partial list of tasks is shown below. A complete list for all tasks in Figure 1 will need to be developed as part of the CE-6 model development. However, only the first six tasks in Figure 1 could be associated with windows of opportunity in the FEWS simulation data.

For CE-6 modeling, the goal is to improve the fidelity of the HPM beyond what was accomplished during CTO-8, which includes an improved prioritization scheme. Although the prioritization scheme will be developed with SME input based on the tasks outlined in Figure 1, a key purpose of this project was to outline a process for generating empirical data to support the prioritization scheme as well. In a modeling environment, time available and time required can be determined much more accurately, based on the WO definitions for each task. Hence, it will be possible to define a near-optimal task prioritization scheme that could be implemented in the model to determine maximum traffic levels prior to task overload. Comparison of controller's actual performance, as quantified by Rantanen et al. (2005) and in results of this research, for example, with this optimal will also allow measurement of the 'goodness' of controllers' temporal mental models and gauge the range of individual differences in temporal performances.

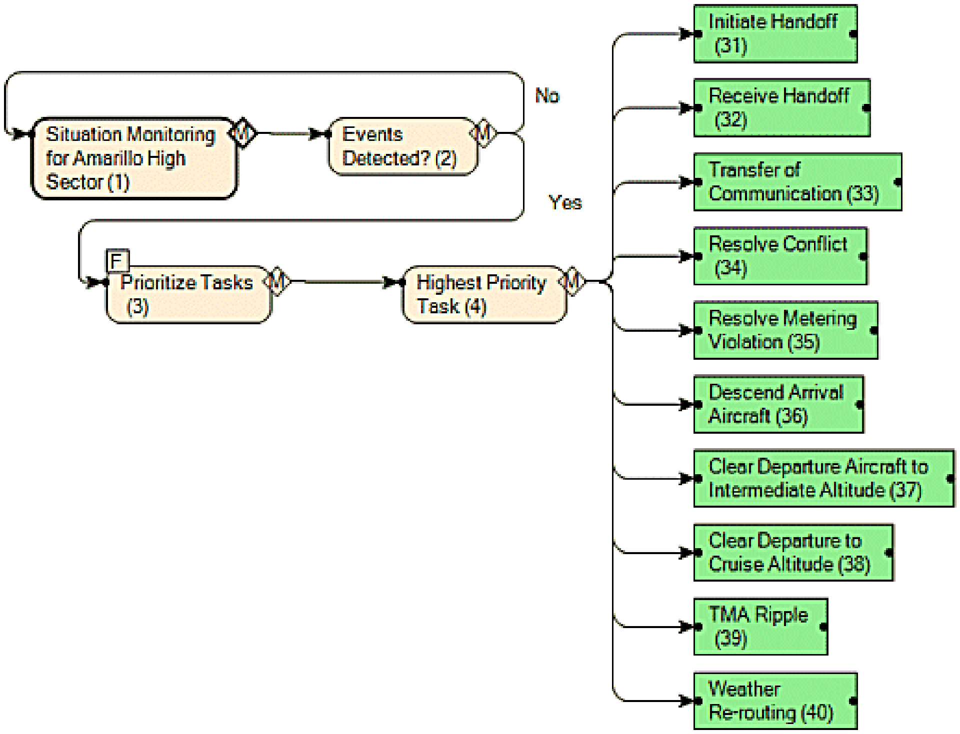

Figure 1. High-level task network for Amarillo sector

1.5. Hypotheses of Task Prioritization Schemes

Five specific hypotheses were formulated concerning possible task prioritization schemes used by controllers:

1. As traffic levels (and, presumably, workload) increase, both mean TFA and variance (standard deviation) of TFA will also increase indicating later responses to tasks as become available and less consistent timing of actions.
2. As traffic levels (and, presumably, workload) increase, mean TRm will decrease and variability in TRm inclease indicating later responses to events to and later completion of tasks, or diminishing margins of error for too late responses, as well as less consistent timing of actions.

The above hypotheses pertain to general awareness of the temporal characteristics of the tasks at hand, and in particular to a breakdown of the ability to anticipate tasks and perform them in a timely and consistent manner at higher levels of workload.

We further hypothesized that if controllers were aware of the temporal characteristics of their tasks and not thwarted by excessive workload, they would have prioritized them according to the following principles:

3. Tasks that have been ‘available’ longer, that is, whose WO had opened earlier, are performed before tasks whose WO had been opened later; this would be signified by longer TFA values of tasks performed before other tasks and describe a ‘first come, first served’ approach to task prioritization;
4. Tasks that are ‘urgent’ by impending closing of WO are performed before tasks that have longer time available before closing of WO; this is signified by shorter TRm values of tasks performed before other tasks.

Finally, task prioritization was examined by individual task pairs to test a final hypothesis:

5. Task prioritization may be driven by task characteristics that are categorical rather than continuous and quantifiable; hence, certain tasks may be consistently and with high probability performed before other tasks, when both have WO open within the same time epoch.

2. METHOD

2.1. FEWS Simulations

The data analyzed in this effort were collected from the FAA’s evaluation simulations of Future En route WorkStation (FEWS) under four air traffic load configurations at the WJHTC RDHFL at Atlantic City International Airport, NJ. The simulations took place during the summer of 2005 and were extremely realistic including a complete set of tasks for the participating controllers to perform.

The simulations were conducted using the FAA Target Generation Facility (TGF), an emulator for the HOST computer system and the Display System Replacement (DSR), and the Center Tracon Automation System. The controller environment included full DSR emulations with all operational functions, additional automation functions and modified interfaces. The simulation airspace was the Genera Center ARTCC with IFR in effect (for a complete description of the simulations see Willems, in review). The simulations involved four different conditions: (1) 100% traffic, (2) 133% traffic, no data link (DL), 133% traffic with DL, and 166% traffic.

2.1.1. Participants

Altogether eight teams of controllers (R-side and D-side controllers) participated in the simulations. Reliable data for analyses were available only from five teams (Teams 3–8). The participants were 16 active Full Performance Level (FPL) controllers, with DSR certification and at least one month DSR experience on DSR release BCC22 including the use of URET/CCLD. The participants came from several different Air Route Control Centers (ARTCCs) within the Continental United States, and had 17.8 (SD=7.2) years of experience on average. Their mean age was 41.8 (SD=5.9) years (Willems, in review). The simulation runs were about 50 minutes in duration.

2.2. Data Processing and Reduction

Complete description of the FEWS data processing and reduction methods is provided in Appendix A; description of a conflict detection algorithm is in Appendix B. The data from the FAA TGF output was first processed by the FAA at the RDHFL into two files per simulation run, (1) Event files and (2) Trajectory files. The former contained a timeline of all controller and pilot actions and events during the run, time-stamped to millisecond accuracy. A separate data processing program was developed at the University of Illinois at Urbana-Champaign to derive metrics of air traffic controller performance and taskload. Altogether seven separate tasks could be identified in these data:

1. Accept handoff (RH);
2. Respond to aircraft radio calls (RC);
3. Standard operating procedure (SP), descend arriving traffic to predetermined coordination altitude;\
4. Resolve conflicts (CR)
5. System data entry (DE)
6. Initiate handoff for all traffic prior to sector boundary (IH);
7. Transfer communications (FR).

For each task, the following temporal variables were calculated:

1. Opening time of Window of Opportunity (WO_open)
2. Closing time of Window of Opportunity (WO_close)
3. Time of first action in the task (FirstAct)
4. Time of last action in the task (LastAct)
5. From (1) and (2), window duration (WO_dur)
   WO_dur = WO_close – WO_open
6. From (3) and (4), task duration, or time required (TR) to perform a task could be calculated: *TR = LastAct – FirstAct.*
7. From (1) and (3), Time to First Action (TFA) could be calculated:
   TFA = FirstAct – WO_open
8. From (2) and (4), the time remaining in a WO (TRm) at task completion could be calculated: *TRm = WO_close – LastAct*; a negative value means that the controller was late in performing the task (e.g., accepting handoff after an aircraft had entered the sector).

These definitions of temporal variables are also illustrated in Figure 2 below. Note that not all of the above variables could be calculated for all tasks. For example, no WO_open times could be defined for initiating handoffs as controllers could do so almost any time of their choosing as long as the aircraft no longer needed their attention.

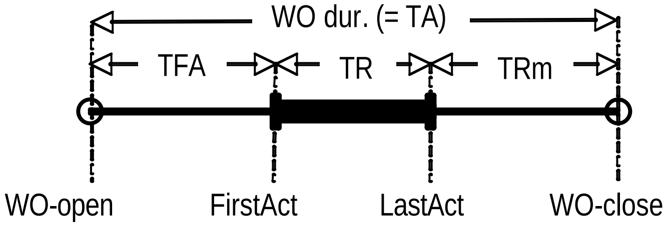

Figure 2. Definitions of the temporal variables used in the analyses. The duration of the window of opportunity (WO) corresponds to time available (TA) for the controller to perform a given tasks. Temporal awareness was examined by the timeliness of actions in each task, or short time to first action (TFA) after opening of thew WO, and ability to complete the task with its given time required (TR) for performance with time remaining (TRm) before closing of the window (positive TRm; negative TRm signified too late action).

2.3. Outliers

The task-relevant data were reviewed for outliers by four derivative variables: (1) duration of the window of opportunity (WO_dur), (2) time to first action after opening of WO (TFA), and (3) time remaining in the WO at the time the action was completed, and (4) duration of action, or time required (TR). The criteria for designating a value as valid were as follows:

Since all the aforementioned variables are dependent on accurate determination of the Windows of Opportunity (WOs) for them, the output from that data reduction program was examined for any discrepancies. The following instances were removed from the data:

1. WO_open = 0; these made the determination of WO_dur and TFA unreliable.
2. TFA < 0; since no tasks could be performed before opening of the WO, there cases were considered unreliable.
3. WO_close = 0; these made the determination of TRm and WO_dur unreliable.
4. By examining the times aircraft spent in the sector, any TFA, TRm, TR, or WO_dur that exceeded 1,200 s (or, 20 min) were considered suspect.

By these criteria, 500 TFA scores (13.81%) and 230 TRm scores (5.03%) were designated as outliers. This left a total of 3120 TFA and 4340 TRm data points for analysis.

3. RESULTS

3.1. The Effect of Temporal Task Characteristics on Task Prioritization

Temporal task characteristics were defined by opening and closing of WO for each individual task and by timing of controllers’ actions on the task within the WO, from which the two main dependent variables , TFA and TRm, were calculated (see Fig. 2 above). We analyzed the data both by the seven different tasks identifiable in the data as well as pooled across tasks. Workload was examined both by the predetermined experimental traffic load conditions as well as by the number of simultaneous tasks available for the controller to perform in a given time interval (epoch).

3.1.1. Individual Tasks

The TFA and TRm values for individual tasks were analyzed by task condition to examine the effect of workload (presumably a function of traffic load) on timely (i.e., small TFA and large TRm values) performance of the tasks. Note that TFA could not be calculated for initiation of handoff (IH) or transfer of communications (FR) tasks, for it was not possible to determine the opening of WO for these tasks; neither TFA or TRm could be calculated for data entry, as no WO could be associated with this task. The results are presented in Table 1. The data were also analyzed by 1-way ANOVAs for TFA and TRm, separately for each task. Pairwise differences between traffic conditions were analyzed by Tukey's tests.

3.1.1.1. Receiving Handoffs (RH)

The TFA for receiving handoffs was much lower in the lowest (100%) traffic condition than under higher traffic loads. A 1-way ANOVA was significant for traffic load, $F(3, 1665) = 268.52$, $p < .001$. However, by Tukey's test, only the 100% condition was different from the 133% condition with data link, $T = 22.77$, $p < .0001$, and without, $T = 23.55$, $p < .0001$, and the 166% condition, $T = 24.21$, $p < .0001$. The results of 1-way ANOVA for TRm were significant for traffic load, $F(3, 1835) = 37.32$, $p < .001$. Tukey's tests showed that the 100% traffic load condition was significantly different from only the 133% condition, $T = 5.62$, $p < .0001$, 133% no-DL condition different from 133% (with DL, as the other conditions), $T = 8.25$, $p < .0001$, and the 133% condition different from 166% condition $T = -9.59$, $p < .0001$. However, when the mean TRm was high in 100% and 133% conditions (115 and 166 s, respectively), and low in the 133% no-DL and 166% conditions (96 and 94 s, respectively), it is difficult to draw any conclusions from these results.

3.1.1.2. Responding to Calls from Aircraft (RC)

Responding to radio calls from aircraft is a task that has some urgency to the controller, both real and perceived; the information transmitted via voice is often time-critical in nature and controllers know that if they do not respond within a reasonably short time, the pilot will call again, adding to frequency congestion and the controller's time stress. It is also clear that as the controller gets busier the time lags in responding to pilots will necessarily increase. Hence, it was hardly surprising to find a very clear, linear trend of increasing TFAs as a function of increasing traffic load. This effect was also statistically significant, $F(3, 566) = 2.85$, $p < .05$. Tukey's pairwise comparisons, however, showed that only the 133% condition was different from the 166% condition, $T = 2.65$, $p < .05$. TRm was also calculated from *opening* of the WO, as no reasonable WO closing time could be estimated; the TRm values contain the duration of the voice message. The results of analyses for TRm were identical to TFA.

3.1.1.3. Standard Operating Procedure: Descend Arrivals (SP)

Results for descending arriving traffic from 1-way ANOVA of TFA for traffic load were significant, $F(3, 494) = 21.25$, $p < .001$. However, the trend is difficult to explain, as the longest TFAs were for the lowest and highest traffic load condition and the shortest TFAs (implying timely attention to the task) for the two 133% traffic load conditions. Tukey's pairwise comparisons showed that the 100% traffic condition was significantly different from both 133% and 133% no-DL conditions, $T = -4.32$, $p < .001$, and $T = -3.25$ $p < .01$, which were different from tbe 166% condition, $T = 5.05$, $p < .0001$, and $T = 7.29$, $p < .0001$. The results on TRm were very similar, overall statistically significant, $F(3, 450) = 10.4$, $p < .001$, but significant differences only between the two medium-traffic load (133% and 133% no-DL) and high traffic load (166%) conditions, $T = -3.31$, $p < .01$, and $T = -5.4$, $p < .0001$ (Tukey's test).

3.1.1.4. Resolving Conflicts (CR)

Conflict resolutions showed a clear and linear trend of increasing TFAs as traffic load increased. This effect was significant, too (1-way ANOVA), $F(3, 382) = 10.65$, $p < .001$. Tukey's pairwise comparisons revealed that the 100%, 133%, and 133% no-DL conditions were all different from the 166% condition, $T = 4.47$, $p < .0001$, $T = 3.15$, $p < .01$, and $T = 4.32$, $p < .0005$, respectively, but not from each other. This trend was not repeated in TRm, however, with no significant differences between conditions but in all cases actions for resolving conflicts were completed over 500 s (over 8 min) before loss of separation.

Table 1

Time to First Action (TFA) and Time Remaining (TRm) in a Window of Opportunity (WO) by task and traffic load. . Key: RH = Receive Handoff, RC = Respond to Aircraft Call, SP = Standard Operating Procedure, descend arrivals, CR = Conflict Resolution, DE = System Data Entry, IH = Initiate Handoff, FR = Transfer Communication; Traffic Load: 101 =100%, 130 = 133% no-DL, 131 = 133%, and 161 = 166%.

		TFA (s)				TRm (s)			
Task	Tfc Load	101	130	131	161	101	130	131	161
RH	N	375	368	409	517	305	384	570	580
	Mean	70.9	509	483.8	487.4	115.6	96.7	166.9	94
	StDev	147.4	266.1	267	291.6	67.7	87.9	156.2	145.4
RC	N	70	207	147	146	70	207	147	146
	Mean	6.02	8.84	5.98	16.74	-9.76	-12.7	-9.73	-20.25
	StDev	4.56	12.7	4.65	66.86	4.78	12.63	4.78	66.92
SP	N	29	220	127	122	23	212	120	99
	Mean	281.9	131.4	164.2	276.5	266.4	430.5	364.5	221.5
	StDev	232.2	155.3	160	209.8	317.7	307.9	330.4	323.9
CR	N	83	87	52	164	97	103	60	207
	Mean	275	287.1	303.3	414.9	501.6	556	570.3	505.3
	StDev	228.1	227.5	219.6	218.1	278.5	305.5	306.1	314.1
IH	N					83	93	95	143
	Mean					262.8	187.8	190.1	143.7
	StDev					171.7	143.4	316.7	271.2
FR	N					57	258	125	89
	Mean					40.7	8.9	107.4	-5.1
	StDev					131.2	83.8	276.3	93.5

3.1.1.5. Initiating Handoffs (IH)

A similar task to accepting handoffs, initiating handoffs before aircraft exit the sector (and the WO closes), the TRm showed similar trend, that is, decreasing as traffic load increased, or handoffs being initiated closer to the sector boundary at higher traffic loads. This trend was significant, $F(3,382) = 4.21$, $p < .01$. However, Tukey's tests showed that only the 100% traffic load condition was significantly different from the 166% condition, $T = -3.55$, $p < .005$.

3.1.1.6. Transfer of Communications (FR)

Transfer of communications showed a somewhat very similar trend as handoff initiation, with decreasing TRm as traffic load increased, to the point where in the 166% traffic condition the controllers were on average 5 s late (i.e., after the aircraft had crossed into the next sector) in transferring communications. A 1-way ANOVA on TRm for traffic load was significant, $F(3, 525) = 13.11$, $p < .001$. An odd finding was a much longer mean TRm in the 133% condition (107 s) than in the other conditions, and only this was significantly different from the others (Tukey's test).

3.1.2. Pooled Tasks

Task prioritization was analyzed by dividing the simulation duration into short time epochs and then examining the order different tasks were performed within an epoch. The epoch length in this analysis was 10 s, which is arbitrary but sufficiently short to consider tasks within epoch to be simultaneously available to the controller but not too long to include too many tasks. A resampling technique was used by shifting the epoch by 5 s and recounting tasks within these new epochs. These data were then combined. Hence, the tasks were paired with different tasks within the different 10 s epochs. The tasks were then numbered within their respective epochs ordered by the time of First Action on a task (i.e., within an epoch, a task with smallest First Action time was labeled 1, the task with the next First Action time 2, etc.). Epochs with zero or only one task within them were excluded from analysis. Figure 3 illustrates this data processing technique and derivation of the dependent variables.

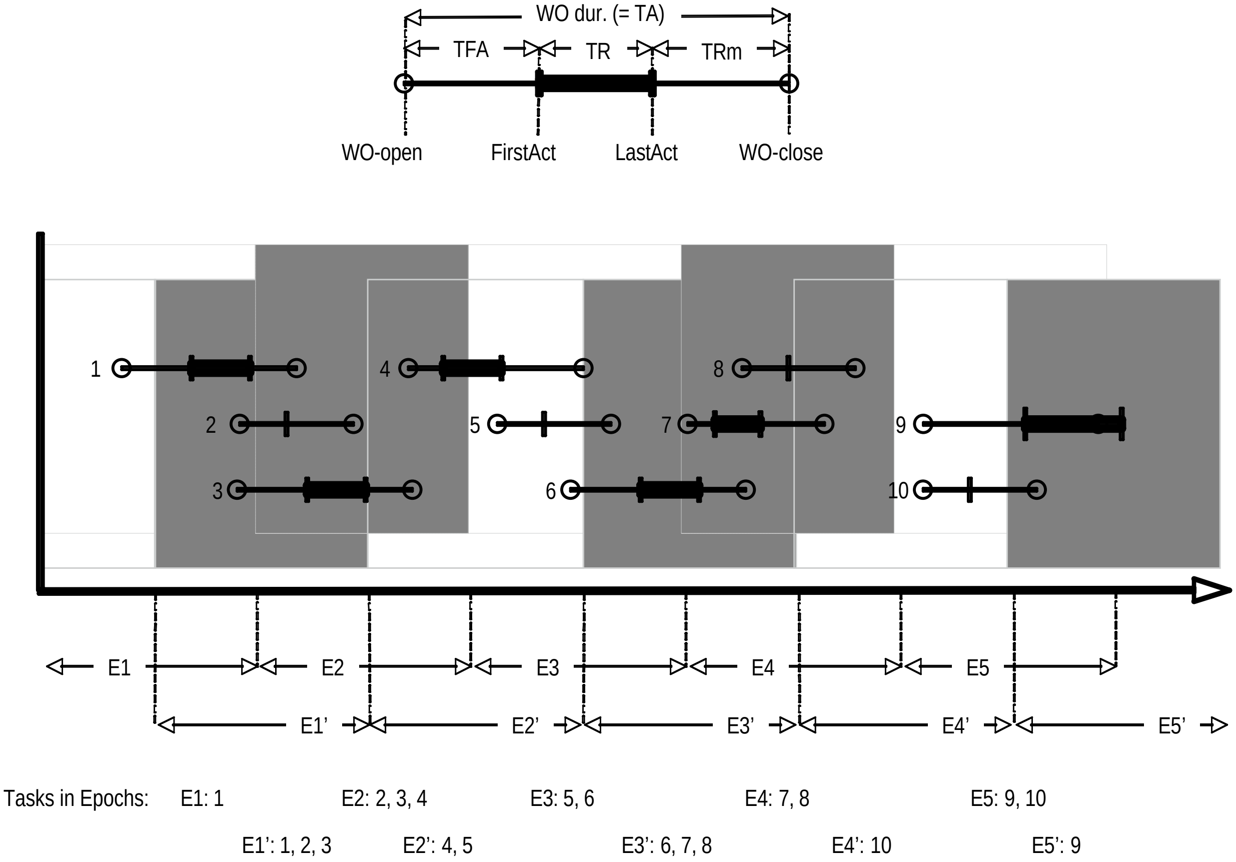

Figure 3. Definitions of the dependent variables and counting and ordering of tasks within epochs, including resampling by shifting the epochs. Tasks were assigned into epochs by the first action times.

The TFA data were transformed separately for each task by subtracting the mean and dividing by standard deviation into z-scores, then pooled across all tasks. The transformation was necessary due to large differences between TFA values between tasks. The transformed data were analyzed separately for TFA and TRm by task order.

The data were also analyzed by taskload in two ways. The first was by the experimentally manipulated traffic load at four levels (100%, 133% with and without DL, and 166%). In another analysis we pooled the data across these experimental conditions but separated epochs with two simultaneous tasks from those with three or more tasks. From the latter group we also excluded epochs with five or more tasks because of the influence of their disproportionately small *n* would have had on the ANOVAs performed; the number of data points by the number of tasks per epoch is given in Table 2 below. The epochs with only two tasks represented a low taskload condition and epochs with three or more tasks a high taskload condition.

Table 2
Number of epochs available for analysis by the number of tasks per epoch; epochs with five or more tasks were excluded due to their disproportionately small n.

No. Simultaneous Tasks	2	3	4	5	6	7
Epochs (data points	3090	1623	684	(195)	(24)	(28)

3.1.2.1. Effect of traffic load condition

The effect of the experimental traffic load condition on TFA and TRm was firs analyzed by a mixed-model ANOVA with team as a random factor and condition and task order as fixed factors. Pairwise differences between traffic load conditions were analyzed by Tukey's tests. The ANOVA showed significant differences between teams, *F*(5, 4051) = 15.05, $p < .001$, conditions, *F*(3, 4051) = 64.13, $p < .001$, and between successive tasks within epochs, *F*(3, 4051) = 4.05, $p = .007$. These trends can also be seen in Figure 4. Tukey's pairwise test showed that the 100% traffic load resulted in significantly lower TFA than in any other condition, $p < .001$; the 133% no DL traffic condition also had significantly lower TFA than the 133% with DL or 166% conditions, $p < .005$, but the latter two were not significantly different from each other.

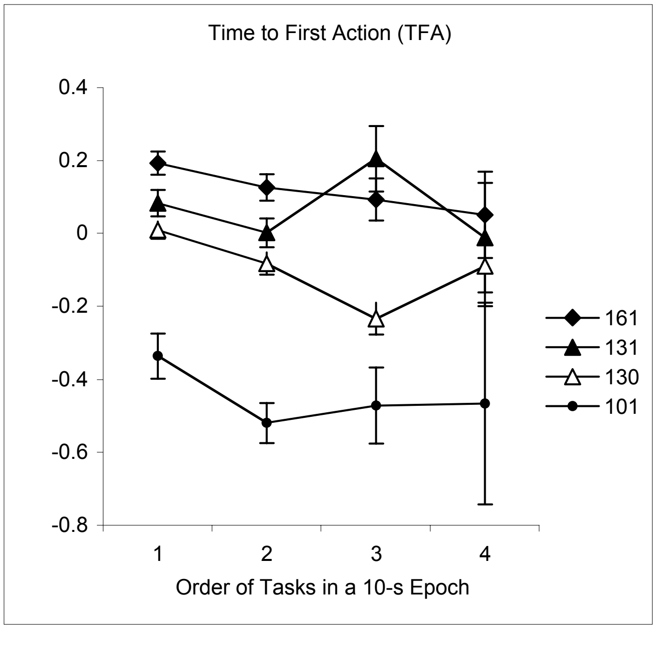

Figure 4. The effect of traffic load (key: 101 = 100%, 130 = 133% without DL, 131 = 133% with DL, and 161 = 166%) on time to first action (TFA) from opening of a window of opportunity (WO). The downward trends apparent in all traffic conditions suggest a first come, first served task prioritization scheme employed by the controllers. This trend was statistically significant on the aggregate level of the data, but not within each traffic load condition. The error bars represent one standard error of the mean.

These results corroborate those from the previous analysis, that increasing workload (as a result of taskload) will result in later actions in each task. Another important trend to note is the decreasing TFA by successive tasks within an epoch, suggesting a ‘first come, first served’ scheme of task propritization.

Differences between successive tasks were analyzed by separate ANOVAs within each traffic load condition. None of these analyses indicated significant reductions in TFA as a function of task order, or significant differences between successive tasks (except for the 133% no-DL condition, between the first and third tasks, $T = -3.89$, $p = .0006$). However, the trends towards lower TFAs with successive tasks in an epoch is quite clear.

Similar analyses were performed on TRm. An ANOVA revealed significant differences between teams $F(5, 4051) = 19.8$, $p < .001$, and traffic load conditions, $F(3, 4051) = 72.24$, $p < .001$. Pairwise comparisons showed that the 100% traffic condition was different from all the others, 133% no DL condition was different from 166% condition, and that the 133% DL condition was different from the 166% condition. The pooled data showed thus an opposite trend than when examined by individual tasks in that the lowest taskload condition (100% traffic) had the smallest TRm values; the effect magnitude was not very big, however (see Figure 5).

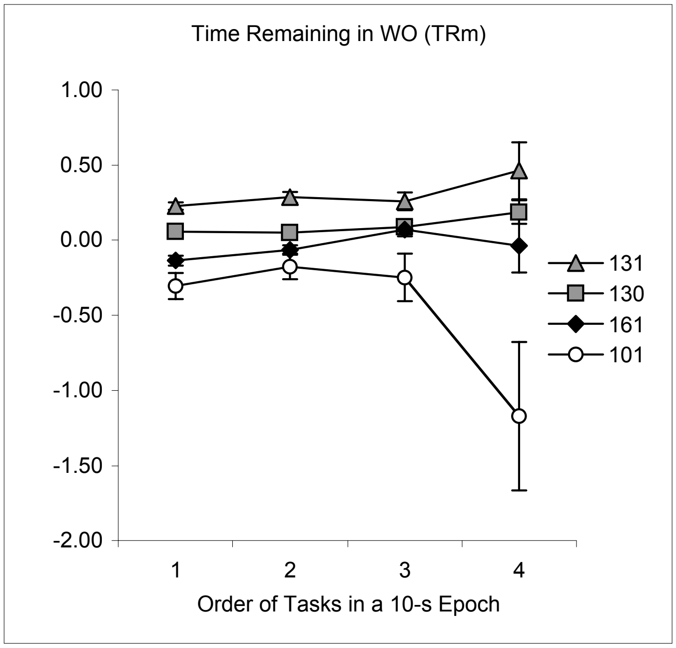

Figure 5. The results on time remaining (TRm) in a window of opportunity (WO), pooled across separate tasks, show trends opposite to those observed on individual tasks. A slight upward trend is apparent within the traffic load conditions, however, suggesting that controllers tended to perform tasks which were more urgent (i.e., had a smaller TRm) before tasks with more time remaining. The significant ‘dip’ in the 100% condition and fourth task in an epoch is associated by a large standard error, resulting from few data points in that particular class. The error bars represent one standard error of the mean.

We also examined the effect of order of tasks within an epoch within each traffic load condition. In the 100% condition, both team and task order had a significant effect on TRm, $F(5, 543) = 34.93$, $p < .001$, and $F(3, 543) = 2.64$, $p = .049$, respectively. TRm for first, second, and third tasks were significantly different from that for the fourth task (Tukey’s test, but given the large variability in the latter condition, no firm conclusions should be drawn from this result. The results were similar for the

133% no-DL condition, with both team, *F*(3, 1265) = 9.27, *p* < .001 and task order, *F*(3, 1265) = 3.66, *p* = .012, significant. First and second tasks were also significantly different from the fourth by Tukey's tests. For the 133% *cum* DL condition only team was significant, *F*(5, 967) = 16.29, *p* < .001; no pairwise significant differences between successive tasks were found. Finally, in the 166% condition, both team and task order were significant, *F*(5, 1254) = 13.16, *p* < .001, and *F*(3, 1254) = 5.25, *p* = .001, respectively. The TRm for the first and second tasks were also significantly different from that for the third task (Tukey's test).

3.1.2.2. Effect of taskload per epoch.

Another way to examine the effect of taskload was to pool the data across the traffic load conditions but to analyze them separately for epochs with two and three or more tasks within them. The effects observed before were much more pronounced in these analyses. The TFA shows steeply declining TFA values from the first-performed task to the subsequent tasks, attesting to the first come, first served prioritization scheme; the TFA values were also higher for the epochs with more than two tasks, showing that the controllers were performing their tasks later under higher taskload (Figure 6).

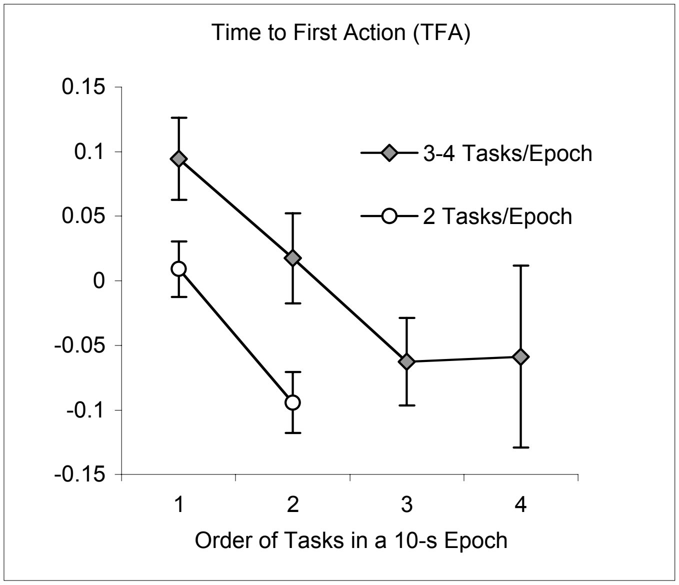

Figure 6. Results from data pooled over traffic load conditions but separated by the number of tasks within a 10-s epoch. The first come, first served task prioritization scheme is apparent in the decreasing time to first action (TFA) values for successive tasks; the controllers were also later in performing their tasks under higher taskload (3-4 tasks per epoch) than under lower taskload (two tasks per epoch). The error bars represent one standard error of the mean.

The data were analyzed separately for each taskload condition by a mixed-model ANOVA (similar to previous analyses). For the two-task condition, both team and task order were significant, *F*(5, 2328) = 8.86, *p* < .001, and *F*(1, 2328) = 6.34, *p* = .012, respectively. The results were very similar for the 3-4 task condition, team being significant at *F*(5, 1719) = 8.53, *p* < .001,, and task order at *F*(3, 1719) = 3.09, *p* = .026. However, only the TFA for the first task was significantly different from the third one, *T* =.2.59, *p* = .016. However, the decreasing trends of TFA are very clear.

Similar analyses were done on TRm. The role of task urgency in task prioritization is quite clear in these results; tasks with less time remaining were on the average performed before those with more time, hence the upward trend in TRm as a function of task order. It is also noteworthy that under higher taskload the tasks were performed with substantially less time remaining (and in fact quite often late, with negative TRm values) than under lower taskload (see Figure 7).

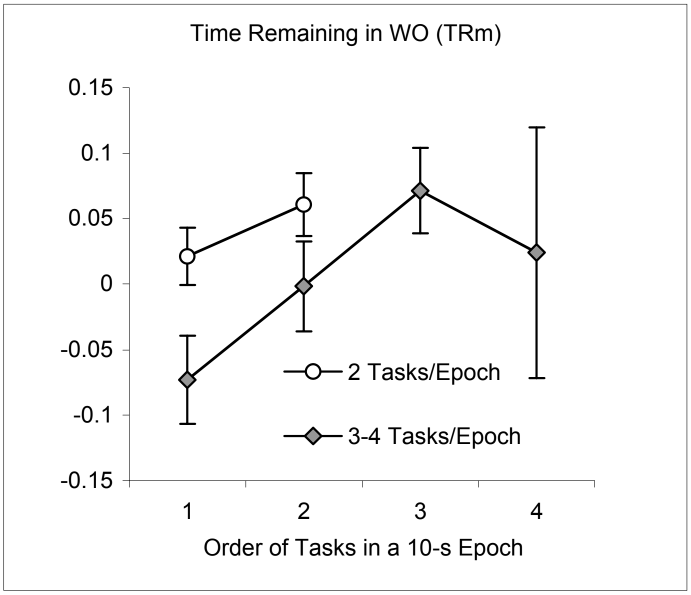

Figure 7. The time remaining (TRm) in the window of opportunity (WO) at the time a task was completed shows that controllers tended to perform more urgent tasks (small TRm) before those with more time. They were also later in their performance under high taskload (3-4 tasks in an epoch) than under lighter taskload (2 tasks in an epoch). The error bars represent one standard error of the mean.

Despite the clear trends present in Figure 7, the effect of task order on TRm was not significant in the two tasks per epoch condition ($F < 1$); team, however, was significant, $F(5, 1719) = 17.23$, $p < .001$. Task order was significant in the higher taskload condition, $F(3, 1719) = 3.01$, $p = .029$, but only the first task was significantly different from the third task in pairwise comparison (Tukey's test), $T = 2.96$, $p = .016$. Team was also significant in this analysis, $F(5, 1719) = 9.2$, $p < .001$.

3.2. Task Prioritization by Task

In addition to the temporal task characteristics influencing task prioritization, it is plausible to hypothesize that certain tasks may simply demand priority by their inherent nature regardless of their temporal parameters. To calculate probabilities for a given task to be performed before another one when both are available simultaneously, tasks were paired within 30 s epochs (30 s was chosen for epoch length rather than 10 s so that as many different pairings of tasks as possible could be obtained). Task pairs were then counted, and from these counts, proportions a task was performed before another calculated. The results are presented in Table 3 by task pairs and traffic load conditions.

Table 3

Probabilities (proportions) a task was performed before another task by an R-Side controller when both were available (i.e., their respective windows of opportunity were 'open') simultaneously within 30 s epochs by task pairs. Key: RH = Receive Handoff, RC = Respond to Aircraft Call, SP = Standard Operating Procedure, descend arrivals, CR = Conflict Resolution, DE = System Data Entry, IH = Initiate Handoff, FR = Transfer Communication.

Traffic Load:		101		130		131		161		All	
Task 1,	2	Prop.	N	Prop.	N	Prop.	N	Prop.	N	Prop.	N
RH	RC	0.708	34	0.648	166	0.589	66	0.640	105	0.640	371
RC	RH	0.292	14	0.352	90	0.411	46	0.360	59	0.360	209
	Sum	1.000	48	1.000	256	1.000	112	1.000	164	1.000	580
RH	SP	0.615	8	0.616	125	0.588	67	0.504	60	0.579	260
SP	RH	0.385	5	0.384	78	0.412	47	0.496	59	0.421	189
	Sum	1.000	13	1.000	203	1.000	114	1.000	119	1.000	449
RH	CR	0.639	46	0.595	44	0.559	19	0.383	79	0.487	188
CR	RH	0.361	26	0.405	30	0.441	15	0.617	127	0.513	198
	Sum	1.000	72	1.000	74	1.000	34	1.000	206	1.000	386
RH	IH	0.000	0	1.000	1	0.800	4	0.318	7	0.387	12
IH	RH	1.000	3	0.000	0	0.200	1	0.682	15	0.613	19
	Sum	1.000	3	1.000	1	1.000	5	1.000	22	1.000	31
RH	FR	0.636	7	0.000	0	0.500	1	0.889	8	0.696	16
FR	RH	0.364	4	1.000	1	0.500	1	0.111	1	0.304	7
	Sum	1.000	11	1.000	1	1.000	2	1.000	9	1.000	23
RC	SP	1.000	6	0.790	109	0.870	40	0.868	33	0.825	188
SP	RC	0.000	0	0.210	29	0.130	6	0.132	5	0.175	40
	Sum	1.000	6	1.000	138	1.000	46	1.000	38	1.000	228
RC	CR	1.000	20	0.653	32	0.750	9	0.857	36	0.789	97
CR	RC	0.000	0	0.347	17	0.250	3	0.143	6	0.211	26
	Sum	1.000	20	1.000	49	1.000	12	1.000	42	1.000	123
SP	CR	0.529	9	0.681	49	0.852	23	0.700	49	0.699	130
CR	SP	0.471	8	0.319	23	0.148	4	0.300	21	0.301	56
	Sum	1.000	17	1.000	72	1.000	27	1.000	70	1.000	186
CR	IH	NA	0	NA	0	NA	0	0.571	4	0.571	4
IH	CR	NA	0	NA	0	NA	0	0.429	3	0.429	3
	Sum	NA	0	NA	0	NA	0	1.000	7	1.000	7
IH	FR	0.438	7	0.435	30	0.389	7	0.296	8	0.400	52
FR	IH	0.563	9	0.565	39	0.611	11	0.704	19	0.600	78
	Sum	1.000	16	1.000	69	1.000	18	1.000	27	1.000	130

4. DISCUSSION

We posited several hypotheses about controllers' task prioritization schemes and the effect of temporal characteristics of their tasks on their overall performance. All of these hypotheses were generally supported by the data.

(1) Both mean TFA and variance (standard deviation) of TFA generally increased as a function of increased traffic load indicating later responses to tasks as they arise and less consistent timing of actions (Table 1). There were, however, large differences between tasks as well as inconsistencies between the traffic load conditions. In particular, descending arriving traffic did not show the hypothesized effects. It is also difficult to know what, if any, difference the absence of DL made in terms of taskload in the 133% traffic conditions. Hence, conclusions are best drawn from comparison of the 100% and 166% conditions, and these generally support the hypothesis.

(2) The results for TRm also generally support the hypothesis that decrease in the mean and increased variability in TRm indicate diminishing margins of error for too late responses as well as less consistent timing of actions as taskload increases. The main exceptions were again descending arriving traffic and conflict resolutions.

(3) Controller awareness of the temporal characteristics of their tasks was manifested in the very strong trends to perform tasks that had been 'available' longer, that is, whose WO had opened earlier, before tasks whose WO had been opened later (Figures 4 and 6). This scheme signifies a 'first come, first served' approach to task prioritization. This trend also seemed to persist as taskload increased, whether the latter was measured by traffic load in the experimental conditions or momentarily by the number of tasks to be performed in an epoch.

(4) Controllers also seemed to have awareness of the urgency of their tasks. Tasks with impending closing of WO were performed before tasks that had longer time available before closing of WO (Figures 5 and 7). This is signified by shorter TRm values of tasks performed before other tasks. Taskload had a similar effect on this measure as on TFA, with the generally later (by TFA) actions on tasks under higher taskload resulting in less time remaining in the WO at the time tasks were completed than under lighter taskload.

(5) Finally, it is clear that task prioritization was also influenced by task characteristics that were separate from the temporal attributes of the tasks (Table 3). Responding to calls from pilots has inherent urgency for reasons described before. It should also be noted that descending arriving aircraft might also serve as a conflict resolution maneuver, which might explain why these tasks were performed more often before conflict resolution tasks than after them.

As exciting as it was to discover evidence of temporal awareness influencing controllers task prioritization schemes, it is important to keep in mind that these results emerged from aggregation of large amounts of data and that there was significant variability between individual tasks and individual controllers. It is quite clear that innumerable factors must have been present in each prioritization decision made by the participating controllers in the FEWS simulations and that the first come, first served scheme as well as the apparent sense of urgency observed in the results were only two possible factors among many. On the other hand, the fact that these patterns indeed so clearly emerged from the data attests to strength of temporal factors and the controllers' awareness of them. With respect to modeling controller's task prioritization behavior, it appears safe to conclude that no feasible model could possibly consider all factors present in each prioritization decision or individual differences among the decision-makers to make accurate case-by-case predictions of task prioritization. However, considering the temporal task characteristics in prioritization algorithms could very well produce realistic aggregate level predictions of controller performance.

REFERENCES

Leiden, K (2000). *Human Performance Modeling of En Route Controllers*. NASA RTO-55 Final Report. Boulder, CO: Micro Analysis & Design, Inc.

Leiden, K., Laughery, K. R., Keller, J., French, J., Warwick, W., & Wood, S. D. (2001). *A review of human performance models for the prediction of human error* (Report prepared for NASA System-Wide Accident Prevention Program). Boulder, CO: Micro Analysis & Design, Inc.

Rantanen, E. M., Levinthal, B. R., & Yeakel, S. J. (2005). *En Route Controller Task Prioritization Research to Support CE-6 Human Performance Modeling* (AHFD-05-03/MAAD-05-3). Savoy, IL: Aviation Human Factors Division.

Willems, B. (In review). *Future En route WorkStation (FEWS)*. [DOT/FAA/TC-TR (number)]. Atlantic City International Airport, NJ: FAA William Hughes Technical Center.

APPENDIX A

A1. Data Reduction Procedure and Algorithms

The data from the FAA TGF output was processed by the FAA at the RDHFL into two files per simulation run, (1) Event files and (2) Trajectory files. The former contained a timeline of all controller and pilot actions and events during the run, time-stamped to millisecond accuracy. A separate data processing program was developed at the University of Illinois at Urbana-Champaign to derive metrics of air traffic controller performance and taskload.

A1.1. Task List

The following list contains all controllers' tasks that can be identified in the data and thus recorded separately by the algorithms specified here:

1. Accept handoff of aircraft entering the sector (ZGN Sector 08)
2. Respond to aircraft radio calls or downlinked messages in the DL condition
3. Descend GEN arrival traffic routed via GEN1 or SFG2 Star to Sector 18 to FL230. CHIGO/IND transition aircraft shall be descended prior to R22 high boundary.
4. Descend eastbound OHO or DETRO landing traffic and DESMN and KANCY arrivals to FL220 as soon as practicable.
5. Resolve all conflicts; conflicts are defined as potential violations of the 1000 ft vertical and 3 nm horizontal separation minima.
6. Update the radar system
7. Hand off all aircraft prior to sector boundary.
8. Transfer communications to the next sector.

Processing of some tasks required additional coding in the Event File; two new columns were added, a column (new column I) titled CommCode and (new column J) CommCont:

a. CommCode: A code for communication type, (1) Init = initiating communication, (2) Clr = clearance, and (3) Ack = acknowledgement.

b. CommCont: Content of the communication will be coded by a letter followed by a number, or a string of letters in case or routing, or a decimal number in case of frequency (in the following, all numbers are 9 and letter X): (1) A999 = altitude, (2) H999 = heading, (3) S999 = speed, (4) RXXXX = route, (5) 999.999 = frequency

A1.2. General Structure of the Data Reduction Program

The general structure and flow of information in the specified data reduction program is depicted in Figure 1 below. The program shall read pertinent information as specified for each task from three different source files and output the results into a single file where each task shall occupy a single row.

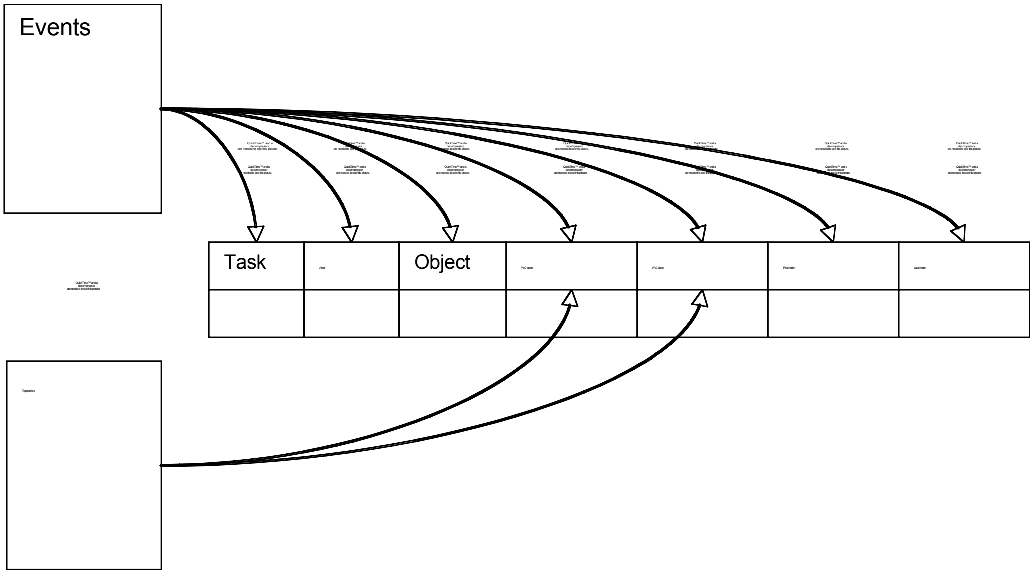

Figure 1. The data reduction program shall read the Event files and Aircraft State files and output the reduced data into a spreadsheet as depicted above. Towards this end, all contents in each column of the output file must be carefully defined in terms of relevant variables in the input files.

A1.3. Filename Syntax and Decoding

The data file names were in a form of a string of letters and numbers, for example T6L133EB13N, which was parsed thus:

T6 = Team 6
L133 = 133% traffic load
E = Experimental simulation run
B = Baseline condition
13 = 13th scenario controllers had worked (including training scenarios)
N = No data link available/

As the data reduction program reads the file, each output row contained the information from the filename in the first 6 columns (note column headings):

A	B	C	D	E	F
Team	TFC_Load	Sim-Run	Cond	Scen	DL
6	133	E	B	13	N

A1.4. Task 1: Receive Handoff (RH)

All necessary data are in the Event Files; the table below specifies the events relevant to the task, keywords identifying the events, corresponding column headings and actor and object words:

Event	File	Keyword [Column Heading]	Actor	Object
Window Opens	EventFile	Flashing [Action]	Callsign	N/A
Window Closes	EventFile	Enters Sector [Action]	Callsign	N/A
Before Fixation	EventFile	Fixates [Action]	R-side	Callsign
After Fixation	EventFile	Fixates [Action]	R-side	Callsign
First Action	EventFile	AcceptsHandoff [Action]	R-side	Callsign
Last Action	EventFile	AcceptsHandoff [Action]	R-side	Callsign
BcnCode	Trajectory File			

Example: The data file (Event File) reads as follows:

```
Time    Actor            Action         Object         Content
2072.722      AAL6006       Flashing
2076.203      R-Side        Fixates                    AAL6006
.
2093.418      R-Side        AcceptsHandOff             AAL6006
..
2182.000      AAL6006       Enters Sector
```

From these data, the program produced an output as follows:

G	H	I	J	K	L	M	N	O
Task	Actor	Object	BcnCode	W_open	W_close	BeforeFix	AfterFix	FirstAction
Accept_Handoff	R-Side	AAL6006	4342	2072.722	2182.000		2076.203	2093.418

P	Q	R	S
LastAction	Action	Complete	Route
2093.418			

A1.5. Task 2: Respond to Aircraft Radio Calls (RC)

This task required manual coding of aircraft call signs from pilot transmission transcripts; the call sign was entered in the ‘Object’ column. This task was aided by Excel automatically filling in a call sign based on call signs in this column elsewhere in the spreadsheet

Event	File	Keyword(s) [Column Heading]	Actor	Object
Window Opens	Event File	Says [Action]	Pilot	Callsign
Window Closes	Event File	Says [Action]	N/A	N/A
First Fixation	Event File	Fixates [Action]	R-side	Callsign
First Action	Event File	[CommCode]	R-side	Callsign
Last Action	Event File	Says [Action + Duration]	R-side	Callsign
Action	Event File	[CommCode] + [CommCont]	R-side	Callsign

Example: The data file (Event File) reads as follows:

```
Time      Actor    Action   Object   Content  CommCode  CommCont  Duration
832.800   Pilot    Says     AAL1001  N/A      Init      A310      2.83
836.000   R-Side   Says     AAL1001  N/A      Ack       A310      4.41
836.243   R-side   Fixates  AAL1001
```

From these data, the program produced an output as follows:

G	H	I	J	K	L	M	N	O
Task	Actor	Object	BcnCode	W_open	W_close	BeforeFix	AfterFix	FirstAction
Resp_Call	R-side	AAL1001	4342	832.800	832.800		836.243	836.000

P	Q	R	S
LastAction	Action	Complete	Route
840.410	Ack+A310		

A1.6. Task 3: SOP--Descend GEN Arrival Traffic (SP)

Descend GEN arrival traffic routed via GEN1 or SFG2 Star to Sector 18 to FL230. CHIGO/IND transition aircraft shall be descended prior to R22 high boundary.

Event	File	Keyword(s) [Column Heading]	Actor	Object
Window Opens	EventFile	AcceptsHandoff [Action]	RPOS	Callsign
OR Window Opens	N/A	Conflict-free	N/A	N/A
Window Closes	Trajectory File	Aircraft is at Top Of Descent (TOD)	N/A	N/A
First Fixation	N/A	N/A	N/A	N/A
First Action	Event File	Clr [CommCode] A(value x 100) {E} < [Alt_ft] at time AcceptsHandoff	R-side	Callsign
Last Action	Event File	Clr [CommCode] A(value) {E} = 230	R-side	Callsign
Action	Event File	[CommCode] + [CommCont] at First Action	R-side	Callsign
Complete	Trajectory File	[Alt_ft] ≤ 23000	N/A	N/A
Route	Trajectory File	Filed		Callsign

Note that the WO_open time is *either* when the aircraft AcceptsHandoff or when the aircraft's trajectory is conflict-free, as determined by a conflict detection algorithm. Note also that the time stamps in the Event file are in seconds to 3 decimal places and in Trajectory file in HH:MM:SS.

The algorithms for outputting the Task 3 data are as follows; the letters in brackets indicate the input file, {E} for Event file and {T} for trajectory file, or whether the variable is a constant {C} or to be calculated by the program {c}

```
current altitude at the LastAction time for *AcceptHandoff*
IF (callsign) flight plan [Filed] {T} has CHIGO..GEN OR DARIO..GEN THEN
    WO_opens at time when aircraft (callsign) AcceptsHandoff {E}
OR (callsign) trajectory is conflict-free;
    WO_closes at time when aircraft (callsign) is at TOD (Top Of Descent);
desired_Alt = 23000 {C}
delta_Alt {c} = current_Alt {T} - desired_Alt {C}
VS_nom = 3000 ft/min {C} (50 ft/sec)
        TOD {c} = Exits sector {E} – (delta_Alt/VS_nom) {c}
FirstAction is at time when [CommCode] = Clr {E}
AND [CommCont] = A(value x 100) {E} < [Alt_ft] at time AcceptsHandoff {T}
LastAction is at time when [CommCode] = Clr {E}
AND [CommCont] = A(value) {E} = 230
Complete is at time when aircraft (callsign) [Alt_ft] {T} ≤ desired_Alt {C}
```

Example:

From Trajectory file (note that the time stamps are in hh:mm:ss):

```
Time      Acid     ..   Alt_ft       ..   Filed
00:16:00  AAL1001  ..   31000.213    ..   IDA120080.BUTTE.INDIN.CHIGO.GEN1.GEN
..
00:26:00  AAL1001  ..   22999.66492  ..   IDA120080.BUTTE.INDIN.CHIGO.GEN1.GEN
```

From Event file (note that the time stamps are in seconds up to 3 decimal places):

```
Time      Actor    Action          Object   Content  CommCode  CommCont  Duration
959.000   AAL1001  AcceptsHandoff
..
1202.65   R-Side   Says            AAL1001  N/A      Clr       A230      3.48
..
1709.000  AAL1001  Exits Sector
```

From these data, the program produce an output as follows (times in seconds to 3 decimal places):

G	H	I	J	K	L	M	N	O
Task	Actor	Object	BcnCode	W_open	W_close	BeforeFix	AfterFix	FirstAction
SOP	R-side	AAL1001	4342	959.000	1709.000			1202.650

P	Q	R	S
LastAction	Action	Complete	Route
1455.56	Clr+A230	1560.000	

A1.7. Task 4: SOP--Descend OHO, DETRO, DESMN, and KANCY Arrivals

Descend eastbound OHO or DETRO landing traffic and DESMN and KANCY arrivals to FL220.

Event	File	Keyword(s) [Column Heading]	Actor	Object
Window Opens OR	EventFile	AcceptsHandoff [Action]	RPOS	Callsign
Window Opens	N/A	Conflict-free	N/A	N/A
Window Closes	Event File	Aircraft is at Top Of Descent (TOD)	Callsign	N/A
First Fixation	N/A	N/A	N/A	N/A
First Action	Event File	Clr [CommCode] A(value x 100) {E} < [Alt_ft] at time AcceptsHandoff	R-side	Callsign
Last Action	Event File	Clr [CommCode] = A(value) {E} = 220	R-side	Callsign
Action	Event File	[CommCode] + [CommCont] at First Action	R-side	Callsign
Complete	Trajectory File	[Alt_ft] ≤ 22000	N/A	N/A
Route	Trajectory File	Filed		Callsign

Note that the WO_open time is *either* when the aircraft enters the sector *or* when the aircraft's trajectory is conflict-free, as determined by a conflict detection algorithm. Note also that the time stamps in the Event file are in seconds to 3 decimal places and in Trajectory file in HH:MM:SS. Hence, the algorithm for outputting the Task 3 data is as follows; the letters is brackets indicate the input file, {E} for Event file and {T} for trajectory file:

```
IF (callsign) flight plan [Filed] {T} ends in OHO OR DETRO OR DESM OR KANCY THEN
WO_opens at time when aircraft (callsign) AcceptsHandoff {E}
OR (callsign) trajectory is conflict-free;
WO_closes at time when aircraft (callsign) is at TOD (Top Of Descent);
desired_Alt = 22000 {C}
delta_Alt {c} = current_Alt {T} - desired_Alt {C}
VS_nom = 3000 ft/min {C}
TOD {c} = Exits sector {E} – (delta_Alt/VS_nom) {c}
FirstAction is at time when [CommCode] = Clr {E}
AND [CommCont] = A(value x 100) {E} < [Alt_ft] at time AcceptsHandoff {T}
LastAction is at time when [CommCode] = Clr {E}
AND [CommCont] = A(value) {E} = 220
Complete is at time when aircraft (callsign) [Alt_ft] {T} ≤ desired_Alt {C}
```

From these data, the program shall produce an output as follows (times in seconds to 3 decimal places):

G	H	I	J	K	L	M	N	O
Task	Actor	Object	BcnCode	W_open	W_close	BeforeFix	AfterFix	FirstAction
SOP	R-side	AAL1001	4342	959.000	1709.000			1202.650

P	Q	R	S
LastAction	Action	Complete	Route
1455.56	Clr+A220	1560.000	

A1.8. Task 5: Conflict Resolution (CR)

Resolve all conflicts; conflicts are defined as potential violations of the 1000 ft vertical and 3 nm horizontal separation minima (LOS).

Check for conflicts each time the controller issues a clearance (Clr in the [CommCode] column) for aircraft receiving the clearance. If a conflict exists, it will be assumed that the clearance was issued as a resolution to that conflict, yielding a First- and Last Action times and a time for closing of WO (i.e., LOS).

Event	File	Keyword(s) [Column Heading]	Actor	Object
Window Opens	EventFile	AcceptsHandoff [Action]	RPOS	Callsign
Window Closes	N/A	LOS	N/A	N/A
First Fixation	N/A	N/A	N/A	N/A
First Action	Event File	Clr [CommCode] X(value) [CommCont]	R-side	Callsign
Last Action	Event File	Clr [CommCode] X(value) [CommCont]	R-side	Callsign
Action	Event File	[CommCode] + [CommCont]	R-side	Callsign
Complete	Conflict File	TLS Time to Loss of Separation	N/A	N/A
Route	Conflict File	Other aircraft in conflict pair	N/A	Callsign #2

The conflict resolution algorithm was based on Song and Kuchar (2001).

A1.9. Task 6: System Data Entry (DE)

The controllers frequently updated the system by entering data into it. These events are marked by [Action] column keyword ‘PressesKey’. The last action is marked by [Content] column keyword ‘Return’; the first action is counted backwards from here to the first ‘PressesKey’ in conjunction with a callsign in the [Object] column.

Event	File	Keyword(s) [Column Heading]	Actor	Object
Window Opens	N/A	N/A	N/A	N/A
Window Closes	N/A	N/A	N/A	N/A
First Fixation	N/A	N/A	N/A	N/A
First Action	Event File	PressesKey [Action]	R-side	Callsign
Last Action	Event File	PressesKey [Action] + Return [Content]	R-side	Callsign
Complete	N/A	N/A	N/A	N/A

Example:

From Event file:

```
Time     Actor    Action         Object    Content

876.349  R-Side   PressesKey     AAL1001    4
876.459  R-Side   PressesKey     AAL1001    space
876.829  R-Side   PressesKey     AAL1001   2
876.980  R-Side   PressesKey     AAL1001    3
877.130  R-Side   PressesKey     AAL1001    4
877.320  R-Side   PressesKey     AAL1001   Return
```

From these data, the program shall produce an output as follows:

G	H	I	J	K	L	M	N	O
Task	Actor	Object	BcnCode	W_open	W_close	BeforeFix	AfterFix	FirstAction
DataEntry	R-side	AAL1001	4342					876.349

P	Q	R	S
LastAction	Action	Complete	Route
877.320			

A1.10. Task 7: Hand Off All Aircraft Prior to Sector Boundary (IH)

All necessary data are in the Event Files; the table below specifies the events relevant to the task, keywords identifying the events, corresponding column headings and actor and object words:

Event	File	Keyword [Column Heading]	Actor	Object
Window Opens	N/A	N/A	N/A	N/A
Window Closes	Event File	Aircraft exits sector	Callsign	N/A
First Fixation	N/A	N/A	N/A	N/A
First Action	EventFile	Handsoff [Action]	RPOS	Callsign
Last Action	EventFile	Handsoff [Action]	RPOS	Callsign

Note: opening of WO cannot be defined for this task.

Note: Missing ‘Handsoff’ keyword indicates that handoffs were performed automatically and were not controllers’ task.

Example: The data file (Event File) reads as follows:

```
Time          Actor       Action           Object      Content
1531.863      RPOS        Handsoff         AAL1001
..
1709.000      AAL1001     Exits Sector
```

From these data, the program shall produce an output as follows:

G	H	I	J	K	L	M	N	O
Task	Actor	Object	WO_open	WO_close	FirstFixation	FirstAction	LastAction	Action
Exit_Handoff	RPOS	AAL1001		1709.000		1531.863	1531.863	

A1.11. Task 8: Transfer Communication

All necessary data are in the Event Files; the table below specifies the events relevant to the task, keywords identifying the events, corresponding column headings and actor and object words:

Event	File	Keyword(s) [Column Heading]	Actor	Object
Window Opens	N/A	N/A	N/A	N/A
Window Closes	Event File	Aircraft exits sector	Callsign	N/A
First Fixation	N/A	N/A	N/A	N/A
First Action	Event File	[CommCode]	R-side	Callsign
Last Action	Event File	Says [Action + Duration]	R-side	Callsign
Action	Event File	[CommCode] + [CommCont]	R-side	Callsign

Note: opening of WO cannot be defined for this task.

Example:

The data file (Event File) reads as follows:

```
Time     Actor   Action   Object   Content  CommCode CommCont Duration
1668.190 R-Side  Says     AAL1001  (trans.) Clr      120.18     5.26
..
1709.000 AAL1001 Exits Sector
```

From these data, the program produced an output as follows:

G	H	I	J	K	L	M	N	O
Task	Actor	Object	BcnCode	W_open	W_close	BeforeFix	AfterFix	FirstAction
Comm_Xfer	R-side	AAL1001	4342		1709.000			1668.190

P	Q	R	S
LastAction	Action	Complete	Route
1673.45	Clr+120.18		

A1.12. Additional Comments

(1) Missing ‘Handsoff’ keyword means that handoffs were performed automatically; hence we were not interested in these as human tasks.

(2) Missing ‘Says – Actor Pilot’; this looks like pilot transmissions were not transcribed and hence we must deal with this as a case of missing data (i.e., we lost Task #2).

(3) Missing ‘Fixates’ simply mean missing data. At this time this is less important, as the eye movements were only a peripheral issue and redundant with other recorded actions.

(4) Missing call signs from pilot transmissions had to be manually entered into the Object column from the transcript.

(5) ‘Leaves Sector’ is the same as Exits Sector; these keywords had to be found and replaced.

(6) Missing ‘In Sector at End’ entry; we assumed that all aircraft without ‘Exits Sector’ entry are In Sector at End

(7) Missing ‘R-Side—Says; the actor should be R-Side on rows where the ‘Source’ column (H) has R Transcript entry.

(8) ‘AcceptMine’ = ‘AcceptsHandoff’.

A2. Task Prioritization

The data output from the above data reduction program was yet processed by another program to investigate the controllers' task prioritization schemes and strategies. This task prioritization program paired each task in a predetermined time epoch (30- or 15 s) with each task after it (tasks were ordered by FirstAction time) and wrote an output as in the following example:

```
INPUT

Team   Cond   Task   Actor  WO_o   WO_c   E30    FAct   LAct   TR     WO_dur TFA    TRm

3      11     RH     R-side 105.22 238.00 4      115.24 115.24 0.00   132.78 10.02  122.7
3      11     RH     R-side 114.22 225.00 4      117.13 117.13 0.00   110.78 2.90   107.88
3      11     RH     R-side 104.77 270.00 4      118.72 118.72 0.00   165.23 13.95  151.28
3      11     RH     R-side 109.75 245.00 4      120.80 120.80 0.00   135.25 11.06  124.20
```

```
OUTPUT

Team Cond Task1 Task2 Actor1 Actor2 TFA1 TFA2 TRm1 TRm2

   3    1     5     7 R-side R-side    *    *    * 66.20

   3    1     5     5 R-side R-side    *    *    *    *

   3    1     5     5 R-side R-side    *    *    *    *

   3    1     7     5 R-side R-side    *    * 66.20   *

   3    1     7     5 R-side R-side    *    * 66.20   *

   3    1     5     5 R-side R-side    *    *    *    *
```

APPENDIX B

B1. Conflict Detection Algorithm

B1.1. Conflict Detection Algorithm

Conflicts are defined as potential violations of the 1000 ft vertical and 3 nm horizontal separation minima (LOS). The code for this is based on the following algorithm.

B2. Minimum Separation Algorithm for ATC Simulation

B2.1. Notation and Variable Names

DCPA = distance to the closest point of approach
MD = miss distance
MDL = miss distance—lateral
MDV = miss distance—vertical
MLS = minimum lateral separation
MVS = minimum vertical separation
DLLS = distance to loss of lateral separation
TLLS = time to loss of lateral separation
DLVS = distance to loss of vertical separation
TLVS = time to loss of vertical separation

B2.2. Computation of DCPA and MDL

This algorithm computes distance to the closest point of approach (DCPA) and the miss distance—lateral (MDL), as presented in Song and Kuchar (2001). A representative scenario appears in the figure below (Fig. 1).

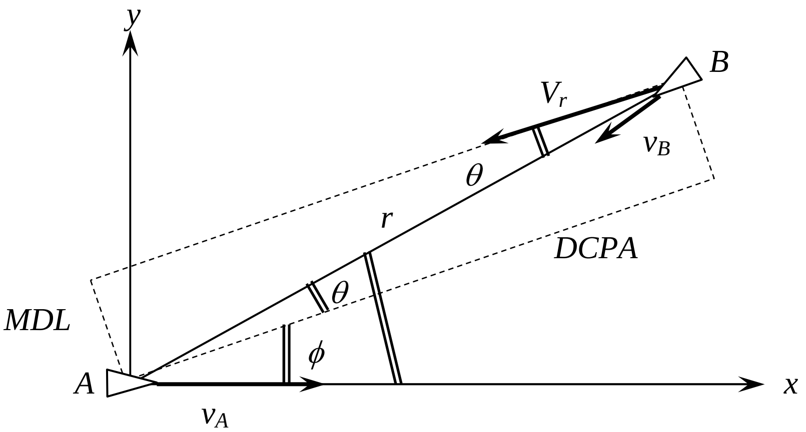

Figure 1: A schematic of a conflict geometry for computation of the DCPA and MDL

The range *r* between the aircraft at any given time can be computed as

$$r = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2} \qquad (1)$$

and their relative bearing χ as

$$\chi = \tan^{-1}\left(\frac{y_A - y_B}{x_A - x_B}\right) \tag{2}$$

The velocity components in the x- and y-dimensions of each aircraft in a pair should be calculated as follows:

$$\begin{aligned} v_{x_A} &= \frac{x_{A_{T+t}} - x_{A_T}}{t} \\ v_{y_A} &= \frac{y_{A_{T+t}} - y_{A_T}}{t} \end{aligned} \tag{3}$$

where x_{A_T} and y_{A_T} are the x and y coordinates of aircraft A at time T and $x_{A_{T+t}}$ and $y_{A_{T+t}}$ aircraft coordinates after time t.

The angle ϕ is computed as

$$\phi = \tan^{-1}\left(\frac{v_{B_y} - v_{Ay}}{v_{B_x} - v_{A_x}}\right) \tag{4}$$

The angle θ is given by

$$\theta = \chi - \phi \tag{5}$$

Finally, the *DCPA* is given by

$$DCPA = r\cos\theta \tag{6}$$

and the *MDL* by

$$MDL = r\sin\theta \tag{7}$$

B2.3. The Vertical Dimension

Conflict between two aircraft shall be determined in a hierarchical fashion, first examining the horizontal dimension and the minimum lateral separation, and then the vertical dimension and the minimum vertical separation (*MVS*). After the *MDL* has been determined and if *MDL* <*MLS*, the vertical dimension is examined. If the planes are on different altitudes (i.e., $alt_A \neq alt_B$), in level flight, and $|alt_A - alt_B| \geq 1{,}000$ ft, a vertical separation exists. If the planes are on the same altitude (i.e., $alt_A = alt_B$), or $|alt_A - alt_B| \leq 1{,}000$ ft, and *MDL* <*MLS*, the aircraft are in conflict. It is therefore important to verify the vertical distance between the aircraft at *MDL*. This shall be done in terms of time: If one of the aircraft is climbing or descending through the other aircraft's altitude, the program shall compute the *TLLS* as well as the time interval when $|alt_A - alt_B| \leq 1{,}000$ ft; if *TLLS* occurs in that time interval, the time to loss of separation is the time to loss of vertical separation, *TLVS.* In other words

IF *TLVS* >*TLLS* THEN *TLS* = *TLVS*

IF *TLVS* < *TLLS* THEN *TLS* = *TLLS*

B.3. References

Song, L., & Kuchar, J. K. (2001). Modeling and analysis of conflicts between alerting systems. *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* Montreal, Canada, August 6-9, 2001. Cambridge, MA: MIT.