

DOT/FAA/ND-97/15
Volume 7 of 9

Office of Communication,
Navigation, and Surveillance
Washington, DC 20591

Operation Heli-STAR - Cargo Simulation System

Ellen Bass
Charles Stancil

Georgia Tech Research Institute
Atlanta, Georgia 30332

September 1997

Final Report

This document is available to the public
through the National Technical Information
Service, Springfield, Virginia 22161.



U. S. Department
of Transportation

Federal Aviation
Administration

DISTRIBUTION STATEMENT A
Approved for Public Release
Distribution Unlimited

DTIC QUALITY INSPECTED 4

19991129 055

NOTICE

**This document is disseminated under the sponsorship of the United States
Department of Transportation in the interest of information exchange.
The United States Government assumes no liability for the contents or use thereof.**

Department of Commerce
Bureau of Economic Analysis
Washington, D.C. 20540

Technical Report Documentation Page

1. Report No. DOT/FAA/ND-97/15 Volume 7 of 9		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle Operation Heli-STAR - Cargo Simulation System				5. Report Date September 1997	
				6. Performing Organization No.	
7. Author (s) Ellen Bass and Charles Stancil Georgia Tech Research Institute, Atlanta, Georgia, 30332				8. Performing Organization Report No.	
9. Performing Organization Name and Address Under contract to: Science Applications International Corporation (SAIC) 1213 Jefferson Davis Highway, Suite 1500 Arlington, Virginia, 22202				10. Work Unit No. (TRAIS)	
				11. Contract or Grant No. DTFA01-93-C-00030, Task Order A3	
12. Sponsoring Agency Name and Address Federal Aviation Administration General Aviation and Vertical Flight Program Office, AND-710 800 Independence Avenue, S. W. Washington, DC 20591				13. Type Report and Period Covered Final Report	
				14. Sponsoring Agency Code AND-710	
15. Supplementary Notes AND-710 General Aviation and Vertical Flight Program Office					
16. Abstract Operation Heli-STAR (Helicopter Short-Haul Transportation and Aviation Research) was established and operated in Atlanta, Georgia, during the period of the 1996 Centennial Olympic Games. Heli-STAR had three major thrusts: 1) the establishment and operation of a helicopter-based cargo transportation system, 2) the management of low-altitude air traffic in the airspace of an urban area, and 3) the collection and analysis of research and development data associated with items 1 and 2. Heli-STAR was a cooperative industry/government program that included parcel package shippers and couriers in the Atlanta area, the helicopter industry, aviation electronics manufacturers, the Federal Aviation Administration (FAA), the National Aeronautics and Space Administration (NASA), and support contractors. Several detailed reports have been produced as a result of Operation Heli-STAR. These include 4 reports on acoustic measurements and associated analyses, and reports on the Heli-STAR tracking data including the data processing and retrieval system, the Heli-STAR cargo simulation, and the community response system. In addition, NASA's Advanced General Aviation Transport Experiments (AGATE) program has produced a report describing the Atlanta Communications Experiment (ACE) which produced the avionics and ground equipment using automatic dependent surveillance-broadcast (ADS-B) technology. This latter report is restricted to organizations belonging to NASA's AGATE industry consortium. A complete list of these reports is shown on the following page.					
17. Key Words Helicopter Rotorcraft Automatic Dependent Surveillance Heliport Datalink Community Involvement Security Short-Haul Transportation Helicopter Acoustics			18. Distribution Statement This document is available to the U. S. Public through the National Technical Information Service, 5258 Port Royal Road, Springfield, Virginia 22161.		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 92	22. Price

Operation Heli-STAR Technical Reports

Volume 1 DOT/FAA/ND-97/9	Operation Heli-STAR - Summary and Major Findings; S. Fisher, FAA AND-710, Washington DC and E. McConkey, Science Applications International Corporation, Arlington, Virginia; Principal Editors; September 1997
Volume 2 DOT/FAA/ND-97/10	Operation Heli-STAR - Helicopter Noise Levels Near Dekalb Peachtree Airport; Krishan Ahuja, Robert Funk, Jeffrey Hsu, Marcie Benne, Mary L. Rivamonte, and Charles Stancil; Georgia Tech Research Institute, Atlanta, Georgia; September 1997
Volume 3 DOT/FAA/ND-97/11	Operation Heli-STAR - Helicopter Noise Annoyance Near Dekalb Peachtree Airport; Krishan Ahuja, Marcie Benne, Mary L. Rivamonte, Robert Funk, Jeffrey Hsu, and Charles Stancil; Georgia Tech Research Institute, Atlanta, Georgia; September 1997
Volume 4 DOT/FAA/ND-97/12	Operation Heli-STAR - Helicopter Noise at Heliports; Krishan Ahuja, Robert Funk, Jeffrey Hsu, and Charles Stancil; Georgia Tech Research Institute, Atlanta, Georgia; September 1997
Volume 5 DOT/FAA/ND-97/13	Operation Heli-STAR - Effects of Buildings on Helicopter Noise; Krishan Ahuja, Robert Funk, Jeffrey Hsu, Michael Heiges, and Charles Stancil; Georgia Tech Research Institute, Atlanta, Georgia; September 1997
Volume 6 DOT/FAA/ND-97/14	Operation Heli-STAR - Aircraft Position Data; Michael Heiges, Shabnam Khan; Georgia Tech Research Institute, Atlanta, Georgia, September 1997
Volume 7 DOT/FAA/ND-97/15	Operation Heli-STAR - Cargo Simulation System; Ellen Bass, and Charles Stancil; Georgia Tech Research Institute, Atlanta, Georgia, September 1997
Volume 8 DOT/FAA/ND-97/16	Operation Heli-STAR - Community Involvement; Christine Eberhard and Bobbi Rupp; CommuniQuest, Inc., Manhattan Beach, California; September 1997
Volume 9 DOT/FAA/ND-97/17	Operation Heli-STAR - Atlanta Communication Experiment (ACE), AGATE Flight Systems Communication Work Package 1.4, (AGATE Restricted Information) (AGATE Flight Systems Communication Team), December 1996.

FOREWORD

This is Volume 7 of a 9-volume report documenting the activities and results of Operation Heli-STAR, the Atlanta Short-Haul Transportation System (ASTS). ASTS was a cooperative government/industry program that established a helicopter transportation system to support community of Atlanta during the 1996 Olympic games. Volumes 2 through 5 of this set of reports documents the noise studies that were performed during Operation Heli-STAR. The noise research was performed by Georgia Tech Research Institute (GTRI). GTRI also produced two additional reports documenting Operation Heli-STAR. Volume 6 describes the aircraft position data processing research, and Volume 7 documents a Cargo Simulation System that was used in support of Heli-STAR cargo operations. The research and development elements of Operation Heli-STAR were funded by the Federal Aviation Administration through Science Applications International Corporation (SAIC).

The GTRI manager of the overall ASTS program was Mr. C. Stancil. The Principal Investigator of the noise studies, reported in volumes 2 through 5, was Dr. K. K. Ahuja of GTRI. GTRI personnel responsible for making and analyzing day-to-day noise measurements were Dr. R. Funk and Mr. Jeff Hsu who were assisted by a team of 20 researchers. Ms. Marcie Benne, a graduate student from the School of Psychology lead the effort on the community survey reported in Volume 2. She was assisted by Ms. Mary Lynn Rivamonte, a student in the School of Aerospace Engineering. The authors are particularly grateful for Dr. Mike Heiges of GTRI for providing the helicopter altitudes and flight paths and to Mr. Stephen Williams, also of GTRI, for setting up the microphone locations for noise contour measurements.

The titles of the four volumes reporting noise research are:

Volume 2 - Helicopter Noise Levels Near Dekalb Peachtree Airport

Volume 3 - Helicopter Noise Annoyance Near Dekalb Peachtree Airport

Volume 4 - Helicopter Noise at Heliports

Volume 5 - Effects of Buildings on Helicopter Noise

The titles of the other two volumes authored by GTRI are:

Volume 6 - Aircraft Position Data

Volume 7 - Cargo Simulation System

Table of Contents

SUMMARY.....	1
ACRONYMS.....	2
BACKGROUND	3
THE DOMAIN - ASTS CARGO DELIVERY OPERATIONS.....	6
HELIPORTS USED FOR CARGO DELIVERY	6
CARGO DELIVERY FLIGHTS AND ROUTES.....	7
LOADING ZONE OPERATIONS.....	9
SPECIAL LOADING ZONE OPERATIONS - TRANSFER.....	10
HELIPORT CLOSURE.....	10
LOGICAL CARGO HOLDING AREAS AT THE HELIPORTS	11
SIMULATION MODEL.....	11
SHIPPERS	12
HELICOPTERS.....	13
LOADERS.....	16
CURRENT SIMULATION ARCHITECTURE	18
NOMINAL CARGO HELICOPTER SCHEDULING PROCESS.....	20
SELECT INITIAL ROUTE DATA	21
FORMAT AND VALIDATE SHIPPER DEMAND DATA	21
CHECK FOR TRANSFERS	23
SOLVE CAPACITY PROBLEMS.....	25
CHECK FOR CONFLICTS AND LATE CARGO DELIVERY	29
TRIM UNUSED FLIGHTS AND UPDATE FLIGHT TIMES	31
SUBSTITUTE SMALL HELICOPTERS FOR LARGE ONES	33
CREATE REPORTS	34
DETAILED SIMULATION USAGE DESCRIPTION.....	34
FIRST PLANNING DAY	36
SECOND PLANNING DAY	38
THIRD PLANNING DAY	39
FOURTH PLANNING DAY	42
FIFTH PLANNING DAY	43
SIXTH PLANNING DAY	47
SEVENTH PLANNING DAY.....	50
EIGHTH PLANNING DAY	52
NINTH PLANNING DAY	57
TENTH PLANNING DAY	58
ELEVENTH PLANNING DAY.....	61
TWELFTH PLANNING DAY	63
REQUIREMENTS FOR AN INTELLIGENT AGENT.....	64
ROUTE PLANNING AID	64
INPUT VALIDATION AID	67
CHECKING FOR TRANSFERS AID.....	67
CAPACITY PROBLEM AID.....	67

CONFLICT RESOLUTION AID	68
FLIGHT UTILIZATION AID	68
HELICOPTER UTILIZATION AID	68
ARCHITECTURE IMPLEMENTING AN INTELLIGENT AGENT	68
FUTURE WORK.....	70
ACKNOWLEDGMENTS.....	70
REFERENCES.....	71
APPENDIX A. INPUT FILES REQUIRED BY THE SIMULATION	I
AIRPORT.DAT - HELIPORT CONFIGURATION	I
DEMAND.DAT - SHIPPER DEMAND	I
FLIGHTTIMES.DAT - FLIGHT TIMES BETWEEN HELIPORT PAIRS	II
HELO.DAT - AIRCRAFT CHARACTERISTICS	II
ROUTE.DAT - ROUTE SCHEDULE.....	III
SHIPPER.DAT - SHIPPER IDENTIFICATION	III
WEATHER.DAT - HELIPORT CLOSING TIMES	III
APPENDIX B. REPORTS AVAILABLE FROM THE SIMULATION.....	V
APPENDIX C. GLOSSARY.....	VII

List of Figures

FIGURE 1. APPROXIMATE LOCATIONS OF THE ASTS HELIPORTS RELATIVE TO MAJOR HIGHWAYS AROUND ATLANTA ..	7
FIGURE 2 ORDER OF THE HELIPORTS ON A TYPICAL ROUND ROBIN TRIP	8
FIGURE 3 LOGICAL REPRESENTATION OF CARGO LOCATIONS IN THE SHIPPER SERVICE AREA	11
FIGURE 4 SHIPPER BRINGING CARGO TO THE ORIGIN.....	13
FIGURE 5 ACTIVITIES FOR THE HELICOPTER	15
FIGURE 6 ARCHITECTURE FOR THE ASTS CARGO HELICOPTER DECISION SUPPORT TOOL.....	19
FIGURE 7 NOMINAL CARGO HELICOPTER SCHEDULING PROCESS.....	21
FIGURE 8 CONTENT OF THE FILE REPORTING THE STATUS OF CARGO WAITING TO BE TRANSFERRED WHEN NO CARGO IS WAITING	25
FIGURE 9 PERCENTAGE OF TIME PERFORMING TASKS WITH THE AID OF THE SIMULATION.....	35
FIGURE 10 PARTIAL FLIGHT TIMES REPORT WITH SMALL HELICOPTERS OVERFLYING ATL AT 11:59 PM.	46
FIGURE 11 TRIP GENERATION LOGIC FLOW	66
FIGURE 12 ARCHITECTURE WITH AN INTELLIGENT AGENT	69

List of Tables

TABLE 1. HELIPORT IDENTIFIERS AND AVAILABLE AIR TRAFFIC SERVICES.....	6
TABLE 2 SHIPPER DEMAND DATA VALIDATIONS OBSERVED DURING EXPERIMENT.....	23
TABLE 3. ROUTE DATA VALIDATIONS OBSERVED DURING EXPERIMENT.....	33
TABLE 4 SUMMARY OF THE PLANNERS' USAGE OF THE SIMULATION TO SOLVE PROBLEMS.....	35
TABLE 5 SUMMARY OF SIMULATION RUNS ON DAY 1.....	37
TABLE 6 SUMMARY OF SIMULATION RUNS ON DAY 2.....	38
TABLE 7 SUMMARY OF SIMULATION RUNS ON DAY 3.....	39
TABLE 8 SUMMARY OF SIMULATION RUNS ON DAY 4.....	43
TABLE 9 SUMMARY OF SIMULATION RUNS ON DAY 5.....	44
TABLE 10 SUMMARY OF SIMULATION RUNS ON DAY 6.....	47
TABLE 11 SUMMARY OF SIMULATION RUNS ON DAY 7.....	50
TABLE 12 SUMMARY OF SIMULATION RUNS ON DAY 8.....	53
TABLE 13 SUMMARY OF SIMULATION RUNS ON DAY 9.....	57
TABLE 14 SUMMARY OF SIMULATION RUNS ON DAY 10.....	58
TABLE 15 SUMMARY OF SIMULATION RUNS ON DAY 11.....	61
TABLE 16 SUMMARY OF SIMULATION RUNS ON DAY 12.....	63
TABLE 17 DESCRIPTIONS OF THE FAA APPROVED HELICOPTER ROUTES.....	IX
TABLE 18 PARTICIPATING SHIPPERS.....	X

Summary

This report describes a year long effort to develop an architecture for distributed simulation that integrates intelligent, agent-based, operator aids with models of entities in the airspace system. The problem used as a basis for the effort is part of the Atlanta Short Haul Transportation System, a joint experiment to use specially equipped helicopters in a partial free flight environment to move cargo between eleven sites around metropolitan Atlanta during the 1996 Summer Olympic Games. The focus of the simulation was to provide the cargo delivery operations personnel the ability to evaluate route plans and to investigate the impact of potential weather related airspace closure.

The cargo operations decision support simulation was developed based on requirements from the planning personnel and on issues uncovered during implementation of the simulation. These requirements, the model to capture them, and the simulation infrastructure used to create the simulation are described in the report.

In order to determine requirements for intelligent aiding in this domain, I collected data on how the planners used the simulation during the cargo helicopter planning periods. From that data, I determined what the nominal daily cargo helicopter scheduling process is (i.e., select route data, validate shipper demand data, check for transfers, solve capacity problems, check for conflicts, trim under-utilized flights, substitute smaller aircraft when possible, and create reports).

I provide a description of the nominal process as well as the details of how the planners used the simulation during the experiment to create and to modify schedules. The data collected during the experiment illustrates that the planners need better support in order to complete their tasks during the time allowed. Solving capacity problems take the planners the most time to solve with the existing support. Validating data and trimming flights are second and third in required planner time. To make the most drastic performance improvements, the decision support simulation environment should better aid these three tasks. The other tasks were supported well enough although small improvements are possible.

I discuss the requirements for an intelligent agent to support cargo helicopter operations planners based on the data collected during the experiment. I describe the additional knowledge and the functionality the agent would require to provide better information and aid to the planners. Finally, I propose an improved decision support tool architecture including intelligent aiding and simulation to support cargo delivery operations.

Acronyms

ABX	Airborne Express
ACD	Air Courier Dispatch
AJC	Atlanta Journal Constitution (shipper)
ASTS	Atlanta Short Haul Transportation System
ATC	Air traffic control
ATL	Atlanta Hartsfield International Airport
BNK	Bank
BUC	Wachovia Bank (heliport in the Buckhead neighborhood of Atlanta)
CRC	Courier Communications
CRD	Courier Dispatch
CRP	Corporation
CRT	Cathode Ray Tube
DHL	DHL
EDT	Estimated departure time
ETA	Estimated time of arrival
EXC	Executive Courier
FAA	Federal Aviation Administration
FDX	Federal Express
FTY	Fulton County Airport
GA	Georgia
GAL	Galleria
GBH	Georgia Baptist
GPS	Global Positioning System
GTRI	Georgia Tech Research Institute
LDC	Long distance carrier
MARTA	Metropolitan Atlanta Rapid Transit Authority
MIT	Nations Bank Mitchell Street
MLQ	MLQ
MSL	Mean sea level
NAS	National Airspace System
NASA	National Aeronautics and Space Administration
NBE	Nations Bank Northeast
NBK	Nations Bank
NBS	Nations Bank Southwest
NOR	Atlanta Journal Constitution (heliport in Norcross, GA)
OOP	Object-Oriented Programming
PDK	Dekalb Peachtree Airport
RAF	North Fulton Hospital (heliport in Roswell, GA)
SDC	Short distance carrier
TCP/IP	Transmission Control Protocol, Internet Protocol
UPS	United Parcel Service
VFR	Visual Flight Rules
VHF	Very High Frequency
WAC	Wachovia Bank (shipper)

Background

The research topic for this fellowship is to develop an architecture for distributed simulation that integrates intelligent, agent-based, operator aids with models of entities in the airspace system. The idea is to model entities that are part of the National Airspace System (NAS) and intelligent agents with autonomous decision making capabilities and to integrate the models in a distributed simulation of a busy terminal area. In the research, a client-server model supporting concurrent processing should enable representation of entities represented at levels appropriate for their context.

The research proposal identifies seven tasks in completing this goal:

1. Identify a problem within the terminal area including the role of human decision makers, airspace constraints, air traffic control (ATC) and other procedural constraints, and the pertinent hardware and software.
2. In consort with the Federal Aviation Administration (FAA), analyze the nature of decisions in that problem area including those that are amenable to modeling using intelligent agents and identify the knowledge required of such agents.
3. Design and implement models for ATC and aviation hardware and software for the problem area (building on existing simulation(s) if possible).
4. Design and implement a client server model and simulation infrastructure.
5. Design and implement the identified intelligent agents.
6. Integrate the ATC, aviation, and intelligent agents models within the simulation infrastructure.
7. Test the architecture using data from the literature, if there is any, and with consultation with the FAA.

On initial contact with the FAA (personal conversation with Peter Hwoschinsky, technical manager in the Vertical Flight Program Office) in the fall of 1996, I learned that a one focus in the agency's research entails investigating issues related to free flight, a concept still undergoing evolution with the aim to increase flight efficiency by allowing more flexibility in flight planning and in returning more aspects of air traffic separation responsibility to pilots. The idea for the future of NAS is to have safe and efficient flight under instrument flight rules where pilots have freedom to select both their flight path and their speed in real-time (Nordwall, 1995).

With free flight, air traffic controllers would only intervene to avert potential problems. The FAA is investigating many issues whose solutions can be understood by modeling and simulation. A small set of examples follow:

1. What communication, navigation, and surveillance equipment is required on aircraft for safe free flight?
2. What is the effect of mixing aircraft with this equipment with aircraft that do not?
3. What data is required at what update rate in order to predict potential aircraft conflicts under various free flight procedures?
4. What information should be transmitted between aircraft and at what rate for free flight to be effective?
5. What information should be transmitted between pilots/aircraft systems and flight dispatchers/airlines operations center computer systems for free flight to be effective in allowing in-flight plan changes for large civil transports? Currently flight dispatchers at major airlines can build fuel efficient flight plans that are checked for interference with bad weather. These plans are given to pilots before takeoff. With free flight, there will be more opportunity for flight dispatchers to help with in-flight replanning as opposed to only helping with pre-flight planning.
6. What can be done to accommodate general aviation, regional airlines and corporate aircraft that may not have the same level of equipage as the major air carriers?

The problem area that the FAA suggested investigating in order to develop and to test the simulation architecture for this fellowship is part of the Atlanta Short Haul Transportation System (ASTS) project. ASTS is a joint experiment between the FAA, NASA, several Georgia state agencies, Georgia Tech Research Institute (GTRI), and industry. One purpose of ASTS is to use specially equipped helicopters in a partial free flight environment to move cargo between eleven sites around metropolitan Atlanta during the 1996 Summer Olympic Games. Because the roads and highways in the Atlanta metropolitan area were projected to be congested during the Olympics, the project planners and the participating shippers expected great dependence on this airborne cargo transportation system.

For ASTS, later renamed Operation Helistar, helicopters are equipped with Amnav Systems' GeoNet - an integrated combination of VHF datalink, GPS, and CRT-type display. Thus equipped, the helicopters provide automatic dependent surveillance for a low altitude point-to-point transportation system.

The ASTS helicopters only operate under visual flight rules (VFR) because of the lack of radar coverage and the nature of the project. However, even with this constraint, the experiment provides insight into:

- Community noise sensitivity during low altitude operations,
- Data link communication techniques used in a degraded environment,
- "See and be seen" collision avoidance capabilities applicable to future free flight air traffic management,
- The effectiveness of GPS in providing precise navigation information, and
- Design requirements for helicopters used in civil short haul operations in uncontrolled and controlled airspace.

ASTS has a limited budget for helicopter flight time. Although participating shippers pay fees for the cargo delivery service, their contribution is not intended to cover expenses. Rather, the shippers' payments are set to encourage participation in the experiment. The majority of the costs are paid from the fixed ASTS budget. The allocated flight time portion of the experimental budget imposes a tight constraint. Because of the limited flight time budget, two types of helicopters move cargo. One helicopter type, the B0 105, has a smaller cargo load carrying capacity (i.e., 61 cubic feet and 1000 pounds of cargo) and a low fixed and hourly operating cost. The other, the Bell 412, has a larger carrying capacity (i.e., 228 cubic feet and 2500 pounds of cargo) but a much higher fixed and hourly operating cost. Determining a good mixture of the two aircraft types to meet cargo demand while remaining within the flight time budget is crucial to successful cargo delivery operations.

My FAA contact suggested that I contact the ASTS program manager, Charles Stancil, at Georgia Tech Research Institute (GTRI). When I described the simulation architecture that I wanted to design and to build to fulfill the requirements for the fellowship, he suggested that I work with the planners in charge of the cargo delivery operations because they would have use for a decision support simulation with intelligent aiding. Also he realized that planning cargo helicopter operations in the congested airspace in a partial free flight environment clearly meets the problem area objectives for the fellowship. Together we determined that I would have to build the simulation and its architecture from scratch and would not be able to build upon an existing simulation. I agreed to work with the cargo delivery operations planning team, but I also explained that I may not be able to design and to implement the entire simulation architecture with intelligent aiding in time for the Olympics. The program manager understood this development time issue.

The GTRI planning team is responsible for the planning and execution of the cargo delivery system including both air and ground operations. Before the start of the actual shipping operations, the planners created preliminary schedules for the cargo transport helicopters based on shippers' estimated demand. Several months before the Olympics, the participating shippers provided estimated cargo demand between heliport pairs at various times during the Olympic period. Using the shipper demand data, the GTRI planners generated tentative helicopter flight schedules. After checking the tentative schedules, the shippers provided new cargo demand data with assignments of cargo to specific flights. The GTRI researchers worked with the new demand data and negotiated repeatedly with the shippers to fine-tune the schedules in order to meet demand and to stay within the flight time budget.

At the end of this process, each shipper committed to send minimum and maximum cargo loads on the different flights in the schedule. The researchers used this information to create an initial set of operational schedules for the helicopters. These operational schedules included the actual flights (with estimated departure and arrival times at the heliports), helicopter type assignments to flights, and cargo assignments to flights.

In addition to working with the shippers, GTRI also planned and coordinated the cargo loading ground operations. GTRI researchers hired and trained the cargo handlers who themselves were military helicopter maintenance personnel. GTRI worked with a scanning technology company to develop procedures and the hardware/software infrastructure to support cargo handling operations. Together they developed strategies that included using a unique identifier (i.e., a preprinted Interleaved 2 of 5 bar code) for each cargo item along with a paper manifest

(Gurin, 1996). Each cargo item would be scanned four times by a Symbol Technologies 3805 portable data collection terminal:

1. When the shipper dropped off the cargo at its origin,
2. When the loaders at the origin put the cargo on the helicopter,
3. When the loaders at the destination unloaded the cargo from the helicopter, and
4. When the shipper picked up the cargo at the destination.

The packages would be tracked using Genisys Operation's Receive Sentry Software.

GTRI researchers also contracted with a helicopter operator to provide aircraft, pilots, and maintenance personnel for the experiment. With the helicopter operator, they determined aircraft and crew time requirements and other constraints so they could plan appropriately. They also provided flight path information to the helicopter operator based on the FAA approved helicopter routes to be flown during the Olympics and on noise abatement procedures.

GTRI was also responsible for the heliports. For private (i.e., non-airport) heliports, they gained permission to use the pads. Once they had such commitments, they re-configured the heliports for helicopter and loading operations. Each heliport had its own particular set of issues such as how to get cargo to and from the roof top (if there was one) and where the shippers were allowed to park.

While the GTRI researchers were engaged in the schedule creation, cargo handling procedures development, helicopter contracting, and heliport procurement and configuration processes, they incrementally worked on the requirements for the decision support simulation. The purpose of the simulation is to help them to evaluate proposed schedules and to investigate the effects of heliport closures by varying the number of helicopters to be used, the mix of each type of helicopter, the timing of heliport availability, and shipper demand distributions. The simulation also provides a test bed for air and ground procedures development. With procedures implemented in code, the simulation tests for inconsistencies, "holes", and similar problems.

The lead cargo operations planner at GTRI and one of the other planners provided high level requirements while I designed and implemented the simulation. During the development process, the two researchers modified requirements based on their work on their other responsibilities, their evolving understanding of what decision support they would like, and on issues uncovered during the simulation design.

The simulation requirements for normal operations and the running model were completed about three months before the actual shipping operations. Two months before the Olympics, the GTRI lead planner started using the simulation in order to gain familiarity and to suggest changes in its reporting (i.e., decision support) capabilities.

During the two months before the actual cargo operations, the planners provided requirements for simulating heliport closure. As the helicopters were constrained to VFR operations, weather could potentially interfere with cargo operations. Thus, the planners wanted to be able to simulate the impact of heliport closures. The implementation of the heliport closure requirements was only partially completed at the start of the actual cargo operations period. The simulation was tested with one helicopter pad closing at a single heliport, but not for multiple closures. As the heliports were geographically close to each other, the possibility of weather impacting more than one heliport within a short period of time was likely so the state of the simulation was limiting in this regard.

Immediately before the experimental period, no "perfect" operational schedules for the maximum cargo loads emerged. When running the operational schedules, the planners encountered problems predicted by the simulation as follows:

- Volume capacity problems (i.e., the cargo volume desired by the shippers to be loaded on to the assigned helicopter for a particular flight exceeds the maximum capacity),
- Helicopter conflicts jeopardizing the time schedule (where a conflict arises when more than one helicopter tries to land at the same non-airport heliport at the same time and the later ones must hold),
- Cargo delivered later than expected (usually caused by helicopter conflicts),
- Unused and underutilized flights (i.e., flights that have little or no cargo to load at the origin and little or no cargo to deliver at the destination)

- Attempted cargo loading at the wrong heliports and cargo deliveries to the wrong heliports (i.e., cargo assigned to flights that start and/or stop at heliports other than those the shippers had intended)

The researchers thought that the first type of problem was minor because the actual cargo demand would vary enough from the maximum so capacity problems would either not realize themselves during the experiment or would be trivial to fix on a case by case basis. The second and third problems were also thought to be minor because not all of the scheduled flights would really be flown during the actual cargo operations; these conflicts would not really occur. The researchers thought they could solve these problems while solving the fourth problem - by eliminating unused flight from the schedule each day. The last problem was always attributed to errors in the shipper demand data which could be resolved by coordinating with the shippers. So at the start of the experiment, the initial baseline plan from which the GTRI researchers worked had small potential problems that the researchers thought could be easily fixed during each day's planning period.

In the rest of this paper, I first describe in more detail the ASTS domain with respect to cargo delivery operations. Then I describe the model embedded in the decision support simulation. I then describe the simulation architecture. Then I describe the nominal daily cargo helicopter scheduling process. Then I provide a more detailed description of how the GTRI researchers used the simulation during the experiment to create and to modify schedules. I discuss the requirements for an intelligent agent to support cargo helicopter operations planners based on the data collected during the experiment. Finally I propose an improved decision support tool architecture including such an intelligent agent.

The domain - ASTS cargo delivery operations

The ASTS operations supported by the decision support simulation consisted of using at most seven helicopters at a time flying between eleven heliports to deliver cargo. The planned mixture of helicopters included using up to two large helicopters and four small ones at any time. One additional small helicopter (i.e., not one of the four) was always available in case of a maintenance or other problem. This "back up" helicopter could be called into service to replace another helicopter or to fly a short mission on an "as needed" basis.

Each helicopter had two occupants - the pilot and the observer. The pilot was responsible for flying the aircraft and communicating with air traffic control when in controlled airspace. The pilot also communicated with a company flight dispatcher. As operations were conducted under visual flight rules, the observer helped look for traffic. The observer also handled the paper manifests for all cargo onboard the helicopter.

Heliports used for cargo delivery

The eleven heliports used for ASTS cargo delivery varied in configuration and capability. See Table 1 for the list of heliports and the type of air traffic control services available. Figure 1 shows the approximate locations of the ASTS heliports. Heliports are identified by their three letter designators. Major highways are also identified.

Table 1. Heliport identifiers and available air traffic services

Heliport name	Heliport designator	Control
Atlanta Hartsfield International Airport	ATL	Class B
Wachovia Bank	BUC	Uncontrolled
Fulton County Airport	FTY	Class D
Galleria	GAL	Uncontrolled
Georgia Baptist	GBH	Uncontrolled
Nations Bank Mitchell Street	MIT	Uncontrolled
Nations Bank Northeast	NBE	Uncontrolled
Nations Bank Southwest	NBS	Uncontrolled
Atlanta Journal Constitution	NOR	Uncontrolled
Dekalb Peachtree Airport	PDK	Class D
North Fulton Hospital	RAF	Uncontrolled

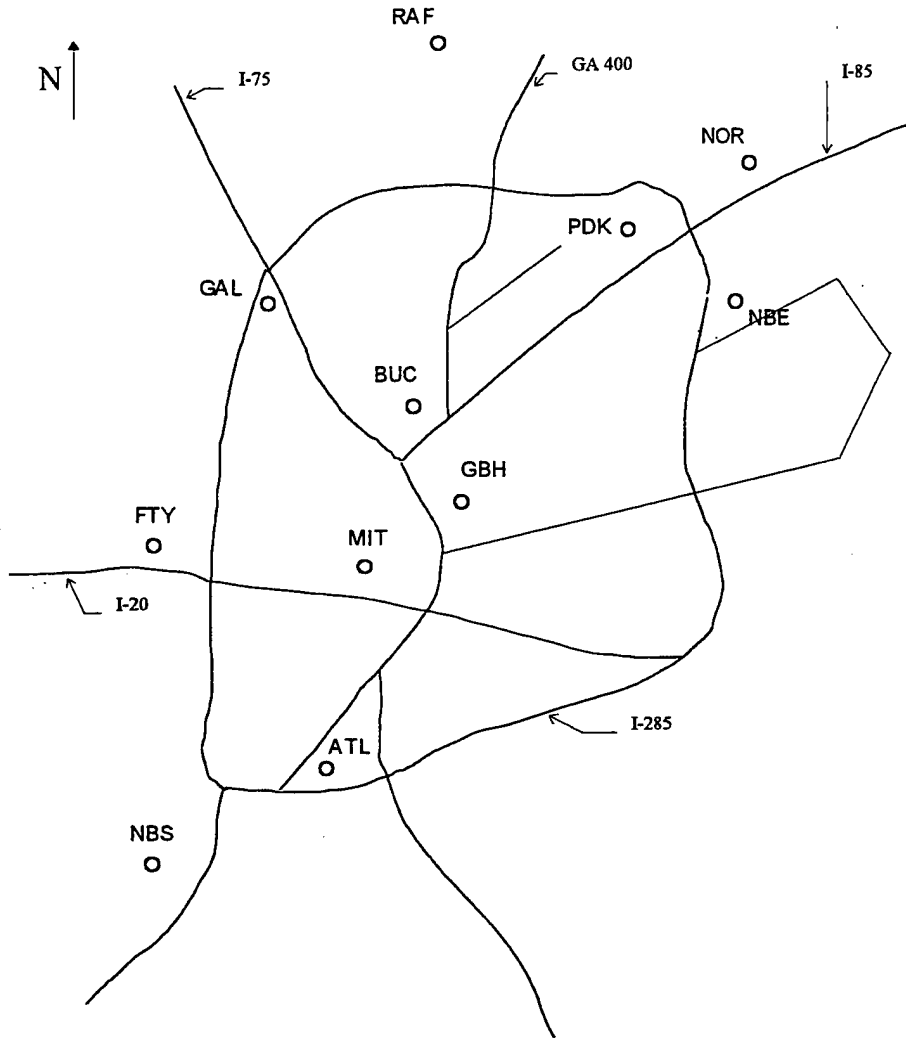


Figure 1. Approximate locations of the ASTS heliports relative to major highways around Atlanta

Three of these sites are controlled airports (i.e., ATL, FTY, and PDK). They have unlimited space for helicopters to wait on the ground between flights. One of these airports, PDK, served as the "home base" for the helicopter fleet so each route started and ended there. Also, all refueling was scheduled to occur at PDK. The other eight heliports are private, uncontrolled pads. They only have space for one aircraft at a time. This space constraint means that if a helicopter approaches when another helicopter is already on the pad, the former has to hold in the air. This holding in the air was referred to by GTRI as a "conflict". Conflicts were to be avoided as they wasted flight time.

Cargo delivery flights and routes

The planners use the term "flight" to specify one leg from the origin to the destination and the term "route" to either mean a set of flights from PDK to PDK or sets of such routes. See the entries for "flight" and "route" in the glossary in Appendix C. To eliminate some of the confusion in the use of the term "route", the term "trip" is introduced here as a substitute for the first meaning of "route" in Appendix C - a circuit of flights beginning and ending at PDK with no intervening stops at PDK.

The planners created two kinds of routes - "round robin" and "dedicated". Flights on round robin routes are available to all shippers, while flights on a dedicated route are only available to the shipper specifically requesting and paying for them.

Flights beginning with the letters A, B, C, and D composed the round robin routes. During the experiment, a round robin trip from PDK to PDK usually was nine total flights. The A and B routes typically were composed of sets of trips with the following heliport order:

PDK BUC MIT ATL NBS FTY GBH GAL NOR PDK

while the C and D routes are in the opposite order. In some trips, RAF is visited instead of NOR. In some trips, MIT is visited instead of GBH and vice versa. Some trips have an additional stop at NBE on the way to or from PDK. This flight path creates a modified figure eight pattern with the center of the "eight" at the downtown heliports (see Figure 2 which illustrates the order of the heliports for a typical round robin flight, although the aircraft would follow the approved FAA route structure as opposed to flying directly from location to location). The planners created this figure eight flight path because they noticed that a majority of the cargo either should be moved from the downtown heliports to the other heliports or vice versa. The figure eight structure provides the opportunity to visit a downtown heliport twice during each round robin trip.

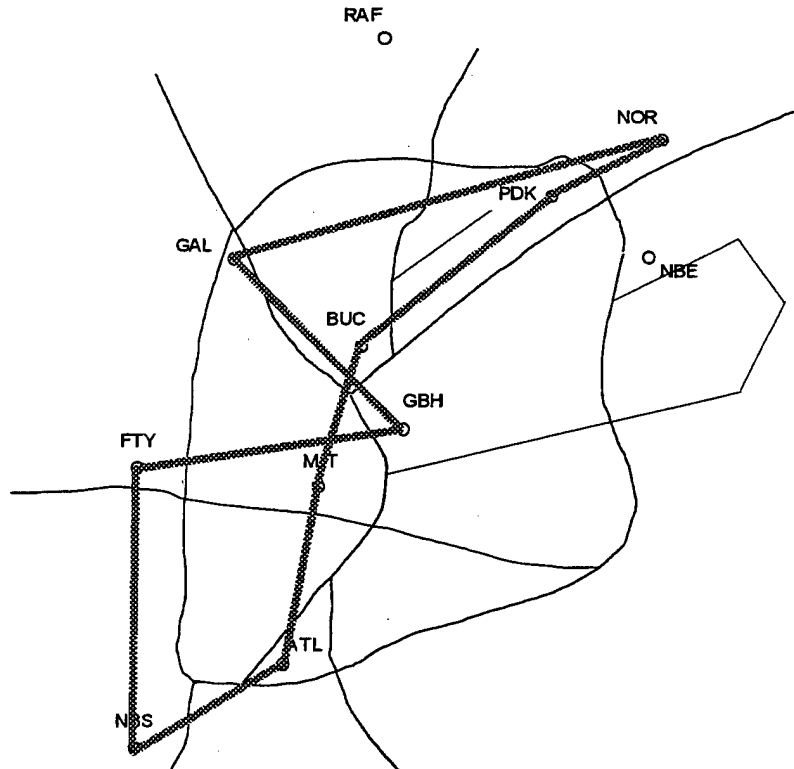


Figure 2 Order of the heliports on a typical round robin trip

The dedicated routes, available to the shippers that have exclusively requested them, are specifically tailored to the need of the requesting shipper. The dedicated routes differ between the weekdays and the weekend. They began with the letters G (weekday), H (weekday), J (Saturday only), K (Saturday only), L (Saturday only), and M (Sunday only).

As part of their responsibilities, the planners had to define the exact flight path that the pilots should follow when flying between the heliports. The paths had to conform to the FAA identified routes over which the helicopter

should fly for noise abatement purposes and to avoid restricted areas such as the Olympic Village (where the athletes were staying). The FAA published a special helicopter route chart for the Olympics that defined FAA routes following highways and other busy roads. See the third definition for term "route" in the glossary in Appendix C for more detail concerning the eight FAA routes.

The FAA routes were very significant because they constrained where the helicopters could fly. When the GTRI planners decided which heliports composed a particular cargo delivery trip, they could easily identify exactly how that trip should be flown. The pilot should fly the shortest path from the departure heliport to the FAA route in a direction favoring the destination. Then the pilot should fly along the FAA route structure until reaching the closest point between the FAA route structure and the destination in the direction of approach. Then the pilot should fly directly to the destination. These constraints meant that the planners could estimate the flight distances and times between heliports because there was usually one "best" path to follow. One difficulty in making precise time estimates is the variability in the times in and out of the controlled airports due to the high traffic volume and the need to get verbal clearances on busy radio frequencies.

Loading zone operations

Each heliport has two main cargo delivery service areas - a shipper service area (known as the "fence") and the helicopter loading area (known as the "pad"). Together these two areas and the space between them are called the "loading zone". Shippers are allowed at the shipper service area but only the loaders, and of course the pilots and observers, are allowed at the helicopter loading area. The loaders at each heliport has two carts for moving cargo between the two areas. One cart, the "load cart", is used for cargo taken from the shippers to be loaded on to the helicopter. The other, the "unload cart", is used for cargo unloaded from the helicopter to be given to the shippers.

Each heliport has one scan gun to be used for scanning cargo. The scan gun therefore limits loader operations; the scan gun cannot be at a pad and at the shipper service area at the same time. Also the scan gun is not equipped with direct transmission capability. For billing and reporting, the loaders periodically download the scanned data from the scan gun onto a low-end computer at the heliport. The information is then sent to the main Receive Sentry data base at GTRI.

At the shipper service area, shippers bring cargo to be delivered. Shippers have to provide a cargo shipping label with a bar code, cargo destination, and cargo size information before the loaders accept the cargo. Shippers also give the loaders a paper manifest when dropping off cargo. Loaders take the cargo from the shippers and either put the cargo in a cargo waiting area or directly on the load cart. The loaders scan cargo coming from the shippers. This initial scanning is called the "scan 1" operation.

When a helicopter arrives, two loaders (with the scan gun and the load and unload carts) approach the helicopter. First the helicopter observer hands the loaders the manifests for the cargo to be unloaded. The loaders remove the cargo identified by the manifests from the helicopter and put this cargo on the unload cart. As the loaders remove the cargo from the helicopter, the loaders scan it. This scan is called the "scan 3" operation because this is the third time the cargo is scanned (note that the cargo receives its second scan when it is loaded at its origin).

After the cargo to be unloaded is processed, the loaders hand the observer the manifests for the cargo to be loaded. The loaders scan the cargo as it is loaded. The scan is called "scan 2" as it is the second time the cargo is scanned.

The loaders have an option in executing the unloading procedure. If a helicopter falls behind its schedule (either before it arrived or at the loading zone) or if another helicopter was waiting for service (either on the ground or in the air), the loaders can unload all the cargo from the helicopter without executing the scan 3 operation. In the case where the helicopter is late, the loaders can perform the scan 3 after the helicopter is loaded and is ready to depart (and thereby expedite the helicopter's departure). In the case where a helicopter is waiting, the loader can perform the scan 3 during a slack period after servicing the waiting helicopter.

After servicing a helicopter, the loaders wheel the load and unload carts back to the shipper service area. If no shippers are waiting for the cargo, the loaders put the cargo that was on the unload cart in the cargo waiting area in order to make room on the unload cart for cargo from the next helicopter. Otherwise, the loaders scan the cargo for the fourth time and gave the cargo with its paper manifest to the shipper. This final scanning operation is called "scan 4".

Special loading zone operations - transfer

At PDK, transfers are permitted. A transfer is when a cargo item is picked up at the origin by one helicopter, is transferred to another helicopter at PDK (the only heliport where transfers are allowed), and is delivered to its destination by a second aircraft.

The planners thought transfers would come into play if a large helicopter flew one trip of a round robin route and then a small helicopter took its place on the next trip or vice versa. Cargo loaded at the tail end of the first trip and delivered near the beginning of the subsequent trip would require the transfer process. Transfers never actually occurred in the real cargo operations but they did occur during the planning periods. The GTRI planners were always concerned about transfers and eliminated transfers whenever possible. The issue is that since the helicopter that originally picks up a piece of cargo can be late by the time it returns to PDK, it is possible for the second helicopter to leave PDK before the first helicopter arrives. Such a set of events would leave cargo stranded at PDK.

Heliport closure

The GTRI planners developed procedures for the pilots and loaders to implement if the weather at a particular heliport drops below VFR minimums. A pilot en route to or holding at a closed heliport overflies (i.e., bypasses) the closed heliport and proceeds directly to the next heliport called the "overfly destination". The loaders at the overfly destination unload the cargo destined for the closed heliport. This cargo, called "orphan cargo" would remain at the overfly destination heliport until the closed heliport re-opens. When the closed heliport re-opens, the loaders at the overfly destination can load this cargo on the next helicopter stopping there bound for the previously closed heliport.

If a helicopter were on a pad at a heliport when the heliport closed, the planners were to coordinate with the helicopter operator to dispatch the "back up" helicopter to continue the mission plan of the helicopter stuck at the closed heliport. When the heliport re-opens, the previously trapped helicopter would return to PDK.

Logical cargo holding areas at the heliports

For the shipper service area, the GTRI planners created special names for queues of cargo that were not associated with the carts. Although these queues had no specific place in the shipper service area, the planners used this nomenclature for discussion purposes. These fictitious areas became part of the teams' shared view of how the operations at the shipper service area would occur. See Figure 3.

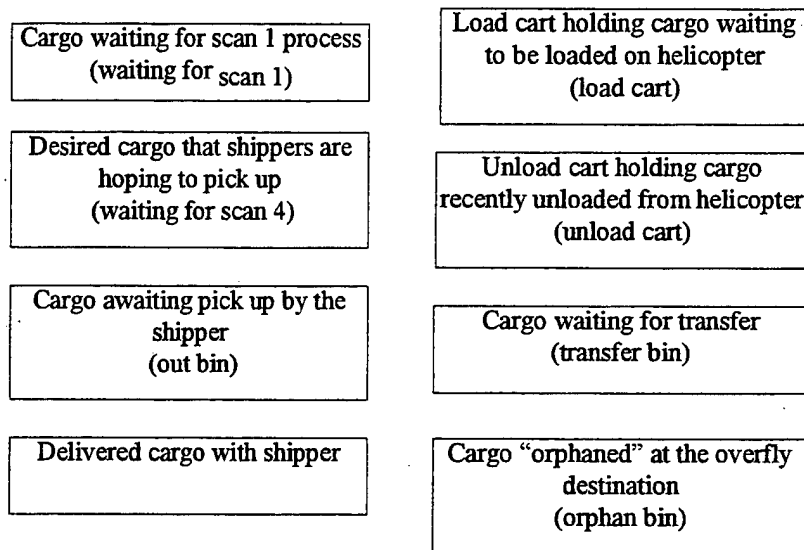


Figure 3 Logical representation of cargo locations in the shipper service area

When shippers want to bring cargo to the origin, but the loaders are busy, the shippers wait with the cargo for the scan 1 operation in the "waiting for scan 1" queue. When shippers come to pick up delivered cargo and either the loaders are busy or the cargo has not yet arrived, the shippers wait by adding desired cargo to the "waiting for scan 4" queue. When loaders wheel the load and unload carts from the pad to the shipper service area and the shippers are not waiting for the cargo, the loaders empty the unload cart and put the cargo in an area for cargo waiting to be picked up by the shippers called the "out bin". When loaders transfer cargo off a helicopter at PDK, they put the cargo in the "transfer bin" before they transferred the cargo on to the second helicopter. When the loaders at the overfly destination took cargo destined for a closed heliport from a helicopter, they put the cargo in the "orphan bin".

Simulation model

The simulation has two main purposes for the planners. One focus of the simulation is to provide predictions for the GTRI cargo scheduling planners under normal operations. For normal operations, the simulation helps the planners to evaluate proposed route schedules to deliver cargo with respect to the following areas:

1. Late cargo delivery,
2. Loader workload at each heliport,
3. Helicopter conflicts at non-airport heliports,
4. Helicopter capacity problems, and
5. Unused and under-utilized flights.

The planners wanted to be sure that each cargo item could make its desired delivery time. Cargo might be late because its assigned helicopter got behind schedule due to conflicts, refueling or loader workload. The planners wanted to eliminate helicopter conflicts or at least minimize the impact of such conflicts. They also wanted to model loader workload to predict busy periods that might impact helicopter schedules or reduce service to

shippers wanting to drop off or pick up cargo. They also wanted to be able to negotiate with the shippers when cargo capacity problems arose. The planners also wanted to eliminate unused and under-utilized flights as much as possible in order to use the limited flight time budget in the best manner.

The second focus of the simulation is to provide the potential impact of heliport closures. As the helicopters operate in visual flight rule conditions, weather changes could temporarily close a particular heliport. The GTRI planners wanted to know what cargo would be left "orphaned" (i.e., never picked up) at the closed heliport. They also wanted to know how late cargo "orphaned" (i.e., left) at the overfly destination would be if a heliport were closed and then re-opened and the loaders at the overfly destination sent the cargo with the next helicopter having spare capacity flying to the previously closed heliport.

The lead planner and the planner who best understood the modeling capabilities of a discrete event simulation identified the simulation requirements. These requirements focused on the procedures for three main actors (shippers, loaders, and helicopter pilots). Therefore the discussion of the model is couched with reference to these objects.

In discussing the models for the three main objects, other objects and their models are also be mentioned:

- Air traffic control to control the pads at each heliport and to assign aircraft to and from holding patterns
- Cargo with bar code data, other attributes from the shipper demand data, scan times, transfer information, and overfly information
- Carts used by the loaders to move cargo between the helicopters and the shipper service areas
- Heliports
- A holding pattern associated with each heliport
- A mission plans for each helicopter
- Pad(s) at the heliports
- A scan gun used by loaders at each heliport to scan the cargo
- A shipper service area composed of several logical cargo holding locations at each heliport

Shippers

The GTRI planners were only interested in modeling shipper activity to the extent that shippers created workload for the loaders. Therefore a detailed model of shipper activity is not necessary. With respect to loader workload, the shippers have two main functions in the simulation. They drop cargo off at the origin and they pick cargo up at the destination. The simulation does not model these activities with the limitations that exist in the shippers' real world. For example, there is no limit on the number of heliports at which a shipper can be, regardless of how many shipper representatives are actually participating in the experiment. In the simulation, it is possible for the same shipper to be dropping off and picking up cargo at all of the heliports at the same time.

In the simulation, a shipper brings a cargo item to the origin based on a parameter time before the shipper thinks that the origin flight arrives at the heliport. This parameter time is defined by user input. The planners use fifteen minutes because that is the time before each flight arrival by which they told shippers to bring cargo to the origin. Note that in the simulation, a shipper uses as a basis time (from which the parameter time is subtracted) the estimated departure time (EDT) that the shipper has entered in the demand data for that item of cargo. (For a description of the data that the shippers provide, see the input file demand.dat in Appendix A.) The shipper also delivers cargo to the origin heliport based on the origin item in the demand data. Therefore, if the shipper has entered the incorrect EDT and/or location in the demand data, the error(s) are propagated throughout the simulation. These types of input errors are flagged by the simulation.

The shipper cargo drop off operation is illustrated in Figure 4. When a shipper arrives at the origin and there is an idle loader at the shipper service area, the loader performs the scan 1 operation based on the number of bags to be scanned. The number of bags is defined in the cargo's demand data. The planners estimated that each bag takes five seconds to process for a scan 1 operation. The model uses a minimum scan 1 time of five seconds (for partial bags) and a maximum time of one minute for large loads (i.e., one cargo item composed of more than twelve bags). If a shipper has multiple cargo items listed in the demand data for the same heliport at the same time, each

cargo item is processed separately. If the shipper arrives at the origin but there is no available loader, the shipper must wait. In the model, each heliport has a "waiting for scan 1" queue for the cargo that the shippers are waiting to drop off (refer also to Figure 3).

The model is very similar for shippers picking up their cargo at the destination. A shipper arrives at the destination at the estimated arrival time (ETA) for the cargo item. The ETA is defined in the cargo's demand data. If a loader is available, the loader tries to perform the scan 4 operation by searching for the cargo in the shipper service area. If the shipper's cargo is on the unload cart or the out bin, the loader performs the scan 4 operation and returns the cargo to the shipper. If there is no idle loader with a scan gun at the shipper service area, the shipper waits by putting the desired cargo item in the "waiting for scan 4" queue. It is possible that the cargo never is delivered because the cargo exceeded the capacity of the helicopter to which it was assigned, it missed a transfer flight, it was "orphaned" at its origin or it was "orphaned" at the overfly destination. In this case, the "desired cargo" will still be in the waiting for scan 4 queue at the end of the simulation.

To help the GTRI planners solve capacity problems, each shipper object in the simulated world keeps track of its "shipper plan" which is composed of three lists for each flight:

1. What cargo the shipper wants to load on each flight (whether or not the cargo actually meets the capacity constraints of the helicopter),
2. What cargo the shipper wants on the flight (whether the shipper wants it loaded at the flight's origin or the shipper wants it loaded on a previous flight but not yet unloaded), and
3. What cargo the shipper wants to unload from the flight (whether or not the cargo actually would have been loaded).

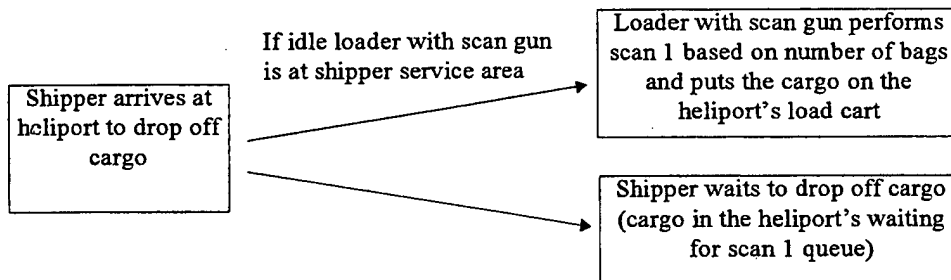


Figure 4 Shipper bringing cargo to the origin.

Helicopters

The simulation has three main foci with respect to helicopters and their normal operations:

1. To model the ability to meet departure and arrival times based on loader workload, helicopter conflicts, helicopter refueling time constraints, and estimated flight times between heliport pairs,
2. To model planned cargo over-capacities, and
3. To identify unused and under-utilized flights.

Based on the needs of the planners, all helicopter operations are modeled with respect to what the loaders do with the cargo on the helicopter and with respect to helicopter operations around the heliport. Modeling detailed pilot activity is not necessary. To limit the complexity of the simulation, all pilot activities are modeled via "anthropomorphic" helicopters; the helicopters do all of the action instead of modeling a pilot that acts on the helicopter. Observer functions such as passing of the paper manifest is not modeled at all.

Air traffic control is modeled for the purpose of controlling the pads both at controlled and uncontrolled heliports and therefore also for clearing helicopters to and from holding patterns. Although the uncontrolled heliports do not have air traffic controllers, assignments of pads to helicopters and of helicopters to and from holding patterns are all handled by air traffic control for consistency.

Each helicopter is initialized based on the helicopter input data in a file called helo.dat (for a description of this file, see Appendix A). For ASTS, each helicopter starts at PDK, the home base. Each helicopter is on one of PDK's "pads". Therefore PDK must have as many pads as helicopters in the simulation (seven total - five small helicopters and two large ones). The number of pads at a heliport is defined in airport.dat, a file of input data defining each heliport (see the description of airport.dat in Appendix A). Note that "pad" in this context is a place for a helicopter on the ground but not necessarily where it lands.

Each helicopter has a mission plan that identifies each flight the helicopter will fly, what cargo is to be picked up at the origin of each flight and what cargo is to be delivered at each flight's destination. The cargo to be picked up and delivered on each flight is determined by the origin and destination flights identified in the shipper demand data (see demand.dat in Appendix A). The mission plan also has estimated departure and arrival times for each flight based on the route data (see the description of route.dat in Appendix A). Thus, a helicopter's mission plan is a combination of the subset of route data assigned to the helicopter plus the cargo identifiers of cargo with flights assigned to that helicopter. During the simulation, the mission plan is updated with actual departure and arrival times for reporting purposes.

The idea of using the mission plan is for simulation convenience as no such artifact existed in the real world. The planners mentioned that the pilots would have a detailed schedule of their routes to fly but neither the pilots nor the observers would have a listing of all scheduled cargo loads. The planners said that the observers would know what to deliver based on the paper manifests and the loaders would know what to load based on what was at the heliport. However, representing the information in the mission plan turned out to be useful to the planners for several purposes - all discussed later in this report.

Figure 5 and the following description summarizes the helicopter's activities modeled in the simulation. Note that in the simulation, all loading zone operations and refueling occur when the helicopter is on the "pad". In reality, loading really only occurs on the pad at the non-airport heliports. At an airport, the helicopter must taxi from the real landing pad to the loading zone. However, this time is accounted for by the flight times to the airports instead of explicitly modeling the taxi separately. Thus in the simulation, helicopters on the ground are serviced while on the pad no matter what type of heliport they are visiting.

Refueling can only occur at PDK. Refueling time is defined by a parameter normally set to thirty minutes. The first activity that begins when a helicopter lands at PDK is the refueling operation. Helicopters are "topped off" at the end of the previous day, so helicopters do not require refueling before the first flight of the day. Other activities such as spooling down and loading can occur while the aircraft is refueling.

Land on an unoccupied pad at the heliport and record the actual arrival time in the mission plan		
Start refuel if at PDK and it is not the first flight of the day		
Spool down if it is not the first flight of the day		
Request unload if there is something to unload		
Update the current flight in the mission plan after unloading operation		
Request load if there is something to load		
If it is past the EDT for the current flight, the aircraft isn't refueling or being serviced, and the pad is open, spool up		
If the pad is open, request departure clearance from ATC		
If ATC has provided a clearance, calculate a new arrival time based on the estimated flight time between the origin and the destination		
Request a landing clearance from ATC at the calculated arrival time at the destination		
If the heliport is closed, ATC clears overflight of this destination	If the heliport is open but all the pads are occupied, ATC clears entrance into the holding pattern	If the heliport is open and ATC assigns an unoccupied pad, perform final approach and land
Calculate the new arrival time at the overfly destination	When ATC gives the clearance to leave the holding pattern, perform final approach and land	
At the new arrival time, request a landing clearance at the overfly destination		

Figure 5 Activities for the helicopter

When a helicopter first lands at any heliport (including PDK), the actual arrival time is noted in the mission plan. Also, for reporting purposes, each landing is recorded by the heliport. After landing, the helicopter's blades must spool down before the loaders can leave the fence and walk to the helicopter. When the spool down period has passed, loaders can approach the aircraft and begin unloading. Note that for the first flight of the day, the loaders can immediately approach the aircraft (because the blades are not yet moving).

If loaders are not available to unload the helicopter, the helicopter waits. When the loaders become available, they service the helicopter. If the helicopter has no cargo destined for the heliport, the unloading operation is skipped.

Unloading occurs before loading because each helicopter has a limited capacity. Once the helicopter's total capacity is reached, no more cargo can be loaded. Unloading first therefore allows more cargo to be loaded. In the helicopter's mission plan, the cargo to be unloaded at the destination is listed for each flight.

Loaders can immediately load the helicopter once unloading is completed. However, if there is nothing to load, this operation is skipped. The cargo to load is listed in the helicopter's next flight's information in the mission plan. Since the mission plan lists for each flight what is loaded at the origin and what is unloaded at the destination, the load that follows an unload is really the load for the next flight. Therefore when the load operation begins, the helicopter's mission plan is updated to reflect that the current flight is the next one.

Each helicopter has maximum volume and weight capacities defined in the helo.dat input file. Every time a loader attempts to load cargo, the cargo's volume and weight is compared to the helicopter's remaining capacity. Cargo that is too heavy or too large is not allowed on board. Partial cargo items are not loaded unless the analyst has explicitly split up the cargo into multiple items.

The planners do not want helicopters to depart until the estimated departure time. In the simulation, the helicopter does not depart when the unloading and loading operations are completed unless the flight's estimated departure time has passed and the refueling operation is completed. If the conditions for departure are achieved, the helicopter spools up for its spool up time and then requests a departure clearance from air traffic control. The air traffic controller clears the helicopter for departure. The actual departure time is noted in the mission plan.

At this point, the pad that the helicopter had been occupying is now available for another aircraft. If any helicopter is holding, the air traffic controller clears the one waiting the longest for final approach to the available pad.

Flying between heliports is only modeled with respect to the time it takes to fly from location to location. According to the flight times in the flight time data (see flightTimes.dat in Appendix A), the helicopter "flies" to its next location. When it approaches its destination, the helicopter requests a landing clearance from ATC. If there is an available pad at the heliport, the air traffic controller assigns the pad to the helicopter and clears the aircraft for final approach. Otherwise, the air traffic controller clear the aircraft to hold. When the helicopter is cleared for final approach, the aircraft lands and the process starts again.

The simulation also models the effects of heliport closures. The simulation uses a file called weather.dat to determine when each pad at a heliport opens and closes (see Appendix A for a description of this input file). If a helicopter requests a landing clearance at a heliport that has no open pads, the helicopter overflies the heliport and flies to the next heliport in its mission plan. It calculates the actual time of arrival based on the flight times data. At the same time, the helicopter "marks" all of the cargo supposed to be delivered at the closed heliport with the overfly destination so the loaders know to unload it and to put it in the heliport's "orphan bin". At that next heliport, during the unload operation, the loaders unload all cargo destined for the closed heliport so the helicopter does not fly around with this cargo and therefore reduce remaining capacity unnecessarily. If a helicopter is "stuck" at a pad when it closes, the air traffic controller sends the helicopter back to PDK once the pad re-opens. The planners asked for this capability in the simulation because helicopters stuck at a pad will be replaced by a back up helicopter.

Loaders

In the simulation, each heliport has two loaders. One loader is the "chief" and has the scan gun. The other is called the "handler". Loaders can either both be at the shipper service area called the "fence" or at the "pad" (i.e., any of the heliports pads). Regardless of how many pads a particular heliport may have, when the loaders are servicing a helicopter, they are at the "pad". The two loaders move between the fence and the pad together because one wheels the load cart and the other the unload cart. At the beginning of the simulation, these loaders are idle at the fence.

Loaders have several tasks that they accomplish at the fence:

1. Perform scan 1 operation for 5 seconds per bag (i.e., take cargo from a shipper that has just arrived at the fence or has cargo waiting in the "waiting for scan 1 queue", scan it for the first time, and put the cargo on the load cart).
2. Perform scan 4 operation for 5 seconds per bag (i.e., take cargo from the out bin or the unload cart, scan it for its fourth time, and give the cargo to the shipper).
3. Remove cargo from the unload cart that was unloaded from a helicopter and put the cargo in the out bin taking 2 seconds per bag.

These tasks are accomplished on a cargo item basis. In other words, if a shipper brings three items, each cargo item is processed separately. Each time a loader completes a scan 1 or a scan 4 operation, the loader can be interrupted to perform a higher priority task.

Loaders have several tasks that they accomplish at the pad:

1. Perform scan 2 operation for 10 seconds per bag with a minimum of 10 seconds and a maximum of a minute (i.e., remove cargo from the load cart, scan it, and put it on the helicopter). Loaders load all cargo by volume from smallest volume to highest volume. Cargo with an asterisk ("*") in the cargo identifier are loaded after cargo that do not have the asterisk. In this way, the planners can prioritize the cargo loading (i.e., items with an asterisk are lower priority).
2. Perform scan 3 operation for 10 seconds per bag with a minimum of 10 seconds and a maximum of a minute using the preferred method and half that time otherwise (i.e., remove cargo from the helicopter, scan it if doing the preferred method, and put it on the unload cart).

3. Remove cargo to be transferred at PDK from the first helicopter (i.e., remove cargo from the helicopter, mark the cargo with the time the cargo was transferred, and put the cargo in the transfer bin).
4. Put cargo to be transferred on to a second helicopter at PDK (i.e., remove cargo from the transfer bin, mark the cargo with the time the cargo was transferred on to the delivery helicopter, and put the cargo on the helicopter).
5. Remove cargo from a helicopter that just flew over a closed heliport and put the cargo in the orphan bin (i.e., remove cargo from the helicopter, mark the cargo with the time the cargo was put in the orphan bin, and put the cargo in the orphan bin).
6. Take cargo from the orphan bin and put it on a helicopter that is flying to a newly re-opened heliport (i.e., remove cargo from the orphan bin, mark the cargo with the time the cargo put on to the helicopter, and put the cargo on the helicopter).

The loaders service one helicopter at a time. When a helicopter lands, spools down, and gets the loaders attention, the loaders walk for a parameter time with the carts to the pad. Before starting the unload operation, the loaders check if the helicopter is behind schedule, if any other helicopters are waiting for service, or if any helicopters are holding. If not, the loaders perform the normal scan 3 operation as they unload. Otherwise, they take the cargo off the helicopter more quickly but do not scan the cargo. The loaders use the helicopter's mission plan to determine what to unload although in reality they would look at what cargo is at the heliport for the helicopter. All "regular" unloaded cargo is put on the unload cart. The loaders unload all of the cargo without interruption.

During the unload process, the loaders perform "special" operations that take no time. The planners did not believe that these rare operations would add much workload so the performance times for these actions are not modeled. In the simulation, cargo to be transferred off at PDK are marked with a special indication. At PDK, all cargo transferred off the helicopter is placed in the transfer bin. Cargo is marked with the "transfer off" time.

When a helicopter overflies a closed heliport and the loaders must remove orphan cargo, the loaders take no time to place the cargo in the orphan bin. Similar to the transfer cargo, orphan cargo has a special indication. Cargo is marked with the time of this operation.

When the unloading is completed, the loaders load the helicopter using the normal scan 2 process. The loaders load cargo from the load cart using the same amount of time as the normal scan 3 process. Loading transfer cargo from the transfer bin and loading orphan cargo going to a previously closed heliport take no time to load in the simulation. The loaders load all of the cargo without interruption.

Every time the loaders finish servicing a helicopter, they return to the fence in order to empty the unload cart. If the shippers are waiting to pick up cargo, the loaders give them the cargo from the unload cart using the scan 4 procedure. Otherwise the loaders remove the cargo from the unload cart and put it in the out bin. Moving cargo from the unload cart to the out bin takes 2 seconds per bag.

The loaders have a priority scheme to determine what to do next if there are several entities wanting service. The events that precipitate loader activity are listed in priority order:

1. A helicopter lands on a pad and there is cargo to unload. If the loaders are idle at the fence, they immediately go to the pad and service this helicopter. The loaders unload and then load the helicopter without interruption.
2. A helicopter lands on a pad and there is no cargo to unload but there is cargo to load. If the loaders are idle at the fence, they immediately go to the pad and load this helicopter.
3. A helicopter that had landed while the loaders were busy doing other tasks is waiting to be unloaded. If the loaders are at the fence, they check to see if there are shippers waiting to drop off cargo for this waiting helicopter. If not, the loaders immediately go service this helicopter when they complete the task that they were doing when the helicopter landed. Otherwise, they perform the scan1 operations for all cargo for the waiting helicopter and then go service the helicopter.
4. A helicopter that had landed while the loaders were busy doing other tasks is waiting to be loaded. The same rules apply here as for 3.

5. The loaders have finished servicing a helicopter so the loaders walk back to the fence and empty the unload cart. If the shippers are waiting for cargo on the unload cart, the loaders give the cargo straight to the shippers (via the scan 4 operation). Otherwise they put the cargo in the out bin.
6. A shipper brings cargo to the shipper service area for delivery. The loaders perform the scan 1 operation if they are idle at the fence. Otherwise the shipper waits until the loaders are idle at the fence and the loaders have no higher priority tasks to accomplish.
7. The loaders finish a scan 1 or a scan 4 operation and there are shippers waiting to drop off cargo. The loaders perform the scan 1.
8. A shipper comes to the shipper service area to pick up cargo. The loaders try to perform the scan 4 operation if they are idle at the fence. Otherwise the shipper waits until the loaders are idle at the fence and the loaders have no higher priority tasks to accomplish.
9. The loaders finish a scan 1 or a scan 4 operation and there are shippers waiting to pick up cargo at the destination. The loaders perform the scan 4 if they can find the cargo in the out bin or on the unload cart.

In the real world, the scanning operations are recorded on the scan gun and are downloaded on to a workstation and then sent to the data base repository. The loader with the scan gun is responsible for making sure that the scanning data gets back to the planners. The planners were not interested in modeling these downloading activities in the simulation. However, the planners were interested in seeing the simulated scan times. Therefore for simplicity, the scanning information for each cargo item is stored with that cargo item. Transfer times and overflight cargo operation times are not stored in the real world. However, the planners also wanted to see these times. Like the scan times, these times are stored as attributes of the cargo objects.

Current simulation architecture

When simulating complex systems, it becomes increasingly difficult to build analytically tractable performance models. Stochastic discrete event simulations, where systems are typically modeled by a network of queues, provide an analysis technique for studying complex systems (Franta, 1977). A discrete event simulation simplifies the problem by assuming that the system of changes state at discrete points in simulated time. Event-oriented discrete event simulations move from state to state upon the occurrence of an "event".

Creating a discrete event simulation in a flexible, comprehensible, and extensible architecture is not a simple problem. The Object-Oriented Paradigm (Booch, 1991) provides an effective way to model and to implement the complexity of large, non-trivial systems. An "object" is a well-defined entity with state (i.e., data attributes) and behavioral (i.e., operations or methods) information. The object-oriented programming (OOP) paradigm allows structural and behavioral aspects of similar objects to be captured in a "class". To create an entity of a particular class, the object is instantiated as an "instance" of the class. With the concepts of data encapsulation, inheritance, and polymorphism, OOP provides the ability to enhance modularity and to increase software re-usability.

With the ability to capture both structural and behavioral mechanisms in the models, object-oriented system simulation provides a way to facilitate representation with the benefit of OOP (Narayanan, Bodner, Sreekanth, Govindaraj, Mitchell, & McGinnis, 1994). Using OOP in simulation modeling allows a natural mapping that provides understandable, fundamental abstractions upon which to structure the simulation architecture itself and the entities modeled (Govindaraj, McGinnis, Mitchell, & Platzman, 1990).

Based on OOSIM (Govindaraj, McGinnis, Mitchell, Bodner, Narayanan, & Sreekanth, 1993), the ASTS simulation uses a discrete event simulation architecture employing an event calendar that maintains a time-ordered list of events and a clock to track simulated time. As the event-oriented simulation is written in C++ (Ellis & Stroustrup, 1990), the event calendar, the clock, the events, and all of the simulated entities (e.g., ATC, helicopters, heliports, loaders, and shippers) are modeled as objects having attributes and methods that manipulate the attributes. Simulated objects can schedule events (i.e., add events with associated execution times) to the event calendar's list of events. The event calendar then processes the events at the appropriate time. See Figure 6.

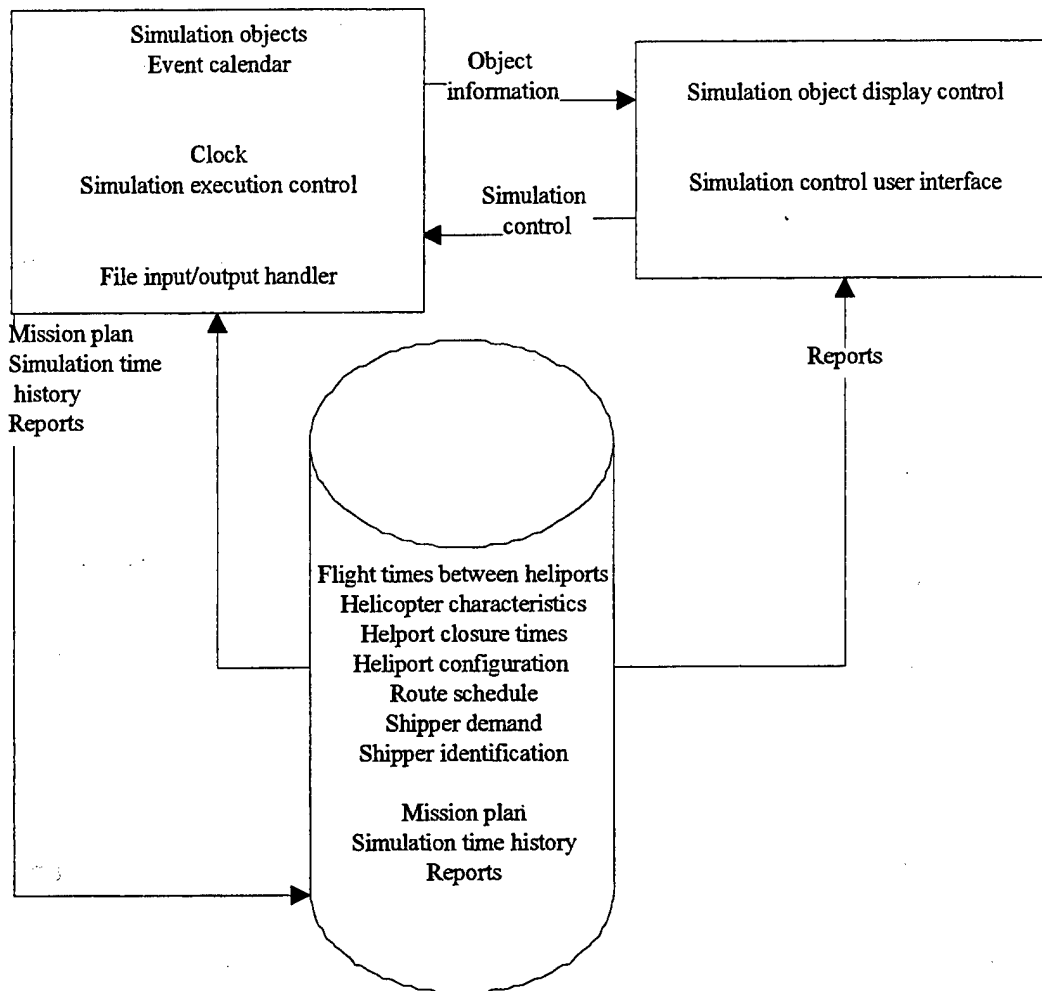


Figure 6 Architecture for the ASTS cargo helicopter decision support tool

Each event is defined by its execution time, the simulation object that is to perform the event, the method that the object is to use to perform the event, and any parameters (i.e., arguments) needed to execute the method. Thus when the event calendar processes an event, it has the event's associated object execute the method using the associated arguments at the appropriate time. That method then modifies the appropriate attributes in the model in order to update the simulation's state.

The event calendar has to check that the simulated time has reached the first event's execution time before it can process the event. It uses the simulation clock for this purpose. Based on how the analyst wants to run the simulation, this clock can either update simulated time at a fixed interval or based on the associated time of the earliest event.

With the fixed interval method, the analyst can set the interval to be the same as the system clock, thus having the simulation run in real-time. The analyst can set the interval so that the clock updates faster than real-time. For example, the analyst can set the interval such that each second of simulated time is a minute of real-time. In this way, the simulation runs sixty times faster than real-time.

To run the simulation as fast as possible, the analyst can set the clock to update based on the next event's time in the event calendar. On each simulation cycle, the simulation sets the time to the time associated with the next (i.e., earliest) event. In this way, one or more events are executed on each simulation cycle. During the ASTS

experiment, this latter mode of the running the simulation is used exclusively. On the simulation platform, a Silicon Graphics Challenger 10000, an entire simulated day executed in about four seconds including reading the input files and writing the output files. This fast execution time meant that making many simulation runs was not a limiting factor.

Although the ASTS planners did not use it during the Olympics, I built a computer-human interface using Tcl version 7.4/Tk 4.0, and iTcl version 1.5 (Ousterhout, 1994). The interface runs as a client process of the simulation. The two processes communicate via TCP/IP (Stevens, 1993). When running the simulation, the analyst must first start the simulation server process and then start the client computer human interface process.

On initialization of the user interface process, the analyst sees each heliport as two scrolling lists. One list represents the heliports' pad(s) and the other a consolidated waiting for scan 1 queue and load cart. Helicopters that land at the heliports are added to the appropriate pads scrolling list. When at a pad, a helicopter is represented by its N number and its flight name. Each cargo item waiting to be loaded is added to the other scrolling list and is represented by its cargo identifier. The idea is that the analyst can see for each heliport what helicopter(s) is(are) on the heliport's pad(s) and what cargo is waiting to be loaded on the helicopter(s) coming to the heliport.

The interface also has two other lists. One list shows all helicopters en route to a heliport. The other list shows all helicopters in holding patterns. Each helicopter is represented in the either list by its associated N number, its current flight name, and its current destination heliport name.

The analyst can get more information about each object in the interface by selecting the object (i.e., moving the mouse over an object and pressing the mouse button). In this way, the interface is not cluttered with information about every object - only the one(s) the analyst is currently considering.

While the simulation is running, the Tcl interface provides the analyst with the ability to change how the simulation executes. The analyst has access to buttons and pull-down menus that pause the simulation and change the speed factor (i.e., the relationship between the number of seconds of real-time per second of simulated time). The analyst can also run the simulation as fast as possible for a specified amount of time. In this way, the analysts can focus on the part of the day that is of interest and can skip over other time periods.

Whichever way the analyst chooses to run the simulation, he must make certain that the input files described in Appendix A are available on the platform running the simulation server process. The files tailor the operation of the simulation for the evaluation that the analyst wants to make. Also, whichever way the analyst runs the simulation, the output reports described in Appendix B are written. When not using the user interface, the analyst can view the files in a spread sheet. When using the user interface, the analyst requests the output by selecting the appropriate buttons. The simulation writes and displays the output based on the current state of the simulation when the analyst selects the output report. The simulation automatically writes a version of all of the output files to disk when the simulation ends.

Nominal cargo helicopter scheduling process

The "planning cell" at GTRI is the area for all cargo helicopter planning. The four planners (including the lead planner) plus one helicopter operations consultant participated in the planning period each day. The lead planner had the responsibility to coordinate the planning process and to disseminate the final results of the planning process. Each of the four planners was responsible for getting the demand data from three of the participating shippers and for coordinating any changes during the planning period. Usually the lead planner delegated part of the shipper coordination task for his assigned shippers to one of the other three planners.

The helicopter operations consultant advised the planners about the impact to the pilots and the helicopters from proposed changes to helicopter routes. This consultant provided guidance about flight time, refueling time, and crew scheduling requirements. He ensured that no trip from PDK to PDK included too much scheduled flight

time. He estimated flight times for heliport pairs for which there was no actual flight time data. He also coordinated with the helicopter and pilot provider to determine crew time constraints.

The planning period for the next day's schedule was supposed to begin no later than 4:00 PM and to end at 6:00 PM. The planners had previously instructed the shippers to fax, e-mail or telephone in their demand data for the next day before 4:00 PM so that the planners could begin planning on time. The planners were to complete the final version of the plan by 6:00 PM so that the lead planner could give the final route schedule to the person in charge of pilot scheduling and the loading zone report to the person in charge of the loading zones. In general, the planners did not meet the two hour time limit at the beginning of the experiment but improved over the experimental period. This improvement was partly caused by the shippers sending in less demand data with fewer errors in it. Also, as the days passed, the planners re-used parts of other schedules and other cargo capacity problem solutions. Additionally, the planners developed better strategies and the simulation reports were reformatted to accommodate these strategies.

No two planning periods were exactly alike. However, the planners followed a general procedure. Figure 7 and the rest of this section describe the nominal cargo helicopter scheduling process.

Select initial route data
Format and validate shipper demand data
Check for transfers
Solve capacity problems
Check for conflicts and late cargo delivery
Trim unused flights and update flight times
Substitute small helicopters for large ones
Create reports

Figure 7 Nominal cargo helicopter scheduling process

Select initial route data

The first step of the planning period is to select the initial route data to be used for the planning period. The route data includes the locations of, the scheduled times for, and the helicopter assigned to each flight for the entire day. During the week, the planners usually start with the complete weekday route schedule that had been developed after the first day of the experimental period. On Saturday, the planners use the unique Saturday route which had been developed before the experimental period began. One time a planner began with reduced routes but this strategy was not preferred.

The lead planner maintained the set of possible routes on his computer. To select the initial route for the simulation, the lead planner transferred a file with the appropriate route data into a file called route.dat on the simulation platform (see Appendix A for a description of route.dat).

Format and validate shipper demand data

The next step is to format and to validate the shipper demand data. Each of the four planners is responsible for getting the demand data from three of the shippers although the lead planner usually delegates his three shippers to one of the other planners.

The shippers each send their data to the appropriate planner. Most of the shippers mail electronically a spread sheet with demand data to the responsible planner. For the shippers who send faxes, the appropriate planner types the data into an electronic form. Most shippers send the data well before the 4:00 PM deadline but others send the data close to or after the appointed time.

The planners usually process each shipper's data as it becomes available. Each planner transfers the shippers' data files to the simulation platform. One planner, most familiar with the simulation environment, its input and output files, and its run-time messages, incrementally creates the demand data input file for the simulation from the separate shipper demand data files. In Appendix A, see the description for demand.dat for the desired format of the demand data coming from the shippers.

Validating the shipper data is critical in assuring that the simulation tool works properly. The planner usually runs the simulation each time new data is available so that any input problems can be fixed as soon as possible. The shippers' data is generally correct although the planners found typographical errors from time to time. Also, over the course of the experimental period, the lead planner changed the flights to reflect actual demand (e.g., unused and under-utilized flights were removed from the schedule and flight times were adjusted). Although the planners communicated the new route schedules to the shippers, the shippers did not always reflect these changes in their demand data. Flight name and time errors are the most frequent types of problems in the input data. For example, if the lead planner changes a flight to bypass its original destination because no cargo is going there, he eliminates the next flight and changes the previous flight's destination. The shippers' demand data then has the wrong destination flight.

The simulation stores input validation messages in a file called inputCheck.out. See Table 2 for a description of the simulation generated demand data validation messages seen by the planners during the experiment. In the table, capitalized words (i.e., CARGO, FLIGHT, HELIPORT, HELIPORT1, HELIPORT2, TIME1, TIME2) are place holders for values substituted during the running of the tool. Note that in the simulation, each cargo has an identifier that is a concatenation of the shipper designator, shipper type and an integer identifier (e.g., for CRD's cargo item 7, the identifier is CRDBNK7).

Table 2 Shipper demand data validations observed during experiment

Message	Meaning
Expected parameters in demand.dat are id, shipper, departure-airport, destination-airport, desired-scan2-time, scan4-time, weight, bags, volume Missing data parameters in demand.dat - see [echoes line of data] Fix demand.dat	Data fields are missing in a line of data. Shippers sometimes left out the shipper type and/or the number of bags in the spread sheets that they sent to the planners.
The origin airport HELIPORT for cargo CARGO in demand.dat does not exist. Not adding CARGO to the simulation.	The origin is not a valid three letter heliport name. Shippers sometimes make typographical errors with the heliport name.
The destination airport HELIPORT for cargo CARGO in demand.dat does not exist. Not adding CARGO to the simulation.	The destination is not a valid three letter heliport name. Shippers sometimes make typographical errors with the heliport name.
Pickup flight FLIGHT for CARGO does not exist - cancel.	The origin flight is not a valid flight name. The flight does not match any flight in the input route file route.dat. The shipper may have entered the flight name incorrectly or the flight may have a problem in the input file route.dat.
Delivery flight FLIGHT for CARGO does not exist - is CARGO on trimmed route? ¹	The destination flight is not a valid flight name. The flight does not match any flight in the input route file route.dat. The shipper may have entered the flight name incorrectly or the flight may have a problem in the input file route.dat.
CARGO in demand.dat should be picked up at HELIPORT1 but is assigned to FLIGHT originating at HELIPORT2.	The origin listed in the input file demand.dat does not match the origin for the origin flight. Shippers made typographical errors. Also the origin flight may have been changed during the route trimming process.
Destination for CARGO in demand.dat is HELIPORT1 but FLIGHT has destination of HELIPORT2	The destination listed in the input file demand.dat does not match the destination for the destination flight. Shippers made typographical errors. Also the destination flight may have been changed during the route trimming process.
Origin time for CARGO in demand.dat is TIME1 but FLIGHT has EDT of TIME2	The estimated departure time listed in the input file demand.dat does not match the estimated departure time for the origin flight in route.dat.

One planner generally fixed all of the typographical errors. However, sometimes the planner could not determine what was incorrect. In this case, he either called the shipper directly or asked the planner responsible for that shipper to determine how to fix the data.

Check for transfers

During one or more points in the scheduling process, the planners checked for transfers. A transfer is when a cargo item is picked up at the origin by one helicopter, is transferred to another helicopter at PDK (the only heliport where transfers were allowed to occur during the project), and is delivered to the destination by a second aircraft. Transfers generally occurred because the lead planner originally assigned a large helicopter for one trip from PDK to PDK and then assigned a small one for the next trip. A cargo item scheduled to be picked up at the tail end of the first trip (with the large helicopter) but delivered at the beginning of the next trip (with a small helicopter) would have to be transferred between the helicopters at PDK. The same type of transfer would occur if the first trip were flown by a small helicopter and the next trip by a large one.

¹ Note that originally this message was
Delivery flight FLIGHT for CARGO does not exist - cancel.
but it was reformatted after a route trimming problem.

Here is a fictitious transfer for illustration purposes. Suppose a shipper wanted to send cargo from NBE to BUC on early A flights A10 and A11. The shipper would indicate such a cargo item in the demand data in the regular way. Here is such an example.

33 SDC MLQ NBE BUC 8:52 9:35 A10 A11 70 5 15

Now, suppose the lead planner had assigned N001 (a small helicopter) to the first A trip and N006 (a large helicopter) to the second A trip in the input file route.dat as follows (with flights A04 through A08 and flights A12 through A18 left out for brevity):

A02	PDK	MIT	6:53	7:02	N001
A03	MIT	ATL	7:09	7:15	N001
...					
A09	NOR	NBE	8:40	8:45	N001
A10	NBE	PDK	8:52	8:56	N001
A11	PDK	BUC	9:30	9:35	N006
...					
A19	NOR	PDK	11:27	11:30	N006

The combination of these two events lead to a transfer. To deliver MLQ's cargo item 33 from NBE to BUC, helicopter N001 carries it from NBE to PDK, the loaders transfer it off N001 at PDK (around 8:56 AM), the loaders load it on N006 (around 9:30 AM), and N006 brings it to BUC.

The planners wanted to avoid transfers because there was an added opportunity for a mistake at PDK. The loaders might forget to unload the cargo at PDK as it is marked with another destination. The loaders might unload the cargo at PDK and forget to transfer it to the second helicopter. Another potential problem is that the first helicopter could be late and the second helicopter could leave PDK before the first one arrives. For this reason, the planners tried to eliminate transfers as much as possible.

Most of the time, the same planner that runs the simulation to validate the input data checks for transfers after the demand data has been validated. However, this same planner might instead check for transfers after the route trimming process. In either case, checking for transfers meant checking the simulation generated files called transfer.out and waitingTransfer.out. The transfer.out file includes all information about transfers:

- Existence of planned cargo transfers,
- Simulated time when cargo is transferred from the first helicopter, and
- Simulated time when cargo is transferred on to the delivery helicopter (assuming the timing is favorable).

The waitingTransfer.out file had a list of any cargo waiting at each of the heliports at the end of simulated time. If a cargo item is listed in this file, it means that the cargo item has been transferred off the helicopter that picked it up, but has not made it onto the delivery helicopter. Items are in the waitingTransfer.out file if the second helicopter involved in a transfer is not on the ground at PDK at least five minutes after the cargo has been transferred off the first helicopter (thus not giving the loaders enough time to move the cargo).

Most of the time, the file transfer.out had a one line entry:

No cargo has been scheduled for transfer.

When there is a scheduled transfer, transfer.out has the status of those transfer operations. A successful transfer would include three pieces of information:

1. The plan for the transfer,
2. The time when the cargo is transferred off the helicopter that picked it up, and
3. The time when the cargo is transferred on to the delivery helicopter (if the delivery helicopter was on the ground at PDK for at least five minutes after the cargo was transferred off the first helicopter).

Here is an example of the contents of transfer.out for a successful transfer operation:

```
The mission plan includes a transfer for cargo MLQSDC33 from N001 to N006 flight A11
Cargo MLQSDC33, which should be delivered on flight A11 of N006, removed from N001 at PDK at 08:56:48
Cargo MLQSDC33 transferred on to flight A11 of N006 at PDK at 09:30:00
```

If the transfer is not successful, the last line would be missing.

Most of the time no cargo is waiting for transfer so the waitingTransfer.out file looks like Figure 8:

```
For ATL
There is no cargo waiting for transfer.
For BUC
There is no cargo waiting for transfer.
For FTY
There is no cargo waiting for transfer.
For GAL
There is no cargo waiting for transfer.
For GBH
There is no cargo waiting for transfer.
For MIT
There is no cargo waiting for transfer.
For NBE
There is no cargo waiting for transfer.
For NBS
There is no cargo waiting for transfer.
For NOR
There is no cargo waiting for transfer.
For PDK
There is no cargo waiting for transfer.
For RAF
There is no cargo waiting for transfer.
```

Figure 8 Content of the file reporting the status of cargo waiting to be transferred when no cargo is waiting

If cargo does not get transferred, the cargo identifier of the cargo waiting for transfer would appear in the list under the appropriate heliport. Note that this report really should only have output for PDK because transfers are only allowed at PDK. However, the simulation tool is designed to allow transfers at any heliport and this design did not change to reflect the rule that transfers could only occur at PDK. The planner reading the report ignored the data for the other heliports.

Solve capacity problems

Once the demand data is validated, the planner running the simulation checks to see that all of the cargo has fit onto the desired flights. Capacity problems in the planning period are caused by shippers requesting cargo to be carried that exceeds the volume capacity of the assigned helicopter (weight was never the issue during the experiment). Solving these volume capacity problems is difficult for the planners. A special report, capacity.out, prepared to identify all of the cargo that cannot be loaded onto a helicopter because of a capacity problem, proved to be a helpful but not a sufficient report. During the experimental period, a second report was developed.

Before describing the additional report, here is a description of the strategy to solve capacity problems before the addition report was created. The capacity report, capacity.out, provides a listing of each cargo item that cannot be loaded. The order of the entries in the report is based on the simulated time that the loaders try to load the cargo. Note that as the planners often solved each capacity problem in the order of the report, this order could have been more helpful (e.g., the entries could have been ordered by flight or helicopter).

At first, the text in each line entry included the heliport, the flight name, the cargo item and its volume, the helicopter and the amount of cargo currently carried by the helicopter when the loading attempt occurs. Later a planner asked for the message to be shortened and for the remaining capacity to be listed instead of what is currently carried. The change allows the message to fit in a smaller window and on one line of a printed page. Also there is no need to subtract the carried load from the helicopter's maximum capacity. Here is an example of what a line item looked like at first:

At BUC B22, CRDBNK3 volume 18.0 exceeds N002 max. volume (carrying 48.5 cubic feet).

Here is what the same line item looked like after the format change, where "vol" means volume and "rem. cap." means remaining helicopter capacity:

At BUC B22, CRDBNK3 vol 18.0 exceeds N002 max. vol(rem. cap. 12.6 ft*3).

The second format is more helpful as it eliminates the need to remember which helicopter "N numbers" are large helicopters and which are small and how much capacity large and small helicopters have. The second format also eliminates the subtraction required to determine the remaining capacity. The second format is also shorter and thereby fits in narrower display window.

With either format, a planner has to figure out what to do - remove or reduce the cargo item in the report, and/or remove or reduce another cargo item (not in the report) so that the former item can be loaded. Usually the planner running the simulation starts to solve the capacity problems himself although he asks other planning team members for support as needed.

After printing the capacity report, the planner starts with the mission plan. The mission plan, stored in missionplan.dat, is a file generated by the simulation. It stores what cargo is to be picked up and delivered on each flight. Although it was created for internal use by the simulation, the planner consulted it to solve capacity problems because it consolidates the route data and demand data in a helpful way. The mission plan starts with the route data, sorts it by helicopter and estimated departure time, and adds which cargo is to be picked up and delivered by each flight. Cargo to be picked up at the flight origin is prefixed by a "P" and cargo to be delivered at the destination is prefixed by a "D".

Here is an example of the portion of the mission plan for helicopter N003's flight C32. Cargo items MLQSDC512, EXCSDC26, CRDBNK15, CRDBNK67, and CRDBNK70 are scheduled to be picked up at NBE at 3:26 PM. MLQSDC512 is scheduled to be delivered at RAF at 3:38 PM.

```
C32 NBE RAF 1526 1538 N003
P MLQSDC512
P EXCSDC26
P CRDBNK15
P CRDBNK67
P CRDBNK70
D MLQSDC512
```

By tracing through the mission plan, the planner can determine what cargo is on the helicopter during each flight. For example, to see what was on N002 which was causing CRDBNK3 to be bumped, the planner looked at the mission plan for flight B22 and for enough of the preceding flights to be sure he knew what caused the problem. Here is a portion of the mission plan for N002 during the CRDBNK3 capacity problem solving episode:

```
B21 PDK BUC 1245 1250 N002
P CRDBNK4
B22 BUC MIT 1257 1302 N002
P NBKBNK5
P CRDBNK3
B23 MIT ATL 1306 1310 N002
```

D CRDBNK3
D CRDBNK4

The planner usually prints the mission plan file. The planner crosses out cargo that was picked up and delivered in a few flights before the flight in question. The planner annotates the mission plan with the cargo volumes from the demand data for cargo that was on the aircraft at the flight in question. In this way, the planner determines what cargo is on the flight (both loaded on the current flight and loaded on a previous flight and still not unloaded). This process is quite tedious.

Once the planner knows what is on the aircraft, he checks if a single shipper has requested too much cargo for the flight. In fact, this is what happened in the example. CRDBNK4 and CRDBNK3 combined volumes add up to 65.25 cubic feet (which exceeds the helicopter's capacity of 61.0 cubic feet). The planner made this volume calculation by searching the demand data for those cargo items:

3	BNK	CRD	BUC	ATL	12:57	13:10	B22	B23	95	2	18
4	BNK	CRD	PDK	ATL	12:45	13:10	B21	B23	160	3	47.25

In the situation where one shipper is requesting more capacity than possible, the planner is justified in reducing the shipper's load by a bit more than necessary to get that shipper's load below the helicopter's capacity. If a planner could solve the capacity problem by reducing the shipper's demand by a small amount to solve the problem, the planner would "bump" (i.e., remove completely) or reduce the cargo item in the report and let the shipper know. Reducing in this context has a special meaning - split the cargo in the demand data by making two entries for the cargo in question. One entry would have a volume value that would fit while the other entry would have the "reduction". This reduced entry would have a "*" with its integer identifier which indicates to the simulation to load this cargo last (i.e., give it lower priority during the loading process).

In this case, CRDBNK3 changed from:

3	BNK	CRD	BUC	ATL	12:57	13:10	B22	B23	95	2	18
---	-----	-----	-----	-----	-------	-------	-----	-----	----	---	----

to

3	BNK	CRD	BUC	ATL	12:57	13:10	B22	B23	95	2	12
3*1	BNK	CRD	BUC	ATL	12:57	13:10	B22	B23	95	2	6

Note that the planner usually only changes the volume value (even though the number of bags and weight should be changed along with the volume). The other changes take too much time to calculate and do not effect the simulated results in a significant way.

If one shipper alone is not overloading the helicopter, the planner sometimes consults the spread sheet with the contracted amounts each shipper wanted to send. He sometimes asks the lead planner for help here because the lead planner created the spread sheet and knows how to interpret the information. The planners check if any of the shippers have exceeded the contract amount on the current flight in question or a previous flight impacting the current flight. If so, the planner removes or reduces the appropriate cargo. The combination of the mission plan, the demand data, and the contractual information help with this process.

Some capacity problems cannot be solved by reducing the shipper's total to the helicopter capacity or by consulting the contract information. The lead planner oversold some of the flights in the original contracts. The planner then tries to move cargo from one set of flights to another and still meet or be close to the requested times. He usually asks another team member for help with this task. He reads the origin, destination, and times to another team member who scans the route data and suggests an alternate flight pair (if one could be found). For example, flights A22, A23, and A24 go from BUC to ATL a little earlier than B22 and B23.

A22	BUC	GBH	12:27	12:31	N006
A23	GBH	MIT	12:38	12:40	N006
A24	MIT	ATL	12:47	12:53	N006

If the alternate flights have excess capacity, the appropriate planner coordinates with the shipper to see if the cargo can be sent on the alternate flights.

Into the experimental period, the planner used better methods to solve the capacity problems. One strategy is to run each shipper's demand data separately. In this way, any capacity problems are caused by that shipper and therefore are easier to resolve. Because the original demand data from each shipper is in separate files at the start of the planning period and because the simulation runs very quickly, this method is a simple way to solve the problems where a single shipper overloaded the helicopter. The planner simply copies a shipper's data into the input file demand.dat and runs the simulation.

Another strategy that emerged after a few days takes advantage of the fact that one particular shipper never really sent any cargo, even though it sent in demand data. The planner removes that shipper's cargo first when solving capacity problems. By consulting the mission plan, it is easy to see when that shipper's cargo is loaded on the same flight as a bumped piece of cargo.

With really difficult problems, the planner uses a related strategy. He asks the lead planner to check what cargo has actually been sent on that trip in the past few days to see if he can remove cargo that will most likely not materialize.

After a few days had passed, the planner also tried to reuse previous solutions. Sometimes the same cargo (or at least the same flight) would appear in the capacity report from one day to the next. If the planner notices this repetition, he checks the previous solutions. The original and incrementally changed demand and route data from all previous days is stored on the simulation platform so checking previous solutions was always an option.

For the eighth planning period, a new shipper plan report was available in a file called shipperPlan.out. This file stores the planned amount of cargo each shipper wants to have on each flight, either because the cargo is loaded on the flight or is still on the aircraft from a previous flight. Each line on the report is composed of the shipper designator, flight name, the shipper's total requested weight, the shipper's total requested volume, and all cargo identifiers planned to be on that flight by that shipper. Each field in the file is followed by a comma because the planners wanted to use the information in a spread sheet. In this way, the planners can sort by flight (when solving capacity problems) or by shipper designator when answering questions from management. As an illustration, the shipper plan report has the following entries for flight B22

Shipper	Flight	Wt	Vol	Cargo
CRD	B22	255	65.2	CRDBNK3 CRDBNK4
NBK	B22	15	1.1	NBKBNK5

The planner used this report to solve capacity problems once it was available. After transferring the file to his computer, he opened the spread sheet and added a column next to the volume column. In this column, he inserted a calculation that put an "x" if the volume column was greater than 61.0. Then the planner reduced or removed cargo for any shipper trying to single-handedly overload an entire small helicopter on a round robin flight. Note that the planner could have asked for the report to be reformatted with a new column to include the "x" but he performed this operation so quickly that he never made the request.

The planner also uses this report when there are capacity problems caused by the interaction of the demand of more than one shipper. The planner saw capacity problems in capacity.out such as these for flights C32, C33, and C34 (although entries for other flights were interspersed in the report but are not added to this example):

- At NBE C32, MLQSDC512 vol 15.0 exceeds N003 max. vol(rem. cap. 10.0 ft*3).
- At RAF C33, CRDBNK21 vol 9.0 exceeds N003 max. vol(rem. cap. 3.2 ft*3).
- At RAF C33, MLQSDC513 vol 15.0 exceeds N003 max. vol(rem. cap. 3.2 ft*3).
- At GAL C34, CRDBNK130 vol 9.0 exceeds N003 max. vol(rem. cap. 0.8 ft*3).
- At GAL C34, EXCSDC25 vol 10.5 exceeds N003 max. vol(rem. cap. 0.8 ft*3).

At GAL C34, CRDBNK10 vol 12.0 exceeds N003 max. vol(rem. cap. 0.8 ft*3).
 At GAL C34, NBKBNK12 vol 12.6 exceeds N003 max. vol(rem. cap. 0.8 ft*3).
 At GAL C34, MLQSDC514 vol 15.0 exceeds N003 max. vol(rem. cap. 0.8 ft*3).

Looking in the shipper plan report (sorted by flight), the planner saw the total weight and volume and all of the cargo the shippers want to put on those flights:

Shipper	Flight	Wt	Vol	Cargo
CRD	C32	288	40.5	CRDBNK15 CRDBNK67 CRDBNK70 CRDBNK44 CRDBNK24 CRDBNK49 CRDBNK60 CRDBNK109 CRDBNK17
EXC	C32	150	10.5	EXCSDC26
MLQ	C32	70	15	MLQSDC512
CRD	C33	376	56.2	CRDBNK62 CRDBNK22 CRDBNK21 CRDBNK15 CRDBNK67 CRDBNK70 CRDBNK44 CRDBNK24 CRDBNK49 CRDBNK60 CRDBNK109 CRDBNK17
EXC	C33	150	10.5	EXCSDC26
MLQ	C33	70	15	MLQSDC513
CRD	C34	414	63.8	CRDBNK55 CRDBNK74 CRDBNK158 CRDBNK130 CRDBNK10 CRDBNK62 CRDBNK15 CRDBNK67 CRDBNK70 CRDBNK44 CRDBNK49 CRDBNK60 CRDBNK109
EXC	C34	300	21	EXCSDC25 EXCSDC26
MLQ	C34	70	15	MLQSDC514
NBK	C34	290	19.6	NBKBNK41 NBKBNK12

Using the previous method (i.e., tracing through the mission plan and looking up each cargo item in the demand data) would have been very time-consuming. The shipper plan report provides a faster way to get this same information.

The type of problem illustrated above requires a planner to use several strategies in combination. As the capacity report showed problems for three flights in a row, the planner also used a strategy to start at the earliest flight, hoping that solving some of the capacity problem here would help downstream. Also the planner solved part of the problem, ran the simulation again to be sure his changes were fixing the problem, and continue iterating until the issues were resolved. During these iterative problem solving episodes, the planner checked the input validation report from time to time to be sure he did not enter any mistakes.

Check for conflicts and late cargo delivery

Either before or after solving the capacity problems, the planner checks to see if the helicopters are meeting the estimated times and to see if cargo meets the desired delivery times. If the helicopters are meeting the estimated flight times, then the loaded cargo would most likely be on time.

One reason a helicopter would not make an estimated arrival time is if there is a "conflict" at the pad. If another helicopter is on the pad at a non-airport heliport when another arrives, the helicopter cannot land because the non-airport heliports can only accommodate one helicopter at a time. The pilot must hold in the air near the pad in such a situation. This waste of costly flight time is to be avoided.

In the initial route schedule, all the flights times were selected so no conflicts would occur. However, after the first day, the lead planner changed the route schedule to increase the refueling time at PDK. This new route schedule had conflicts built into it. The lead planner hoped that as unused flights were removed, these conflicts would be resolved so he did not spend the time and effort to create a new route schedule free of conflicts.

A report, stored in a file called conflict.out, keeps track of every time a helicopter enters and leaves a fictitious holding pattern surrounding the heliport. The first column in each entry in the report has the flight name for the helicopter arriving at a non-airport heliport when another helicopter is on the ground. The second column is the holding helicopter's identifier. The third column is either "enter" for the start of the hold or "leaving" at the end.

The fourth column is the heliport. The fifth column is the start hold time or the end hold time. The final column, only filled in when the third column has "enter", is a remark stating which helicopter is at the pad when the arriving helicopter must hold.

Here is what that reports looks with the increased refuel time schedule:

C04	N003	enter	GBH	07:37:02	N002 at airport
C04	N003	leaving	GBH	07:38:02	
A46	N006	enter	GBH	19:11:02	N003 at airport
A46	N006	leaving	GBH	19:13:02	

In general, the planner considered holding times less than three minutes to be acceptable because he could fix such problems manually. In the final route schedule, the planner fixed these short conflicts by changing the ground times of the two helicopters to resolve the conflict.

A helicopter flight times report, stored in a file called flightTimes.out identifies when helicopters are late for any reason. The report has an entry for each flight by helicopter. Each entry has the planned and simulated (actual) departure time and arrival time. If the helicopter departs late, the difference between the departure times is displayed in minutes. If the helicopter arrives late, the difference between the arrival times is displayed in minutes. During the experiment, helicopters were usually late because of a conflict. Here is a portion of the flight times report showing the conflict at GBH for flight A46:

Helo	Flight	Leg	EDT	ETA	ADT	ATA	ADT-EDT	ATA-ETA
N006	A45	NBS-FTY	18:48	18:58	18:48	18:58		
N006	A46	FTY-GBH	19:05	19:11	19:05	19:13		2 MINS
N006	A47	GBH-GAL	19:18	19:25	19:18	19:25		

Helicopters could also be late if the loaders could not handle the cargo fast enough for the helicopters to meet the departure time. Normally this situation did not happen because the fifteen minute buffer period that shippers arrive to drop off cargo before the flight's EDT is sufficient. Helicopter could also be late if not enough time was scheduled for refueling. The simulation had a parameter for the minimum refuel time and no helicopter would depart from PDK until that time had passed. This situation occurred when the lead planner was validating a new route schedule and he noticed the discrepancy using this report.

During the design of the simulation tool, the planners requested a cargo summary report. This report was used very little during the experiment, because the same information was available in the demand data, the capacity report, and the conflict information. However, from time to time, the planners would consult it to see if the cargo met the delivery schedule when there were conflicts because a conflict does not necessarily lead to cargo being late (e.g., it is possible that the arriving helicopter was not delivering any cargo so the hold would not make cargo late).

The information in the report, stored in a file called cargo.out, is the same information as in the demand data plus a remarks field. The remarks field is usually blank. However, for cargo that was late (mostly because of conflicts), the remarks field has "HELO-LATE". For cargo that exceeds the helicopter's volume capacity, the remarks field has "OVER-V". Here are some example entries from the cargo summary report:

ID	SHIPPER	TYPE	EDT	ETA	ORG FLT	DES FLT	ORG	DES	WT	VOL	BAGS	REMARKS
MLQSDC321	MLQ	SDC	15:00	15:05	A31	A31	PDK	BUC	70	15	5	
MLQSDC354	MLQ	SDC	7:30	7:37	C04	C04	GAL	GBH	70	15	5	HELO-LATE
MLQSDC512	MLQ	SDC	15:26	15:38	C32	C32	NBE	RAF	70	15	5	OVER-V

The entries in the report are in the same order as in the demand data. The planner usually sorts this report by the remarks field. This way, the few cargo items with remarks are close together. The planner then can see all the "problem" cargo at one time.

Trim unused flights and update flight times

After processing the demand data, the planner eliminates unused flights. This process is called "trimming routes". In the design of the simulation tool, the planners had requested a report identifying all flights that had no cargo to pick up and nothing to deliver. The report generated for this purpose is in the file noAction.out. This report, however, was not preferred to using the mission plan for this purpose.

At the start of the route trimming process, the planner prints out the mission plan. He then asked for help from one of the planning team members. The team member looked for groups of unused flights and suggested changes while he typed the changes into a new route data file.

In removing flights, the planners want to be sure that they do not introduce conflicts. To meet this goal, they initially used two strategies for removing unused flights. They would hold helicopters at airports until the first necessary flight occurred. Also they would send helicopters to airports from the last productive heliport. By looking for the "P" and "D" entries in the mission plan, the planners knew what heliports were productive.

Here is an example of holding the helicopter at an airport until the first useful heliport. On one day, no activity was planned to occur on the B route until a pick up at ATL on flight B14. So the planners sent the helicopter from PDK directly to ATL. The mission plan for the early morning B flights looked as follows:

```
B01 PDK BUC 712 717 N002
B02 BUC GBH 724 728 N002
B03 GBH ATL 738 744 N002
B04 ATL NBS 756 815 N002
B05 NBS GBH 822 837 N002
B07 GBH GAL 847 854 N002
B08 GAL NBE 901 910 N002
B10 NBE PDK 917 921 N002
B11 PDK BUC 951 956 N002
B12 BUC MIT 1003 1008 N002
B13 MIT ATL 1015 1021 N002
B14 ATL NBS 1028 1047 N002
P CRDBNK104
```

To eliminate the unnecessary B flights, the planners changed the route data by eliminating flights B01 through B12. They made B13 the first B flight and changed this flight's times. The helping team member checked the flight time from PDK to ATL and mentally subtracted the flight time to calculate the new departure time from PDK to get to ATL by 10:21 AM. The new route data entry for flight B13 became:

```
B13 PDK ATL 1007 1021 N002
```

Here is an example where the planners sent a helicopter from the last useful heliport to an airport. In this example, there is a delivery at RAF but the helicopter is scheduled to visit NBE before returning to PDK. There is no cargo to deliver or pick up at NBE

```
B18 GAL RAF 1130 1140 N002
D MLQSDC412
B19 RAF NBE 1147 1158 N002
B20 NBE PDK 1205 1210 N002
```

In this situation, the planners eliminated flight B20 from the route data. They changed flight B19 to

B19 RAF PDK 1147 1204 N002

In this case, the team member looked up the RAF to PDK flight time and mentally added it to the original departure time.

Some flight changes meant that the demand data also had to change. If two flight were combined with cargo activity on each, the flight information on one of the cargo items had to change. Here is an example where two flights can be combined because there is a pick up on the first flight and a delivery on the second flight. The two flights can be combined to originate at BUC and end at ATL.

B22 BUC MIT 1257 1302 N002
P NBKBNK5
P CRDBNK3
B23 MIT ATL 1306 1310 N002
D CRDBNK3
D CRDBNK4

No matter which flight is eliminated, the demand data of something must change. If the two flights become flight B22 the demand data for CRDBNK3 and CRDBNK4 must have the delivery flight listed as B22. For example, B22 could be:

B22 BUC ATL 1257 1306 N002

The pick up flights for NBKBNK5 and CDRBNK3 would have to change if the two flights become B23:

B23 BUC ATL 1257 1306 N002

Making all of these flight changes is error-prone. The planners have to change a flight's origin or destination and to adjust the arrival or departure time to accommodate the change. To adjust the times, they check a hard copy listing of the flight times. For a new destination, they add this time to the departure time to get the new arrival time. For a new origin, they subtract this time from the arrival time to get a new departure time. Of course, they have to fix the demand data as well when certain changes are made.

The simulation generates messages to help validate the flight changes. These messages are stored in inputCheck.out along with the demand data validation messages. See Table 3 for the different types of route data validation messages seen by the planners during the experiment. In the table, capitalized words (i.e., FLIGHT, FLIGHT1, FLIGHT2, HELICOPTER, HELIPORT, HELIPORT1, HELIPORT2) are place holders for values substituted during the running of the simulation. The planners check the input validation report when they trim flights to be sure that they do not remove a flight needed by one or more cargo items.

Some of the demand data validations illustrated in Table 1 are designed to help the planners notice when flight changes have to be reflected in the demand data. The two messages related to flights not existing and the two messages related to mismatching locations direct the planners to cross-check flight changes with the demand data.

The planners also have to be sure that the simulation had the proper flight time parameters for the new heliport pairings created as a result of eliminating flights. In some cases, the planners were creating flights between heliports for which they had no flight time information. Although they had flight times for each heliport pair in the original schedule, flights between new heliport pairs were sometimes required after unnecessary flights were removed. The simulation uses a default value of thirty minutes for flights when there is no valid flight time data. The simulation displays a message on the screen when it uses the thirty minute value to remind the planners to get a better flight time value.

Table 3 Route data validations observed during experiment

Message	Meaning
The departure time for FLIGHT in route.dat is later than the arrival time so not added.	In the input file route.dat for the flight, the EDT is later than the ETA so that flight is not added to the mission plan of the assigned helicopter
Flight FLIGHT in route.dat has airport labeled NONE so not added.	In the input file route.dat for the flight, the origin and/or the destination is NONE so that flight is not added to the mission plan of the assigned helicopter
The origin and destination for FLIGHT in route.dat are both HELIPORT so not added.	In the input file route.dat for the flight, the origin and the destination are the same so that flight is not added to the mission plan of the assigned helicopter
[echo line in route.dat] For HELICOPTER, the previous flight ends at HELIPORT1 but the next flight FLIGHT2 starts at HELIPORT2 Previous flight is FLIGHT1 from HELIPORT to HELIPORT1 Fix route.dat See other input errors in inputCheck.out	In the input file route.dat, there are 2 consecutive flights for a helicopter where the destination of the first does not match the origin of the next.
Two route at same time? HELICOPTER [on display] FLIGHT2 for HELICOPTER in route.dat coincides with another flight at that time so FLIGHT2 is not added. Previous flight is FLIGHT1 from HELIPORT1 to HELIPORT2 Fix route.dat	The assigned helicopter has two flights occurring at the same time. The first processed flight (i.e., the previous flight) is added to the helicopter's mission plan but not the current one.

When the planners determine a better flight time between a pair of heliports, they must update the file used by the simulation for this purpose. The file flightTimes.dat stores the flight times for each heliport pair. This file has the time in seconds to fly between each pair of heliports. At the start of the experiment, all known flight times were entered and all unknown flight times were set to 1800 (30 minutes).

After the planners trim as many flights as they can by using the two airport strategies, they try to trim other unused flights. The strategy they use is to check the conflicts report, remove one flight, and then check the conflict report again. If they do not create a new conflict, they proceed with the next candidate removal. If they create a conflict, they either change the flights back to the way they were or try to manipulate the ground times of the two flights to eliminate the conflict. If the manipulation creates another conflict, they usually decide to add the flight back and move on to another candidate removal. Because changing the route data and running the simulation are fast, the planners do not mind using this approach.

Substitute small helicopters for large ones

Before preparing the final schedule, the planner checks the utilization of the flights assigned to large helicopters to see if small ones could be substituted. The planner uses the aircraft utilization report for this purpose. This report, stored in a file named aircraftUtil.out, contains for each flight, the basic flight information, the amount of cargo flown on this flight, the remaining capacity, and shipper information at the origin and the destination. The planner looks at this file in the volume column for trips from PDK to PDK for the large helicopters, N006 and N007. The planner scans the column. If he sees a trip with all volumes less than 61.0 cubic feet, he consults with the lead planner. If they both agree, he assigns the trip to a small helicopter available at that time. For example, here is the aircraft utilization report for a heavily loaded A trip

FLT	ORG	DES	DEP TIME	ARR TIME	HELO	WT	VOL	BAG	WT REM	VOL REM	ORIG SHIP	ORG WT	ORG VOL	ORG BAG	DES SHIP	DES WT	DES VOL	DES BAG
A51	PDK	BUC	20:35	20:40	N006	870	75.6	29.5	1630	152.4	3	670	50.5	19.5	1	70	15	5
A52	BUC	MIT	20:47	20:52	N006	870	75.6	29.5	1630	152.4	1	70	15	5	2	270	40	15

FLT	ORG	DES	DEP TIME	ARR TIME	HELO	WT	VOL	BAG	WT REM	VOL REM	ORIG SHIP	ORG WT	ORG VOL	ORG BAG	DES SHIP	DES WT	DES VOL	DES BAG
A53	MIT	GBH	20:59	21:01	N006	740	62.9	24.5	1760	165.1	2	140	27.3	10	1	70	15	5
A54	GBH	ATL	21:08	21:14	N006	740	62.9	24.5	1760	165.1	1	70	15	5	2	130	19.5	6
A55	ATL	NBS	21:22	21:41	N006	1255	160	34.8	1245	68.4	2	645	116.2	16.2	2	680	58.3	23.5
A56	NBS	GBH	21:47	22:02	N006	645	116	16.2	1855	111.8	1	70	15	5	2	645	116.2	16.2
A57	GBH	FTY	22:09	22:15	N006	120	23	6	2380	205	2	120	23	6	1	70	15	5
A58	FTY	GBH	22:22	22:28	N006	120	23	6	2380	205	1	70	15	5	1	70	15	5
A59	GBH	ATL	22:35	22:41	N006	120	23	6	2380	205	1	70	15	5	2	120	23	6
A60	ATL	PDK	22:48	23:01	N006	70	15	5	2430	213	1	70	15	5	1	70	15	5

In this case, the large helicopter remains in use because the first six flights have volumes greater than 61 cubic feet.

Create reports

The last step of the planning process is to create the reports for the pilots and the loading zones. The pilot report is the trimmed route data. The loading zone report is generated from a specially created file in the simulation called lzCaptain.out. This report lists a line entry for each flight to be loaded and unloaded at each loading zone. There were separate entries for transfers. Each entry includes the loading zone name, the flight name, the transaction (LOAD, UNLOAD, XFER ON, XFER OFF), the estimated time to begin the operation, the helicopter, the total cargo weight, the total cargo volume, the total number of bags, plus a break down of this total by shipper. If the helicopter has no cargo to pick up for a particular flight, the cargo values are all zero. With this report, the loaders know how early they need to be at the loading zone each day and what to expect throughout he day.

Here is an example of the report (with the shipper break down removed) for the early morning ATL activity. In this example, the first real activity is a load operation at 6:15 AM.

ZONE	FLT	L/U/TRANS	ET	AT	HELO	WT	VOL	BAGS
ATL	G02	UNLOAD	5:55	5:55	N007	0	0	0
ATL	H01	UNLOAD	6:08	6:08	N006	0	0	0
ATL	G03	LOAD	6:15	6:15	N007	600	80	20
ATL	H02	LOAD	6:20	6:20	N006	350	45	4
ATL	G05	UNLOAD	6:48	6:49	N007	0	0	0
ATL	H03	UNLOAD	6:59	6:59	N006	0	0	0
ATL	H04	LOAD	7:10	7:10	N006	1000	150	15
ATL	G06	LOAD	7:15	7:15	N007	2150	200	20
ATL	A03	UNLOAD	7:15	7:15	N001	170	33	7
ATL	A04	LOAD	7:22	7:22	N001	170	33	7

Detailed simulation usage description.

The planners never exactly followed the nominal daily cargo helicopter scheduling process as described in the previous section. There were always slight variations due to factors such as the quantity and quality of the shipper demand, knowledge they had acquired over time, extenuating circumstances such as a Presidential visit closing the airspace, time pressure, and upgraded simulation reports.

Overall the planners ran the simulation 144 times during twelve planning periods. In thirty nine of the runs, the planners either added new demand data or changed the route data to be used for the day. The planners used the simulation to solve problems in the other 105 runs. Table 4 and Figure 9 illustrate that the planners spent more time solving capacity problems than any other. Not only did the planners spend the most time on these problems,

but they also spent a lot of time per cargo item. In other words, there were not as many over-capacity cargo items as there were items with input data problems. However, each capacity problem took a lot longer to solve than each input problem.

The next most time-consuming task was validating the shippers' demand data including fixing the data as a result of the changing route schedule. Fixing each input error was not particularly difficult but there were many of these kinds of problems to solve. When the lead planner changed the schedule and during the route trimming process, the demand data did not match the route data so the simulation could not successfully add the cargo.

The planners spent a lot of time trimming unused and under-utilized flights. Figuring out which flights to trim and then figuring out how to manipulate the times of the remaining flights were time-consuming processes.

The planners ran the simulation four times when trying to get the simulation to behave as expected after an unusual outcome. One day the planners had to deal with a problem in the simulation because they did not fix the time errors

Table 4 Summary of the planners' usage of the simulation to solve problems

Task	Runs	Total Time in minutes
Eliminate transfers	1	4
Debug simulation	4	46
Solve over-capacity	27	547
Trim flights	31	368
Validate data	42	492
TOTALS	105	1457

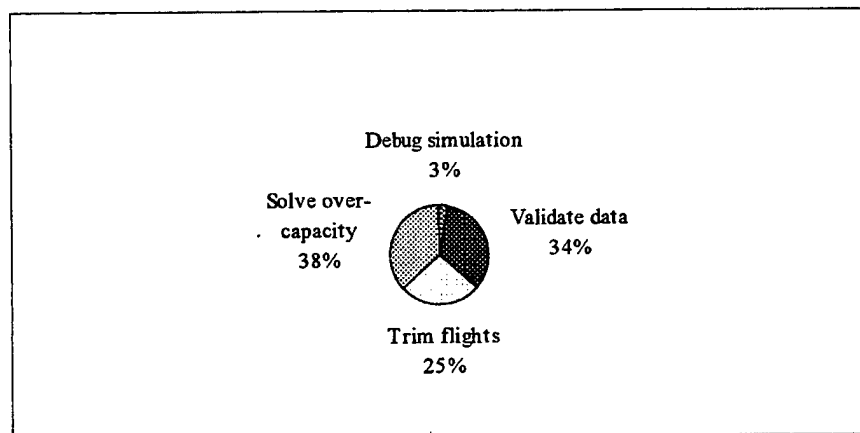


Figure 9 Percentage of time performing tasks with the aid of the simulation

in the demand data. The time errors caused unexpected behavior in the simulation. They were not sure why the simulation behaved as it did, so they tried changing the input in different ways to get the behavior they wanted.

In one run (not shown in Figure 9), the planners used the simulation to eliminate a transfer. This change went very smoothly.

Now I provide a more detailed description of how the GTRI researchers used the simulation during the experiment to create and to modify planned schedules. There is more detail for the first few days of planning; then there is a higher level of description for the latter days to minimize repetition.

First planning day

On the first planning day, the day before the first actual cargo operations, the planners started very late. They had trouble getting the data from the shippers. Some of the shippers sent in the data late because they were having problems with their electronic mail.

When the data arrived, it was not all formatted correctly. The lead planner used a spread sheet to validate the data instead of using the simulation tool as he was less familiar with the simulation validation messages. Setting up the spread sheet for each shippers data was time-consuming. Also, the spread sheet calculations did not run very fast.

Around 9:30 PM, the data was ready. See Table 5 and the description below it for a summary of how the planners used the simulation on this first day.

Table 5 Summary of simulation runs on Day 1

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 1 21:42	261 items-all shippers	Mon.-Fri. routes (15 min. refuel)	Validated data in spread sheet	14 items total 1 DED 4 ESTCNT 9 OVRCAP	Used capacity report, mission plan, and demand data to work on capacity issues
Day 1 22:45	Put * on 4 items Split 9 with * on overage			2 unexpected items on 1 flight	Searched demand data for items with *. Determined it was loading order issue. Added * to other part on 9 items
Day 1 22:51	Put * on other part			As expected	Wanted to eliminate item in capacity report on dedicated flights
Day 1 22:54	Reduce volume of 1 item			As expected - all items have *	Use aircraft utility report to make loading zone report

Here is an explanation of the symbols used in the tables detailing the planners' use of the simulation:

CNTRCT	Shipper requests more cargo volume on a flight than the maximum level in the contract
DED	Shipper requests more cargo volume on a dedicated flight than what fits on helicopter (planners let shipper know and shipper solves the problem)
ESTCNT	Planner estimated that this item must be over the contract amount because items from other shippers are either known to be within contract limits or are too small to make a difference
OVRBK	Lead planner overbooked the flights based on the maximum contract amounts.
OVRCAP	One shipper requests more cargo volume on a non-dedicated flight than what fits on helicopter

One planner ran the simulation. In the first run, the planner saw 14 items in the capacity report. He ignored the one on a dedicated flight because the shipper using that entire flight can allocate the cargo. To solve the other problems, the planner used the mission plan data and the demand data. He used the strategy of printing the mission plan, starting at the beginning of a trip where there is a capacity problem, crossing out items that are picked up and delivered before the flight in question, and hand marking the cargo volumes of the remaining cargo from the demand data in order to calculate what is on the helicopter.

There were 9 cargo items that could not be loaded because a single shipper requested to load more cargo on a flight than could fit. When the planner saw in the mission plan that a single shipper put too much cargo on a flight, he broke up the cargo that showed up in the capacity report into two items - one that would fit and one that would not. He marked the portion that would not fit with an asterisk ("*") so the loaders would try to load the overage after all the other cargo.

There were 4 cargo items that could not be loaded because a single shipper most likely violated the maximum contract amount. However, the planner did not check the contracts to make this determination. Instead, he saw that more than one shipper had cargo on the flight. He knew that one particular shipper always put the same volume on each flight and that this shipper's volume was always within contractual constraints. Therefore he used a process of elimination to determine which other shipper caused the capacity problem. To fix these problems, he used the asterisk method described above.

It took the planner a little over an hour to solve the 13 problems. When he ran the simulation, he saw that two new items showed up in the capacity report. Both of these items were loaded on the same flight. He looked at the other cargo loaded on the flight and the changes he had just made in the demand data by searching the demand data for asterisks. He saw that the other cargo had more volume than the portion of the original over-capacity cargo that did not have the asterisk. He realized that since cargo is loaded by volume from low to high, the

original over-capacity cargo was “bumping” these other items (i.e., the cargo without the asterisk was now making it on the flight because it was loaded before the cargo with the larger volume). He decided to change the demand data by putting an asterisk on both portions of the cargo that caused the original over capacity problems so that cargo would all be lower priority. In this way, that cargo (both portions) would be loaded after all the other cargo.

It only took the planner about six minutes to solve and to implement the solution. He looked at the capacity report and decided to reduce the one dedicated cargo item that was still in the report so it would show up in the reports as making it to the destination.

As it was so late in the day, after making that last change and running the simulation, the planner told the lead planner that the plan was completed. Using the route data, the lead planner created a report for the helicopter operator. The lead planner took the aircraft utilization report and created a report for the loading zones. This process was finished by 12:30 AM.

Second planning day

The next morning the planners found out that the President’s visit to Atlanta for the Olympic Games opening ceremony would close the airspace for most of the afternoon. They needed to coordinate with the shippers so they would know what flights would be canceled. They also needed to let the shippers know the outcome of the previous night’s planning period (i.e. what cargo was removed from the schedule or reduced in size).

Table 6 and the rest of this section describe how the planners used the simulation for cargo planning on the second planning day.

Table 6 Summary of simulation runs on Day 2

Day/ Time	Demand data	Route data	Validate data	Remarks
Day 2 8:50	Remove 2 items not sent. Change 7 items back			Start with previous night’s solution and contact shippers. One shipper not sending 2 items.
Day 2 9:00		Trim 27 flights including one in error	Saw that D21 was needed	Use mission plan and strategies to stay at airports to trim flights. Afternoon flights canceled as President’s visit closed airspace.
Day 2 9:02		Put D21 back in		
Day 2 16:36	7 items total for Saturday	Whole weekend route	No input errors	Plan for Saturday - initial run Saw a transfer
Day 2 17:56	Change 1 item’s destination flight due to trimming process	Trim 12 flights	Two flights at same time. Saw message to check flight time	Use mission plan to trim flights and checked demand data’s flights
Day 2 17:58		Removed flights intended to be removed in last run to fix two flights at same time		Checked flight times report because of flight time message from previous run
Day 2 18:02	Change demand data to remove transfer			Planning was complete (no transfer)

In their conversations with the shippers, the planners found out that two items that had caused capacity problems during the planning period were not going to be sent. A planner took those items out of the demand data, recombined the items that had been split the night before, and ran the simulation.

The planners had to coordinate with the helicopter company dispatcher to decide what flights to fly and what to cancel. Before making the call, they decided to trim unused morning flights as well. A planner used the mission plan and the "stay at airports" strategy to trim 27 flights. When running the simulation with these flights removed, the planner saw an error message in the input validation report. A cargo item was assigned to a flight that originated at a heliport that did not match the shipper's requested origin. Because the planner had trimmed the flight, the locations changed. The planner added the trimmed flight back, ran the simulation one last time for the day's plan, and gave the final route schedule to the lead planner.

Before the next planing period, the lead planner worked on a new route schedule for the round robin flights. The original routes had a fifteen minute window for refueling at PDK. However, after the first day of flying, it was evident that the pilots would need 30 minutes. Therefore the lead planner developed new Monday-Friday routes that would go into effect for the next planning period (i.e., not for this afternoon, but for the planning period after that). The lead planner distributed the new schedule to all the shippers.

That afternoon, the planners started planning the next day's schedule. The demand for that Saturday was light; there were only 7 items. The planner put all of the data in one demand file and ran into in the simulation. There were no input errors and no capacity problems. The transfer report showed that there was one transfer that the planners wanted to try to eliminate.

First the planner trimmed 12 flights using the mission plan and the strategies of leaving helicopters at airports. The planner ran the simulation after trimming the flights and saw in the input validation report that he had made a mistake (i.e., making two flights at the same time for the same helicopter). Also the new route schedule had a flight between two heliports that were not in the origin routes so there was no flight time information for this flight.

The planner fixed the route data, checked the flight time data, and ran the simulation again. He checked the flight times report to be sure that the helicopters were making the scheduled flight times. The routes were good so he changed the demand data to eliminate the transfer. He ran the simulation one last time. Since everything looked good, he told the lead planner that the simulation reports were ready.

Third planning day

The new 30 minute refuel schedule was used for this planning period. Table 7 and the rest of this section describe how the planners used the simulation for cargo planning on the third planning day.

Table 7 Summary of simulation runs on Day 3

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 3 11:46	163 items from 2 shippers	Mon. -Fri. routes (30 min. refuel)	1 location error, 4 flight names errors, 11 time errors (matched old schedule)	8 items total - all are the same shipper and 7 are same as Day 1	Started early as had some shipper data. Started with 30 minute refueling time routes with built-in conflicts No transfers
Day 3 12:37	Fixed input errors but missed 1		1 incorrect time		Compared route data with demand data to fix input errors.
Day 3 12:41	Fixed 1 time			9 items total - new one same as Day 1	Fixing the input data allowed cargo that did not get loaded previously make it in this run
Day 3 15:30	Added 9 items from 2 shippers		No errors	Same 9 items	
Day 3	Added 68		2 time errors	12 total (some new)	Planner working on 9 capacity

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
16:13	items from another shipper Remove 5		in new data	ones) after 5 CNTRCT removed	errors using mission plan found 4 items were not in contract. 1 item not in report was removed - not part of contact and it was bumping other cargo
Day 3 17:42	Added 11 items from 1 shipper			16 total (so 4 new ones)	No transfers Same 2 known conflicts

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 3 17:43	Remove 2 items			14 total after 2 CNTRCT were removed	Planner working on 12 capacity errors using mission plan found 2 items were not in contract.
Day 3 17:53	Remove 1 item			12 total after 1 CNTRCT removed	1 item not in capacity report was removed-not part of the contact and it was bumping other cargo
Day 3 18:10	2 items reduce			11 total after 2 CNTRCT reduced	Reduced 2 items to contractual amounts
Day 3 18:16	1 remove 3 reduce			8 total after 1 OVRCAP removed 3 CNTRCT reduced	1 item removed even though it could have been reduced
Day 3 18:20	Move 1 item to different flight			7 items after move 1 item in report	Time pressure since past 6:00 PM desired planning period end time. Tried putting cargo on other flights
Day 3 18:23	Broke up 1 item to contract amount and overage			8 items (really 7 since same items has 2 parts) 1 CNTRCT reduced but still in report	
Day 3 19:13		Remove 6 contiguous flights			Use mission plan to trim flights and luckily 1 conflict was solved
Day 3 19:32		Remove 27 flights (7 sets of contiguous flights)	Cargo on flight at wrong location		Use mission plan to trim flights Still have 1 conflict
Day 3 19:35		Added 1 flight back			Still have 1 conflict

Two of the shippers sent their demand data to GTRI before noon. A planner ran these 163 items through the simulation tool for validation purposes instead of using the slower spread sheet. There were 16 errors in the input validation report - 15 of them caused by the fact that the shippers used the old schedules. The planner checked the capacity report and saw 8 capacity problems. The planner also checked the transfer report and saw that there were no transfers.

The planner went through each message in the input validation report to fix the identified problems. The planner fixed the incorrect times by looking at the times in the demand data and comparing these times to those in the route data for those flights. The planner fixed the incorrect flight names by looking in the route data for flights at the times and between the locations the shippers wanted. In all cases, the flight name was off by one number (e.g., D39 instead of D40). The planner fixed the incorrect location by looking at the demand data for the cargo item and comparing the demand data to the route data. He saw that the times, the flight names, and the

destination matched actual flights in the new schedule but that there was a mismatch on the origin. So he changed the demand data to make the origin match the origin flight name. He made a note to double check that the shipper wanted the cargo to be picked up at the origin flight's origin.

After fixing all the input errors, the planner ran the simulation again. He saw that he forgot to make one of the changes so he fixed the last item and ran the simulation again. He looked at the capacity report and saw that there was now one more item (i.e., 9 total).

After lunch, he started analyzing the capacity problems by marking up the mission plan as on the first day. However, at this point, since there was plenty of time, he decided to look at the contract information to determine what items to remove and to reduce. This process was tedious because the planner had to go through the mission plan, cross out items delivered before the flight with the capacity problem, total up the cargo on each flight from each shipper, and then compare these totals to the contract data. Because flight names had changed from the time the contract data was compiled, the planner had to figure out which flights in the new schedule mapped to flights in the old schedule. Finally when the planner found situations where a shipper sent too much, he had to determine which cargo item(s) to remove from the demand data. Sometimes a shipper had many items loaded on to the same flight but getting unloaded at different locations so picking the right one(s) was time consuming.

Around 3:30 PM two other shippers sent their demand data. Together these shippers only had 9 cargo items. The planner added these to the simulation demand data and ran the simulation. There were no new input errors or capacity problems.

By 4:13 PM, the planner analyzing the 9 capacity problems found that 4 of the cargo items in the capacity report were not part of the original contract data. He also found another item, not part of the contract, that was successfully loaded on to a flight bumped cargo that should have been loaded. Because this item was smaller than the legitimate cargo, it made it but the larger cargo was bumped.

Before he made the changes to remove the 5 items, the planner was given the demand data from another shipper. This shipper had 68 items. The planner added these 68 items and removed the other five items from the simulation's data demand. He ran the simulation and looked at the input validation and capacity reports. There were two times errors in the new shipper's data and 12 total capacity problems. The planner did not compare the old capacity report to the new one to see how many of the twelve were new but he knew that at least 7 of them were (removing the 4 items that used to be in the report meant that the old report would have had at most 5 items plus removing the fifth item would have removed more).

The planner started working on the twelve capacity problems using the same method as before. It was getting close to the desired end of the planning period.

The last shipper that wanted to participate in the next day's shipping finally sent in their data. A planner added these 11 items and noticed there were 4 more capacity problems (16 total). This time the planner also looked at the transfer and conflict reports. There were still no transfers. The only conflicts in the report were the two that were built in to the 30 minute refuel schedule.

Right after the last new items were added, the planner working on the capacity problems removed two of the items from the simulation because they were not part of the contract. The number of capacity problems dropped to 14.

The planner who had been working on the capacity problems saw that three of the four new capacity problems were on contiguous flights that already had capacity problems. He determined that the last shipper added put cargo on these flights in excess on the contract amount. Although this item did not show up in the capacity report, it was the cause of two legitimate cargo items getting bumped. The planner removed that cargo from the demand data and ran the simulation. The number of capacity problems dropped to 12.

The desired end time for the planning period was approaching. The planner working on the capacity problems found two cargo items on those two contiguous flights that were over their contract amounts so he reduced those items. He saw 11 items in the capacity report after making these changes.

The time for the planning period to end had passed. The planner found 3 items that were in the capacity report that were above the contract amount. He reduced the cargo to the contract amount. He also found one flight on which one shipper requested more cargo than the helicopter could hold. He removed one item from that flight (although he could have merely reduced it). After making these changes, there were still 8 items in the capacity report.

The planner tried looking at other flights that could accommodate the cargo in the capacity report. He found one flight that was a half hour earlier than what the shipper wanted but could fit the item. He made the change in the demand data and ran the simulation. There were 7 items in the report.

The planner had previously seen that one of the items in the capacity report was over its contract amount, but even reducing this item would not have let it fit. He broke the item up into two entries (one for the contract amount and one for the overage) and put an asterisk on both parts even though he knew this would not solve the problem. When he ran the simulation, there were 8 entries in the capacity report but two of them were from this broken up cargo. He checked the other reports and saw there were still no transfers and there were still the two conflicts.

He printed out the mission plan in order to trim the unused flights from the schedule. He did not check to see which flights were needed by the currently bumped cargo. He went through the mission plan in the order on the paper. He used the strategies to keep helicopters at airports until the first useful heliport and to send helicopters from the last useful heliport to the next airport. He removed 6 contiguous morning flights from the route data and ran the simulation again. He checked the conflict report and saw that one of the conflicts no longer appeared.

The planner checked the mission plan again and saw several flights that could be trimmed. In this pass, he trimmed 27 flights. When he ran the simulation, he saw a message indicating that a cargo item was assigned to a flight originating at the wrong location. He checked that item's demand data and the route data and realized that he had removed a flight that was needed. He added the flight that had been removed. As the lead planner wanted to create the reports for the helicopter provider and the loading zones, planning activities ended at this point.

Fourth planning day

The planning that had taken place on the third planning day was to create a schedule for a Monday. Monday's plans were created on Saturday because many of the shippers did not want to staff personnel for the ASTS project on Sunday. Because all of the capacity issues were not solved during the third day's planning period, the planner that had been working on the capacity problems decided to come in on Sunday to finish. Table 8 and the rest of this section describe how the planner used the simulation for cargo planning on the fourth planning day.

Table 8 Summary of simulation runs on Day 4

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 4 14:01	Put together 1 item from 2 entries	Trimmed routes from previous day	2 time errors that were never fixed from run at 16:13	7 total from previous day 2 OVRCAP 2 OVRBK 1 CNTRCT	Started with last data from previous day but put item that had been broken up in the 18:23 run back together Used mission plan to trace capacity report items
Day 4 14:19	Remove 1 item * 1 item			Still had 7	Removed item not in report bumping another item but item was not large enough to let bumped item on Lowered priority of 1 item in report
Day 4 16:30	Broke up and * 5 items			Had 6 items not making it	Broke up bumped items and lowered priority Accidentally marked the wrong item with lower priority

In the first run, the planner put back into one entry the item he had broken up the day before in the 18:23 run because he wanted to solve the capacity problems from scratch. He saw the two time problems from the 16:13 run but ignored them as the times were only off by two minutes and were not affecting the simulation.

He printed the mission plan and started with the first item in the capacity report. By crossing out items already delivered in previous flights and annotating volumes for the cargo on the helicopter, he saw that one shipper had requested more cargo on the aircraft than it can hold.

The next two items in the capacity report were on contiguous flights so he worked on these together. He saw that two shippers together wanted to load too much cargo but neither one was overloading the helicopter. He checked with the lead planner and found out that this flight was oversold in the contracts. The lead planner thought that neither shipper would actually ship the maximum so that this problem would not really occur.

The planner looked at the fourth item in the report and saw that another item from another shipper with less volume and not in the contract was getting loaded and bumping the cargo. This item could be removed and make room for the legitimate cargo.

For the fifth item, he saw that two shippers together wanted to load too much cargo but neither one alone was overloading the helicopter. He made a note to check with the lead planner about the contract amounts for this flight.

The planner traced the mission plan for the sixth item and saw that one shipper was overloading the helicopter. He could reduce the item in question.

Before working on the last item, he made some changes in the demand data and ran the simulation again. He removed the item bumping the fourth item and lowered the priority of the sixth item (he could have removed it for the same effect but he wanted to use the report later to help remind him what to tell the shippers).

He kept looking at the data but he could not solve the problems. He made all of the bumped cargo lower priority and ran the simulation again. The planner mistakenly marked the wrong item at a lower priority but he did not catch the mistake. He could not stay longer so he left the data as it was; he had already spent 2.5 hours trying to solve seven capacity problems.

Fifth planning day

The planner who had been running the simulation and analyzing the capacity issues was not at work on this day. The lead planner was stressed because this team member was missing.

Table 9 and the rest of this section summarize the simulation runs on Day 5. The lead planner started the planning process at 4:00 PM. He already felt time pressure to complete the planning task because he knew that the other planner always tried to start earlier than 4:00 PM and still did not finish on time.

Table 9 Summary of simulation runs on Day 5

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 5 16:01	1 shipper - 149 items	Mon.-Fri. routes (30 min. refuel)	5 time errors, 3 wrong flights, 1 wrong location		Planner who had run the simulation and handled over capacity issues not there. Did not start early.
Day 5 16:20	Fix 3 wrong flights, 1 location		No new errors	Saw capacity problems but ignore them	Errors fixed here were same as Day 3 11:46 run so they were easy to fix
Day 5 16:33	Add 1 shipper - 64 items		2 new time errors - ignored	Had 6 items but did not analyze	Time pressure to add all shippers
Day 5 16:40	Add 1 shipper - 2 items		No new errors	Add a 7th issue	Time pressure to add all shippers
Day 5 16:45	Add 1 shipper - 8 items		Typo - oh instead of zero	Still 7 and no analysis	Time pressure to add all shippers
Day 5 16:52	Add 1 shipper - 5 items		No new errors		Time pressure to add all shippers
Day 5 16:56	Add 1 shipper - 11 items			3 new ones (10 total). Noticed that C33, C34 flights were full like Day 3	Time pressure to add all shippers
Day 5 17:00	Add 1 shipper - 73 items		2 wrong locations		Lead planner wanted to skip invalid items because of time but another planner saw one was a typo easily resolved and other was resolved by a call to the shipper

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 5 17:11	Fixed 2 items			19 capacity issues not analyzed	A planner looked at full flights (C32-C34) to see what was similar to Day 3's solution. Lead planner started trimming flights. The no action report showed 1 dedicated route could be deleted - that shipper's data was missing.
Day 5 17:48	Add 1 shipper - 5 items		No errors		Using no action report, decided there were 14 flights to remove
Day 5 17:50		Trim 14 flights			Simulation did not behave as expected. The overfly logic was invoked over ATL. Pads closed at midnight (end time in weather.dat). Planner suggested removing dedicated flights so fewer aircraft at ATL
Day 5 17:55		Removed H flights			This did not solve the problem so tried removing other dedicated flight on next run.
Day 5		Removed		All capacity	Simulation behaved as expected. Time had

17:56		G flights		problems not analyzed	run out so the lead planner went to create the reports of the non-dedicated routes.
Day 5 18:29		Ran the G and H flights			Simulation still behaved in unexpected way with only G and H routes. After looking at the detailed cargo report, cargoStatus.out, saw some cargo waiting for scans so tried reducing number of bags
Day 18:36	Reduced bags for dedicated route cargo				Simulation behaved as expected so lead planner created reports for dedicated routes

In the first ten runs, the lead planner kept adding the shipper data as it came in. In two of the ten runs, he fixed errors in the data related to wrong flight names. He was willing to forego fixing mismatches between the flights and their locations because of time pressure but another planner stepped in to resolve these problems. The lead planner did not fix incorrect times although there were seven time mismatches.

Because of the time pressure, the lead planner decided that he would not focus on fixing the capacity problems identified in the capacity report. One of the other planners noticed that there were three contiguous flights in a row that had many of the "bumped" items and started working on these issues by looking at solutions from Day 3.

Because the 6:00 PM planning end time was approaching, the lead planner started trimming flights even though he might have removed flights needed by some of the items in the capacity report (since these items do not get loaded, they will also not be unloaded and these cargo items' destination flights may appear to be unnecessary even though they are needed).

When the lead planner trimmed 14 flights and ran the simulation, the simulation displayed behaviors that were unexpected. The lead planner requested help in solving this problem from the simulation designer (i.e., me). I knew that the simulation had executed the overfly logic because of a debug message that flashed on the display during the run. I looked at the flight time report to see what flights were affected. I saw that the small helicopters never could land at ATL. Figure 10 shows the part of the flight times report that I consulted at this point. It shows that the large helicopters (i.e., N006 and N007) land at ATL in the morning but never leave and the small helicopters (i.e., N001, N002, N003, and N004) overfly ATL starting at the pad closing time (i.e., 11:59 PM). Apparently the small helicopters were holding at ATL and could never land. When all the pads closed at 11:59 PM, the overfly logic caused them to overfly ATL. Because all pads close at 11:59 PM, the small helicopters had to overfly the rest of their scheduled routes.

Helo	Flight	Leg	EDT	ADT	ATA	ATA
N001	A02	PDK-MIT	06:53	07:02	06:53	07:02
N001	A04	OVERFLY-ATL	23:59:00			
N001	A05	OVERFLY-NBS	00:17:30			
...						
N002	B08	PDK-NBE	09:06	09:10	09:06	09:10
N002	B10	NBE-PDK	09:17	09:21	09:17	09:21
N002	B14	OVERFLY-ATL	23:59:00			
N002	B15	OVERFLY-NBS	00:17:30			
...						
N003	C01	PDK-NBE	06:50	06:54	06:50	06:54
N003	C02	NBE-NOR	07:01	07:05	07:01	07:05
N003	C03	NOR-GAL	07:13	07:23	07:13	07:23
N003	C04	GAL-GBH	07:30	07:37	07:30	07:37
N003	C05	GBH-FTY	07:44	07:50	07:44	07:50
N003	C06	FTY-NBS	07:57	08:07	07:57	08:07
N003	C08	OVERFLY-ATL	23:59:00			
N003	C09	OVERFLY-MIT	00:05:00			
...						
N004	D02	PDK-NOR	07:16	07:19	07:16	07:19
N004	D03	NOR-GAL	07:26	07:36	07:26	07:36
N004	D04	GAL-MIT	07:43	07:51	07:43	07:51
N004	D05	MIT-FTY	07:58	08:03	07:58	08:03
N004	D06	FTY-NBS	08:15	08:25	08:15	08:25
N004	D08	OVERFLY-ATL	23:59:00			
N004	D09	OVERFLY-MIT	00:05:00			
...						
N006	H01	PDK-ATL	05:55	06:08	05:55	06:08
N007	G01	PDK-NOR	05:15	05:18	05:15	05:18
N007	G02	NOR-ATL	05:38	05:55	05:38	05:55

Figure 10 Partial flight times report with small helicopters overflying ATL at 11:59 PM.

I pointed out to the lead planner that the dedicated flights were not getting loaded within the allocated ground times. Because flights H01, G01, and G02 left later than their EDTs, I thought that perhaps the number of cargo items and number of bags for these cargo items may be so large that the scanning times model caused the loading to take too long.

The lead planner decided to remove the dedicated H flights and run the simulation again. Because this removal did not solve the problem, he then took out the G flights as well. The simulation behaved as expected. He decided to format the reports for the remaining round robin flights.

After he formatted those reports, he ran the simulation with only the G and H flights. He saw some unexpected behavior again. I remembered that the G and H flights had been late in the previous runs and I suggested that we look at the detailed cargo report, cargoStatus.out, to see if the scan times for the dedicated route cargo were later than the lead planner wanted. I saw that the scan times were not as expected. Knowing how the scan time model works, the lead planner suggested that he reduce the number of bags for these dedicated route cargo items to one.

After changing the number of bags for the dedicated route items, the lead planner ran the simulation and the simulation ran as expected. He formatted the reports and ended the planning session.

It is interesting to note that the lead planner tailored the number of bags attribute in order to run the simulation to create the loading zone report information. The report identified the cargo to be loaded on each flight but did not

accurately represent the number of bags. He did not need that level of description in the loading zone report because the loaders knew to load whatever the shipper wanted on to the dedicated flights. By representing the demand data at a higher level (each cargo item represented as one entity as opposed to being sub-divided into separate entities), the planner "faked out" the loading model which consults the number of bags when calculating loading time. Thus using his knowledge about what information he needed to get from the simulation and his understanding of the imbedded models, he changed the representation of the input data and traded off data he did not need for information that he did need.

Sixth planning day

The planner most familiar with the simulation and its environment was there for this planning period. He started the planning period early.

Since previous days showed that solving capacity problems were time consuming and that several of the problems were caused by a single shipper's demand, the planner considered using a new strategy of running each shipper's data separately and fixing the capacity problems caused by each shipper's data before running all the data together and looking at the interactions. Table 10 and the rest of this section summarizes the use of the simulation during the sixth planning day.

Table 10 Summary of simulation runs on Day 6

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 6 12:25	64 items from 1 shipper	Mon.-Fri. routes (30 min. refuel)	2 time errors	1 item - OVRCAP	Looked at demand data but it was no help. Checked mission plan for 1 capacity problem. Saw 2 items were on the flight. Ignored time errors as they were small
Day 6 14:10	12 items from 1 shipper		No errors	1 item - OVRCAP	Looked at item in demand data and saw an item directly below it for the same flight. Could tell that together they were too large
Day 6 14:15	2 shippers together			Existing 2 items	Ran data from previous two runs together to see if there were any interactions between the shippers' demands.
Day 6 14:42	11 items from 1 shipper		1 time error		The data came via fax. The planner compared it to the previous day's data. It had an updated flight time.
Day 6 15:34	15 items from 3 shippers, 1 item reduced			1 new item 1 existing item	Based on input from shipper, reduced item causing problem in 14:10 run. Looked at mission plan to solve new capacity problem. Tried moving cargo to new flight on next run.
Day 6 15:42	Moved 1 item to another flight			1 new item replaced the previous item	The moved item bumped a larger item.
Day 6 16:09	Moved item back to original flight			2 items as expected	
Day 6 16:11	Broke up item into 2			2 items (1 from Run 1; 1 the part with a *)	The item that had been moved between flights was broken up into two parts - one part with * and the other without.
Day 6 16:17	106 items from 1 shipper		1 typo - same as Day 5 17:00		

Day 6 16:18	Fixed typo			No problems	
Day 6 16:19	Add 1 item from 1 shipper			4 items total - 2 new ones (1 from new shipper)	Saw in mission plan which newly added item bumped the item from a different shipper that had made it before.
Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 6 16:26	Broke up 2 items			4 items total (1 from other shipper still there)	3 of the items had the * as expected. Because the newly added shipper had not really sent any cargo, the planner broke up its items.
Day 6 16:27	Remove portion of 1 item that had been broken up			3 items with * as expected	Planner realized that he had to reduce item loaded on previous flight instead of breaking it into parts
Day 6 16:35	Add 1 item from 1 shipper				
Day 6 17:30		Trim flights	Errors for items on trimmed flights		
Day 6 18:17	Remove 28 items on trimmed flights				

The planner started with the full Monday through Friday route schedule. In the first run, the planner saw 2 time errors in the input validation report, but he ignored them as the time errors were only for 2 minutes. He saw a capacity problem. When he looked at the item in the demand data, it did not help so he checked the mission plan near the full flight D45. He used the usual crossing out method on a printed copy of the mission plan and saw that two items were requested by the shipper to be loaded. The planner knew that one of these items should be reduced.

The planner ran the next shipper's data when it arrived. The data was valid but there was one capacity problem. He looked at the item in the demand data and noticed that the item below it was also to be loaded on the same flight. He noticed that the volume of these two items would overload the helicopter so he did not need to consult the mission plan to figure out what was happening. He knew that one of the items needed to be reduced. He decided to telephone the shipper about this issue later.

He tried out a new strategy of running only these two shippers together to see if there were any new capacity problems. There were no new ones.

The next shipper's data came in via fax. The planner compared it to the previous day's data and it matched except for an updated flight time. The flight time would not affect the simulation. Not wanting to waste time typing the data into electronic form, the planner re-used the previous day's data. There was one time error but no new capacity problems in this run.

The planner called one of the shippers about reducing the load on a full flight. The shipper told him which item to reduce. Around the same time, three shippers sent in their data. He ran the simulation after reducing the one item and adding the data from the three shippers. There was a new capacity problem. The planner looked at the mission plan. In tracing, he saw that there was still cargo on the aircraft from the previous trip. In this case, flight B40 coming back to PDK from NBE had cargo still on the aircraft by the time it went to GBH to pick up more cargo:

B40 NBE PDK 1740 1744 N002
P NBKBNK21
P NBKBNK56
B41 PDK BUC 1815 1820 N002
B42 BUC GBH 1827 1831 N002
B43 GBH ATL 1838 1844 N002
P ACSDC4
D ACSDC4

The planner looked at the route data to see if the item loaded on flight B43 could go on another flight. He saw a flight at a good time but he did not check the aircraft utilization report to see if the cargo would fit. When he moved the cargo and ran the simulation, the cargo made it but bumped a larger item.

The planner moved the item back to its original flight and ran the simulation to be certain that he saw the expected 2 capacity problems. He saw what he expected. Then he broke up the item into two parts - one part without an asterisk and one part with. He expected that the asterisk part would not get loaded and this is what happened when he ran the simulation.

The planner then broke up the item from the first run into two parts - one with an asterisk and one without. He ran the simulation and saw two items in the capacity report - both with asterisks.

He ran a new shipper's data alone. He saw a typographical error in one of the heliport names which was the same as in the 17:00 run of the previous planning day. Another planner told him what the name should be. There were no capacity problems.

He fixed the heliport name, added this data to the existing data and ran the simulation again. He found 2 new capacity problems. One of the items was the new shipper's item. He decided to reduce this item because that shipper had not sent any cargo in the "real world" so far. He thus used a strategy of reducing an item that most likely would not be sent anyway.

For the other item, he found out that a cargo item from the newly added shipper was loaded on a previous flight and bumped this item. He broke up the item from the newly added shipper.

When he ran the simulation, the other shipper's item was still not getting loaded. He realized that breaking up an item only works if the item getting bumped is loaded on the same flight. He realized that he had to reduce the item on the previous flight by the amount of capacity needed by the item loaded on the later flight. He reduced the item loaded on the earlier flight instead of breaking it into parts. When he ran the simulation, he saw what he was expecting.

The last shipper sent in the demand data. There was only one item so the planner added the item to the other demand data and ran the simulation. There were no new problems.

At this point, the planner focused on eliminating unused flights. On this day he decided to also trim flights if the only cargo on the flight belonged to the shipper that had not been using the delivery service in the real world. In deciding what flights to trim, he used the mission plan and the noAction report in the usual way. He also used the loading zone report in a novel way - he sorted it by flight and looked for flights with no cargo to load and unload.

He trimmed all of the flights in one run. The input validation report had messages for all of the items of the shipper that never really sent any cargo that were not going to be shipped because their flights were trimmed. The planner removed all these items from the demand data and ran the simulation for the last time.

Seventh planning day

The shipper plan report was added to the set of output reports for this and all subsequent planning periods.

The planners were told that the traffic conditions around the city were not as bad as had been projected so some of the shippers were not actually sending cargo on the helicopters even though they sent in demand data during the planning period. The planners were also told that a part of the airspace would most likely be closed the next day from 12:00 PM until 10:00 PM and the day after that between 9:40 AM - 1:30 PM and between 5:00 PM and 7:45 PM because of visits from the President. The helicopters would not be able to go to the downtown heliports (i.e., GBH and MIT).

During the morning, one of the planners spoke with a shipper that really wanted a particular flight in the mid-afternoon to go to a different destination. The lead planner checked the shipping data for the past several days and found that this change could be made without impacting any of the shippers that were actually sending cargo. The lead planner put a new Monday through Friday nominal route on the simulation platform with 2 flight changes (i.e., A28 going to a new destination with a new ETA and A29 leaving from that destination with a new EDT and ETA).

The planning period started at 3:00 PM even though the shippers' data was available earlier. The planners were in a meeting to discuss the airspace closing after lunch and could not start on the data earlier than 3:00 PM. They decided to plan as they normally did. If they found out when the airspace would close, they would remove/change the affected flights from the route data and see what cargo is affected.

The planners looked at the data they had received and noticed that the data looked similar to the previous days' data. Table 11 and the rest of this section describes how the planners used the simulation on planning day 7.

Table 11 Summary of simulation runs on Day 7

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 7 15:14	144 items from 6 shippers	Mon.-Fri. routes with A28-A29 change	1 12 minute time error, 1 flight name typo, 2 flight/location mismatch (A28, A29)		A28 to go to NOR instead of RAF and A29 to leave from NOR instead of RAF
Day 7 15:17	Fix typo; remove item from A28		1 flight name typo 1 flight/location mismatch		Item on A28 was item from shipper that was not really sending anything so the item was removed from the demand data
Day 7 15:19	Fix typo; remove item from A29		Still have time error	No problems	Item on A29 was item from shipper that was not really sending anything so the item was removed from the demand data
Day 7 16:20	Add 2 items from 1 shipper		No new errors	No problems	

Day 7 16:22	Add 64 items from 1 shipper		2 new time errors (3 total)	4 problems that looked familiar	One item in capacity report from shipper that never sends anything so it is removed in next run
Day 7 16:28	Broke up 1 item, remove 2 items			1 problem not addressed, 1 item with *	Looked at previous day's solution (split one item into 2 parts using *, removed 1 item from shipper that has not been sending anything)
Day 7 16:30	Broke up 1 item			2 items both with * as expected	Broke up item not addressed in last run into 2 parts - one part with *
Day 7 16:32		Routes for airspace closing	Many items with non-existing flight or mismatch flight/location; 3 existing time errors		Planners ran the data with the routes changed to accommodate airspace closing

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 7 17:00	Remove 104 items	Remove A33 Change A32			Planners used input validation report and removed items with non-existent flight or wrong location. When looking at demand data, saw cargo to MIT should not have made it.
Day 7 17:08		Fix flight time for A05	Messages associated with items on A33		During previous run saw new GAL, NBS flight so fixed flight times data
Day 7 17:12	Remove 7 items from A33 Fix 2 times				
Day 7 17:29		Trim B32-B38			Used mission plan to trim flights. Did not notice that items not going to GBH or MIT were eliminated in the big cargo removal step so trimmed some needed flights.
Day 7 17:37		Trim 5 flights			Used noAction report

A planner ran the data from the six shippers that had sent their data in previous to 3:00 PM. One of the shippers had typographical errors (oh instead of zero). One of the shippers had cargo assigned to the flights that had been changed earlier in the day. Because this shipper never really sent any items, these items were removed. It took three runs to fix the typographical errors and to remove the items assigned to changes flights from the data. The planner ignored the three time errors.

In the hour that the planners were waiting for shipper data, they created a new route file based on the airspace restrictions. No helicopters could go to the downtown heliports GBH and MIT during the restricted hours. Certain flights were removed and others had either their origin or their destination changed.

One shipper sent in data with only 2 items. Another shipper sent in data with 64 items. First a planner added the 2 items and ran the simulation. There were no new problems so the planner added the 64 items. There were 4 capacity problems that had been seen previously. One item was from a shipper that never really sent any cargo, so

this item was removed. The planner looked at the demand data for the previous day's planning period and found the solutions for two other capacity problems. He removed another item of the shipper that never sent anything. He broke up one item into two. When he ran the simulation, he saw that the fourth capacity problem remained so he broke up that item into a part that would fit and a part that would not.

Once the regular cargo planning was completed, the planners tried running the new route data that accommodated the airspace closing. When they tried it, the input validation check showed many cargo items with invalid flights or wrong locations. A planner took the report and removed all of the items from the demand data were identified. While looking at the demand data, he saw an item going to MIT during the airspace restriction period. He looked at the route data and saw that one of the flights going to MIT had not been removed. He changed the route data after he edited the demand data.

Note that the planner made a mistake when removing all of the items in the input validation report. When flights are trimmed, the origins and destinations of some flights change while other flights are removed. The demand data may then have the invalid flight names when other flights are still able to carry the cargo. For example, the demand data may have an item going from BUC to PDK with an origin flight A02 and a destination flight A03. If that item originally went on two legs (say from BUC to MIT on flight A02 and then from MIT to PDK on flight A03) and the two flights were combined to go from BUC to PDK now called A02, the simulation will flag the cargo item as invalid because it refers to a non-existing flight A03. This item should not be removed; its destination flight name should change to A02.

When the planner ran the simulation, he saw a run-time message stating that a flight between a new heliport pair was flown. He checked the flight time data for this flight and determined that it was too long. He consulted with the lead planner and they calculated a new time. He fixed the flight times data and ran the simulation again. This time he saw input validation messages for all the items for the flight going to MIT (i.e., A33) that was accidentally not removed. He removed the items from the demand data and ran the simulation again.

Other planners called the shippers to tell them which items cannot be sent because of the airspace closure. At this point, the planner used the mission plan and trimmed unused flights. In one run he removed the B flights. Then he checked the noAction report and found a few more flights to remove. He ran the simulation and told the lead planner that the information was ready.

At 6:00 PM the lead planner asked if a small helicopter could be used for all of the A route flights. A planner checked the aircraft utilization report and found that all the loads were less than the small helicopter's capacity.

That evening I noticed the problem mentioned previously - that the input validation report identifies items that can still be shipped but the flight names must be verified. I went through the items the planner had removed and showed him six items that were removed unnecessarily. Unfortunately some of the six items were on trimmed flights. One item could still be sent so the planner called the shipper.

Eighth planning day

Because of the flight trimming problem from the previous planning period, I changed the format of one message in the input validation report from:

Delivery flight FLIGHT for CARGO does not exist - cancel.

to:

Delivery flight FLIGHT for CARGO does not exist - is CARGO on trimmed route?

The planners wanted to start planning early so there would be plenty of time. One planner decided to start with the previous day's initial demand data figuring it would not be very different from what the shippers will be sending in their demand.

Because of the President's arrival, some airspace will close between 9:40 AM -1:30 PM and between 5:00 PM and 7:45 PM. No GBH or MIT deliveries can occur during those times. Instead of removing the appropriate flights from the route data, as he had done in the previous planning period, the planner sorted cargo by location and time and removed the cargo going to MIT and GBH during the restricted times. He did not change the routes because he thought the flights would get eliminated in the route trimming process. He did not consider the "lazy" route trimming strategy where only flights going to and from airport heliports are trimmed. He also did not mention his new strategy to anyone (eliminating cargo instead of changing flights). No one knew that he was running the simulation with the old route data.

Table 12 and the rest of this section describe the use of the simulation in planning on Day 8.

Table 12 Summary of simulation runs on Day 8

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 8 11:40	168 items from 8 shippers	Mon.-Fri. routes with A28 change	2 consecutive flights are impossible 1 heliport typo 2 time errors		Planner edited nominal route for A28 change but forgot A29 change

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 8 12:02	Fix 2 items on A28 and A29, fix 2 time errors, fix typo	Fixed A29 Assigned small on A11-A49	1 heliport typo		Planner was told to use smaller helicopter on A11-A49 flights if possible. So planner changed route.dat with N001 on those flights (from N006)
Day 8 12:04	Fix typo		No errors	4 problems- 1 OVRCAP	Planner wrote down how much of each item can fit based on remaining capacity. Used shipper plan for 1st time and mission plan
Day 8 12:45	Remove 2 items; reduce 2 items			7 problems (1 from previous run)	Used aircraft utilization report and shipper plan Reduce 2 items from OVRCAP Remove items of shipper that never sends anything
Day 8 12:59	Remove 1 item			6 problems (all from previous run)	Forgot to reduce item from previous run some more and to remove 5 items of shipper that never sends anything
Day 8 13:12	Reduce 1 item			5 problems (all from previous run)	All capacity problems for items from shipper that never sends anything. Looked at conflict report
Day 8 13:25	Remove 1 Reduce 4			No problems	Looked at flight times report and it agreed with conflict report
Day 8 15:04	Remove cargo from 2 shippers	Trim 37 flights	Saw message to check flying time		Used mission plan and noAction report to decide to trim flights 2 shippers not participating
Day 8 15:50		Adjust flight times Trim 4 flights			Adjusted flight time between one heliport pair Planner iteratively did route trimming Saw problem in flight times and conflict report - helicopter was waiting at non-airport heliport which had caused a long conflict
Day 8 15:59		Fix 1 flight			1 3-minute conflict
Day 8 16:10		Trim 32 flights			All reports looked good - still 3 minute conflict
Day 8 17:39		Remove MIT and GBH flights -	20 messages due to flight change		Because the planner did not remove all MIT and GBH flights during the restricted time, these flights had to be removed
Day 8 17:47	Fix 4 incorrect flight names Remove 14 items		No errors		
Day 8 18:04		Trim 6 flights			

The planner started with the nominal Monday through Friday route data. He made the A28 flight change that the one shipper had requested for the previous planning period. He added 168 items from eight shippers to the

demand data. When he ran the simulation, he saw two message related to the A28 route change - the route discontinuity between A28 and A29 and one cargo item with the wrong location. He also saw a message for one incorrectly entered flight name and two time errors.

The lead planner said to use a small helicopter on flights A11 through A49. The planner made the changes in the route data, fixed the errors and ran the simulation again. He saw another message about a flight name typographical error, fixed it and ran the simulation again. There were four capacity problems. The planner wrote down how much of each item could fit and requested that the capacity report message be changed to show remaining capacity on the helicopter instead of what was carried.

The planner looked at the mission plan for the first item and saw one item loaded onto the flight from the shipper that never sends anything. He decided to remove that item. He also saw that another item from the shipper whose item does not fit is loaded on the flight. He checked the shipper plan report for the first time and saw that the shipper's two items overload the helicopter. He decided to reduce both items.

He started on another item by checking the mission plan and the shipper plan report for the flight. Again he found an item from the shipper that does not send anything so he removed that item.

He looked at the mission plan for the next item in the capacity report and saw that the item was the only one loaded on the flight. The shipper plan showed that the shipper had requested too much cargo to be loaded on previous flights for this item to be loaded. The planner decided to reduce the item in the capacity report

He did not analyze one of the items at all. When he made the four changes to the demand data (removing two items and reducing two items), the original item he had ignored was still there plus six new ones. The planner was surprised. He started thinking about the loading logic - small volume items are loaded before higher volume items. That did not help him understand what happened. He looked at the capacity report and noticed that five of the six new messages referred to consecutive flights. These flights all occur after one of the flights that he had made sure an item that had been bumped would now make it. By making an item loaded earlier in the route make it, later ones were bumped. The planner said he would make a mental note to remember to focus on what flights now get more cargo when solving capacity problems.

Five of the items in the report are from the shipper that never sends anything so he decided to remove those and focus on the other two. The planner used the aircraft utilization report for first time to look at the loaded flights. He saw that the remaining volume for all of the flights was less than 4 cubic feet. He looked at the shipper plan report and saw that the items he had made fit were causing the problem so he decided to reduce one of the items he had already reduced by a little more.

He looked at the capacity report and started working on the item he had ignored in the previous run. He looked at the shipper plan for the affected flight. He saw and removed an item of the shipper that never sends anything.

Before he ran the simulation, he only removed the one item that was bumping an item from a shipper that really sent cargo. He forgot to reduce the item he had decided to reduce. He was puzzled by the result showing six items still in the capacity report (he was expected only the five from the shipper that never sent anything). He started looked at the mission plan and the shipper plan. The shipper plan's total volume for the affected flight reminded him about not reducing the item he had intended.

The planner looked at the conflict report for the first time this day and noticed 2 conflicts - 1 for 1 minute and 1 for 2 minutes. These were considered to be acceptable (note that these are the recurring ones that were built into the route schedule).

The planner removed one item from the capacity report (the helicopter was nearly full) and reduced the rest. When he ran the simulation, there were no more capacity problems. He checked the flight times report and it matched the conflict report (i.e., one flight was one minute late and another 2 minutes late).

After lunch, the planner started trimming flights using the noAction report and mission plan. He used the strategy to go from the last useful heliport to an airport and from an airport to the first useful heliport. A planner responsible for two shippers said neither shipper would participate so the planner removed all of their cargo from the demand file. Because of the interruption from this other planner, the planner decided to remove the cargo first before removing the rest of the flights because more flights could be trimmed based on the deletions. He trimmed 37 flights at this time.

During the run, the planner saw a message to check the flying time between a heliport pair that had not been previously flown. He looked at the flying times of two pairs which in combination were about the same length as this flight. He adjusted the flying time of the appropriate heliport pair in the flight times data and in the route data.

The planner trimmed 4 more flights and checked the reports. The flight times report showed a problem.

Helo	Flight	Leg	EDT	ETA	ADT	ATA	ADT-EDT	ATA-ETA
N003	C48	ATL-MIT	20:00	20:06	20:00	20:18		12 MINS
N003	C50	MIT-PDK	20:13	20:22	20:20	20:29	7 MINS	7 MINS

The conflict report also pointed to the same problem:

C48	N003	enter	MIT	20:06:02	N002 at airport
C48	N003	leaving	MIT	20:18:02	

The planner was not sure what was happening at MIT. He looked at N002's flight times and saw it was scheduled to be on the ground at MIT for 39 minutes (between 7:39 PM and 8:18 PM):

Helo	Flight	Leg	EDT	ETA	ADT	ATA	ADT-EDT	ATA-ETA
N002	B46	FTY-MIT	19:34	19:39	19:34	19:39		
N002	B49	MIT-PDK	20:18	20:27	20:18	20:27		

The planner checked the route data. When trimming the flights, the planner accidentally scheduled the helicopter to be on the ground for that time:

B46 FTY MIT 19:34 19:39 N002
 B49 MIT PDK 20:18 20:28 N002

The planner realized that the helicopter should be at FTY (an airport). He changed the route data and ran the simulation again. The original conflict went away but a three minute one was created:

B46	N002	enter	MIT	20:10:47	N003 at airport
B46	N002	leave	MIT	20:13:02	

The planner decided to continue to trim unused flights because the conflict might disappear. Also the planner felt pressure to finish as the real shipper data was mostly in and he was still working on yesterday's data. He went through the mission plan and noAction report. He trimmed 32 flights and ran the simulation. The reports all looked good although the 3 minute conflict was still there.

The lead planner asked if a small helicopter could handle the entire A route. The planner checked the aircraft utilization report and saw that large helicopters were only needed on the dedicated routes (E, G, H).

As it turned out, the data from the participating shippers was very similar to the previous planning day. Before leaving, the planner who had run the simulation told the lead planner to use the reports. The lead planner started working with the reports. Eventually he noticed that the GBH MIT flights were not all deleted. Another planner

took the old route data and figured out how to fix it. The route data was changed and the simulation was run again.

The lead planner fixed the incorrect flight names in the demand data. He deleted 14 items. He ran the simulation and used the noAction report to trim more flights. Six more flights were trimmed.

When the original planner came back, he was not sure why his cargo deletion strategy at the beginning of the day did not work.

Ninth planning day

A planner had asked for the capacity report message to be shortened and for the remaining capacity to be listed instead of what was currently carried on the helicopter. The change allowed the message to fit in a narrower editor window and to fit on one line of a printed page. Also there would be no need to subtract the carried load from the helicopter's maximum capacity. I reformatted the message from:

At HELIPOINT FLIGHT, CARGO volume FLOAT exceeds HELICOPTER max. volume (carrying FLOAT cubic feet).

to:

At HELIPOINT FLIGHT, CARGO vol FLOAT exceeds HELICOPTER max. vol(rem. cap. FLOAT ft*3).

where CARGO is the cargo identifier, FLOAT is the value to one decimal place, HELICOPTER is the helicopter's N number, HELIPOINT is the heliport name, "vol" means volume and "rem. cap." means remaining helicopter capacity.

In the morning, the weather was below VFR minimums so the helicopters did not fly. The planners used the time to compare the flight times used in the simulation with actual flight times from the experimental period. The planners changed three flight times in the simulation.

One of the shippers that was using the cargo delivery service faxed in a requested replacement for the C route. The lead planner worked on this new route schedule.

Table 13 and the rest of this section describe the use of the simulation in planning a very light day.

Table 13 Summary of simulation runs on Day 9

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 9 17:13	7 items total	Saturday and Sunday routes	Two routes at the same time message		Because Saturday and Sunday routes were in the route data, the same helicopter was assigned to two flights at the same time
Day 9 17:16		Change helo on 1 route	Flight name typos Still had route messages		
Day 9 17:21	Fix flight name typos	Remove routes for wrong day	1 flight incorrect 1 time error		
Day 9 17:23	Fix flight and time errors		No errors	No capacity problems	

Day 9 17:29		Trim 10 flights Assigned helo so no transfer			Used mission plan to trim flights Noticed transfer in mission plan Thought had finished trimming process.
Day 9 17:53		Trim 1 flight			When preparing loading zone report, noticed unload followed by load where no cargo was moved

The shippers were late in sending in their demand data but it was very light - only 7 items. The planner accidentally started with route data for both Saturday and Sunday. When he ran the simulation, he saw messages for multiple flights assigned to the same helicopter. He did not realize that he had the Sunday routes in the route data, so he changed the assigned helicopter on one route.

He ran the simulation but still had problems with the routes. He looked at the route data and noticed that the Sunday routes were in there. He removed them. He also fixed typographical errors that the shippers had made. These changes almost fixed all of the input errors. When he ran the simulation again, he fixed the last few input data errors.

The planner looked at the mission plan to start trimming flights. While he was selecting flights to remove, he saw there was a transfer. He assigned the same helicopter for the consecutive trips in order to eliminate the transfer. He trimmed ten flights.

He ran the simulation and thought everything looked good so he started making the reports for the lead planner. He noticed in the loading zone report that there were consecutive unload and load operations for a helicopter but no cargo was moved:

ZONE	FLT	L/U/TRANS	ET	AT	HELO	WT	VOL	BAGS
MIT	L01	UNLOAD	15:51	16:02	N005	0.0	0.0	0.0
MIT	L03	LOAD	15:59	16:02	N005	0.0	0.0	0.0

He trimmed the flight going to MIT and ran the simulation for the last time.

Tenth planning day

The planners used the new C routes for this planning period. They sent the schedule to the shippers but they did not believe that the shippers would incorporate all of the changes needed in the demand data. The planners expected that the shippers' demand data would have many errors on the new C routes.

These routes were never compared with the remaining routes in the existing nominal schedule to see if there were conflicts in the schedule. Therefore, the likelihood of having conflicts was high.

Because one of the shippers still had not really sent any cargo, the planners decided to use a strategy to only schedule that shipper's cargo on flights where there was "real" from other shippers.

Table 14 and the rest of the section describe how the planners used the simulation on Day 10.

Table 14 Summary of simulation runs on Day 10

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 10 13:38	165 items from 5 shippers	Mon.-Fri. routes with new C routes	Many errors		Planner went through input validation report to fix the flights that were changed on the C route.

Day 10 13:58	Fix 2 items Move 1 item to B Remove 8 items		Still many errors		Planner went through input validation report
Day 10 14:04	Fix 2 items Remove 6 items			2 problems on same flight OVRCAP	Looked at mission plan to fix capacity problem
Day 10 14:06	Remove 1 item			Still have same 2 problems	Looked at shipper plan report and saw than 1 shipper is overloading flight
Day 10 14:41	Add 10 items from 1 shipper		5 large time errors		The simulation did not behave as expected. The 5 conflicts were not very long.

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 10 15:23	Fix time errors			Still have 2 capacity problems	The simulation behaved as expected. Still have 5 conflicts
Day 10 15:28	Add 1 item from 1 shipper Change 2 items to 1 ft*3				Planner wanted to be able to trim flights to eliminate conflicts. He would fix the capacity problem later so he changed the items' volume to 1 ft*3.
Day 10 15:49	Remove 39 items	Trim 64 flights			Trim flights included those with only cargo from shipper that never sends any Tried new strategy of removing cargo in 1 run and flights the next Eliminated 1 conflict
Day 10 15:56	Remove 6 items	Trim 11 flights			Did not like the two pass strategy Down to 3 conflicts
Day 10 16:07		Trim 6 flights	Flights at the same time for 1 helicopter		
Day 10 16:11		Fix flight time mistakes			
Day 10 16:14		Trim 1 flight			Wanted to trim flights between non-airport heliports. Do 1 at a time and check conflict report. Saw new conflict but hoped it would go away later
Day 10 16:18	Remove 1 item	Trim 1 flight			Back to 3 conflicts
Day 10 16:22		Trim 1 flight	Fix cargo's origin flight		Still 3 conflicts
Day 10 16:27	Fix origin flight		Message to fix cargo's dest. flight		Accidentally removed item and did not notice until next day. Still 3 conflicts
Day 10 16:29	Fix dest. flight		No errors	1 item (only 2.2 ft*3)	
Day 10 16:30	Reduce 1 item			No problems	

Day 10 16:37		Trim 1 flight			Still 3 conflicts
Day 10 16:40		Trim 1 flight			Still 3 conflicts
Day 10 16:47		Trim 2 flights	2 flight name errors		Saw message to update one heliport pair flying time so checked with the lead planner and changed the flight time
Day 10 16:49	Fix 2 flight names		No errors		When trimming changed flight so had to update demand data
Day 10 16:55	Remove 1 item	Trim 2 flights	1 flight name error		
Day 10 17:00	Fix 1 flight name Remove 1 item				Still 3 conflicts
Day 10 17:01	Trim 2 flights, remove 3 items		2 consecutive flights impossible		Forgot to factor in ground time in flight trimming

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 10 17:08		Fix EDT for ground time	No errors		
Day 10 17:11	Trim 3 flights, remove 5 items, fix flight names 2 item		4 errors related to 1 flight		
Day 10 17:13	Remove 1 item, fix flight name 3 items				Still 3 conflicts. The planner fixed the conflicts by hand by manipulating the ground times

It took two runs to clean up the demand data from six shippers. The input validation report showed that demand data had items referring to flights that no longer exist. The planner removed items in the input validation report from the shipper that never sent anything and fixed the others. In one case he had to put an item on another route.

Then the planner focused on the two capacity problems. They were both on the same flight. First by looking at the mission plan, he saw he could remove from the flight an item of the shipper that never really sends anything. This removal did not solve the problem because the items getting bumped were large. He consulted the shipper plan report and saw that one shipper was trying to send too much cargo on the flight. The planner decided he would reduce the two items.

Another shipper sent in data so the planner added this information to the demand data. The simulation did not behave as expected. The input validation report identified five time errors; these errors were almost fifteen minutes long meaning that the shippers were barely dropping off the cargo before the helicopters' estimated arrival times. The planner looked at all the reports but could not figure out the reason why the simulation was not working. Eventually he fixed the time errors and the simulation worked properly. The planner knew that the simulation had not been tested with demand data with large time errors and guessed that this was the problem.

One shipper sent in demand data with only one entry and the planner added it to the demand data. He also reduced the two items in the demand data. He wanted to start trimming flights because the new C routes had built-in conflicts that he was hoping he could solve by eliminating flights.

The planner used the mission plan to trim unused flights and flights with only cargo of the shipper that never really sends anything. He made several changes and ran the simulation each time. As he trimmed flights, he checked the conflict report to see if any conflicts were eliminated. In the trimming process, the planner made mistakes that he had to fix in a later run. One time he forgot to factor in ground time; another time he factored it in incorrectly.

At one point the planner tried a strategy of removing items from the demand data in one run and then trimming the flights in the next run. He decided that he did not like this two pass method because he had to look at the same part of the mission plan again and again.

One strategy emerged for trimming flights that do not involve airports. The planner would trim one flight at a time and then check to see if there is a new conflict in the conflict report.

Eleventh planning day

The planners started later than usual in this planning period. Guessing that most routes unused one day will be unused the next, a planners tried a new strategy to start with the trimmed routes from the previous planning period. This ended up being a very poor strategy.

In the all of the runs on this day, the planner saw many input errors. In most cases, the origin and destination flight names in the demand data no longer existed because the flights had been trimmed. In some cases, the flight name was there but that flight's origin and/or destination did not match the cargo's origin and destination. In some lucky cases, the planner could put the cargo on the same logical flights that the shipper had intended (the names mismatch was caused by the route trimming process from the day before but the flights to the proper locations were still in the route schedule).

The planner went through each item and either changed the flight names to the trimmed route flight names, added back flights, or tried putting the cargo on other flights. The planner had to look in the route data to see if there were flights going to the right places at the right times. This was a time consuming process. He also made errors when adding the flights back into the schedule which meant he had to figure out the problem and fix it.

The planner spent over two hours trying to fix the input errors. By the he fixed all the errors, the planning period end time had passed. He had no time to analyze the conflicts or to perform the route trimming process.

Table 15 summarizes how the planners used the simulation during the eleventh day of planning

Table 15 Summary of simulation runs on Day 11

Day/ Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 11 16:13	88 items from 6 shippers	Trimmed route from previous day	Many errors.		Worked with the input validation report, the demand data and the route data
Day 11 17:07	Move 2 items to different flights, fix flights on 2 items	Add back 10 flights	16 flight name errors, 3 mismatch flight with location, 2 time errors		Started with location errors

Day 11 17:18	Fix flight name, change 1 item's origin from GBH to MIT		16 flight name errors, 1 mismatch flight with location 1 time error		Saw 1 flight name had oh instead of zero
Day 11 17:25	Fix oh/zero typo		15 flight name errors		
Day 11 17:32		Add back 1 flight	Still 15 errors		Started with early C flights. 1 item can be moved to D flights. Forgot to change demand data
Day 11 17:34		Add back 7 flights	Error in 1 flight added back in		Started with A routes
Day 11 17:36		Fix 1 flight	10 flight name errors		The departure time was later than the arrival time
Day 11 18:12	Move 2 items to different flights		9 flight name errors		Late!
Day 11 18:30	Move 2 items to different flights		5 flight name errors 2 items have mismatch flight with location		
Day 11 18:36		Change 5 flights	5 flight name errors 2 items have mismatch flight with location 1 time error		

Day/Time	Demand data	Route data	Validate data	Capacity problems	Remarks
Day 11 18:59	Change flights on 6 items		1 flight name error 1 mismatch flight with location		
Day 11 19:06		Change 2 flights incorrect -ly	Two consecutive flights are impossible		Check in demand data that no cargo needed GBH as destination
Day 11 19:08		Trim 7 flights	1 flight name error Arrival time earlier than departure time		While fixing error in route data, trimmed 7 flights
Day 11 19:13		Put flights back in	2 flight name errors 1 mismatch flight with location		
Day 11 19:21	Fix flights on 2 items		1 flight name error		
Day 11 19:24	Removed 1 item				No time to analyze possible route trimming or to fix 3 conflicts

Twelfth planning day

The project manager told the planners to create a schedule that eliminated the under-utilized flights. The idea was that they would publish this flight schedule and use it for the rest of the experiment. Shippers would no longer be required to send in demand data; they would just show up with their cargo at the heliports at the scheduled time.

The project manager suggested eliminating the D flights completely, the morning and evening A flights and the evening B flights. A planner started with this route schedule and used the demand data from the previous day in order to have representative demand.

Table 16 Summary of simulation runs on Day 12

Day/ Time	Demand data	Route data	Validate data	Remarks
Day 12 8:04	Previous day's data	Current nominal Mon.-Fri. routes	C20s and C30s errors plus a few others	
Day 12 8:17		Remove dedicated flights, all D flights, morning and evening A flights, unused C flights	Typo in route data	PDY instead of PDK
Day 12 8:19		Fix typo and added back C routes and dedicated flights so can leave demand data alone	Many A and D route problems	
Day 12 8:33		Changed A28 and 29 to what it should have been	Many A and D route problems	Started with item on A28 which had its destination changed several planning periods back
Day 12 8:46		Should have removed B evening flights so these were removed		
Day 12 10:04	Moved 1 item to different flight	Add back evening A flights Changed C flight to go to MIT	Many A and D route problems - 3 have wrong destination	Saw in mission plan B01-B20 unused In A11-A19 only A16 had 1 item so moved that 1 item 25 items on D flight need analysis - 2 morning items need to go to MIT but C only goes to GBH

Day/ Time	Demand data	Route data	Validate data	Remarks
Day 12 10:17		Changed 1 flight to correct destination	Many A and D route problems - 1 has wrong destination	Started with wrong destination messages
Day 12 10:26		Changed 1 flight to correct destination		At this point the planners printed the input check report, the demand data, and the route data and went through all of the bumped cargo. Hand marked the solution for all but 4 items

The planners started working with the nominal Monday through Friday schedule. A planner ran the simulation with the demand data from the previous day and saw many input validation errors related to the new C routes. In one run, the planner removed the D flights, the morning and evening A flight and the unused C flights. The planner realized that removing the C flights was a mistake because the project manager said those were well-utilized and so he added these flights back into the schedule. The planner also changed A28 and A29 to accommodate the one shipper who had wanted those flights changed. The planner also removed the evening B flights.

At this point, there were many A and D flight problems. The planner added back in the evening A flights because they were well-utilized according to the demand data. He started analyzing the D flight cargo to see if the cargo could be accommodated elsewhere. He started working on it item by item. Then he decided to print the input validation report, the demand data, and the route data. He worked with another planner and went through each item one by one. One planner marked the demand data while the other marked the route data.

At lunch time, the planners went to a meeting where they presented the results of what cargo could be accommodated with the routes generated. The project team decided to use that schedule for the rest of the experiment.

Requirements for an intelligent agent

The data collected during the experiment shows that planning for a cargo delivery system is a complex task. Good decision support tools are necessary. Because there are tasks that are difficult, time consuming, error prone and/or tedious for the planners to perform, there are several opportunities for providing intelligent support. The requirements for intelligent aiding are described in the order of the nominal cargo helicopter scheduling process summarized in Figure 7.

Route planning aid

The planners had to develop the initial route schedule. They analyzed the shippers' estimated demand and crafted the round robin and dedicated routes. Many would jump to the conclusion that an optimization would have been helpful. The issue is to develop a model with a meaningful objective function (for a definition, see Appendix C) and the appropriate constraints to capture the essence of the route planning problem without creating a combinatorial explosion (i.e., having too many alternatives to consider) on one hand or eliminating the possibility of any solution on the other.

When I started developing an optimization tool, the planners had trouble specifying the criteria for an objective function because they only had data concerning costs for the helicopter flight time and the total fixed portion of the flight time budget. The revenues in the experiment were set artificially low so the typical notion of maximizing profit did not apply. Minimizing cost was also not a possible solution because there were no fixed costs so the optimal choice would be not to fly at all. The lack of realistic demand, cost, and constraint information before the experiment made it very difficult to create a valid objective function and constraints. The objective function that the planners and I agreed on was to maximize the volume of delivered cargo without cargo being late and without exceeding a specified cost. We decided that cargo lateness could be a parameter that the analyst should be able to modify. The planners decided how much of the flight time budget they were willing to spend each day and that budget became a constraint in the optimization. Note that with ASTS's actual financial results, creating a realistic objective function would now be possible.

With a workable objective function in hand, Dr. Ellis Johnson suggested that I develop the optimization in a two step process:

1. Build a program to generate a set of feasible trips from PDK to PDK based on the demand data for each day, and
2. Develop a linear program (for a definition of linear programming, see Appendix C) to select among the trips, assign aircraft to the trips, and to determine which cargo would be delivered on which trip.

Dr. Johnson's idea is to run the trip generation program to determine a feasible set of trips and then to run the trips through a linear program to select the optimal set of trips subject to the appropriate constraints. For the linear program to run in a reasonable amount of time, Dr. Johnson explained that the trip generation program could not provide too many possibilities. To limit the number of trips, we decided to let a helicopter leave from PDK every fifteen minutes, starting at a time to get the earliest pick up and ending at the latest pickup time. The route generator would calculate many variants of each trip and prune less promising ones.

The trip generation portion of the problem is as follows (see Figure 11):

1. Set the trip start time to the next fifteen minute trip increment. Set the total flying time for the trip to zero.
2. Search through the demand data for cargo originating at PDK with estimated departure times at or a parameter number of minutes earlier than the trip start time and for cargo originating at heliports with estimated departure times at or another parameter number of minutes earlier than the current time plus the time required to fly there.
3. Branch on each piece of cargo. If the cargo has an origin of PDK, mark the cargo as having been picked up in this trip, and skip to step 6. If the cargo has an origin other than PDK and fly to the destination.
4. Update the total flying time for the trip and the current time.
5. If the helicopter is at a non-airport heliport, mark the beginning of the time on the pad.
6. If there is something to unload, unload all cargo that should be delivered to this destination and update the current time based on a pre-defined 3.5 minute ground time for unloading.
7. If the helicopter is not at PDK, check the total flying time for the trip and see if the flying time required to return to PDK is equal to or greater than the maximum flying time per trip. If so, load any cargo that is going to PDK, update the current time based on a pre-defined 3.5 minute ground time for loading, mark the cargo as being picked up on this trip, mark the end time for being on the pad, and go to step 10. If not, load all cargo that can be loaded (meaning any cargo that has not been picked up on this trip and that has an estimated time of departure that is earlier than the current time but later than forty five minutes ago), mark those pieces of cargo as having been picked up in this trip, update the current time based on a pre-defined 3.5 minute ground time for loading.
8. Mark the end time of being on the pad if at a non-airport heliport.
9. Branch at this point on the following possibilities:
 - a) Fly to a destination of one of the cargo items just picked up. Go to step 4.
 - b) Fly to the origin of any cargo that has not been picked up on this trip if the cargo has an estimated departure time that is later than thirty minutes ago and earlier than the current time plus the time required to fly to the heliport. Go to step 4.
9. Update the total flying time for the trip to PDK and update the current time. Refuel for the refueling time and update the current time. Mark the end time of this trip to the updated current time.

The linear program portion of the optimization is still under development as of this writing. The constraints in the model have to be encoded. The pertinent constraints are as follows:

- Availability of pads at the non-airport heliports - only one pad can be on the ground at a time
- Number of available helicopters
- Capacity of the helicopters
- Total helicopter flight time per day based on crew time requirements

The output of the route planning aid is the route data plus an updated version of the demand data with values for the origin flight, destination flight, estimated departure time, and estimated arrival time.

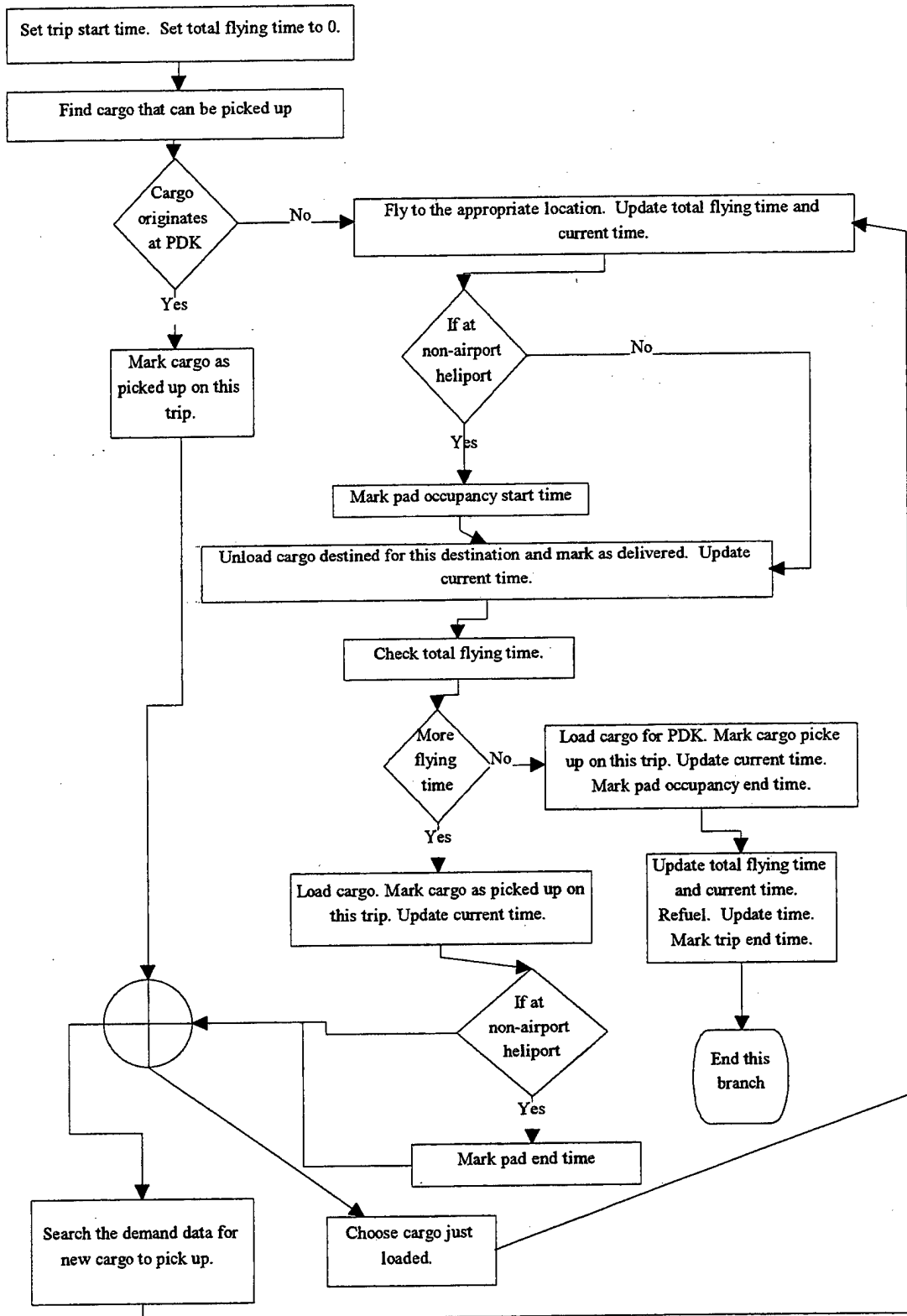


Figure 11 Trip generation logic flow

Input validation aid

The planners were generally pleased with the input validation message from the tool. However, in many cases, the planners had to edit the demand data when the solution to the problem was obvious. For example, in one case, all of the demand data matched a particular flight except that the shipper had typed the name of one of the heliports incorrectly. In another example, a shipper had the correct origin, destination, origin flight, and destination flight, but because the flight times matched a previous schedule's times, they had to edit the times. The tool could provide an option to "autocorrect" the misspelled heliport name in the first case and the times in the second case.

The "autocorrect" capability would not only save time the planners, but would also help focus the planners on "real" problems. However, not all input problems could be solved in this way. For example, if a shipper chose flights, origins, destinations and times that were not close to any known flights, the tool would have to identify the problem without suggested a solution.

The tool could also provide a list for the planners including each item that was corrected (either by the tool or by the planners) for shipper coordination purposes. In this way, the planners would not have to keep track of these changes manually. This capability should allow for incremental changes in the demand data as the planners liked to add different shippers' demand data as it became available and liked to make a few changes in each run. The tool should have access to the set of demand data files used in a particular planning session and look for changes for each particular cargo item. The tool should provide a report by shipper to make coordination easier.

One change to the tool (and to the knowledge available to the simulation) could eliminate the need for some of the validations caused by making flight name changes during the route trimming process. As mentioned previously, the planners like to eliminate unused flights and change the meaning and name of some flights. For example, flight B22 from BUC to MIT and flight B23 from MIT to ATL were combined when MIT was not a useful destination. The planners either called the resulting flight from BUC to ATL either B22 or B23. Shipper data then had to change to meet the new route locations and times.

If the planner could use some type of merged name capability during the route trimming process, such validations and data changes would not be necessary. In other words, when combining two flights, the new flight should have its name be a combination of the two flights (e.g., B22B23). Using the flight combination strategy, where the first name is the flight origin and the second name is the flight destination based on the original route data, would not only eliminate the need to fix the demand data, but would also provide traceability for the planners. They would be able to determine more easily which flights were combined.

Checking for transfers aid

The planners really did not have much experience with transfers during the experiment. The tool identified when transfers were required and the planner had no difficulty in removing transfers from the schedule. One service the tool could have provided is checking the utilization of the two helicopters involved in a transfer and suggesting using the same helicopter across the two routes if one (or both) were free for the other route.

Capacity problem aid

An intelligent aid would be greatly helpful in solving capacity problems as these types of problems were very difficult and time consuming for the planners. One simple enhancement is to compare the shipper plan information with the contract maximums to see if any shipper is sending too much cargo. The agent would need access to the contractual information for each shipper for each flight. In this way, when cargo cannot fit, the tool can determine whether this cargo can be reduced or bumped or whether some other cargo on the aircraft at this time should be removed. The agent could query the helicopter about what cargo is onboard from each shipper and then compare the shippers' total volumes to the contract information. The agent could then suggest which cargo should be reduced and/or bumped and/or possibly moved to other flights. If there are choices, the agent could

either suggest them and let the analyst decide or could incorporate rules for making these choices. More analysis is required about making these kinds of choices.

To keep the contractual information consistent with the evolving flight schedules, some convention should be implemented in the flight name changing regime. If the tool has access to contractual information tied to an old flight name and the name were changed in some way (perhaps in the route trimming process), it would have difficulty matching up the flights and contractual amounts. This issue provides more credence to the idea that the planners should keep the old flight names available (perhaps like the alias notion mentioned above).

The tool may not be able to resolve issues related to oversold flights although it could suggest alternate flights between the two destinations around the same time (if there is remaining capacity). The tool, if provided with proximity data, could also suggest alternate flights between close heliports. For example, if the flight from PDK to MIT could not accommodate a cargo item, there may be a flight around the same time from PDK to GBH.

Conflict resolution aid

If the route planning tool is successful in creating good route schedules, conflicts would most likely not occur. However, if the planners wanted to use a route schedule with conflicts and the tool were provided with minimum and maximum flight times between heliport pairs and a more accurate ground time model, the tool could provide resolutions to conflicts. For performance reasons, this process should occur after route trimming because conflicts often are eliminated by trimming one of the flights causing the conflict.

Flight utilization aid

The tool could easily implement the strategies of staying at airports and adjusting the flight time accordingly. The tool could even create the merged flight names mentioned above when combining flights. For unused flights between non-airport heliports, the tool could check for the potential for creating new conflicts by recording the number of conflicts, making route changes, running the simulation itself, and looking for new conflicts. It could only make changes when no new conflicts are created. With minimum and maximum flight times, the tool would have greater flexibility in resolving these types of conflicts.

Helicopter utilization aid

The tool could easily check the maximum capacity for each large helicopter route from PDK to PDK and suggest using a small helicopter for trips where a large helicopter has been assigned but is not needed.

Architecture implementing an intelligent agent

The approach taken here for integrating an intelligent agent into a decision support simulation to support route planning is user centered (Norman & Draper, 1986). The idea is to create an agent that supports the planners in the tasks that they do by performing the difficult and tedious tasks without taking away control and by making its actions clearly evident. For each of the planning functions summarized in Figure 7 plus the route planning task, the agent makes its actions evident by providing the same information to the planner that is in the original data files and reports.

Figure 12 and the rest of this section describe the functionality and the knowledge requirements for the intelligent agent.

The input checking in the simulation processes the appropriate errors. The agent should help the planners correct the input by including three functions:

1. Suggesting corrections to the demand data,
2. Editing the demand data should the planners agree to the correction, and
3. Prepare a report to help the planners verify changes with the shippers.

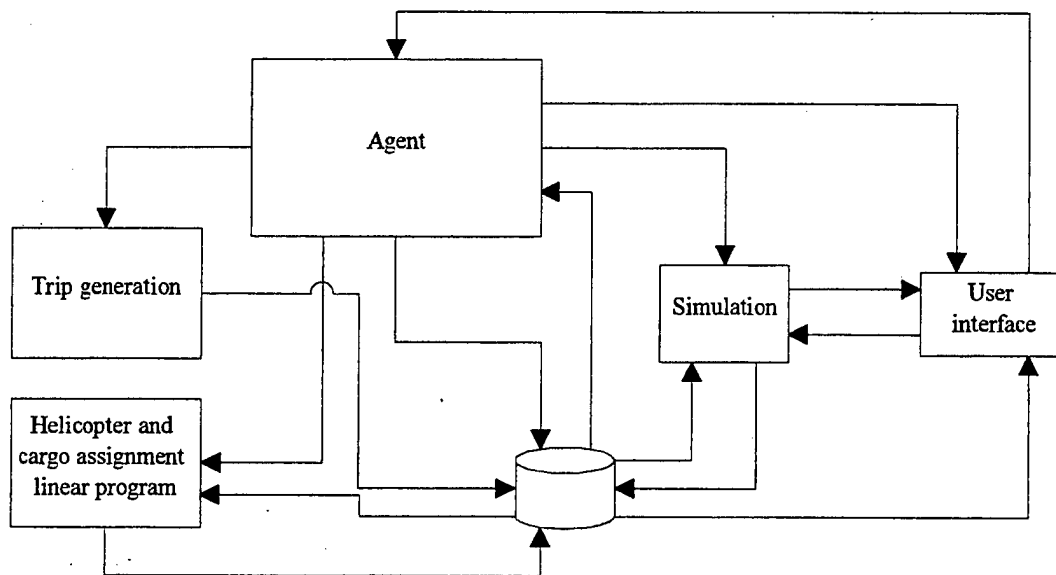


Figure 12 Architecture with an intelligent agent

To suggest corrections, the agent matches a cargo item's origin, destination, EDT, ETA, origin flight name, and destination flight name to the route data. If the origin flight name, EDT, and origin, match two of three of the attributes of a flight in the route data, the agent suggests to the planner that the third item should be changed to match the flight data. The same holds for the destination information.

The agent prompts the planner with these suggestions. The planner can either select that all of the corrections be made or can look at each one separately. For the corrections that the planners wants the agent to make, the agent adds a new line of data into the demand data file and comments the line with the old demand data and the current time. The agent also describes why the change was made (e.g., the flight and location matched but the time was 8 minutes early).

The agent also prepares a report for the planners to describe all the demand data changes that have occurred. The agent searches all of the demand data changes for that day, compares its earliest occurrence with its latest occurrence and, for items that have changed, prepares a report illustrating the changes. The planners can use this report when communication with the shippers.

For the route planning task, the agent runs the trip generation function which reads validated shipper demand data (looking at the origin, destination, EDT and ETA) and creates the set of possible routes. Then the agent launches the linear program that uses the possible routes to determine the optimal route schedule. The outputs of this task are the route data file, updated demand data with assigned origin and destination flights and times, and the report showing what cargo did not get assigned to a trip (see the cargoNoHelo report in Appendix B). The last report shows what demand, if any, could not be schedules based on the constraints in the linear program.

The planners can still use the simulation without running the route planning task. This flexibility is necessary because the planners may not want to change the published routes from day to day. They instead may want to use the original process.

The agent checks for transfers. When a transfer is identified in the transfer report, the agent checks to see if either of the involved helicopters can fly both trips. The agent uses the aircraft utilization report and the same crew time constraints available to the linear program.

The agent helps with capacity problems. If the planners provide a contract data file, the agent can suggest what cargo be removed from overloaded flights because of shippers' requesting more than the contract amount. Also

the agent identifies when a single shipper overloads a helicopter. The agent also suggests alternate flights around the same time, if there are any. If provided with heliport locations, the agent can also suggest flights around the same time to alternate locations that are nearby. If the planner uses the route planning function and finds cargo that has not been assigned to a trip, the agent can try to assign flights after the fact by using the same techniques.

To eliminate conflicts, the agent must be provided with minimum and maximum flights time so it can either shorten or lengthen flights to try to eliminate the problems. The agent must record the information about the current set of conflicts, make changes to the route, and then run the simulation to see if the changes are helpful. If so, the agent suggest the changes to the planner and reports the improvement caused by the change.

The agent can enhance flight utilization by implementing the same strategies as the planners: stay at airports until the first useful heliport and go to an airport from the last useful heliport. Also for non-airport heliports, the agent can trim flights and use the conflict report to look for new conflicts.

The agent can make suggestions to the planners about substituting small helicopters for large ones by looking at the aircraft utilization report. With the planner's permission, the agent should make the changes in the route data.

For all of the functions, the agent uses the same data and the same reports as the planners are used to seeing. In this way, the actions of the agent are understandable to the planners and they do not have to learn to decipher more reports. When the agent interacts with the planner, the agent write to the same files as before and uses enhanced displays on the exiting Tcl interface.

Future Work

The first step is to implement the intelligent agent. After implementing the intelligent agent, the next steps involve both improving the model and improving the simulation infrastructure. In the model, I need to allow more than one pad to close at the same time. I also need to enhance the simulation with a detailed model of the airspace and its restrictions and to enhance the aircraft model while it is flight. These changes are necessary to make the agent able to provide the planners better guidance on route planning. The planners need to be able to re-route aircraft at a finer level of detail; currently only re-routes occur when the aircraft is at its destination. For example, if a heliport closes, the aircraft does not know about it until it reaches that closed destination. In reality, the flight dispatcher can tell the pilot before the pilot leaves the ground at the previous location and can re-route the aircraft to fly directly to the overfly destination.

In the infrastructure, I want to be able to distribute the simulation objects and the agent into separate processes. In this way, the simulation and its architecture more neatly maps to the real world. The planners are in one physical space using one set of planning tools while the pilots are flying around in another physical space with another set of tools. Implementing a parallel discrete event simulation would allow the complexity of the simulation to grow while maintaining good performance (Fujimoto, 1990).

Acknowledgments

I would like to thank the ASTS research team at GTRI. I am thankful for the opportunity providing to me by the ASTS program manager, Charles Stancil. I appreciate the time, help, and confidence of all of the planning team members - especially Cliff Eckert and "soon to be" Dr. Robert Roglin.

I would like to that Dr. Alex Kirlik in the School of Industrial and Systems Engineering at Georgia Tech for his help in guiding my data collection and analysis of the planners' use of the simulation. I would also like to thank Dr. Ellis Johnson in the same department for his help in developing the initial route planning models for the linear program.

I also must acknowledge the current and former students at the School of Industrial and Systems Engineering at Georgia Tech who helped me with some of my design and implementation issues. Mike Albers, Alan Chappell,

Matt Harding, John Morris, Uday Sreekanth, and Dave Thurman helped me with various technical issues from casting pointers to freeing allocated memory to porting code from Sparc stations to Silicon Graphics platforms. Without their help, the implementation would not have progressed as quickly as it did.

References

- Booch, G. (1991). *Object Oriented Design with Applications*. CA: The Benjamin/Cummings Publishing Co., Inc.
- Ellis, M. A. and Stroustrup, B. (1990). *The Annotated C++ Reference Manual*. Reading, MA: Addison-Wesley.
- Franta, W. R. (1977). *The Process View of Simulation*. New York: Elsevier North-Holland.
- Fujimoto, R. M. (1990). Parallel Discrete Event Simulation. *Communication of the ACM*. 33(10), pp. 30-53.
- Govindaraj, T., McGinnis, L. F., Mitchell, C. M., Bodner, D. A., Narayanan, S., and Sreekanth, U. (1993). OOSIM: A Tool for Simulating Modern Manufacturing Systems. In *Proceedings of the 1993 NSF Design and Manufacturing Grantees Conference*, pp. 1055-1062.
- Govindaraj, T., McGinnis, L. F., Mitchell, C. M., and Platzman, L. K. (1990). Manufacturing Simulation Using Objects. In *Proceedings of the 1990 Summer Computer Simulation Conference*, pp. 219-224.
- Gurin, R. (1996). Shipping takes flight. *Automatic I. D. News* 12(9), p. 9.
- Narayanan, S., Bodner, D. A., Sreekanth, U., Govindaraj, T., Mitchell, C. M., and McGinnis, L. F. (1994). Research in Object-Oriented Manufacturing Simulations: An Assessment of the State of the Art. Submitted for publication.
- Nordwall, B. (July 31, 1995). Free Flight: ATC Model for the Next 50 Years. *Aviation Week & Space Technology*. New York: McGraw Hill, pp. 38-39.
- Norman, D. A. & Draper, S. W. (Eds.). *User Centered Design: New Perspectives on Human-Computer Interaction*. Hillsdale, NJ, Lawrence Erlbaum Associates.
- Ousterhout, J. K. (1994). *Tcl and the Tk Toolkit*. Reading, MA: Addison-Wesley.
- Stevens, W. Richard (1993). *UNIX Network Programming*. Englewood Cliffs, NJ: Prentice-Hall.
- Winston, W. L. (1991). *Operations Research: Applications and Algorithms*. Boston, MA: PWS-Kent Publishing Company.

Appendix A. Input files required by the simulation

The simulation requires seven files in order to run:

- airport.dat - heliport configuration
- demand.dat - shipper demand
- flightTimes.dat - flight times between heliport pairs
- helo.dat - aircraft characteristics
- route.dat - route schedule
- shipper.dat - shipper identification
- weather.dat - heliport closing times

The following is a brief description of each input file.

airport.dat - heliport configuration

The file airport.dat contains the heliport specific data. Each line in the file has data for one heliport. The file has three items:

- Heliport three letter designator
- Number of pads
- Number of loaders

The heliport designator is used to identify the heliport. These names are used in the demand data and the route data to identify origins and destinations.

The number of pads identifies how many aircraft can be on the ground at the heliport at the same time. For PDK, this number must equal the total number of aircraft in the simulation because all of the aircraft begin on the ground at PDK as defined in the file helo.dat.

The number of loaders identifies how many loaders are at the heliport. Currently this value should always be set to two as the simulation is designed to have one loader move the load cart and another the unload cart. The simulation code has only been tested with two loaders at each heliport.

demand.dat - shipper demand

The file demand.dat stores all of the shipper's cargo demand data. Each line of data has twelve items:

- Shipper assigned cargo identifier (an integer)
- Shipper type
- Three letter shipper designator
- Origin
- Destination
- Estimated departure time
- Estimated arrival time
- Origin flight
- Destination flight
- Weight in pounds
- Number of bags
- Volume in cubic feet

The integer identifier is a number used to identify the cargo item to the shipper. However, several shippers can use the same integer identifier. Therefore, in the simulation, the integer identifier, along with the shipper type and

shipper designator uniquely identifies the cargo item. Each cargo's identifier in the simulation is a concatenation of the shipper designator, shipper type and an integer identifier (e.g., for NBK's cargo item 56, it is NBKBNK56).

The origin and the destination are heliport three letter names. These items define where the shipper goes to drop off and pick up the cargo.

The estimated departure time is used by the simulation to calculate when the shipper drops off the cargo at the origin. The shipper drop off time entered by the simulation user (usually fifteen minutes) is subtracted from the estimated departure time to define when the shipper will arrive at the origin to drop off the cargo. If this time is later than the estimated time of departure of the origin flight, it is possible that the cargo will not make its flight.

The estimated arrival time is used to define when the shipper come to pick up the cargo at the destination. The origin heliport should match the origin of the origin flight and the destination heliport should match the destination of the destination flight.

The origin flight and the destination flight define which flights will be used to pick up and deliver the cargo. The loaders at the origin of the origin flight will try to load the cargo and the loaders at the destination of the destination flight will try to unload the cargo. The origin and the destination in the cargo demand data should match the origin of the origin flight and the destination of the destination flight as defined in the route data.

The weight defines the weight of the cargo in pounds. This weight is used to determine if the cargo exceeds the helicopter's weight capacity.

The number of bags is used by the simulation to determine how long the loaders will take to scan the cargo. In the model, the scan 1 and scan 4 process take 5 seconds per bag while the scan 2 and scan 3 process take 10 seconds per bag.

The volume defines the volume of the cargo in cubic feet. This volume is used to determine if the cargo exceeds the helicopter's volume capacity.

For example, shipper CRD wanted to send 135 pounds of cargo in three nine cubic foot bags from Galleria at 4:14 PM on flight D34. CRD wanted the cargo delivered to Hartsfield International by flight D37. That cargo appeared in the shipper's demand data as:

```
7 BNK CRD GAL ATL 16:14 17:17 D34 D37 135 3 27
```

flightTimes.dat - flight times between heliport pairs

The flight times data file stores the flight time between heliport pairs in seconds. Each line of data in this file has two items:

- The two heliports for which the time is valid in the format name1_name2_TIME where name1 and name2 are three letter heliport designators
- Time in seconds

In this file, there is one entry for each heliport pair. For example, to specify the 12 minute flying time between ATL and NBE, the entry in this file appears as follows:

```
ATL_NBE_TIME 720
```

helo.dat - aircraft characteristics

The aircraft specific data is contained in helo.dat. There are seven items per aircraft:

- Helicopter identifier or "N number" (N001, N002, N003, N004, N005, N006, N007)
- Size (large or small)

- Location at initialization (always PDK)
- Maximum weight capacity (1000 pounds for a small helicopter and 2500 for a large one)
- Maximum volume capacity (61 cubic feet for a small helicopter and 228 for a large one)
- Time for blades to spool up (i.e., time to reach steady state speed after start up)
- Time for blades to spool down (i.e., time to stop spinning after shut down)

The N number is used to identify the aircraft. This N number is also used in the route data.

The size of the helicopter is used in the heliport landings report to identify how many large and small aircraft land at each heliport.

The location at initialization defines where the helicopter is at the start of the simulation. This value is always set to "PDK" so that the helicopter starts on a pad at PDK.

The maximum weight and volume capacities define how much cargo can be carried. The weight is in pounds and the volume is in cubic feet.

The time for the blades to spool up and spool down define how long it takes for the helicopter blades to go from rest to spinning at steady state and vice versa. These time are used to defines how long the helicopter has to wait for it can request a departure clearance and how long a loader must wait after a helicopter lands before walking up to the aircraft respectively.

route.dat - route schedule

The route file contains the flight information for all of the helicopters. There is a line of data for each flight. Each line has the following items:

- Flight name,
- Origin,
- Destination,
- Estimated departure time,
- Estimated arrival time, and
- Assigned helicopter.

For example, flight D34 for helicopter N004 from Galleria to Mitchell Street at 4:14 PM would appear as the following:

```
D34    GAL    MIT    16:14    16:22    N004
```

shipper.dat - shipper identification

This file contains data for the participating shippers. There is one line of data for each shipper. Each line of data has two items:

- Three letter shipper designator
- Three letter shipper type (BNK for bank, CRP for corporation, LDC for long distance carrier, SDC for short distance carrier)

For example, the data for Airborne Express appears as the following:

```
ABX    LDC
```

weather.dat - heliport closing times

The file has one line of data for each pad opening and closing time. The format of each line of data is as follows:

- Heliport
- Pad (concatenation of heliport name, the word "pad", and an integer where the first pad is 0)
- Time pads opens
- Time pad closes

For example, if the planner wants to simulate what would happen if the only pad at NOR is closed between 9:00 AM and 10:44 AM, while all seven pads at PDK remain open, the NOR and PDK entries in weather.dat should appear as follows (assuming the simulated run begins after 1:00 AM and ends at or before 11:59 PM):

NOR	NORPAD0	1:00	9:00
NOR	NORPAD0	10:45	23:59
PDK	PDKPAD0	1:00	23:59
PDK	PDKPAD1	1:00	23:59
PDK	PDKPAD2	1:00	23:59
PDK	PDKPAD3	1:00	23:59
PDK	PDKPAD4	1:00	23:59
PDK	PDKPAD5	1:00	23:59
PDK	PDKPAD6	1:00	23:59
PDK	PDKPAD7	1:00	23:59

Appendix B. Reports available from the simulation

The following lists a brief description of each of the sixteen reports available from the simulation:

1. **aircraftUtil.out** reports utilization information about each flight. It reports how much weight and volume each helicopter is carrying on each flight. The report shows the amount of excess capacity for each flight. The report also includes information about which cargo shippers are expected to drop off and to pick up cargo for each flight. Here is an example for flight A51 which is carrying 870 pounds and 75.6 cubic feet of cargo. At its origin, there is 670 pound and 50.5 cubic feet of cargo to load and at its destination there is 70 pounds and 15 cubic feet of cargo to unload.

FLT	ORG	DES	DEP	ARR	HELO	WT	VOL	BAG	WT	VOL	ORIG	ORG	ORG	ORG	DES	DES	DES	DES
			TIME	TIME					REM	REM	SHIP	WT	VOL	BAG	SHIP	WT	VOL	BAG
A51	PDK	BUC	20:35	20:40	N006	870	75.6	29.5	1630	152.4	3	670	50.5	19.5	1	70	15	5

2. **capacity.out** reports all cargo items not loaded on to a helicopter because of a volume, weight, or volume and weight restrictions. Each entry in the report has the heliport, the flight name, the cargo identifier, the volume and/or weight of the cargo item, the helicopter N number, and the capacity information. Here is one example entry is this report:

At BUC B22, CRDBNK3 vol 18.0 exceeds N002 max. vol(rem. cap. 12.6 ft*3).

3. **cargo.out** reports the basic cargo data (e.g. shipper, origin, destination, weight, volume, flight) and remarks concerning the status of each piece of cargo. The possible remarks include late delivery by the helicopter, violation of helicopter capacity, missed transfer, and late shipper drop off.
4. **cargoNoHelo.out** reports all cargo that have been assigned to flights with no assigned helicopter. When all cargo has been assigned to flight with assigned helicopters, this reports appears as follows:
All cargo has been assigned to a route.
5. **cargoStatus.out** reports more detail than the cargo.out report including scan times, transfer times, and helicopter names.
6. **conflict.out** reports any helicopter conflicts at a loading zone. A conflict occurs when a helicopter is on the ground at a non-airport heliport and another helicopter approaches to land. Here is an example showing aircraft N003 entering a holding pattern at GBH at 7:37 and landing a minute later because N002 is on the pad when N003 arrives.

C04	N003	enter	GBH	07:37:02	N002 at airport
C04	N003	leaving	GBH	07:38:02	

7. **flightTimes.out** reports helicopter estimated and actual (simulated) departure and arrival times for each flight. For late arriving and departing flights, the report lists discrepancies in minutes between actual and estimated times. Each entry has the planned and simulated (actual) departure time and arrival time. Here is an example portion of the flight times report showing that A45 and A47 are on time as well as the effect of a conflict at GBH for flight A46:

Helo	Flight	Leg	EDT	ETA	ADT	ATA	ADT-EDT	ATA-ETA
N006	A45	NBS-FTY	18:48	18:58	18:48	18:58		
N006	A46	FTY-GBH	19:05	19:11	19:05	19:13		2 MINS
N006	A47	GBH-GAL	19:18	19:25	19:18	19:25		

8. **heliportLandings.out** reports how many landings occurred at each loading zone. Landings are broken down by large helicopter landings and small ones.
9. **inputCheck.out** reports discrepancies in the input route and shipper demand data such as missing data fields, invalid loading zones names, inconsistencies between a cargo's origin and destination and the flight to which it is assigned, and inconsistencies between the shipper entered times and the flight times. See Tables 2 and 3.
10. **lzCaptain.out** reports what activity occurred at each loading zone by describing what is unloaded from, is loaded onto, is transferred on, and is transferred off each helicopter coming to each loading zone. Items reported in the **capacity.out** report are not reflected in this report when they do not get loaded. The report has one line of data for each type of action (i.e., load, unload, transfer on, transfer off) taken for each flight at each loading zone. Each line has the planned and actual (simulated) time of the loader action type. Each line also has the total weight, volume and number of bags for the action as well as the number of bags for each shipper.
11. **missionplan.dat** stores what cargo is to be loaded and delivered on each flight. This file begins with the input file **route.dat** sorted by aircraft and estimated time of departure. For each flight, cargo to be pick up at the origin is preceded with a "P" and cargo to be delivered is preceded with a "D".
12. **noAction.out** reports any flights where no activity has occurred (i.e., no cargo was loaded or unloaded).
13. **shipperPlan.out** reports planned shipper demand (i.e., cargo items and their total volume and weight) for each flight based on the data in the input file **demand.dat**. This report includes all demand data, not only the cargo that would fit based on aircraft capacity. Here is an example portion of the shipper plan report for flight B22 where CRD is overloading the helicopter:

Shipper	Flight	Wt	Vol	Cargo
CRD	B22	255	65.2	CRDBNK3 CRDBNK4
NBK	B22	15	1.1	NBKBNK5

14. **sim.dat** is a listing of each event in the simulation. The report provides the simulation designer a detailed trace of the behavior of the program for debugging purposes. Each line in the report has an event code, the time of the event, and a text message describing the event. The event codes are defined in the simulation for each type of activity that the analyst wants to track. Here are two examples illustrating the start and end of a scan 3 process for cargo CRDBNK86 at PDK:

```

221 11:33:34 chief at PDK starts scan3 Cargo CRDBNK86 off N006 - 5 seconds.
130 11:33:39 Cargo CRDBNK86 scanned from N006 by chief at PDK.

```

15. **transfer.out** reports any planned and actual transfer activity. Cargo can be picked up by one helicopter, transferred to a second helicopter, and then delivered by this second helicopter. Such transfers are noted as achieved or not. Here is an example of the contents of **transfer.out** for a successful transfer operation:

```

The mission plan includes a transfer for cargo MLQSDC33 from N001 to N006 flight A11
Cargo MLQSDC33, which should be delivered on flight A11 of N006, removed from N001 at PDK at 08:56:48
Cargo MLQSDC33 transferred on to flight A11 of N006 at PDK at 09:30:00

```

If the aircraft is not transferred on to the delivery helicopter, the last line would be mission. If it does not get loaded on the first helicopter, the second line would be mission.

16. **waitingTransfer.out** reports any cargo that did not make it to a transfer helicopter because the first helicopter arrived to the loading zone after the second helicopter already left. See Figure 8 and its description.

Appendix C. Glossary

Cargo - The objects that the shippers send on helicopters are called cargo. One cargo item can be composed of one or several bags or boxes. Each cargo item has an associated shipper, assigned cargo identifier (an integer), a three letter shipper designator, a shipper type, an origin, a destination, an estimated departure time, an estimated arrival time, an origin flight, a destination flight, volume, weight, and number of bags. In the simulation, each cargo has an identifier that is a concatenation of the shipper designator, shipper type and an integer identifier (e.g., for NBK's cargo item 56, it is NBKBNK56). See also "flight" and "shipper".

Flight - A flight is one leg of a route. A flight originates at the origin heliport and terminates at the (next) destination heliport. A flight has an associated estimated departure time (EDT) and estimated arrival time (ETA). Associated with each flight is a helicopter. Each flight has a name which is a route letter concatenated with a two digit flight number (e.g., A01 and C50). See also the first two definitions of "route".

Helicopter - A helicopter is an aircraft used to move cargo. In the simulation, there are at most seven helicopters. Two helicopters are large and five are small. The large helicopters can carry 2500 pounds and 228 cubic feet of cargo. The small helicopters can carry 1000 pounds and 61 cubic feet of cargo. In the simulation, seven helicopter aircraft identifiers are N001, N002, N003, N004, N005, N006, and N007. N006 and N007 are large helicopters while the rest are small.

Heliport - A location where a helicopter can take off and land. Some heliports have space for only one helicopter (e.g., the roof of Wachovia Bank). Each heliport has a three letter designator (i.e., ATL, BUC, FTY, GAL, GBH, MIT, NBE, NBS, NOR, PDK, RAF). The airport heliports (i.e., ATL, FTY, and PDK) can accommodate multiple helicopters at one time.

Leg - A leg is one flight from an origin to its destination.

Linear program - Linear programming (as discussed in Winston, 1991) is a tool for solving optimization problems. A linear program has a set of decision variables, an objective function that is a linear function of those decision variables, and a set of constraints that the solution must meet.

Load cart - The cart at the loading zone where loaders put cargo after they take it from the shippers. When a helicopter arrives, the loaders wheel the cargo to the pad in the load cart.

Loading zone - The area at the heliport where loaders service shipper and load and unload helicopters. Each loading zone has a loading zone captain plus other loaders. Each loading zone has a scan gun used to scan the cargo.

Objective function - In operations research, an objective function is a function in terms of the decision variables of a problem that the solution should seek to maximize or minimize (as discussed in Winston, 1991). The objective function for a linear program can either be a maximization problem (e.g., profit or revenue) or a minimization problem (e.g., cost). For a linear program, the contribution to the objective function from each decision variable is proportional to the weight or value of that decision variable and is independent of the values of other decision variables.

Orphan bin - When a heliport closes, helicopters en route to the closed heliport overfly the closed heliport and go to the next heliport in its mission plan. At the overfly destination, cargo destined for the closed heliport is unloaded from the helicopter and put into the orphan bin.

Orphan cargo - Cargo can become "orphan cargo" in two ways:

1. When a heliport closes, all cargo that has not yet been picked up by a helicopter is "orphaned" at its origin.

- When a heliport closes, helicopters en route to the closed heliport overfly the closed heliport and go to the next heliport in its mission plan. At the overfly destination, cargo destined for the closed heliport is unloaded from the helicopter and is thereby "orphaned" at the overfly destination.

Out bin -The logical area at the loading zone where loaders put cargo after they bring the unload cart to the shipper service area and found out that the shipper(s) are not waiting to receive the cargo

Planner - A planner is a member of the team in the planning cell who is responsible for a part of the cargo helicopter planning process. Each planner is responsible for coordinating with three of the twelve shippers. One of the planners is the lead planner. The lead planner is responsible for preparing the final route schedule, loading zone reports, and other planning documentation.

Route - The term route has three meanings.

- One meaning is a set of flights originating and ending at PDK. Such a route is flown by one helicopter. The term "trip" is synonymous with this use of route. Here is an example of such a route:

Flight	Origin	Destination	EDT	ETA	Helicopter
A11	PDK	BUC	9:30	9:35	N006
A12	BUC	GBH	9:42	9:46	N006
A13	GBH	MIT	9:53	9:55	N006
A14	MIT	ATL	10:02	10:08	N006
A15	ATL	NBS	10:15	10:34	N006
A16	NBS	FTY	10:41	10:51	N006
A17	FTY	GAL	10:58	11:03	N006
A18	GAL	NOR	11:10	11:20	N006
A19	NOR	PDK	11:27	11:30	N006

- The term route also refers to the set of all flights beginning with the same letter flown in a day (e.g., the A routes or the B routes). Here is an example of the A routes in this context where the first set of flights from PDK to PDK are assigned to one helicopter and the rest of the flights to another:

Flight	Origin	Destination	EDT	ETA	Helicopter
A02	PDK	MIT	6:53	7:02	N001
A02	PDK	MIT	6:53	7:02	N001
A03	MIT	ATL	7:09	7:15	N001
A04	ATL	NBS	7:22	7:40	N001
A05	NBS	MIT	7:47	8:01	N001
A07	MIT	GAL	8:08	8:16	N001
A08	GAL	NOR	8:23	8:33	N001
A09	NOR	NBE	8:40	8:45	N001
A10	NBE	PDK	8:52	8:56	N001
A11	PDK	BUC	9:30	9:35	N006
A12	BUC	GBH	9:42	9:46	N006
A13	GBH	MIT	9:53	9:55	N006
A14	MIT	ATL	10:02	10:08	N006
A15	ATL	NBS	10:15	10:34	N006
A16	NBS	FTY	10:41	10:51	N006
A17	FTY	GAL	10:58	11:03	N006
A18	GAL	NOR	11:10	11:20	N006
A19	NOR	PDK	11:27	11:30	N006
A21	PDK	BUC	12:15	12:20	N006
A22	BUC	GBH	12:27	12:31	N006
A23	GBH	MIT	12:38	12:40	N006
A24	MIT	ATL	12:47	12:53	N006

Rt	Description	Altitude
		Roswell
5	From Chamblee MARTA station at PDK via tracks SW to Lenox Square at GA 400.	At or below 1500 MSL
6	From Berkeley Hills Country Club via I-85 SW-bound to Loring Heights at I-75.	At or below 1500 MSL within PDK Class D
7	From East Lake where I-20 turns south west of the East Lake Country Club. Via I-20 SE to Covington	
8	From ATL, direct to intersection of Sullivan Rd. and Lee's Mill Rd. Via Sullivan Rd. to GA 85. Via GA 85 south until clear of ATL Class B.	

Shipper - A shipper is a participating courier whose packages are being transported. Twelve shippers participated in the experiment. For the experiment, each shipper was assigned a three letter designator and a three letter shipper type. See Table 18.

Table 18 Participating shippers

Shipper	Shipper designator	Shipper type
Airborne Express	ABX	LDC
Air Courier Dispatch	ACD	LDC
Atlanta Journal Constitution	AJC	CRP
Courier Communications	CRC	SDC
Courier Dispatch	CRD	BNK
DHL	DHL	LDC
Executive Courier	EXC	SDC
Federal Express	FDX	LDC
MLQ	MLQ	SDC
Nations Bank	NBK	BNK
United Parcel Service	UPS	LDC
Wachovia Bank	WAC	BNK

Trip - One set of flights from PDK to PDK. Under normal operations, a trip is always flown by the same assigned helicopter. See also the first definition of route.

Unload cart -The cart at the loading zone where loaders put cargo after they unload it from a helicopter. The loaders wheel the cargo to the shipper service area in the unload cart.

Flight	Origin	Destination	EDT	ETA	Helicopter
A25	ATL	NBS	13:00	13:19	N006
A26	NBS	FTY	13:26	13:36	N006
A27	FTY	GAL	13:43	13:48	N006
A28	GAL	RAF	13:55	14:05	N006
A29	RAF	PDK	14:12	14:22	N006
A31	PDK	BUC	15:00	15:05	N006
A32	BUC	MIT	15:12	15:17	N006
A33	MIT	ATL	15:24	15:30	N006
A34	ATL	NBS	15:37	15:56	N006
A35	NBS	FTY	16:03	16:13	N006
A36	FTY	GBH	16:20	16:26	N006
A37	GBH	GAL	16:33	16:40	N006
A38	GAL	RAF	16:47	16:57	N006
A39	RAF	PDK	17:04	17:13	N006
A41	PDK	BUC	17:45	17:50	N006
A42	BUC	MIT	17:57	18:02	N006
A43	MIT	ATL	18:09	18:15	N006
A44	ATL	NBS	18:22	18:41	N006
A45	NBS	FTY	18:48	18:58	N006
A46	FTY	GBH	19:05	19:11	N006
A47	GBH	GAL	19:18	19:25	N006
A48	GAL	NOR	19:32	19:42	N006
A49	NOR	PDK	19:49	19:52	N006
A51	PDK	BUC	20:35	20:40	N006
A52	BUC	MIT	20:47	20:52	N006
A53	MIT	GBH	20:59	21:01	N006
A54	GBH	ATL	21:08	21:15	N006
A55	ATL	NBS	21:22	21:40	N006
A56	NBS	GBH	21:47	22:02	N006
A57	GBH	FTY	22:09	22:15	N006
A58	FTY	GBH	22:22	22:28	N006
A59	GBH	ATL	22:35	22:41	N006
A60	ATL	PDK	22:48	23:02	N006

3. The third meaning of route is one of the eight routes on the Atlanta helicopter route chart. On the 1996 Summer Olympics helicopter route chart for the Atlanta area, there are eight routes above which helicopters can fly. The FAA defined these routes to follow the highways for noise abatement purposes. During the experiment, the GTRI planners were constrained to design their route fly along these approved routes. Table 17 provides a summary of the approved routes with their altitude restrictions, if any. See also Figure 1.

Table 17 Descriptions of the FAA approved helicopter routes

Rt	Description	Altitude
1	From I-285 and I-20 near Harwell Heights clockwise via I-285 to the intersection of I-20 and I-285 at Panthersville. Direct to Atlanta Beach	At or below 1500 MSL within PDK, FTY, and Dobbins Class D
2	From Wolf Creek Skeet Range, direct to I-20 at the Chattahoochee River east of Six Flags. East via I-20 to East Lake (where I-20 turns south west of East Lake Country Club). Via Memorial Dr. east to Stone Mountain. Via US 78 to I-285 at Clarkston	At or below 1500 MSL within FTY Class D
3	From I-285 and I-75 at Clover via I-75 to Loring Heights. Then I-75/I-85 south to Central intersection north of ATL	At or below 1500 MSL within Dobbins Class D
4	From Roswell south via GA Highway 400 to Lindbergh where it intersects with I-85. Direct to Piedmont Park. Direct to the Georgia Power building.	At or below 1500 MSL within PDK Class D and northbound to