AFIT/GST/OS/ 83M-3

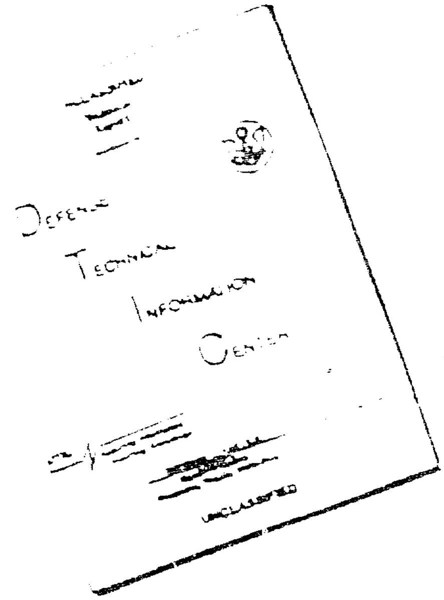
PRIORITY DISPATCH AND AIRCRAFT ALLOCATION: A STUDY OF STRATEGIC AIRLIFT SCHEDULING

THESIS

Vernon H. Hamilton, Maj, USAF
Jerry W. Poe, Maj, USAF

AFIT/GST/OS /83M-3

Approved for public release; distribution unlimited


Defense
Technical
Information
Center
UNCLASSIFIED

AFIT/GST/OS/83M-3

PRIORITY DISPATCH AND AIRCRAFT ALLOCATION: A STUDY OF STRATEGIC AIRLIFT SCHEDULING

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Vernon H. Hamilton
Maj USAF

Jerry W. Poe
Maj USAF

Graduate Strategic and Tactical Sciences

March 1983

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

1

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GST/OS/83M-3	2. GOVT ACCESSION NO. AD-A135730	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PRIORITY DISPATCH AND AIRCRAFT ALLOCATION: A STUDY OF STRATEGIC AIRLIFT SCHEDULING		5. TYPE OF REPORT & PERIOD COVERED MS Thesis
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Vernon H. Hamilton Jerry W. Poe		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT-EN) Wright-Patterson AFB, Ohio 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE March, 1983
		13. NUMBER OF PAGES 185
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

DTIC
ELECTE
DEC 1 2 1983
S D
B

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Approved for public release: IAW AFR 190-17.
LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (ATC)
Wright-Patterson AFB OH 45433

9 SEP 1983

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Airlift
Computer Simulation
Scheduling
Strategic Airlift
Strategic Mobility

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Computer simulation was used to test the effects of strategic airlift scheduling policies on system throughput and cargo tardiness in a "surge" airlift operation. Priority rules from job shop scheduling are adapted and used with alternate aircraft allocation procedures for scheduling C-5 aircraft. A simulation model was written in SLAM (Simulation Language for Alternative Modeling) to test the scheduling policies in a dynamic, pipeline network of airfields. Multiattribute utility theory was used to develop a Scalar Scoring Function(SSF) which combined the response variables into a single value for --

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

each policy to facilitate comparisons among the various scheduling policies. A full factorial experiment was performed--the two factors were the cargo priority rules (five levels) and the aircraft allocation rules (three levels). Thirty replications were accomplished for each policy. Oneway analysis of variance was used to compare the mean SSF values for each policy. The most effective priority rule used ranked cargo requirements by earliest due date. Next in effectiveness was the slack-per-operation rule. The most effective aircraft allocation rules released C-5s for missions with bulk and oversize cargo. Least effective was the rule reserving C-5s for outsize cargo. The report ends with recommendations for further research.

Preface

The intent of the authors in undertaking this thesis project was to apply elements of job shop scheduling theory to strategic airlift scheduling. In particular, several priority dispatching rules from job shop applications were adapted for use in assigning priorities to cargo requirements. These priority rules were combined with three rules for allocation of C-5 aircraft to establish airlift scheduling policies. Computer simulation was used to test the effects of these policies on airlift system throughput and cargo tardiness in a dynamic environment. It is hoped that the results of this study will prove useful to strategic airlift planners.

Among the many factors contributing to the completion of this project, people were the most important. We acknowledge our sincere gratitude to Lt Col Gerald Armstrong, our thesis advisor, whose guidance and cooperation were invaluable. Special thanks to Col Donald Stevens and Maj Joseph Coleman, our readers, for their many helpful suggestions. Maj Charles Dillard, Capt Wayne Stanberry, and Mr Tom Kowalsky of the Operations Research office at the Military Airlift Command headquarters (Hq MAC/XPSR) were constant sources of inspiration throughout. We are also thankful to several AFIT professors and fellow classmates who freely offered their knowledge and opinions.

It would not be possible to overstate our gratitude to our wives and children, whose support and compassion were so important during the busy difficult work of both research and report preparation. Special thanks, again, to Becky Poe (Jerry's wife) for her outstanding assistance in preparing the final report.

Contents

List of Figures

List of Tables

Abstract

The primary emphasis of this thesis was to develop and demonstrate a flexible methodology for examining strategic airlift resource allocation and cargo priority rules. This research was based on a need for heuristics which would increase the cargo tonnage delivered and reduce the amount of cargo delivered late in a contingency operation.

Priority rules derived from job shop scheduling are adapted and used in conjunction with alternate aircraft allocation procedures to develop an improved method for scheduling C-5 aircraft. To test the different scheduling policies in a dynamic situation, a simulation model was developed using SLAM (Simulation Language for Alternative Modeling). Multiattribute utility theory was also utilized to develop a scalar scoring function (SSF) which effectively combined the response variables into a single value for each policy to facilitate comparisons among the various scheduling policies.

A full factorial experiment was performed--the two factors were the cargo priority rules (five levels) and the aircraft allocation rules (three levels). Thirty replications were accomplished for each policy. One-way analysis of variance was used to compare the mean SSF values for each policy.

The most effective priority rule used ranked cargo requirements by earliest due date. Next in effectiveness was the use of slack per operation. The most effective aircraft allocation rules released C-5s for missions with bulk and oversize cargo. Least effective was the rule reserving C-5s for outsize cargo. The report ends with recommendations for further research.

PRIORITY DISPATCH AND AIRCRAFT ALLOCATION: A STUDY OF STRATEGIC AIRLIFT SCHEDULING

I. Introduction

Strategic airlift plays a significant role in current United States (U.S.) national policy.

> "The overall mission of the Military Airlift Command is to maintain, in a constant state of readiness, the military airlift system and other systems and services to perform all tasks the Joint Chiefs of Staff (JCS) and the Secretary of the Air Force assign. In addition to the numerous services MAC performs as an Air Force major command, it is the Department of Defense's (DOD) single manager for airlift and, as a specified command, it is responsible to the Secretary of Defense through the JCS for airlift matters." (Ref 14:7)

> "Through our strategic and tactical military airlift, we can deploy our forces to any part of the world and support them there. Airlift embodies a key facet of a fundamental Air Force capability--rapid, long range mobility." (Ref 1:3)

To accomplish this mission, MAC operates a fleet of both strategic and tactical airlift aircraft. The strategic arm of this airlift fleet includes 70 wide-bodied C-5s whose theoretical maximum capacity is some 100 tons and practical capacity is 50 tons; 234 C-141s whose maximum capacity is approximately 40 tons; 250 Civil Reserve Air Fleet (CRAF) passenger configured aircraft and 123 CRAF cargo configured aircraft (Ref 5). MAC also maintains a network of aerial ports, support bases, and other facilities throughout the world.

In this thesis we will present an intertheatre airlift scenario which requires the efficient and timely delivery of cargo and passengers from the Continental United States (CONUS) to locations in various parts of Southwest Asia in response to a hypothetical contingency plan. Although of limited scope, it will provide realistic methods to allocate strategic airlift resources for this particular scenario.

Headquarters MAC/XPSR sponsored this thesis effort based on their need for a study method to determine how to allocate airlift resources to cargo and passenger requirements during a deployment. The objective of this study method, which includes heuristic algorithms, would be to increase the tonnage delivered and reduce late cargo deliveries, while at the same time attempting to meet scheduled closure dates. Algorithms for allocating aircraft to airlift requirements which meet or attempt to meet this objective are critical to MAC operations; however, very little work has been expended in evaluating this problem. Data and background information for this research effort were provided by Mr. Thomas E. Kowalsky, Chief, Operations Research Division, as well as other members of his staff.

Although Headquarters MAC currently possesses an airlift model (M-14), this model does not provide the flexibility nor the ease of implementation which is necessary for the study of aircraft allocation. The M-14 model is highly complex and requires six to eight weeks reconfiguration time for relatively minor modifications. If a simpler model could be developed which would permit MAC to be able to experiment with various scheduling algorithms before making major changes to M-14, the planning and scheduling processes could be significantly enhanced (Ref 16).

A model was developed for this thesis using the Simulation Language for Alternative Modeling (SLAM) to facilitate studying scheduling effectiveness. Heuristic scheduling algorithms were developed and tested using the SLAM model to determine the most efficient allocation of aircraft resources. The analysis is directed toward providing Headquarters MAC operations planners and aircraft schedulers with a study method for evaluating scheduling algorithms.

Formal Statement of Problem

A determination must be made on how to allocate aircraft to individual cargo requirements in order to increase cargo delivered (throughput), reduce cargo tardiness, and meet closure dates given a particular cargo data set, air base network, and aircraft set.

Aircraft allocation procedures presently used by MAC are based on aircraft cargo preference. These preferences are as follows:

1. C-5s prefer outsize, oversize, bulk and passengers.
2. C-141s prefer oversize, bulk, and passengers.

A search is made of available cargo in order of aircraft preferences, oldest cargo having priority (cargo having the earliest due date). If no preferred cargo is available, the search is reinitiated for the next preferred cargo type. This process continues until all cargo has been scheduled for airlift.

Scenario

A hypothetical contingency plan requires the transportation of cargo and passengers from known locations within CONUS to several locations in Southwest Asia. The types of cargo which will be addressed are:

1. Outsize cargo--cargo which can only be transported on C-5 aircraft.
2. Oversize cargo--cargo which may be transported on either C-5 or C-141 aircraft.
3. Bulk cargo--cargo which may be transported on either C-5 or C-141 aircraft.
4. Passengers may be transported on either C-5 or C-141 aircraft.

Each cargo requirement has a designated pick up point (origin) and a designated delivery point (destination). In addition, each cargo requirement will have time windows associated with it. In other words, each requirement will have an available-to-load-date (ALD) at its origin and a latest arrival date (LAD) at its destination. There may be multiple airlift requirements at a particular base of origin, each of which may be an individual cargo type.

The aircraft to be utilized during the first twelve days of this contingency will be the C-5 and C-141. Although the CRAF may be activated to augment military strategic airlift in supporting contingency operations, it was excluded from our scenario to limit problem complexity. The military aircraft have varying speeds, varying cargo capacities and variable ground times which impact the time cargo spends in the system.

When a cargo airlift requirement is generated at a particular base, aircraft available are allocated to satisfy the airlift requirement. Once tasked for a mission, the aircraft departs its home station, proceeds to the cargo origin base, loads cargo, and departs for Lajes, Azores. The time from the aircraft's departure from its home station to its arrival in international airspace off the U.S. coast is called set-up time. Due to projected cargo loads, and the distances involved in deployment to Southwest Asia, aircraft must make two enroute stops before arrival at cargo destination bases. For the return flights, aircraft must again make two enroute stops at other airfields.

The air base network consists of aircraft home stations, onload bases, offload bases, and enroute stations. Each base possesses a limited amount of ramp space and either one or two runways. The airfields to be utilized for required enroute stops consist of Lajes, Azores; Cairo International Airport, Egypt; Jeddah, Saudi Arabia; and Prestwick, Scotland. (The first two airfields will be utilized for flights enroute to destinations while the latter two will be used for flights returning to the CONUS.)

Assumptions

The above discussion can be summarized by formally stating the following assumptions:

1. Although only the first twelve days of the deployment period are considered, critical measures of performance (closures, cargo tardiness, and tonnage delivered) may be obtained to evaluate the effectiveness of algorithms tested.

2. Each individual cargo requirement at a base of origin will have an ALD of one-to-eleven and an LAD at destination equal to ALD + five.

3. Aircraft allocated to transport cargo always depart from home stations. This implies that aircraft returning to CONUS after mission completion proceed to home stations for post-mission maintenance and preparation for subsequent missions.

4. Distributions of ground time for each type aircraft and airfield are based upon the best information available and accurately reflect ground time for such a contingency. These distributions represent the total time consumed for all activities at a given base which may include time for taxi, maintenance, loading operations, and crew duties. As noted earlier, airfields include cargo onload bases, enroute stations, and offload bases. True ground times for an actual deployment of this type are unknown.

5. The only critical resources in the problem under study are aircraft, ramp space, and runways. This assumption is based primarily on the analyses previously performed by our MAC contacts. Since the

origin and destination groups in the model are aggregates of numerous airfields, it is inferred that the dispersal of aircraft for onloads and offloads will preclude congestion; however, the use of a specific network of enroute airfields would probably cause various degrees of congestion at those particular bases. Although support resources do affect the potential activity level, this relaxation follows the stated objective of evaluating aircraft selection and allocation schemes rather than estimating the system capability.

6. Aircraft are generated for the contingency over a period of 48 hours. This takes into consideration those aircraft away from home station when the mobilization process is initiated and which must return to home stations for deployment mission preparation.

Research Objectives

The objective of this research is to develop and demonstrate a flexible study methodology for examining various algorithms used to allocate strategic airlift resources. This entails the development of heuristic scheduling algorithms and a simulation model to be used as a test vehicle for the algorithms. The contention of this research effort is that the study method developed can assist MAC/XPSR and MAC/DOOF in deployment planning and scheduling processes.

The heuristics tested includes cargo priority dispatch rules, in addition to, aircraft allocation rules based upon cargo available for airlift.

General Methodology

This study consists of three separate phases:

1. Literature search.
2. Algorithm and model development.
3. Analysis.

The first phase, or literature search, delves into the background of the problem and reveals the complexity of aircraft allocation during a military force deployment which relies extensively upon MAC's strategic airlift resources. This phase also includes a review of scheduling theory with emphasis on job shop scheduling and heuristics.

The second phase of this study consists of development of heuristic algorithms and a model to be utilized to test these algorithms. The primary method of studying dynamic job shop phenomena is through simulation models (Ref 11). The heuristics tests are a variety of priority dispatching rules which determined the order in which cargo is scheduled for airlift, and which aircraft is assigned to perform the mission. In line with this, a computer simulation model of the airlift system is defined and developed using SLAM techniques and FORTRAN computer codes. Although many priority rules have been proposed in literature, no analytical formulation has been made which assures optimality (Ref 11). As a result, there is no known optimal standard against which to test the performance of the model. Therefore, substantial reliance is placed upon HQ

MAC/XPSR staff members in validating the model and its underlying assumptions.

The output for each simulation experiment and applicable scheduling algorithm is analyzed in the third phase. From this information, conclusions and recommendations are made in regard to the aircraft allocation problem.

Overview of Thesis

Chapter II of the thesis acquaints the reader with the importance of airlift to deployment operations. It also includes a discussion of organizations which have a valid interest in strategic airlift support for contingencies. Chapter III deals with the scheduling literature research, methodology (algorithm development), and experimental design for this thesis. Chapter IV includes a discussion of the SLAM model, which was used as a test vehicle to evaluate algorithms. In Chapter V, the results and analysis are covered. The conclusions and recommendations of this thesis effort along with ideas for future research are discussed in Chapter VI.

II. Historical Background

Introduction

The history of warfare is filled with examples where a handful of forces, moved to the right place at the right time, swayed the tide of battle. In recent history, airlift has played an increasing role in deployment operations.

During the 1950's a Pan-Arabist movement in the Mideast polarized states in that region into two camps. At that time, the "Eisenhower Doctrine" promised U.S. support would be available if any attempt by pro-communist insurgents was made to overthrow any regime in the area. On July 14, 1958, the pro-west Hashemite monarachy of Iraq was overthrown by pro-Nasser military officers (Ref 26). Lebanon, meanwhile, was experiencing problems with internal strife and requested American assistance to avoid a similar political crisis. The situation caught military strategists and leaders off guard.

President Eisenhower ordered Marine amphibious units of the Sixth Fleet to land at Beirut, but only the 2nd Battallion of 2nd Marines was immediately available. This small force landed 24 hours after notification, and a day later, Marine units from the 3rd Battalion, 6th Marines, landed ashore. On July 18, the 1st Battalion, 8th Marines, landed at Beirut to guard the beachhead and 800 men of the 2nd Battalion, 8th Marines were airlifted in 54 hours from Camp Lejeune to Beirut (Ref 26). The next day, the Army's 24th Airborne Brigade followed. By August, there were nearly 15,000 Marines and Army troops in the area. Although only providing policing action, they injected a measure of calm among the opposing factions within the country.

Another example of the importance of strategic mobility's impact ocurred in the Dominican Republic in 1965. Two thousand paratroopers from the 82nd Airborne were airlifted by C-130s to San Isedro Air Base on April 28 to supplement an already present 1600 Marines. Seven additional battalions were airlifted to the area between May 1 and May 4, 1965 (Ref 26). Although the troops' stated purpose was to protect American lives, their presence effectively stopped further military action and paved the way for resolution of internal problems.

Our Allies also rely upon our airlift resources. French and Belgian troops were airlifted into Shaba Province, Africa, in 1978 to ward off surrogate forces from nearby Angola. In 1980 American resources were utilized in airlifting British Commonwealth troops into Zimbabwe-Rhodesia to supervise elections. During the Yom Kippur War of 1973, the Isreali government requested and received assistance in the form of material and supplies to support their military efforts against the attacking Syrian and Egyptian forces. The bulk of this logistics support was provided by strategic airlift.

The importance of strategic airlift in our national strategy, therefore, cannot be overstressed. Whenver military assistance is required, it is essential that adequate airlift be available to provide rapid deployment of resources before opposing forces can build up their defenses as ocurred during the recent Falkland War. Because British troops were unable to deploy rapidly, Argentina had ample time to build up its defense of the Falkland Islands.

Recognition of the importance of strategic airlift has resulted in improvements being made in regard to our own airlift resources. To help offset a significant shortfall of aircraft, the C-5 is presently undergoing wing modification to extend its service life to 30,000 flying hours, while the C-141 fleet has been modified to permit an additional 30% increase in oversize cargo payloads.

To maximize the use of these limited resources, it is imperative that plans be developed which will result in efficient utilization of aircraft during stategic deployment. The Rapid Deployment Force (RDF) has been tasked with significant responsibilities with regard to non-NATO contingency operations, one of which is Southwest Asia, the scenario selected for this thesis effort.

Rapid Deployment Joint Task Force

The development of the Rapid Deployment Joint Task Force (RDJTF-renamed Central Command, CENTCOM, in January, 1983) places heavy emphasis on the rapid deployment and mobility of forces to areas throughout the world. Deterring overt Soviet aggression in the Persian Gulf region and preserving uninterrupted access to oil from this region was the primary rationale behind the RDJTF formed by the Carter Administration in the wake of Soviet invasion of Afghanistan in December 1979. In his State of the Union Address on January 23, 1980, President Carter stated:

> "Any attempts by any outside force to gain control of the Persian Gulf Region will be regarded as an assault on the vital interests of the United States of America, and such an assault will be repelled by any means necessary, including military force." (Ref 14:9)

The Persian Gulf Region is sometimes referred to as Southwest Asia. Geographically, this area forms a crescent from Pakistan through Turkey and includes Arabia. One should realize that the Persian Gulf is the most demanding contingency for the United States in terms of deployment primarily because of distances (6000 nautical miles from the east coast) involved and the necessity of overflight and landing privileges. (In the 1973 Arab-Israeli Conflict, differing perceptions of that conflict led several countries to deny the U.S. military forces overflight and landing priveleges.)

Since 1979, at which time the Imperial Regime in Iran fell, the U.S. has observed the progressive destabilization in the northern portions of the Middle East--from Turkey to Iran, Afghanistan and Pakistan. This area was once noted as the strategic "Northern Tier" during the Eisenhower-Dulles era. However, due to the recent developments in this area, it no longer prevents the projection of Soviet military power as it did before the political changes in Iran and the Soviet invasion of Afghanistan (Ref 24). As a consequence of these developments, the political structures of the Arabian Peninsula are severely threatened--thus endangering vital sources of oil supplies. Access to Persian Gulf oil is imperative to the western world and its allies since they import large quantities of oil from this region. (The U.S., Western Europe, and Japan import 20%, 67%, and 75% respectively from the area.) Therefore, there is no area more critical to the economic and political survival of the U.S. and its western allies than the Arabian Peninsula (Ref 24).

The RDF relies on a preemptive strategy that, according to the RDJTF commander, permits the U.S. to "get forces into an area rapidly, irrespective of size." (Ref 19:374). Strategic airlift implies the use of those aircraft designed for long range deployment of military forces in response to a national emergency. These aircraft, as previously stated, include 304 military aircraft (C-5s and C-141s), 113 cargo aircraft (Civil Reserve Air Fleet-CRAF), and 250 passenger aircraft (CRAF). Strategic mobility forces move people, equipment, and supplies to wartime locations; provide sustaining support; and allow our forces to respond to unpredictable shifts in the demands of combat. Mobility programs include airlift, sealift, and the prepositioning of equipment and supplies to reduce movement requirements. Airlift and sealift provide flexibility necessary to respond to the unexpected. During a sustained conflict, sealift could carry the bulk of necessary supplies and equipment. Existing sealift, however, cannot provide sufficient rapid response in many scenarios and, therefore, strategic airlift is required.

Because of the distances involved for a contingency in the Persian Gulf and the requirement for quick insertion of forces (preemptive strategy), the U.S. has embarked upon the task of prepostioning certain types of heavy equipment. Presently, 7 cargo ships are anchored at Diego Garcia, an island in the Indian Ocean approximately 2000 miles from the Straits of Hormuz. (The Straits of Hormuz connect the Indian Ocean to the Persian Gulf.) These ships hold much of the equipment which would be utilized by a marine amphibious brigade. Aircraft and other equipment which do not lend themselves to this type storage and most of the necessary personnel would have to be airlifted into the area to marry-up with this prepositioned equipment (Ref 19).

Although initially designed to cope with potential conflicts in Southwest Asia, the RDJTF mission requires planning employment of designated forces, joint training and exercising of these forces, and ultimately deploying and employing them in response to contingencies threatening U.S. interests anywhere in the world (Ref 19). When a conflict arises, the RDJTF will control all the U.S.-based Army, Air Force, Navy, and Marine units it needs. The RDJTF is a central reservoir comprised of units based primarily in the U.S. from which forces can be drawn for a specific contingency. The actual force size and composition would depend upon the crisis and nature of the threat.

Once a particular mission has been assigned to the RDJTF, units appropriate to that mission will also be assigned. It should be noted that with the exception of small, forward-deployed Marine Corps elements, practically all the RDJTF's combat forces, whether Army, Air Force, or Marines, will require transportation from the U.S. (Ref 34). This implies that a majority of equipment and personnel will require strategic airlift to our support the preemptive srategy. The RDJTF has no units permanently assigned to it, but some units have a high likelihood of participating in such a mission. Those units which have been publicly identified are (Ref 26):

ARMY
    82nd Airborne Division
        9 infantry battalions
        3 105 mm artillery batteries
        1 air defense battalion
        1 light armor battalion
        1 air cavalry squadron and
          divisional support units
    101st Airborne Division (Airmobile)
        9 infantry battalions
        1 air cavalry squadron

- 3 105 mm howitzer battalions
- 1 aviation group

24th Infantry Division (Mechanized)

- 6 mechanized battalions
- 3 armored battalions
- 4 artillery battalions
- 1 armored cavalry squadron
- 1 air defense battalion
- 1 aviation battalion

MARINE CORPS

- 3 marine amphibious forces (MAFS)
- 1 marine amphibious brigade (7th MAB)
  - 2-3 infantry battalions
  - 1 tank battalion
  - 1 155 mm howitzer battalion
  - 1 air component
- 1 marine amphibious unit (MAU)

AIR FORCE

SAC - 57th Air Division

MAC - strategic and tactical airlift resources including Civil Reserve Air Fleet (CRAF)

TAC - tactical fighter wing resources

RESERVES

50,000 personnel -- support units (maintenance, transportation, and medical)

To improve our nation's capability to deploy and sustain any of these mobilized combat forces, the Joint Deployment Agency (JDA) was organized in 1979.

Joint Deployment Agency

The JDA's mission, as assigned by the Department of Defense (DOD), is to insure that American forces and their equipment are available to military commanders conducting operations anywhere in the world. Furthermore, its mission includes planning, coordinating, and monitoring deployments/redeployments and movements of mobilized resources necessary to meet military objectives (Ref 22). Recently, an additional decision-making authority during the deployment phase was added by JCS. The JDA will act as the agent of the JCS and supported commanders during deployment execution and sustainment of forces. Based on their guidance, the JDA will adjust movement plans, schedules, and transportation modes, and then direct implementation of these deployment decisions. A harmonious working relationship between JDA and supported/supporting commanders worldwide, the three U.S. military transportation operating agencies (TOAs - Military Airlift Command (MAC), Military Sealift Command (MSC), and Military Traffic Management Command (MTMC)), the Services, Service logistic agencies, and other DOD agencies (Ref 22). Collectively, these organizations are referred to as the joint deployment community--those responsible for planning and executing the movement of forces and materiel from origins to destinations.

The JDA coordinates transportation support provided by the TOAS to move forces, equipment, and supplies. This deployment planning integrates aircraft provided by MAC, ships from MSC, and ground transportation resources from MTMC, to identify actual transportation capabilities for supporting a plan. This helps the supported commander to realize how fast the forces and supplies will be delivered so he can make adjustments during a contingency. During a crisis, the JDA's role is to assist in the JCS decision making process. A Deployment Action Team (DAT) is formed when it is determined that a deployment of forces is required. This team is the focal point for coordinating all deployment actions required by the deployment community. The DAT directs the actions required to modify existing contingency plans or develop a new one for the crisis. Working closely with the TOAs, the team provides JCS with information on how quickly a fighting force will be in place. JCS can then advise the NCA of the options available if the decision to commit military forces is actually made. When a deployment of forces is necessary, the JDA coordinates with commanders providing forces (supporting commanders) and with the TOAs to insure that the operations plan is being supported. When deployment problems arise, JDA makes necessary adjustments to maintain the flow of materiel and personnel by applicable modes of transportation and insure closure profiles (time allocated for cargo delivery) are met. As stated above, MAC, as a TOA, must insure that its allocation of strategic airlift resources complies with the contingency plan.

MAC's Involvement

In 1980, the Secretary of Defense and Air Force Chief of Staff report to Congress stated that the "current airlift deficiency is judged to be the greatest problem the Air Force faces in executing national military strategy" (Ref 29). The magnitude of the problem surrounding transport aircraft has surfaced on several occasions. In July 1980, for example, the U.S. deployed a single squadron of F-4 aircraft to Egypt. Airlift resources required for this deployment totaled 33 aircraft (5 C-141s and 28 C-5s). These aircraft were required to transport approximately 2000 tons (4 million pounds) of cargo and 450 support personnel (112,500 pounds) for the deployment phase alone.

In late November 1980, personnel from the 101st Airborne Division deployed to Egypt in an exercise called Bright Star '81. This was the RDF's first deployment of ground forces to a foreign land and required a total of 90 sorties of C-5s and C-141s in order to transport 625 combat and 775 support personnel into the area (Ref 25).

An even larger Bright Star exercise was conducted in 1982. In this exercise, forces were deployed from several CONUS locations to four Southwest Asian countries. MAC airlifted elements of the Army's 82nd Airborne and 24th Infantry Divisions, as well as Air Force A-10 and F-16 units, to Egypt for a combined U.S./Egyptian field training exercise.

Army combat engineers were transported to Somalia to conduct engineering projects. U.S. and other unconventional warfare units were deployed to Sudan for combined U.S./Sudanese warfare maneuvers. Simultaneously, the Marines and communications support forces were conducting amphibious operations in Oman. This Bright Star exercise involved more airlift aircraft than any exercise since 1960, and it also exercised all four services in joint operations for the first time in Southwest Asia. For the aircraft involved, numerous intermediate support airfields were required to provide necessary fuel, maintenance, and crew staging facilities. The final destination for cargo and people participating in the Egyptian part of the exercise was Cairo West, an austere fighter base approximately 30 miles west of Cairo. Cairo International Airport served as an enroute facility for those missions whose destinations included Wadi Seidna, Sudan; Berbera, Somalia; and Masirah, Oman.

To support this exercise, MAC flew a total of 542 missions (2794 sorties) to airlift a total of 7461 tons of cargo. These missions included 52 C-5 missions, 394 C-141 missions, and 85 C-130 missions (Ref 6). Additionally, CRAF was also exercised. Ten commercial, wide-bodied aircraft assisted in transporting troops to and from Cairo West and another was used to transport paratroopers from the U.S. to Sigonella, Italy. Although exercise employment periods lasted from a few days to a few weeks for rapid deployment forces, MAC aircraft were involved for the entire time period (November 25-December 11). These airlift forces were utilized for deployment, employment, and redeployment exercise phases.

Some problem areas identified and related to airlift resources were as follows (Ref 6):

1. Timely decisions are critical to an orderly airlift flow.
2. Numerous changes to airlift requests resulted in some degradation of the mission.
3. Late approval and denial of overflight and landing rights also caused changes to schedule flow. Instead of a single enroute stop, multiple enroute stops were required which necessitated additional support personnel and equipment.

In Bright Star '82, 98% of the goods and people were delivered prior to or on the desired delivery date (Ref 6).

The details involved in transporting U.S. fighting men all the way to the Persian Gulf are staggering. It is estimated by the Pentagon that "...using all existing air and sealift forces, it would take up to six months to transport 200,000 men to the Persian Gulf" (Ref 25:198). This effort alone would preclude the use of ships and planes for any other lift operation.

Summary

This chapter has presented a brief history of the RDJTF and the JDA, an agency responsible for insuring that American forces and their equipment are available to military commanders conducting operations anywhere in the world. It has also addressed MAC's strategic airlift involvement in recent mobility exercises. Furthermore, it has revealed the magnitude of the strategic airlift problem with regard to deployment distances, airlift requirements (tonnage), and the critical factor of time. With this historical background, it is apparent that planning and scheduling for potential contingencies will continue to play a significant role in supporting national strategic objectives. The authors address the problem of airlift scheduling by developing algorithms to be tested by computer simulation.

The following chapter contains the results of the scheduling literature search, the methodology, and the experimental design for this research effort.

III. Application of Scheduling Theory To Aircraft Allocation

Introduction

The typical job shop has operations/jobs which wait for machines to process them. In the stated problem, the cargo requirements are the operations which must wait to be processed by the airlift resources (aircraft). Because the subject of this research effort dealt directly with allocation of resources, an extensive review of scheduling literature was conducted.

Baker defines scheduling as the allocation of resources over time to perform a collection of tasks (Ref 4). Scheduling is both a decision-making function (process of determining a schedule) and a body of theory (collection of principles, models, techniques, and logical conclusions that provide insight into the scheduling function). The planning function addresses three questions: (1) What product or service is to be provided? (2) On what scale will it be provided? (3) What resources are to be made available? The scheduling function assumes answers already exist. These two functions in many cases are interrelated. Once the tasks and resources available are provided to the scheduler, he formulates a tentative schedule and evaluates it. The schedule may then be returned to the planner, who may alter the tasks to be performed or the number of resources available. This process may continue for some time before a planning decision is reached. The scheduling process, therefore, arises where resources available are fixed as a result of prior planning decisions. Models are available to assist decision makers in a variety of scheduling problems. The Gantt chart, a graphical representation of resource allocation over time, is one of the most simple and widely used. When a model represents reality, it can become an integral part of the scheduling function. Baker also notes that coarse and oversimplified models may be of considerable value in representing the general structure and essential properties of scheduling problems (Ref 4).

Scheduling Theory

Scheduling theory is concerned primarily with mathematical models that relate to the scheduling function. These mathematical models are quantitative approaches which translate decisionmaking goals into objective functions and constraints. An objective function normally contains all system costs impacted by a scheduling decision. Sequencing, the ordering of tasks to be performed (Ref 4), is a specialized scheduling problem in which an ordering of jobs determines a schedule.

Elmaghraby outlines four generally accepted methodologies regarding sequencing theory (Ref 18):

1. Combinatorial approach--this approach changes one permutation to another by switching around jobs which satisfy a given criterion.

2. General mathematical programming--a set of

theories which collectively includes linear, dynamic, quadratic and convex programming, integer programming, networks of flow, and Langrangian methods.

3. Reliable heuristics--called "combinatorial programming" or "controlled enumeration" (Ref 18). On the basis of two principal concepts, problem solving procedures are developed: controlled enumeration concepts are used to consider all potential solutions, and those potential solutions are eliminated which are known to be unacceptable due to bounding, dominance, or feasibility considerations.

4. Monte Carlo--the boundaries of the system under study are defined, and the decision rules which will be followed are specified. This includes any priority rules. The functional behavior of each component of the system is determined along with the mode of interaction among the various components.

The theory of scheduling clearly includes a variety of techniques useful in solutions to scheduling problems. The scheduling field has become a focal point for development, application, and evaluation of combinatorial procedures, simulation techniques, network methods, and heuristic solution approaches. Problem complexity, nature of the model, and criterion choice will determine the actual technique to be employed.

Feasibility constraints are normally found in both scheduling and sequencing problems and include limits on capacity of available resources and restrictions on the order of task accomplishment. A solution to the scheduling problem is any feasible resolution of these two types of constraints (Ref 4). Therefore, to solve a particular scheduling problem one must answer two questions: (1) Which resources will be allocated to perform each task? (2) When will each task be performed? Hence, scheduling is concerned with both allocation and sequencing.

Two areas of scheduling theory surfaced during the literature review which merited consideration for solving the aircraft allocation problem: vehicle routing and job shop scheduling.

Vehicle Routing

Since the stated problem focuses on the allocation of aircraft (vehicles) to cargo airlift requirements (customers), it was determined that vehicle routing and scheduling should be investigated. In the scenario, aircraft were to proceed from their home bases to pick up cargo at a cargo onload base and transport it to a designated offload base via a specified route structure. The vehicle routing problem (VRP) is a constrained version of the travelling salesman problem (TSP) (Ref 3). In the VRP, the primary consideration is vehicle capacity although other type constraints may be considered. The TSP seeks to find a minimum cost path that, from an initial point, visits each city or stop once and only once, ending at the initial stop.

A class of sequencing or routing problems where the key facet of the routing is that a pickup must precede the corresponding, subsequent delivery is represented by the pickup and delivery problem (PUDP). Other constraints are possible and depend upon the particular application. These may include service time windows, service quality parameters or operational constraints on either the driver or the vehicle. Therefore, the PUDP is also a constrained version of the TSP.

The PUDP represents many practical routing situations and includes both dial-a-ride service (DARS) and courier type services. The DARS is characteristic of many public service organizations today which provide transportation to customers for various purposes. DARS vehicles must pick up people at their individual origins and transport them to other locations (destinations). The primary objectives of such problems is to satisfy all customer service requests in the most economical manner.

As noted earlier, the route stucture for the stated contingency operation was essentially fixed. Aircraft were not required to proceed to more than one cargo onload base (i.e., service more than one customer). A minimum cost route was, therefore, not appropriate for this research problem. The authors, therefore, devoted their attention to job shop scheduling, an area believed to hold more promise in resolving the aircraft allocation issue.

Job Shop Scheduling

Most early research in the scheduling field involved manufacturing, therefore, this vocabulary is normally employed when describing scheduling problems. Resources are normally called machines, whereas tasks are called jobs. If jobs comprise several subtasks, these subtasks are known as operations. These operations are normally interrelated by precedence restrictions.

Job shop scheduling deals with resources, capacity constraints, and dynamic job arrivals--characteristics of the problem being addressed with regard to allocation of airlift resources. A job shop is the set of all the machines that are identified with a particular set of operations; a job shop process consists of the machines, the jobs, and the operations, and a statement of the disciplines that restrict the manner in which operations can be assigned to specific points on the time scale of the appropriate machine. "A machine in this process is intuitively a device or facility capable of performing whatever it is that has to be done in an operation, but abstractly, a machine is just a time scale with certain intervals available" (Ref 8).

The majority of published job shop articles reviewed were concerned with the effects of scheduling and sequencing (dispatching) on various measures of shop performance criteria. Usually, these effects are studied in a given context, a context which can be described by making the appropriate choice from each of the following three classifications (Ref 11):

1. Number of component parts (operations) comprising a job.
    a. Single-component jobs

   b. Multiple-component jobs

2. Production factors possessed by the shop.
   - a. Machines
   - b. Labor and machines
3. Jobs available for processing
   - a. N jobs to be scheduled, or sequenced where N is finite. As previously noted, this is the static sequencing or scheduling problem.
   - b. An undetermined (infinite) number of jobs arrive continuously, but randomly, at the shop for service (dynamic sequencing).

In Figure 1, a schema for classifying sequencing problems is shown (Ref 11:12). In this figure, one can see that sequencing problems are classified by: (1) the nature of job arrivals (fixed batch size or continuous arrivals given by a probability density function); (2) the number of machines involved (single stage, M=1; or multistage production, with M>1; and (3) the nature of the job route. In the closed job shop, each job must have one of a number of specified routings, representing a fixed line of products. The open job shop, on the other hand, accommodates practically any possible machine routing. An example of the latter is custom-ordered products. An aggregrate description of the machine routings is normally contained in a routing matrix.

In the 1960's, a large number of simulation experiments suggested advantages to be gained by considering the job shop as a network of waiting lines with fixed short run capacity. The recognition that the job shop could be represented as a

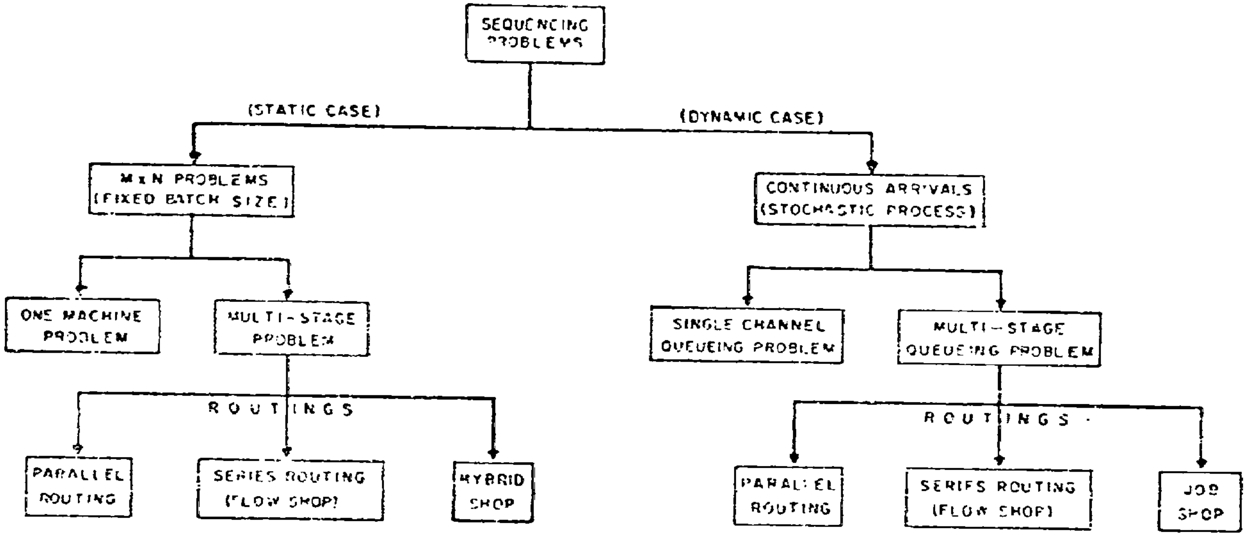


Figure 1 Schema For Scheduling

system or network of queues was an important one and research in this area is continuing. An important variable which can be manipulated in sequencing models is the priority dispatching rule (queue discipline), and much of the research has focused upon comparative studies of dispatching rules through computer simulation.

In a shop scheduling problem, a job is released to a shop with an associated due date (desired completion time), in order that it be shipped to the customer on time. The release time or arrival time is the time at which the job is released to the shop by some external job-generation process. If a job is not completed by its due date, a penalty (cost, customer dissatisfaction, etc.) is incurred. A job which is not completed on its due date is considered late. Lateness may be negative or positive and is obtained by subtracting the due date from the actual completion date. Tardiness of a job is defined as positive lateness (a job is completed after its due date). The aim is to minimize the penalty through good scheduling procedures.

The output of a shop is the number of jobs completed. A single job may involve work on only one productive facility (machine) or on many machines. The job generally pertains to the work that must be performed and the physical entity(s) that are objects of the work. Generally, a job consists of one or more operations and each operation is performed on a machine.

A job file is a listing of operation times and due dates for jobs. For a static scheduling problem, the entire job file is given, however, for the dynamic problem there is an initial set of jobs and more jobs appear from time to time.

To lay out a schedule, a rule or set of rules is required. Because the scheduling problem is one of determining precedence or ordering in time among a number of jobs, an operational scheduling rule includes a means of determining priorities of jobs. The following definitions are provided:

> 1. Priority rule - that rule which assigns to each waiting job a scalar value, the minimum of which, among the jobs waiting at a machine, determine the job to be selected over all others for scheduling.
>
> 2. Scheduling rule - dictates which job among those waiting for service is to be scheduled in preference to the others (scheduling a job means scheduling the next operation of a job).

Good priority rules should be utilized but tailored to the problem at hand. If, for example, a job slack rule is being used but by following the rule it will cause a job of lesser priority to be late, then the rule should be broken. Thus, one or more priority rules are combined with certain heuristics. Heuristics which seem reasonable and worthy of effort should be tested. Heuristics should answer the following question: If the schedule isn't good enough, or if we believe it can be improved, what should be done?

A scheduling rule may include one or more heuristics (rules of thumb) in addition to or instead of a priority rule.

Heuristics as used throughout the remainder of this research effort implies the use of practical experience, observation, and logic, to develop priority dispatch rules to replace exact mathematical relationships. The scheduling rule is a function of the job file when the job selected by the rule is dependent upon number of operations, operation times or due dates.

The effectiveness of a scheduling rule may be measured in a number of ways. Theoretically, these measures of performance for effectiveness have been the average or the maximum of the values of completion-time, flow-time, lateness, or tardiness. These are all examples of regular measures of performance. A regular measure is a value to be minimized that can be expressed as a function of the job completion-times, and which increases only if one or more completion-times increase.

Job priorities may be set before scheduling begins and either remain unchanged or vary in some way independent of the schedule and of the job file (static). An effective priority rule should be dynamic and reflect the status of jobs from time to time as the schedule progresses. Examples of static priority rules include priority equals due date and priority equals reciprocal of operations (a job comprises one or more tasks or operations). In the static case, priority is established before scheduling commences and the order remains unchanged. According to Gere, if a static priority rule appears to be good, then there should be a better rule which is a dynamic analogue of that static rule (Ref 20). The dynamic analog of the due date rule is job slack. For the static case of due date, priority is equal to due date minus starting time. For the dynamic case, priority equals due date minus present time minus processing time remaining. One normally seeks the best priority rule, the one which will be most effective most of the time, regardless of the associated job file.

As noted earlier, Monte Carlo simulation has been the principle tool of analysis for the dynamic job shop because of the stochastic nature of the parameters. Several factors appear common among these simulations, including shop load parameters and operational characteristics of the system. Shop load parameters include: (1) the mean arrival rate of jobs in the shop, (2) the mean processing rates at the various machines or machine centers, and (3) the number of machines or machine centers (including the number of machines in each center) in the shop (Ref 11). Operational characteristics of the system include: (1) the distribution of arrival of jobs in the shop, (2) the processing time distributions at the individual machines or machine centers, and (3) a procedure for generating job routings (Ref 11). Queue discipline, the sequencing policy for jobs in the queue, is another common factor.

Past Research on Priority Rules

Job shop literature is filled with a variety of proposed priority rules but analytical formulation doe not guarantee optimality. Rowe states that even small scheduling problems are very difficult to formulate and solve in terms of all the variables. Furthermore, he said complete enumeration of large-size scheduling problems is virtually impossible. An example of this is a job shop with ten machines and ten jobs. For this problem, there would be $(10!)^{10}$ possible job permutations which makes enumeration prohibitive (Ref 31). However, priority rules have been found, in general, to accelerate jobs through the shop and also affect aggregate measures of shop performance. Priority rules may be classified into four categories:

1. Lateness rules - priority determined according to some increasing function of lateness.

2. Arrival order rules - priority assigned based on the order in which jobs arrive at the machine under consideration.

3. Job based rules - priority based on some property of the job.

4. Random rule - priority of jobs assigned randomly.

The actual effectiveness of a given priority rule is determined by how it meets predetermined criteria. Day states that much of the work done on priority rules in relation to job shops has been with respect to the effect of the rules on the mean and variance of distributions of the following measures of shop performance (Ref 11):

1. Job lateness with respect to due dates.
2. Percent of jobs late with respect to due dates.
3. Work-in-process inventory.

4. Order flow time.

5. Waiting time at each machine or for the shop as a whole.

6. Number of orders in the shop.

7. Machine utilization.

8. Setup time.

Gere conducted a study of eight priority dispatching rules and eight heuristics in 1966 (Ref 20). The sixteen dynamic problems involved 20 to 60 jobs, 1 to 16 operations, and 4 to 16 machines. The priority rules that he tested are listed below:

1. Job slack--the number of free hours available before the due date. If a job is necessarily going to be late, the slack is negative.

2. Job slack per operation--the number of free hours available before the due date divided by the number of operations remaining. If a job is necessarily going to be late, the rating is the (negative) amount of slack.

3. Job slack ratio--the number of free hours available before the due date divided by the number of hours remaining until the due date. When a job is necessarily going to be late, the rating is the (negative) amount of slack.

4. Modified job slack ratio--the number of free hours available before the due date, after operation times have been inflated to include expected delay times, divided by the number of hours remaining until the due date. If a job is necessarily going to be late, the rating is the (negative) amount of modified slack.

5. Shortest imminent operation--length in hours of the next operation of the job.

6. Shortest imminent operation-job slack ratio--the job slack ratio is calculated for each job and a check is made to see whether any job is necessarily

late (has negative slack); if so, the priority rating of each job equals job slack; if not, then the priority rating of each job whose slack ratio is no greater than twice that of the tightest job (job with minimum slack ratio) is equal to the length in hours of its next operation; the rating of any other job is irrelevant.

7. First-come, first-served (FCFS)--the first arrival in the queue of the machine receives the highest priority: priority values are assigned to jobs in a decreasing sequence as they arrive in a queue.

8. Random--select the job at random.

Gere supplemented these priority dispatch rules with heuristic algorithms. While discussing eight potential algorithms, only the following three were tested:

1. Alternate operation--Schedule the operation according to the priority rule, but check to see if this makes another job critical (slack has become negative or, if positive, has reached a critical level). If so, revoke the last operation, and schedule the next operation on the critical job. Check again for lateness. If scheduling the second job does not make another job critical, then schedule it, otherwise, schedule the first one dictated by the rule.

2. Look ahead--if there is a critical job due to reach a machine at some future hour, yet before the scheduled operation is completed, then schedule the critical job instead. Check the effect on other jobs and then make the final decision.

3. Re-do with adjusted due dates--when a schedule is completed and at least one job is late, decrease the due date of each late job (for priority calculations) by the hours the job was late; then lay out the schedule again. For the second schedule, the rule will tighten up the jobs previously late, hopefully just enough to complete them on time in the revised schedule.

Those heuristics not included in his evaluation were:

1. Insert--when a look ahead job has been scheduled, there is normally a period of idle time on the machine. This rule says if there is an operation which will fit into this idle time period, it should be scheduled. Gere questions whether this type of situation occurs often enough to require its use.

2. Time-transcending schedule--determine the priority rating for each job, and schedule the next operation of the job with the top priority. Priorities are reevaluated and the above process repeated with the most critical job being scheduled. Gere concluded that this rule cannot anticipate future conflicts except by iteration, and a time-progression program with a look-ahead feature is as effective as a time-transcending program. In addition, time-transcending programs take longer to program.

3. Subset of critical jobs--requires the selection of a subset of critical jobs, with these jobs scheduled according to some priority rule(s), and the remaining jobs scheduled around these. This rule may force a non-critical job to wait for extended periods becoming first critical and then late. Gere concluded this heuristic provided no additional features over the time-progression schedule with a look-ahead routine.

4. Flexibility--using a Gantt chart, an operation is discovered which will not quite fit between two other operations on a given machine which means this operation will experience a lengthy delay. If this operation could be squeezed into the open time period, this job would not experience such a delay, although subsequent jobs would encounter slight delays. Gere determined there is no need for flexible scheduling if time-progression scheduling is employed without subset scheduling.

5. Manipulation--this requires the utilization of a Gantt chart also. This heuristic is also called the "jig saw puzzle" approach (Ref 20). This heuristic involves trial and error scheduling as follows: if a gap on a machine exists, try to move a late job into it; if a string of operations exist on a particular machine, try to move up the first operation to loosen up the others; if there is a long wait before a job can get on a machine, try to get that job or another on the machine earlier.

The conclusions drawn by Gere as a result of this study were: (1) the selection of a priority rule for discriminating between jobs competing for time on the machines is not as important as the selection of a set of heuristics which bolster the rule. (2) Since there is little difference in selection of priority rules after they are combined with two or more heuristics, a simple rule should be used. Two simple rules he recommended were job slack and shortest imminent operation rules. (3) Those heuristics that anticipate future progress of a schedule, improve schedules significantly in both a statistical and practical sense. The job slack per operation rule performed better than any other priority rule (with and without heuristics).

In an extensive computer simulation experiment in 1965, Conway tested the performance of 39 priority rules. The shortest operation time (SOT or SIOT for Gere) proved to be the least sensitive of the priority rules studied. It should be noted that this rule does not even consider due date but gave better overall performance. Of the due date priority rules, slack per operation was the best (Ref 9). One of the biggest objections to the use of the SOT rule, however, is that while the mean time which jobs spend in the system is minimized, individual jobs (those with long operations) will be intolerably delayed with an increase in lateness variance (Ref 11). Since the variance of the lateness distribution is the basic disadvantage of the SOT rule, Conway illustrates three variance reduction methods: (1) Alternate the SOT rule with a low variance rule with respect to flow time to clean out the shop. (2) Forcibly truncate the SOT rule by imposing a limit on the delay that individual jobs will tolerate. (3) Divide jobs into two classes, preferred and regular. When the next job in the queue of the machine in question is to be chosen, the preferred job with the shortest operation time is selected. If there are no preferred jobs in this queue, then the regular job with the shortest processing time is selected (Ref 10).

The COVERT rule was another attempt to find a rule which retained the SOT performance but which tended to minimize the extreme completion delays of a few orders (Ref 4). The COVERT rule establishes, for each job in the queue, a ratio whose numerator is the delay cost rate for a particular job (c) and whose denominator is the processing time of that job on the machine in question (t). The priority assigned to a waiting operation is the ratio c over t (c/t) and the dispatching rule selects the operation with the largest ratio for earliest processing. Critical jobs are those without slack which implies even if remaining operations were completed without delay, the job could not be completed early. Such jobs were assigned a delay cost of c=1. For those noncritical jobs in which the length of time until the due date exceeds the amount of waiting time anticipated plus processing time, c=0. Jobs between these two extremes possess positive slack, but it is insufficient to meet the waiting time requirement that is anticipated (Ref 4) and a formula for computing c must be used. If one takes c=1 and t=2 the ratio c/t becomes 0.5 which becomes this operation's priority. "The total delay cost is minimized if the task with the highest ratio is dispatched, since, in this process, one maximizes the tradeoff of the resource being allocated (processing time) against the potential gain (delay avoided)"(Ref 33:66). Results obtained by Carroll showed the mean tardiness of the COVERT rule to be superior to that of the truncated SOT rule, to be discussed in the following paragraph.

According to Baker, SOT and LWKR (Least Work Remaining) are static with respect to a particular operation, but dynamic with respect to a particular job in the sense that individual operations of the same job acquire different relative priorities. This applies to other rules also, which are defined below (Ref 4):

1. FCFS--highest priority given to the waiting operation that arrived at the queue first.

2. MST--highest priority given to the operation with minimum slack time (previously defined by Gere as job slack).

3. OPNDD--highest priority is given to the waiting operation with the earliest operation due date. An operation due date is determined by dividing the interval between the job due date and its shop arrival time into as many subintervals as there are operations. The end of each subinterval represents a due date for the corresponding operation. (A dynamic version of this rule occurs if the shop arrival time is replaced by the current dispatching time.)

4. LWKR--highest priority given to the waiting operation associated with the job having the least amount of total processing remaining to be done.

Baker also defines TSPT (Truncated SPT or Truncated SOT) to mean that the highest priority is given to the waiting operation with the shortest imminent operation time (as under SPT) except when an operation in the queue has waited in this queue more than W time units. Operations with queue times larger than W are given overriding priority and are dispatched under FCFS (Ref 4).

This scheduling review provided us with the necessary information from which a solution strategy for the stated problem could be developed.

Methodology

The most critical area to the development of the methodology was the selection of measures of performance. Although these were mentioned in Chapter I, they are restated at this time:

1. Maximize cargo requirements meeting closure dates (those requirements arriving at destinations prior to or on the latest arrival date)--percentage.

2. Minimize amount of tardy cargo-- measured in ton-days.

3. Maximize tonnage delivered-- measured in tons for the twelve day period.

The selection of these three measures was made after interviews with HQ MAC/XPSR staff members and the authors' interpretation and evaluation of the problem's scope. They are listed in order of importance to a theatre commander, who desires that unit requirements arrive at destinations in close proximity to one another and within appropriate time windows.

As mentioned earlier in this chapter, the planning and scheduling functions are interrelated. The planning function (HQ MAC/XP) has addressed the service to be provided, the scale for which this service was to be provided, and the resources to be made available. The service provided is the expeditious airlift of cargo and passengers from their origins in the U.S. to destinations in Southwest Asia. This service requirement was in response to the hypothetical contingency plan described in Chapter I (Scenario). The scale of the problem was also specified in the Scenario. Four types of cargo were to be transported by airlift resources from the CONUS to Southwest Asia and these cargo requirements were to be delivered within certain time limits. The route structure was also specified. The resources made available were the two types of aircraft (C-5s and C-141s), and each type aircraft possessed capacity limits.

With the planning function completed, the scheduling function was initiated. As stated before, this function addresses efficient utilization of resources, meeting customer demands (cargo airlift requirements having specified time windows for pick up and delivery), and meeting due dates. The allocation of aircraft to transport cargo requirements may be compared to a job shop where jobs are assigned to machines in some prescribed order.

The typical job shop has jobs consisting of one or more operations which wait for machines to process them. In this problem the machines were the two types of aircraft, each having a cargo capacity constraint. Cargo requirements were equated to operations which were required to wait in queues for aircraft resources. Operations consume time on machines, whereas for aircraft, individual cargo requirements consume a percentage of available aircraft capacity. As noted earlier, machine time was the primary constraint in the job shop. For aircraft, the primary constraint is capacity measured in tons. Since cargo requirements arrived at their origins over time, it was determined that a dynamic sequencing problem existed. The problem was also defined to be a multistage production type where M>1 (M was number of aircraft).

The assumptions for this dynamic sequencing problem were then developed and are listed below:

1. Cargo requirements enter the system incrementally over an eleven day period.

2. The time required for delivery of cargo was a function of the priority dispatch rule utilized in scheduling, and the type aircraft allocated to transport the cargo. Although flight times for routes were specified, the ground service times were stochastic.

3. Once a particular cargo requirement was scheduled for transport and loaded aboard an aircraft, it was not preempted (i.e., removed and replaced by another cargo requirement).

4. Due dates were fixed.

5. There were two groups of similar machines, but with different capacity constraints.

6. Aircraft were not allocated to replace other aircraft which experienced maintenance difficulties during a scheduled mission. Once cargo was loaded on an aircraft, it remained on that aircraft until delivery occurred. Repair time was included in ground service distribution times.

7. No subcontracting was permitted (i.e., fixed shop capacity).

Having specified these assumptions, the aircraft allocation rules and cargo priority dispatch rules are provided.

Current HQ MAC scheduling policy dictates that C-5 aircraft only be allocated (scheduled for a mission) if outsize cargo requires airlift. There must be outsize cargo requiring transport or the mission is not scheduled. Since the purpose of this research effort was to identify improved allocation policies, a determination was made to test two other allocation policies for the C-5. The first of these involved scheduling C-5s if either outsize or oversize cargo were available. The second required the allocation of C-5s if outsize, oversize, or bulk cargo were available. Therefore, three aircraft allocation rules were tested along with cargo dispatch rules.

The cargo priority dispatch rules selected for testing were as follows:

1. Aircraft preference--cargo requirements were grouped by cargo type (outsize, oversize, bulk or passengers) and dispatched according to aircraft preference. Entries in each of these groups were ranked by EDD.

2. Earliest due date (EDD)--cargo requirements ranked by due dates in non-decreasing order.

3. Smallest weight (SWT)--cargo requirements in the job file were ranked according to weight in non-decreasing order.

4. Largest weight (LWT)--cargo requirements in the job file were ranked according to weight in non-increasing order.

5. Slack per operation--the number of days available before the due date divided by the number of operations.

The aircraft preference dispatch rule was selected as the base case since this is present HQ MAC policy. Recall, C-5 aircraft prefer cargo in the following order: outsize, oversize, bulk, and passengers. C-141 aircraft prefer cargo as follows: oversize, bulk, and passengers. The reader should also remember that the C-141 cannot transport outsize cargo. For the particular aircraft allocation rule described above, cargo requirements were scheduled for airlift based upon aircraft preference. For example, assume an outsize cargo requirement existed at an origin base. The scheduler would allocate a C-5 to transport this requirement and would search the outsize group (file) for other outsize cargo with the same origin/destination pair provided the aircraft's cargo capacity had not been exceeded. Assuming the first outsize cargo assignment did not fill the aircraft's capacity and other outsize cargo with the same origin/destination was not available, the scheduler would first search the oversize file for filler cargo with the same origin/destionation pair. If such oversize cargo were not available, the scheduler would search the remaining two cargo preference files (in order) in an attempt to fill the aircraft. This rule also applies to the C-141, however, recall the C-141 may be allocated for any cargo requirement except outsize and the search for filler cargo, if required, would begin in the oversize file. Within each of these cargo type files, cargo requirements are listed according to EDD. For the remaining four dispatch priority rules, a single job file was established. Because of the transformation utilized in the interpretation of the job shop for the stated problem, only certain priority rules appeared reasonable for evaluation. The second cargo priority selected was EDD.

The EDD appeared to be one of the most practical rules for testing. Cargo requirements were ordered according to due date and scheduled for airlift accordingly. The scheduler attempts to fill the aircraft within weight limitations as previously discussed. The next priority rule selected was the SWT (smallest weight) rule.

The SWT rule was selected because of its similarity to the LWKR (least work remaining) rule. Recall the LWKR rule ranks operations associated with jobs according to the least amount of total processing remaining to be done. The scheduler attempts to fill the aircraft as previously discussed but with smallest weight cargo first. The fourth rule selected was LWT (largest weight).

It was determined that the LWT rule should be tested in anticipation that it would increase the tonnage delivered. In addition, a decision was made to compare this rule against the SWT rule. The final dispatch rule chosen was the slack per operation rule.

Both Gere and Conway discovered the slack per operation rule performed better than other dynamic priority rules. Therefore, a decision was made to include this priority rule in the test. For this particular rule, the slack (due date minus today) was computed for each cargo requirement in the job file. A new variable called NOPS (number of operations) was calculated. NOPS was computed to be the cargo requirement quantity divided by the aircraft capacity and rounded to the next highest integer value. The slack per NOPS was then calculated to obtain a particular priority. For example, if the cargo requirement quantity equalled 20 tons and the aircraft capacity was 60 tons, NOPS = 1. If one assumes the slack equals 4, the cargo requirements priority in the job file equals a value of 4. Similarly, each of the other cargo requirement priorities were calculated. Combinations of aircraft allocation rules along with cargo priority rules were tested using the SLAM model (Chapter IV).

Experimental Design

The design of this experiment required the identification of the following response variables:

1. Closures - those cargo requirements delivered on or before their due dates.

2. Tardiness - the number of tons delivered late times the number of days tardy.

3. Tonnage delivered - the actual amount of cargo delivered in tons over the twelve day period.

Because three response variables were involved, a method to combine these variables into a single value was needed. The authors chose to utilize a multiple attribute utility theory (MAUT) technique to accomplish this task. "MAUT is a decision making theory which requires the analyst to elicit preference information concerning attributes (response variables) of proposed alternative policies for the decision maker (DM)" (Ref 13:10). Using the DM's preferences, a scalar scoring function can be developed by the analyst. A policy alternative has associated with it a set of events and outcomes. Using the SSF in conjunction with the corresponding outcomes, each alternative can be scored and the set of alternative policies ranked for the decision making step. A MAUT technique was selected to obtain the SSF and then rank the alternative scheduling policies.

This technique, called simple multiattribute rating technique (SMART), is described by Edwards (Ref 17). The basic idea of multiattribute utility measurement implies that every outcome of an action may have value on a number of different attributes. The values, therefore, must be determined for each attribute. These values are then aggregated across the attributes using a suitable aggregation rule and weighting procedure. SMART uses the simplest aggregation rule and weighting procedure which consists of simply taking a weighted linear average. This technique includes ten steps:

1. Identify the key players in the decision making process. For the stated scenario, HQ MAC/XP performs the planning function, while the theatre commander is the decision maker whose values in the scaler scoring function are to be maximized.

2. Identify the decision(s) to which the values needed are relevant. This step was described in the problem statement of Chapter I as the need for allocating aircraft to unit cargo requirements in order to increase cargo deliveries, reduce cargo tardiness, and meet closure deadlines.

3. Identify the alternatives to be evaluated. These are represented by the fifteen possible scheduling policies to be discussed in a later chapter.

4. Identify the relevant attributes whose values are to be evaluated for the alternative policies. These were previously mentioned and include: closure, tardiness, and cargo delivered.

5. The attributes are ranked in order of importance. Operations and transportation planners at HQ MAC rated these attributes in the following order based upon their interpretation of a theatre commander's desires during an actual deployment: closures, tardiness, and deliveries (Ref 15). These attributes are listed in descending order of importance.

The remaining steps (6-10) are discussed below in greater detail:

6. The attributes are rated in importance, preserving ratios. The DM (HQ MAC/XP, representing the theatre commander) was asked to assign a weight to the least important attribute on a scale of [0,1]. The DM specified a weight of 0.2 for cargo deliveries. The remaining attributes were then compared to this attribute's weight by the DM. The DM stated that tardiness was three times as important as deliveries, and closures four times as

important. Therefore, on a scale of [0,1] the weights of 0.8 (closures), 0.6 (tardiness), and 0.2 (deliveries) were obtained.

7. The importance weights from Step 6 are summed and individual weights are divided by the sum to obtain normalized weights for the SSF. The following relationships then exist:

$$\text{Sum of all weights} = 0.2 + 0.6 + 0.8 = 1.6$$

$$\text{WT1} = \text{normalized closure weight} = \frac{0.8}{1.6} = 0.5$$

$$\text{WT2} = \text{normalized tardiness weight} = \frac{0.6}{1.6} = 0.375$$

$$\text{WT3} = \text{normalized delivery weight} = \frac{0.2}{1.6} = 0.125$$

8. In consultation with the DM a value for each attribute is obtained. The plots obtained were based on a linear value function and are shown in Figure 2. The x-axis represents the plausible range of values for each of the attributes; while the y-axis represents the value [0,1] for each level of the plausible range. The values are denoted by the variables C, T, and D respectively. The plausible range for closures extends from zero to a maximum value of 170, which represents total unit requirements. For tardiness, the range extends from 150,237 ton-days to zero. The cargo delivery range was zero to 37,600.

Example: Assume only 0.5 of unit closure requirements are met (i.e., 85 of 170). Then the value (C) for this outcome equals 0.5.

$$C = \frac{\text{Total unit closures}}{\text{Total unit requirements}} = \frac{85}{170} = 0.5$$

Example: There are five days of deployment requiring 20 tons of cargo to be transported each day. Each 20 ton airlift requirement has a due date of five days after its available to load date. All cargo, therefore, must be delivered not later than the tenth day. If one assumes none of these requirements are delivered, the total ton-days tardy would be computed by using day eleven in the computation. For example, Day1 cargo has a suspense date for delivery of Day6. The suspense date is subtracted from Day11 (i.e., 11-6 = 5). This number of days tardy is multiplied by the 20 ton cargo requirement to obtain the value of 100

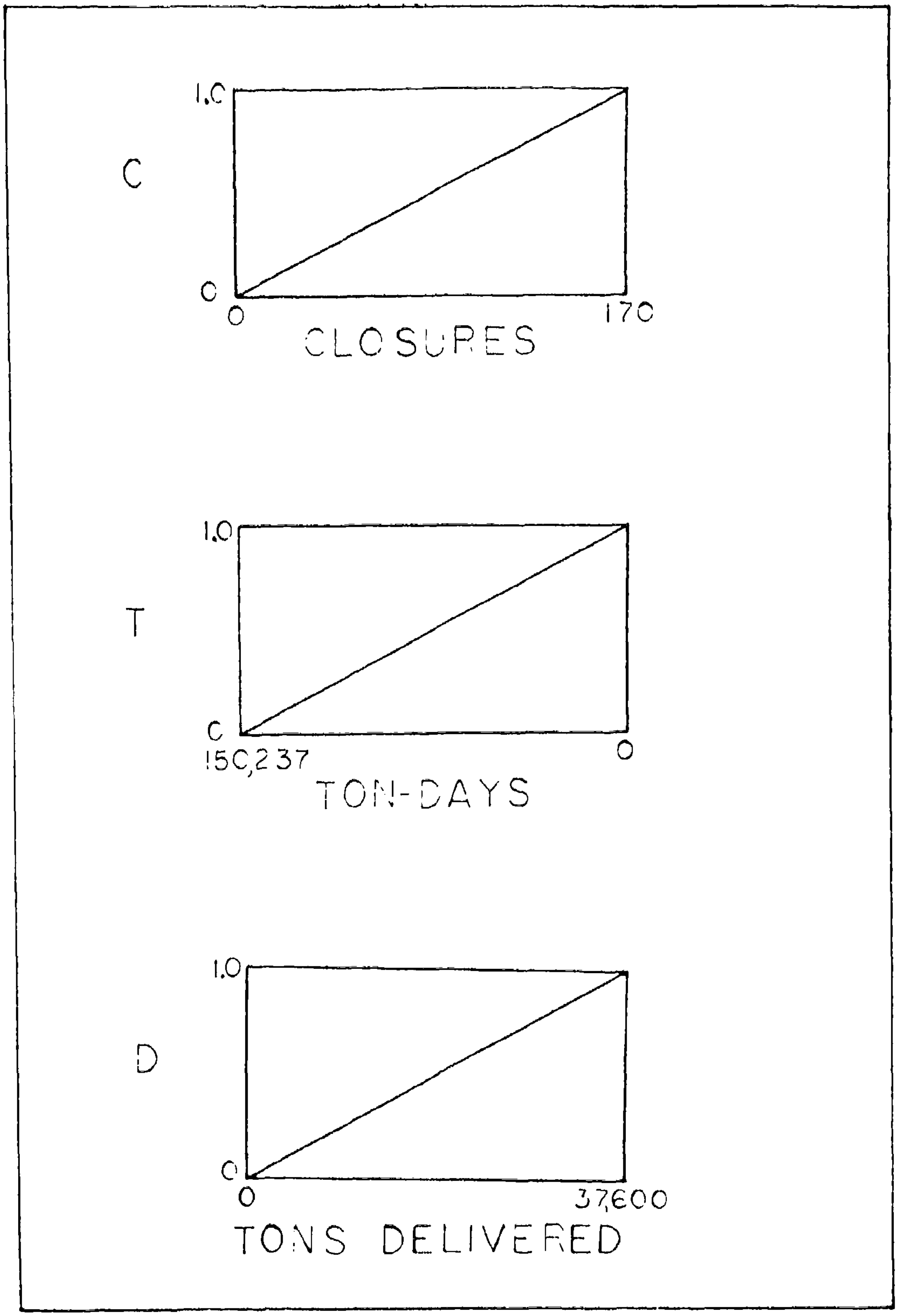


Figure 2 Attribute Value Graphs

ton-days. Similarly, ton-days tardy may be computed for the remaining days of deployment.

Day1 (ton-days tardy) = 100
Day2 (" ") = (11 - 7)(20) = 80
Day3 (" ") = (11 - 8)(20) = 60
Day4 (" ") = (11 - 9)(20) = 40
Day5 (" ") = (11 -10)(20) = 20
Total (ton-days tardy) = 300 ton-days

This total represents the value of the denominator when computing T. If it is assumed that the actual ton-days tardy = 100, then,

$$T = \frac{\text{Actual ton-days tardy}}{\text{Total ton-days tardy}} = \frac{100}{300} = 0.333$$

Example: To compute D, one simply divides total tonnage delivered by tons available for airlift. Using the above example, and assuming only 20 tons of cargo was delivered of the total 100 available,

$$D = \frac{\text{Actual tons delivered}}{\text{Total tons available}} = \frac{20}{100} = 0.2$$

9. The scalar scoring function (SSF) is then specified and its value calculated for each alternative policy:

$$\text{SSF} = (\text{WT1})(C) + (\text{WT2})(1-T) + (\text{WT3})(D)$$

$$= (0.5)(C) + (0.375)(1-T) + (0.125)(D)$$

10. The alternative policy possessing the best SSF is then selected.

The authors made several major assumptions in implementing the methodology discussed above. The first of these was that the three attributes comprised the total set of attributes that the decision maker was interested in and formed a complete set (no other attributes were considered significant). Secondly, the values of the attributes were assumed to describe the scheduling system's output in its entirety. Finally, based on these considerations, it was assumed that the problem was modeled as completely as required. Some implicit assumptions were also necessary in developing the scalar scoring function.

The first of these was that mutual preferential independence (MPI) was assumed. This implies that a change in value of one attribute for a particular alternative does not affect the output values for the other attributes. By assuming MPI, the authors were able to utilize the linear additive form of the SSF. Edwards contends that "weighted linear averages yield extremely close approximations to very much more complicated nonlinear and interactive 'true' utility functions, while remaining far easier to elicit and understand" (Ref 17:328). Limited access to decision makers and time constraints prevented more indepth investigation of the actual form of the SSF. Additionally, the authors recognized that some interaction existed between the attributes; and consequently, the assumption of MPI may be in error. There is in fact some double counting between the attributes in calculating the SSF. A more complex form of the value function should, therefore, be investigated and additional research conducted to determine the degree of relationships among these attributes. Despite these shortcomings, the authors firmly believe that the use of MAUT is a valid means of evaluating problems involving multiple criteria. The sensitivity of the SSF to variations in weight parameters is included in Chapter V, Experimental Results.

Having developed the methodology for handling the multiple attributes, attention was directed toward the factors of the experiment: the cargo priority rules and aircraft allocation rules.

In order to investigate the interactive effect of each factor, a full factorial design was employed combining all levels of each factor with the levels of all other factors. This design varied the levels of only one factor at a time while keeping the others constant. This routine was repeated until all levels of all factors were examined. Since there were three factor levels for aircraft allocation rules and five factor levels for cargo priority rules, fifteen treatments (scheduling policies) were evaluated. A one-way analysis of variance (ANOVA) was used on the output from the simulation of these fifteen scheduling policies to test the hypothesis: H_0: group means are equal, H_A: group means differ.

The sample used in the model was a twelve day period. Multiple runs of each scheduling policy were made in the SLAM model to test the outcome on the response variables and SSF. The sample size, or number of replications per scheduling policy, was determined using output from the SLAM model (base case rules) during trial runs. The sample variance was obtained by averaging the outputs from ten runs of three different scheduling policies; and the value obtained was 0.025. The t statistic for a 95% confidence level was determined to be 1.70 for 29 degrees of freedom. The sample size was calculated using the following formula (Ref 32:189):

$$n = \frac{t^2 s^2}{d^2} = \frac{(1.70)^2 (0.025)^2}{(0.05)^2} = 28.9$$

where,

t = tabulated t value for the desired confidence level and the degrees of freedom of the initial sample.

d = the half-width of the desired confidence interval (specified as 0.05)

s = the estimate of the variance obtained in the sample or pilot run

Since the sample size obtained in this formula was 29, the authors chose to invoke the central limit theorem and used a sample size of 30. Each sample (12 day period) is itself a mean; therefore, the central limit theorem holds and normality of the response can be assumed (Ref 32).

Summary

This chapter has included the results of the literature search, the methodology utilized in trying to solve the stated problem, along with the design of the experiment. Attention was focused on relating the problem to that of a job shop and the development of cargo priority dispatch rules. Chapter IV provides the reader with an indepth discussion of the SLAM model, the vehicle chosen to test the scheduling policies.

IV. THE MODEL

Introduction

Careful construction of a representative model is a key factor in performing a simulation experiment. While it is desirable to accurately represent the system being modeled. some judgment is necessary to determine the level of detail captured in the model. In this respect, the reason for building the model must be considered foremost at every stage from design to validation and implementation.

For this experiment, the primary purpose of the simulation model was to provide a framework for comparing the relative effects of factors (cargo dispatching and aircraft allocation) on specific response variables (unit closures, cargo tardiness and system throughput) under a set of conditions specified by MAC operations research analysts. These conditions and the manner in which they were represented will be discussed later in this chapter.

In particular, this chapter first presents an abstract structural model for the system to be simulated. Discussion then proceeds to the SLAM network and the relationships of locations and activities represented by the network. This is followed by an explanation of the FORTRAN subroutines which implement special network features, including the scheduling routine which is the heart of the experiment. Finally, model verification and validation are discussed.

Structural Model

The modeling of a system is made easier if a pictorial representation can be made of it. The structural model provides such a pictorial representation (Figure 3). Pritsker (Ref 30) defines models as descriptions of systems. A model is also an abstraction of the system, which requires model builders to determine the system elements to be included in the model. A system is considered to be a set of interdependent objects united to perform a specified function. The system modelled here was a strategic airlift network for a specified contingency operation which required the allocation of airlift resources for transporting cargo. This model was developed to define the boundaries of the system and establish the modeling detail desired.

The dotted line depicts the boundary of the MAC aircraft allocation process, and reveals that some parts of the input and output are external to the area. This suggests that the allocation system does not operate in a vacuum but is related to both exogenous elements (inputs) and endogenous factors (outputs). The input block for this system includes the base structure, aircraft resources available, ground service time, cargo airlift requirements and set up time.

Inputs. The base structure consisted of four types of airfields: aircraft home stations, cargo onload bases (origins), enroute bases, and cargo offload bases

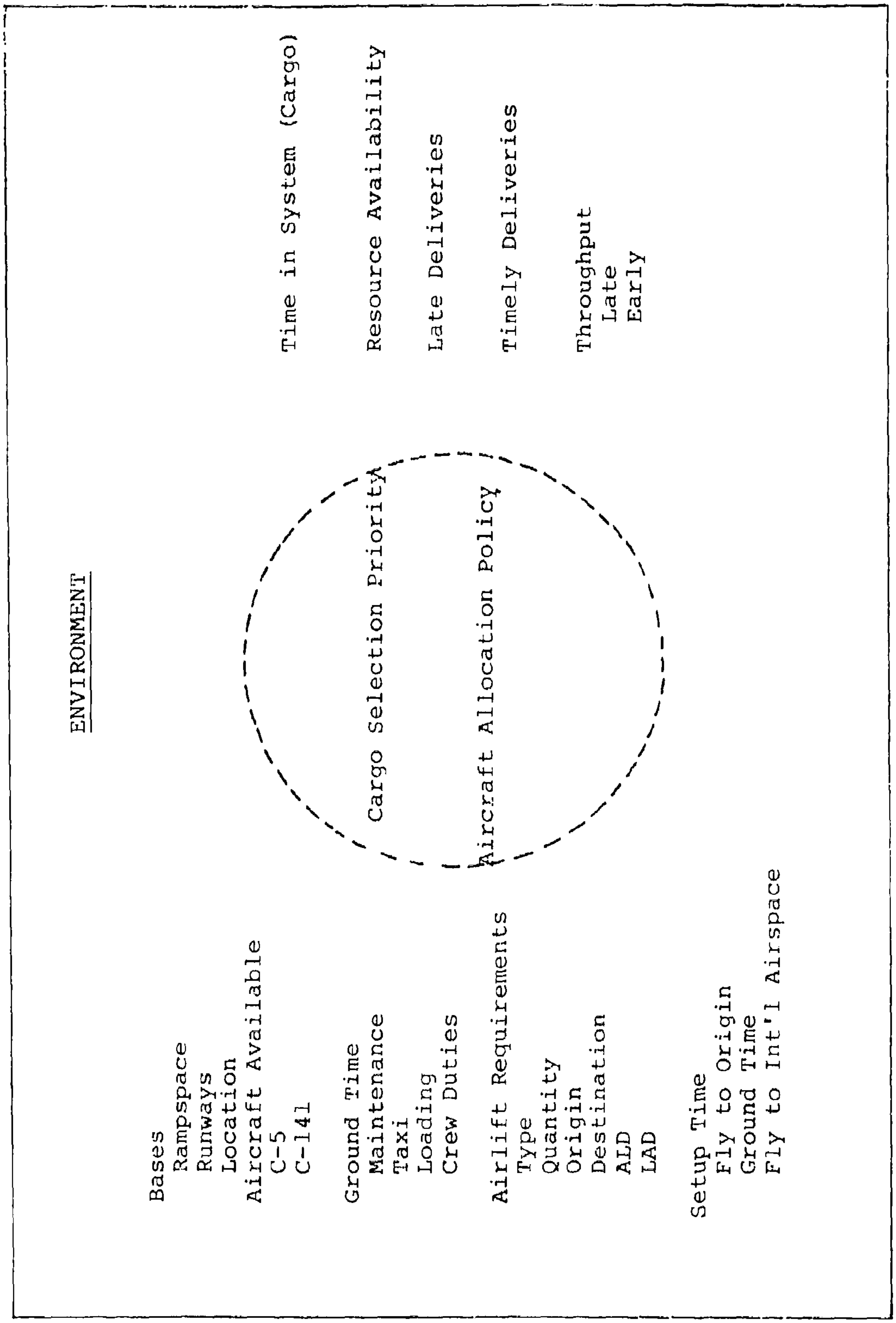


Figure 3 Structural Model Diagram

(destinations). Each aircraft type possessed its own home station. The cargo onload bases represented ports of embarkation (POE), those bases designated in operations plans where cargo was gathered for air shipment. Enroute bases consisted of those bases required for enroute aircraft support (refueling, minor maintenance, etc.) due to the deployment distances involved. Destination airfields were those bases designated to receive the cargo shipments (ports of debarkation). Enroute bases and destination airfields possess limited ramp and runway resources; consequently, the number of aircraft which could be on the ground at any time was limited. Some of the airfields transited by the aircraft only had one runway, which contributed to arrival and departure delays because of runway congestion. Furthermore, aircraft arrivals could be delayed until ramp space was available.

Aircraft may be performing other missions away from home stations when deployment is initiated and must return to their bases for necessary maintenance and mission preparations. This aircraft "generation" process would result in a uniformly distributed sequence of aircraft becoming available over some time period. According to operations planners, no more than 48 hours normally elapse before all strategic airlift aircraft (C-5s and C-141s) are available to support the contingency plan.

Ground service times during contingency operations would probably be considerably different from normal service times.

Actual data from which one could make a realistic estimate of this stochastic variable for such an operation was not available. The variables within ground service times alone can follow different distributions (i.e., maintenance, traffic delays, taxi, etc.); consequently, estimates were obtained from HQ MAC/XPSR for ground service times which were based upon estimates made by appropriate functional offices at MAC headquarters. These estimates were specified for each type aircraft and base type.

Cargo airlift requirements are of four types: oversize, outsize, bulk, and passengers. Organizational units being deployed during a contingency may possess one or more cargo types requiring transport, and these in combination were called a unit requirement. Unit requirements needing airlift support were specified in terms of weight (tons). In addition to weight, cargo requirements possessed other attributes. These included their origin, the ALD (available for load date), destination, and the LAD (latest arrival date). Ground transportation delays, storage requirements, and processing requirements prevent all scheduled cargo requirements from being simultaneously available for airlift at a given base on a particular deployment day. The delay experienced by cargo scheduled for a particular mission was considered to be uniformly distributed over a period of several hours. This specification was based upon the authors' airlift experience, as well as that of the MAC operations research staff.

The total time from an aircraft's home station departure to the point at which it entered international airspace was defined as set-up time. This time included the following: flight time to cargo origin base, ground service time at origin base, and flight time to international airspace.

Process. The MAC scheduling process converts the inputs into the desired outputs to meet its assigned mission. This process includes the manner in which airlift resources are allocated to cargo requirements having certain priorities. In any contingency operation, some type of cargo prioritization will exist. The cargo requirements used for this project were in the form of a time-phased force deployment list (TPFDL), which prioritizes cargo according to the required latest arrival date (LAD). This research effort explores other cargo priority dispatch rules along with aircraft allocation rules as discussed in the Methodology, Chapter III. The scheduler selects cargo requirements and allocates available aircraft to support the requirements based upon the various levels of factors involved.

Output. The primary output variables relevant to this study are those which directly reflect the movement of cargo. An obvious choice is system throughput, or the total quantity of cargo delivered in a set time period. This may be further refined by considering the amount of cargo delivered on or before its due date versus the amount tardy. Cargo tardiness, as well as the degree of tardiness, are both related to the time that a shipment remains in the airlift system. Various measures of system performance may also be related to the utilization (or availability) of key resources, which observably affect cargo movements. A simplified diagram portraying the sequence of events and network flow for aircraft and cargo entities is provided in Figure 4. The actual SLAM network is discussed in the following section.

SLAM Network

As mentioned earlier, Pritsker's Simulation Language for Alternative Modeling (SLAM) was selected to serve as the test vehicle for those priority dispatch rules noted in Chapter III, pages 46-49. A brief general description of SLAM networks is included in Appendix A, and definitions of SLAM user support and callable subprograms are provided in Appendix C. For a more detailed discussion, the interested reader is referred to Pritsker and Pegden (Ref 30).

The SLAM network was divided into four interrelated subnetworks. These networks are discussed under the following topic headings: aircraft generation, mission generation, and operations. Resources (aircraft, runways, and ramp space) were defined in both the network and control statements. The time unit utilized in the simulation was hours. The first subnetwork, aircraft generation, is shown in Figure 5.

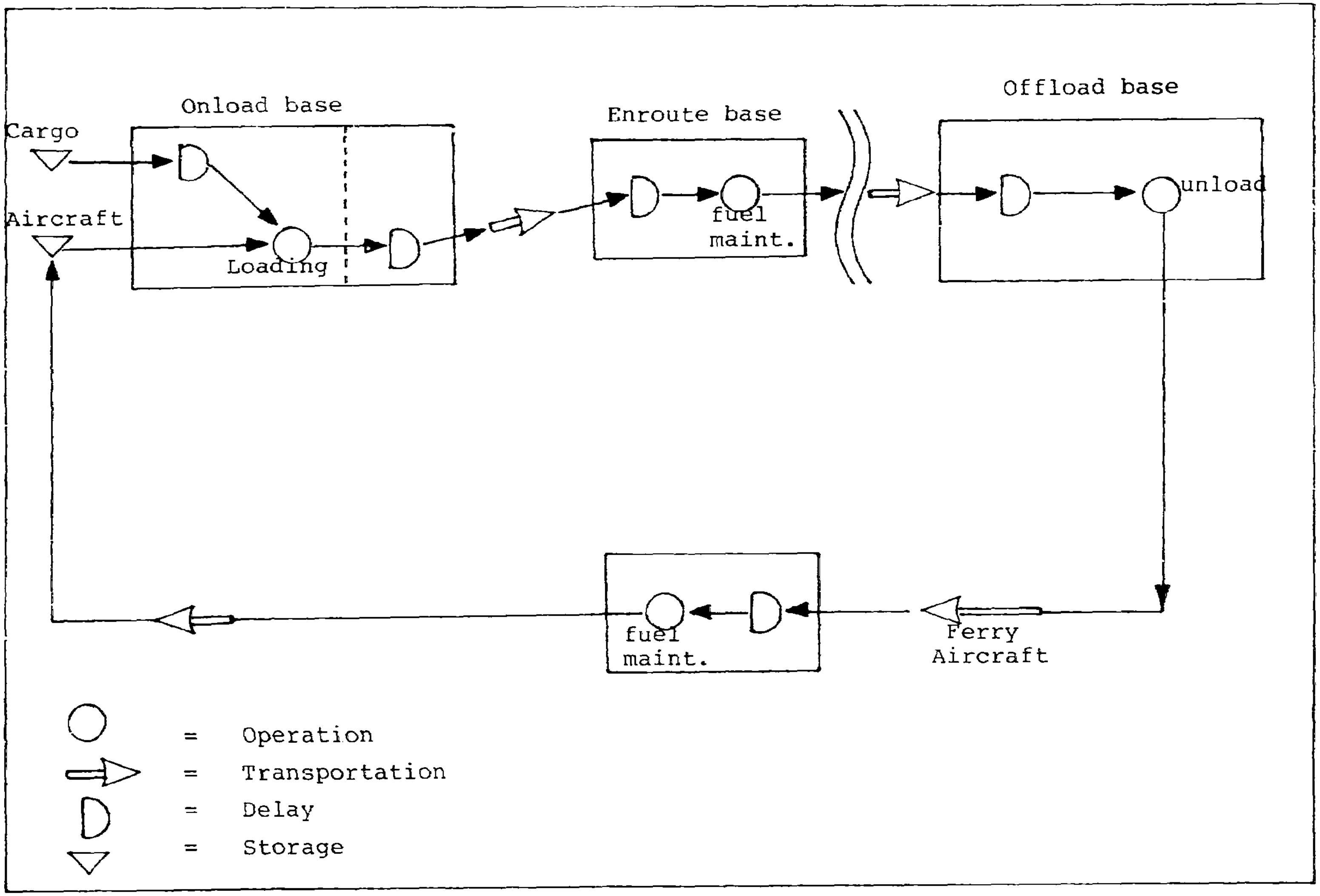


Figure 4 Simplified Aircraft and Cargo Flowchart

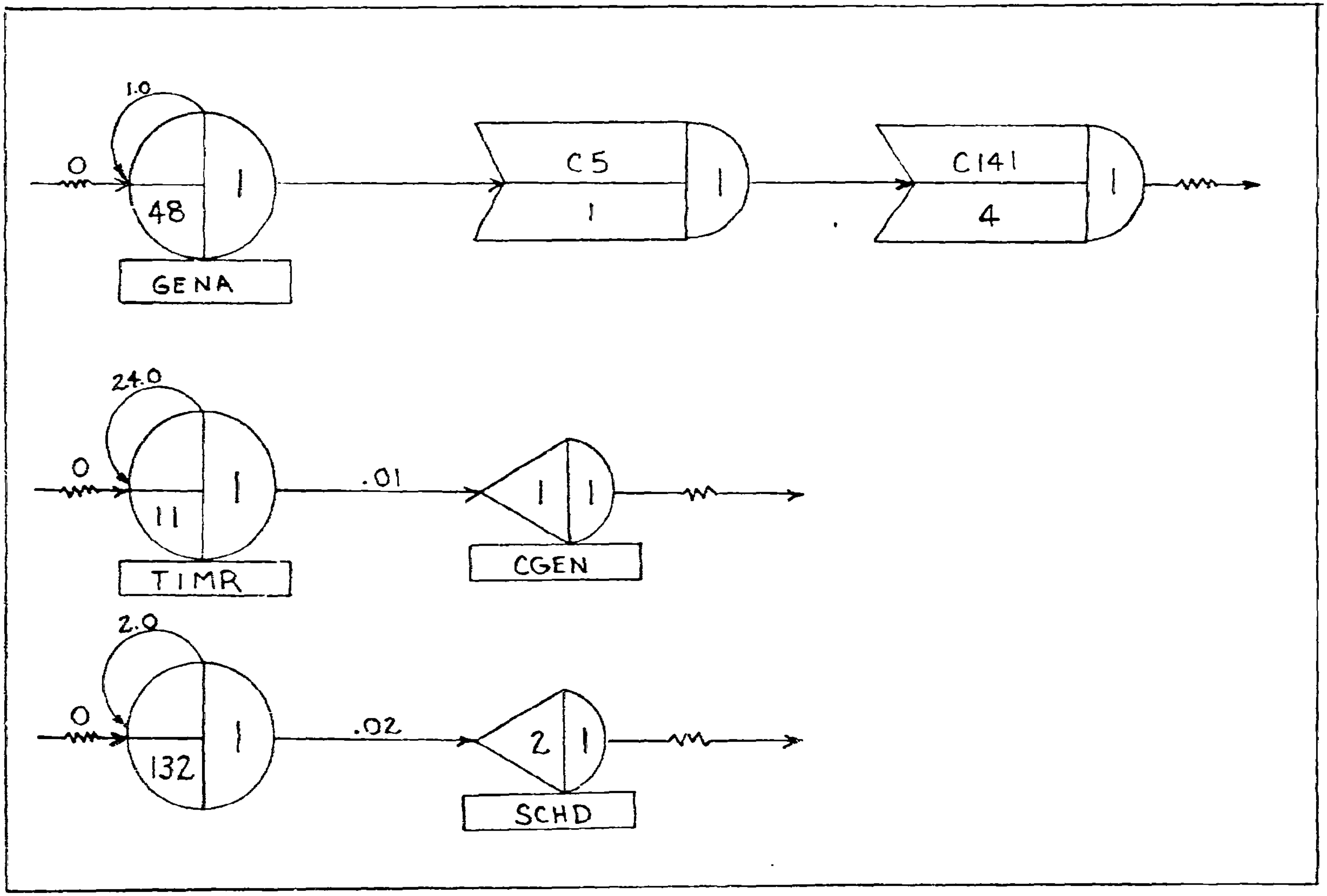


Figure 5 Aircraft Generation and Mission Generation Subnetworks

Aircraft Generation. The purpose of the aircraft generation network was to phase in the number of aircraft available to support the contingency plan. This network was based upon the assumption that a full complement of aircraft would be made available deterministically over a period of 48 hours. The network consists of three nodes: a CREATE node which serves as a timer, and two ALTER nodes used to increment the number of aircraft resources available.

Beginning at time zero and each hour thereafter, the CREATE node (GENA) releases an entity which proceeds through the network and passes through both C-5 and C-141 ALTER nodes. As the entity passes through each ALTER node, the number of each type aircraft resource associated with that ALTER node increases. The incremental increase in aircraft was based upon an assumed uniform distribution. This resulted in one additional C-5 and four additional C-141s entering the system for each of the forty-eight entities created hourly by GENA. Table I illustrates this incremental increase for C-5 aircraft. The mission generation portion of the network consisted of two subnetworks which are discussed below.

Mission Generation. The purpose of the first subnetwork (Figure 5) was to generate cargo requirements for each of the first eleven days of the deployment period. This network consists of only two nodes connected by one activity. The CREATE node (TIMR) releases an entity at the beginning of each

TABLE I
AIRCRAFT GENERATION (C-5)

Time	Not Available	Available	Total
0 hour	48	10	58
1 hour	47	11	58
2 hours	46	12	58
↓	↓	↓	↓
48 hours	0	58	58

24-hour period (for eleven periods) and effectively initiates each day of the deployment. Each entity created by TIMR proceeds to an EVENT node (CGEN), which calls a FORTRAN subroutine (GTREQ, Appendix D) to select cargo requirements for the current day from a master file created before the simulation began (Figure 6). GTREQ also files these requirements in the priority order specified by a SLAM card. These requirements are copied to other files for processing by a second EVENT node (SCHD) to be discussed below.

The purpose of the second subnetwork was to generate mission entities by scheduling available aircraft to transport those cargo requirements generated in the previous network. This network (Figure 5) consists of two nodes and a single activity. The CREATE node, at time zero and every two hours thereafter for 11 days (132 total creations), creates an entity which proceeds to SCHD. Node SCHD calls subroutine SCHED (Appendix D) which checks the number of aircraft currently available and schedules missions until either cargo or airlift resources are depleted. These mission entities were entered into the operations network where they seized appropriate aircraft resources and began a mission.

Operations. The primary purpose of the operations network was to control mission progress according to specified conditions and to provide a framework for collection of statistics. Subroutine SCHED returns mission entities to ENTER

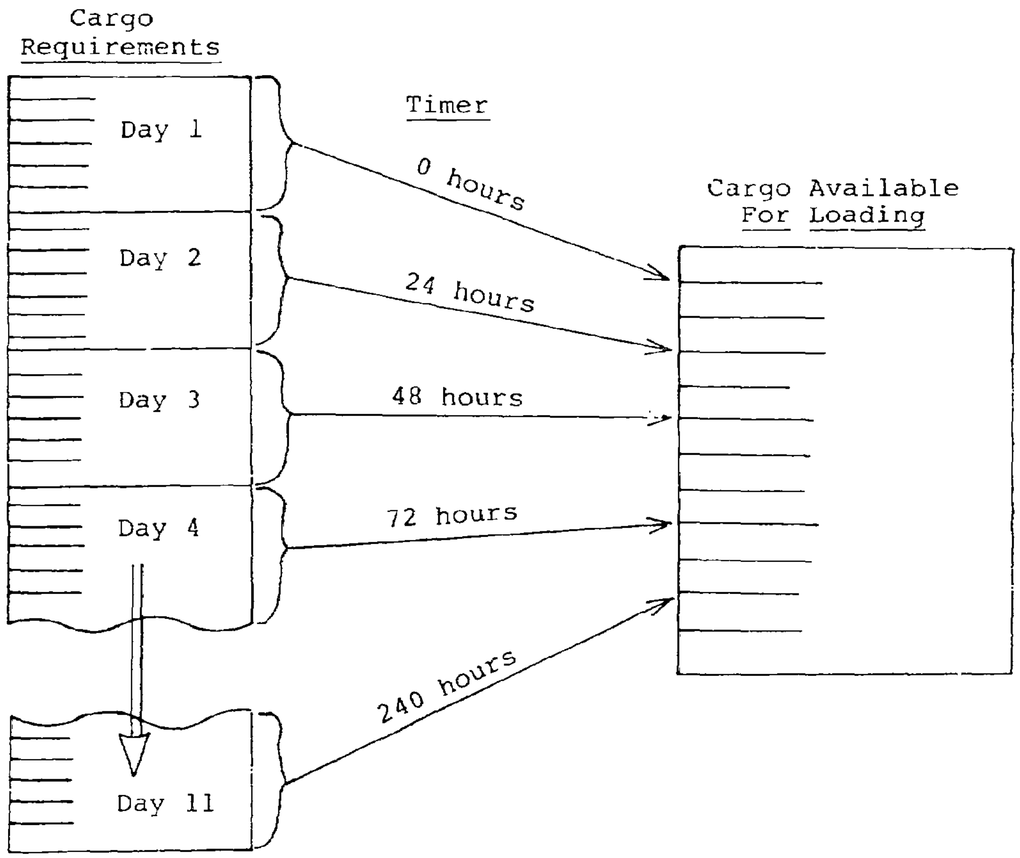


Figure 6 Simplified Illustration of Subroutine GTREQ

nodes one or two based upon which type aircraft was projected to transport the mission entity.

Mission entities are required to flow through this network, which represents the underlying network of airfields utilized for the contingency operation addressed. A number of factors were aggregated, including cargo origin bases, aircraft home stations for each type aircraft, and cargo destination bases. To illustrate the operation of this network, the authors elected to provide the reader with a step-by-step explanation of a portion of the network.

Figure 7 depicts mission entities returned to ENTER nodes one (for C-5s) and two (for C-141s). After passing through their respective ENTER nodes, entities are assigned attributes based on the type aircraft which provides the airlift service. If entities are to be transported by C-5s, their twelfth attribute is assigned the value of one, whereas if transportation is by C-141s, the twelfth attribute is given a value of two. Mission entities then travel to AWAIT nodes (BIG for C-5s and SMAL for C-141s) where they wait for aircraft resources to become available. Once an aircraft resource is available, it is seized by the mission entity. The activity between BIG and HBC5 has a duration which comes from a uniform distribution with a minimum value of 2.0 hours and a maximum value of 14.0 hours. This time period represents the possible delays which could be experienced by cargo requirements before they are actually loaded on an aircraft. The fact that the

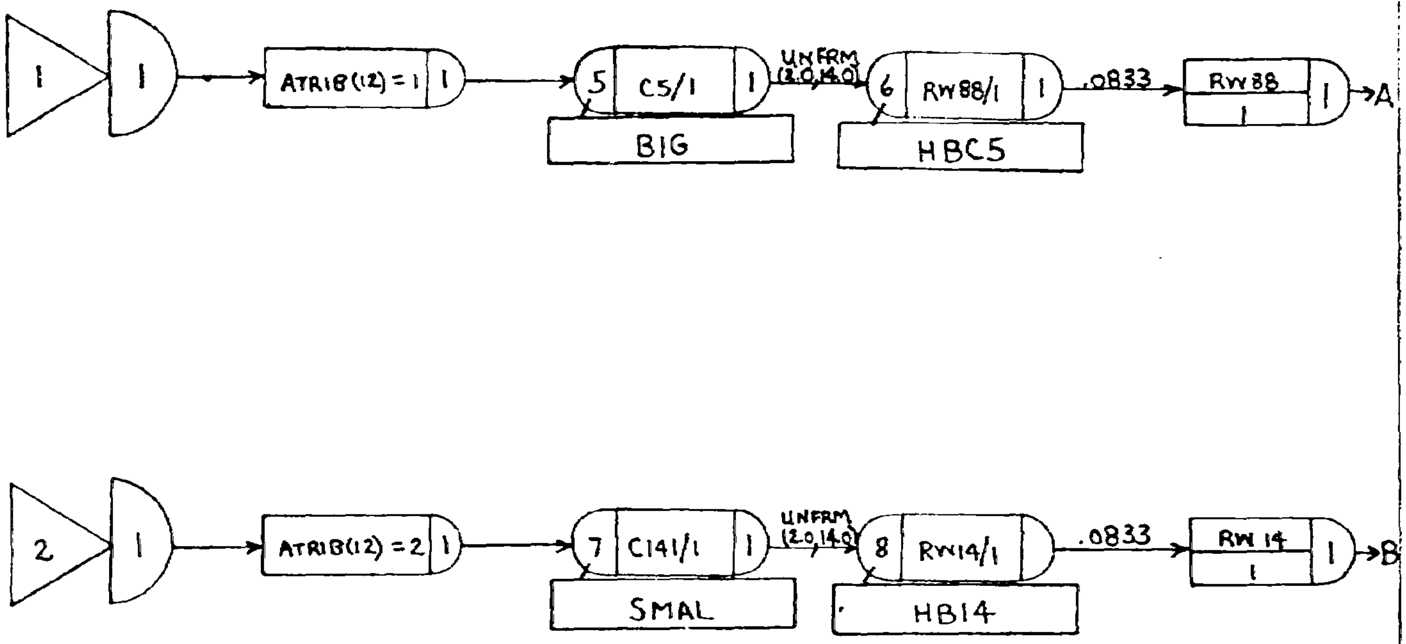


Figure 7 SLAM Operations Subnetwork

aircraft is seized before this delay is known represents the allocation of aircraft against projected cargo availability. This same distribution and times are specified for the activity between SMAL and HB14. HBC5 and HB14 (AWAIT nodes) represent queues for aircraft desiring takeoff from their respective home base runways. When a runway resource becomes available, it is seized by the aircraft. The activities immediately following HBC5 and HB14 represent the time required for takeoff by each aircraft (0.0833 hour). Once the takeoff activity is completed, the runway resource is freed (RW88 for C-5s and RW14 for C-141s).

Figure 8 depicts the segment of the model from aircraft home station departures to the runway departure queue at Lajes, Azores. The activities from A to GOON and B to GOON represent setup time as previously defined. User function one (Appendix D) computes this time for each type aircraft. This activity time includes the flight time to cargo origin bases, ground service time, and flight time to international airspace for each aircraft. The flight time from the point the aircraft enters international airspace to the landing queue at Lajes is represented by the activities between the GOON node and LD67. These flight times are specified as constants for each type aircraft (4.0 hours for C-5s and 5.0 hours for C-141s). Both runway and ramp resources are required to be available before an aircraft is permitted to land. Delays encountered at this

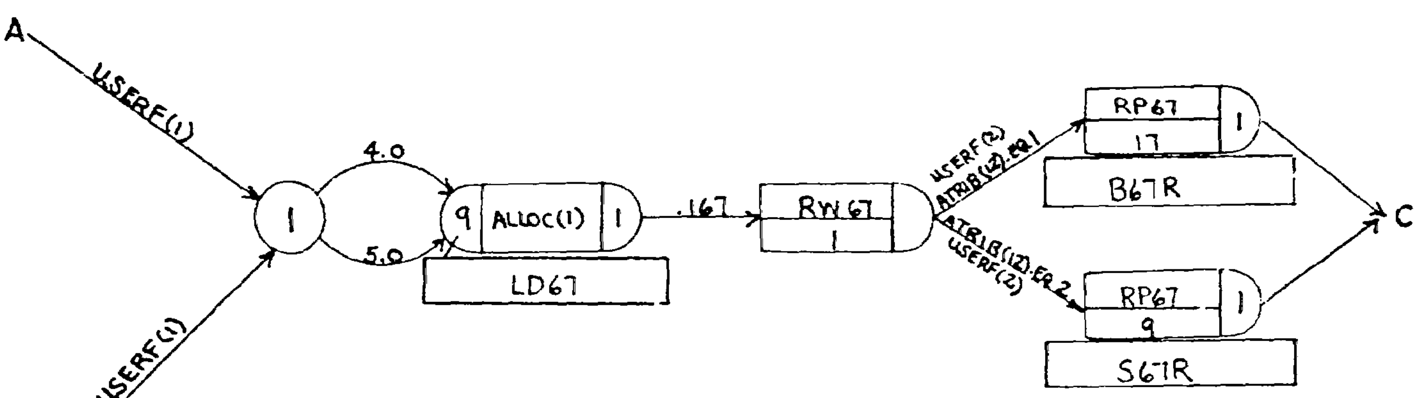


Figure 8 Operations Network (cont'd)

point represent a combination of takeoff delays and landing delays. Once appropriate amounts of each type resource are available, the affected aircraft accomplishes its landing. The activity immediately following LD67 represents the time required for landing for either aircraft and is specified as a constant (0.167 hour). After completing the landing, the runway resource is freed, and this is depicted by the RW67 FREE node. The activities depicted after the FREE node for RW67 represent the total ground time consumed for each type aircraft. Since Lajes was specified as an enroute base, user function two was utilized to calculate the ground time for each aircraft (Appendix D). After completing ground service, ramp resources are freed. The freeing of ramp resources is represented by FREE nodes B67R (17 units) for C-5s and S67R (9 units) for C-141s. The activities emanating from B67R and S67R route the aircraft to the takeoff AWAIT node (TO67) at Lajes.

Figure 9 depicts the AWAIT node (TO67), where each type aircraft awaits a runway for departure from Lajes, and illustrates mission entity flow through the ground service phase at Cairo International Airport, Egypt. Once a runway resource is available at Lajes, the aircraft is permitted to takeoff. Again, the time for takeoff is specified as a constant (0.0833 hour). After completion of the takeoff activity, the single runway resource is freed, and the aircraft fly to Cairo. The activity times following the FREE node are specified as constants (i.e. flight times of 7.23 hours for

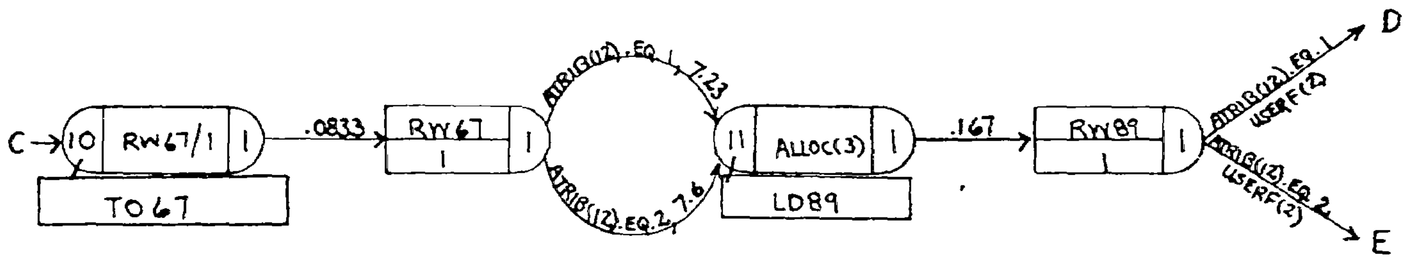


Figure 9 Operations Network (cont'd)

C-5s and 7.6 hours for C-141s). Upon arrival at Cairo (LD89), each aircraft is permitted to land if both ramp and runway resources are available. If not available, the aircraft is required to hold. Once allocated these resources, the affected aircraft completes its landing, which consumes a constant time period of 0.167 hour. This time is represented by the activity following LD89. After completion of the landing activity, one runway resource is freed. Values of the twelfth attribute determine which activity the mission entities follow after the FREE node. Ground service times are again specified by user function two. After completion of the activities, ramp resources are freed at FREE nodes B89R and S89R (Figure 10).

As previously noted, B89R and S89R free ramp space (33 units for C-5s and 16 units for C-141s). The entities leaving these nodes flow to TO89, the departure AWAIT node where aircraft await one runway resource for takeoff. Once this resource is available, the aircraft accomplishes its takeoff. This is represented by the activity leaving TO89 (0.0833 hour). Upon completion of this activity, one runway resource is freed, thus making it available for another aircraft's departure. The activities depicted after the FREE node for the runway resource represent the flying times for each type aircraft to the destination base (LDXX). These times are represented as constants (3.0 hours for each type aircraft). When a runway resource is made available at LDXX, an aircraft seizes the resource and completes its landing, consuming 0.167 hour.

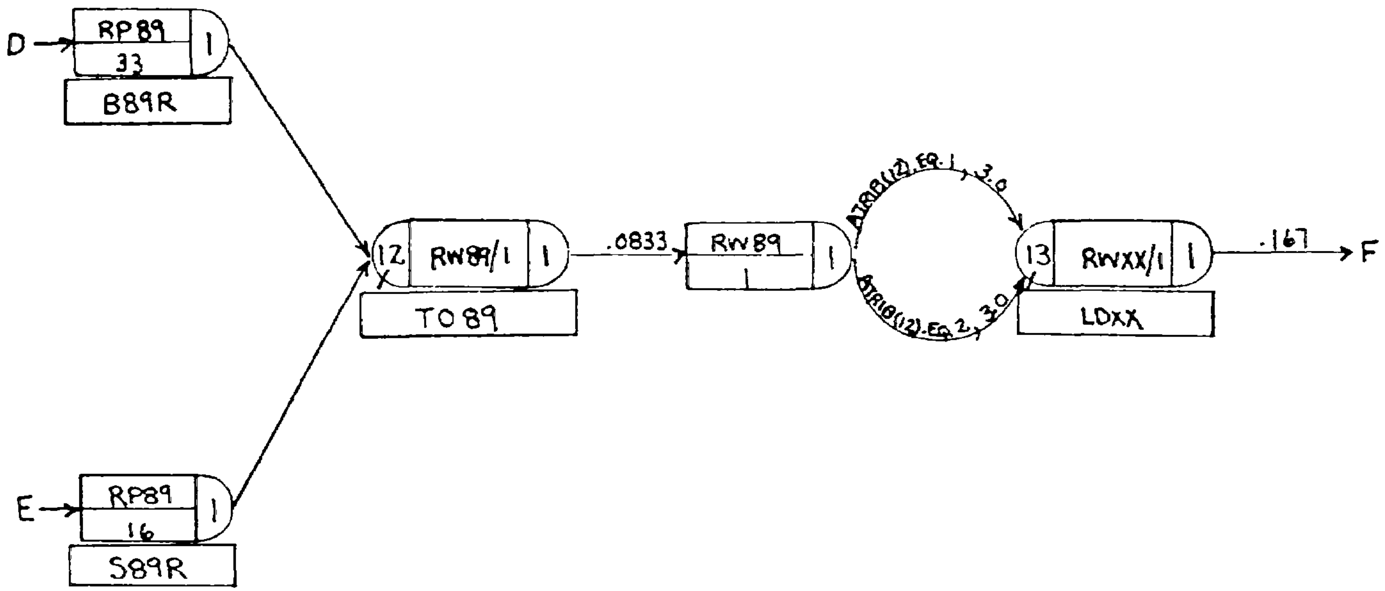


Figure 10 Operations Network (cont'd)

In Figure 11, one unit of runway resource is freed at the FREE node each time a mission entity passes. Again, based upon the value of the twelfth attribute, the duration of the ground service time is calculated by a user function. Since this is the destination base, user function three (Appendix D) is utilized to compute the ground time for each type aircraft. After completion of ground service activities, each type aircraft proceeds to TOXX, an AWAIT node for a runway resource. The activity leaving TOXX has zero time duration and connects TOXX to DLVR. The EVENT node DLVR is activated by an aircraft entity just prior to the mission's departure from the destination airfield and calls subroutine DLIVRY (Appendix D).

DLIVRY checks the number of requirements served by the mission, and posts the delivery with the original cargo requirement data in the master file as either early (on time) or tardy. If the delivery is early, and completes the delivery of the total quantity of cargo associated with that unit requirement, a flag is set in the master file to indicate that closure was met for that requirement. DLIVRY then updates the daily statistics on cargo delivered and cargo tardiness.

The activity following DLVR in Figure 11 represents the time required for takeoff from the destination base (0.0833 hour). Once the takeoff is completed, the runway resource is freed and the aircraft proceeds to the first enroute base on

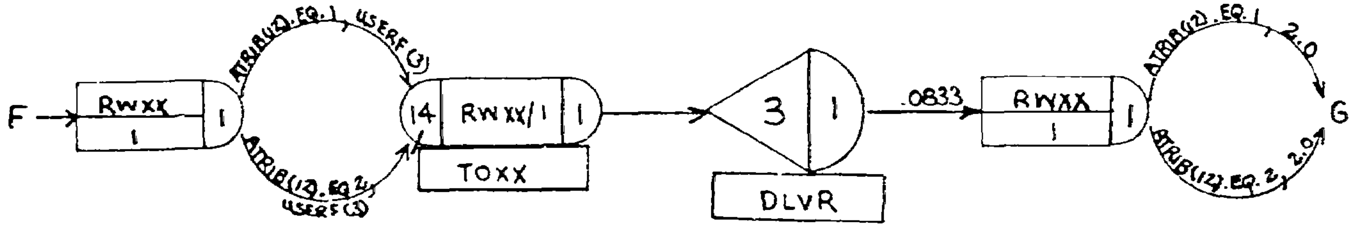


Figure 11 Operations Network (cont'd)

the return trip to the CONUS (Jeddah, Saudi Arabia). The times for this activity are specified as constants (2.0 hours) for each type aircraft.

The remaining network (Figures 12-14) for the return trip to the CONUS operates in essentially the same manner as that previously discussed. After making two enroute stops (Jeddah and Prestwick, England), each type aircraft proceeds to its assigned home station where post mission maintenance is performed. The ground time at aircraft home stations is calculated by user function four (Appendix D). Upon completion of ground service time, aircraft are made available for other missions.

From the foregoing discussion, one may see that airways were represented as activities between bases. Enroute bases were represented in terms of ramp and runway resources with ground time representing total ground time. Runways and ramp space were represented by AWAIT nodes to control the flow rate of mission entities into and out of an airfield.
Onload/offload bases could be considered source/sink nodes in normal network terminology. All takeoff times for either type aircraft were specified as 0.0833 hour, and landing times were given a duration of 0.167 hour. These takeoff and departure estimates were based upon the combined experience of the authors. The flight times between enroute bases were all specified as constants. These times were approximated by using

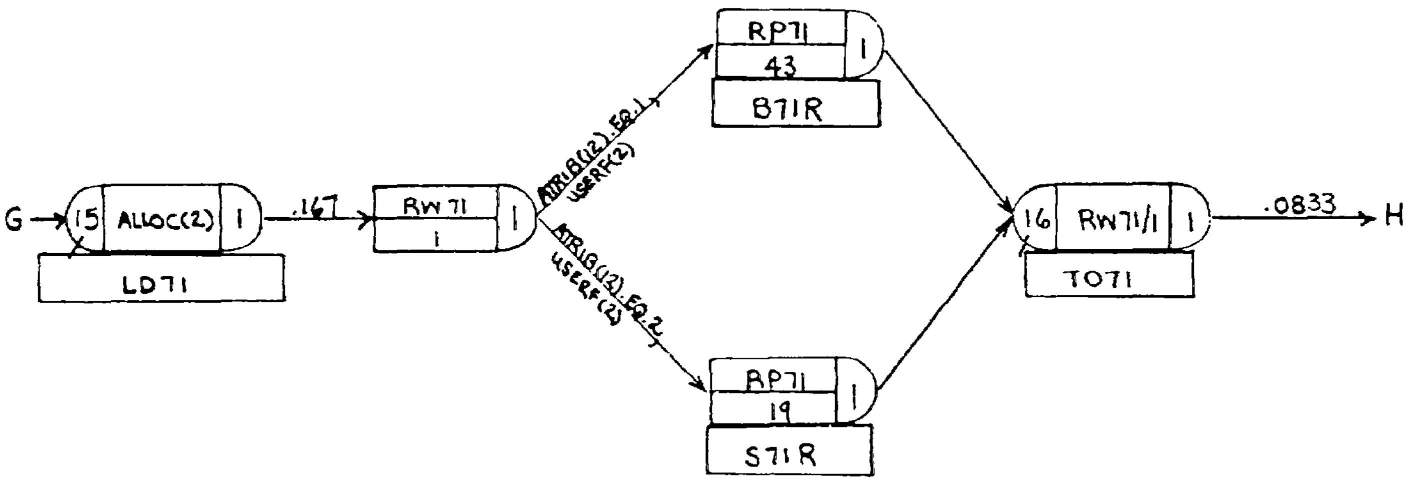


Figure 12 Operations Network (cont'd)

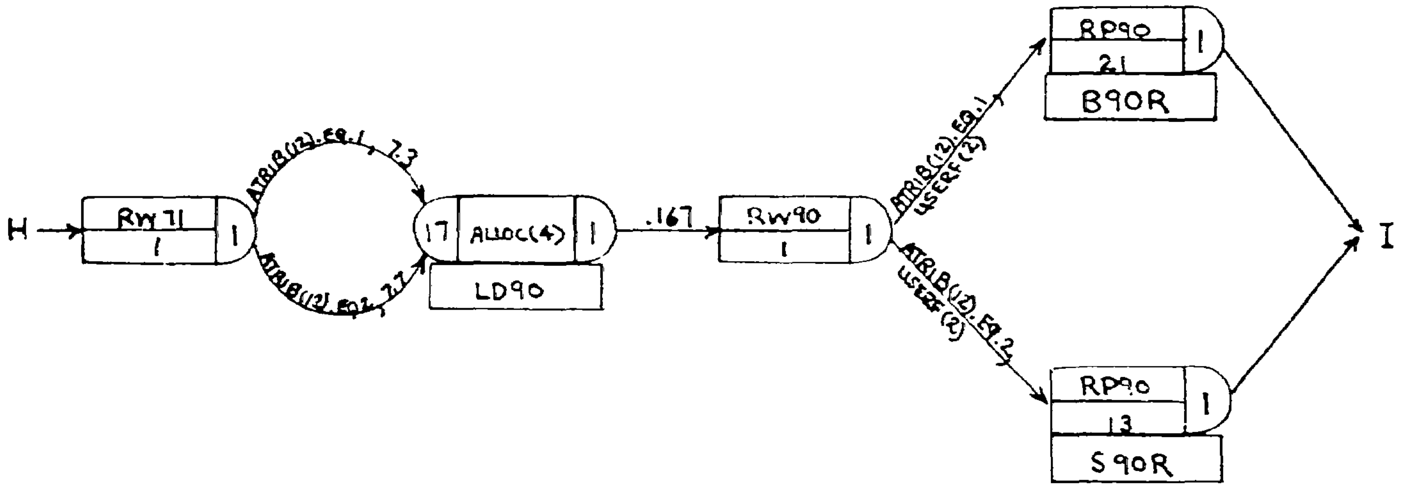


Figure 13 Operations Network (cont'd)

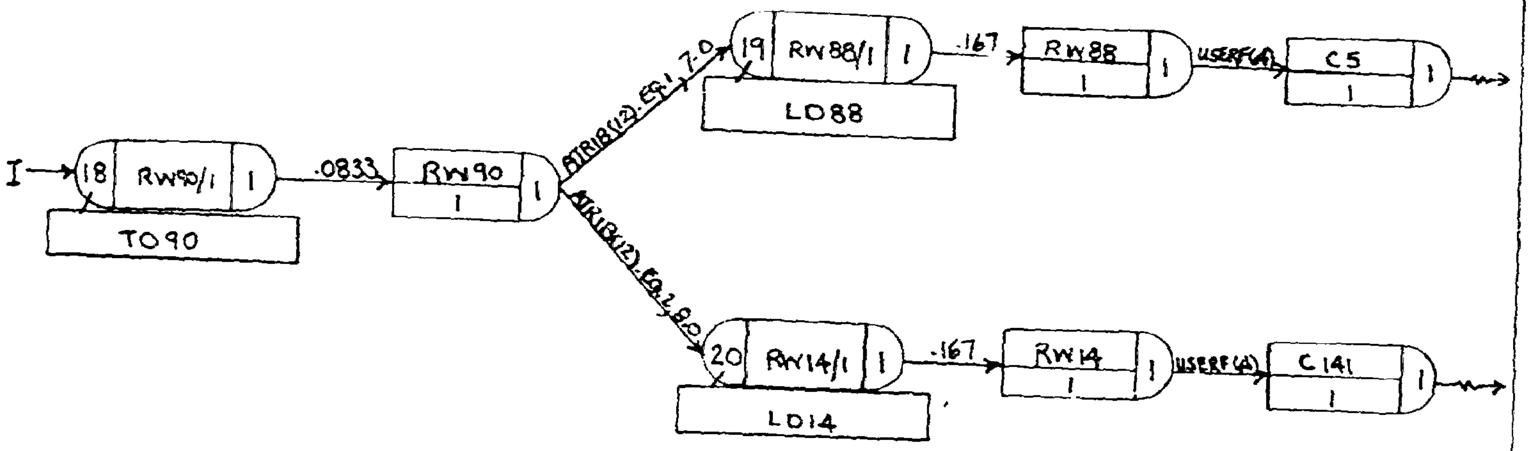


Figure 14 Operations Network (cont'd)

distances and average cruise speeds from AFR 76-2, Military Airlift--Airlift Planning Factors (Ref 2). At this time it is appropriate to discuss the SLAM program inserts developed in the FORTRAN code.

SLAM Program Inserts

The SLAM simulation language permits optional user-written FORTRAN inserts to be used and replace a number of its standard features. This feature made it possible to reduce the complexity of the network where the logic was not straightforward. An example of this was the AWAIT node which is normally capable of only allocating one type of resource to an entity. By writing an ALLOC subroutine, it was possible to allocate both a runway resource and a required number of ramp space units to a mission entity. Author developed codes were substituted for the following intrinsic SLAM features:

1. Program Main
2. Subroutine INTLC
3. Subroutine EVENT
4. Subroutine ALLOC
5. Subroutine OTPUT

Program Main (Appendix D) was substituted only to redimension the array used for SLAM system files (called NSET/QSET). This was necessary because more memory was required than the default value permitted.

Subroutine INTLC (Appendix D), an optional user subroutine, was used to initialize variables and parameters, including some values for later program inserts. It also called subroutine RDCGO (Appendix D), a program insert to establish a SLAM file of cargo requirements to be used throughout the simulation run. This was necessary since SLAM clears all files prior to each run. Subroutine INTLC was called at the beginning of each simulation run to establish initial conditions for that run.

Subroutine EVENT (Appendix D) was used to call subroutines which altered the state of the system according to external logic. Special (EVENT) nodes in the network caused the SLAM executive program to branch to this subroutine, which halted all normal processing until control was returned. This was necessary for the following reasons: to file requirements, to schedule mission entities, and establish cargo priority. The subroutines called from EVENT are discussed below in Discrete Event Inserts.

Subroutine ALLOC (Appendix D) was used to control the allocation of both runway and ramp resources at all network airfields except aircraft home stations and cargo destination bases. It was assumed (because of origin and destination base dispersal in the input data) that there would not be sufficient congestion at these bases to affect the mean response. Therefore, AWAIT nodes were utilized for only the runway resource to provide realistic aircraft flow at these locations.

A mission entity arriving at one of the enroute bases was not permitted to land until both ramp and runway resources were available. In actual practice, the aircraft would be scheduled for a departure time which would contribute to an orderly flow of traffic and prevent unnecessary congestion upon terminal arrival. The relaxation used here is permissable under the assumption that ramp congestion should be a limiting factor only at enroute airfields. Subroutine ALLOC calls subroutine SEIZE, an internal SLAM subroutine, to obtain ramp and runway resources, if available, when a mission entity arrives at an AWAIT node requiring the ALLOC subroutine.

Subroutine OTPUT (Appendix D) was used to print user-collected statistics at the end of a simulation run. A tabular printout of daily cargo deliveries, cargo tardiness, and requirements meeting closure were provided by OTPUT. Summary cargo delivery data for each simulation run were also written to a disk file for subsequent analyses.

Discrete Event Inserts

The EVENT subroutine discussed above permitted complete freedom in controlling the status of SLAM entities. This subroutine allowed entities to be entered into or removed from files, permitted attributes to be read and altered, activities to be scheduled or stopped, and statistics to be collected. Intrinsic SLAM functions and subroutines were available to perform most of these functions from within a FORTRAN subroutine. Discrete event inserts were used to select daily cargo requirements, to schedule missions against these requirements, and to collect statistics upon delivery of an increment of cargo.

Before beginning the simulation, subroutine INTLC was used to establish a master file of cargo requirements to be used throughout the simulation. Cargo requirements were read from an existing file in subroutine RDCGO (read cargo). These requirements possessed various attributes of size and weight. Each cargo requirement was assigned a requirement number and placed (with appropriate attributes) in the SLAM master file. By size of cargo, the cargo type was implied (i.e., outsize, oversize, bulk, or passengers).

The functions of EVENTS 1 and 2 (Figure 5) were discussed earlier in the Mission Generation section of the SLAM network (page 67). EVENT 3 (Figure 11) was included in the discussion of the Operations section of SLAM network (page 80).

Function USERF (Appendix D) assigns the ground service times for the four types of bases included in the model. As noted earlier in the Structural Model (page 59) , ground service times depend not only upon the activities performed at a base, but also upon the type aircraft. These estimates were obtained from HQ MAC operations planners, and represent the best information available. As noted in the Operations portion of the SLAM network, this function contains four sections depending upon the type base. These sections are as follows:

1. USERF(1)--computes the ground time for cargo origin bases for each type aircraft.

2. USERF(2)--computes the ground time for enroute bases for each type aircraft.

3. USERF(3)--computes the ground time for destination bases for each type aircraft.

4. USERF(4)--computes the ground time for aircraft home station bases for each type aircraft.

Each time USERF is called, a ground time is assigned from a discrete probability function appropriate for the type of aircraft and loading/servicing activity.

The primary purpose of this experiment was to evaluate various scheduling policies; and, therefore, a detailed discussion of the scheduling routine is provided.

The Scheduler

The experimental factors are controlled by subroutine SCHED, which is called by EVENT 2 at two hour intervals. SCHED is by far the longest, most complex, and most important of the subroutines. Its operation may be described with regard to three facets: aircraft allocation, initial selection of mission cargo, and selection of filler cargo.

Aircraft allocation, considered separately, is rather straightforward. Each level of this experimental factor is represented by reserving C-5 aircraft for the appropriate cargo type as follows:

Level 1 = Type 1 cargo only
Level 2 = Type 1 or 2 cargo
Level 3 = Type 1, 2, or 3 cargo

Where cargo type one is outsize, type two is oversize, and type three is bulk cargo. Passengers are only loaded on a C-5 if it cannot be fully loaded with cargo. To reserve these aircraft, a conditional statement prevents their selection for a mission unless the next requirement to be serviced is of the appropriate cargo type(s). Once selected for a mission, however, a C-5 with remaining capacity may be filled with any cargo type.

The C-141 aircraft may be selected to service any cargo type, with the obvious exception of outsize cargo. This constraint grows from the definition of outsize cargo as items with dimensions too large for the C-141 cargo compartment.

The initial selection of cargo requirements refers to the selection of the first requirement to be loaded on a new mission entity. (Once the entity is selected, it may fill several aircraft, or it may only partly fill a single aircraft.) Two schemes were used: a "base case" which mimics the MAC M-14 concept of aircraft preference for cargo types (Ref 16); and, a more straightforward approach which selects the requirement having the highest assigned priority.

Aircraft preference indicates the ranking of cargo requirements according to type (or size) of cargo. The order of preference is: outsize, oversize, bulk, and passengers. The due date (LAD) is a secondary ranking attribute within each cargo type.

Alternately, the priority of each cargo requirement may be based upon an intrinsic or computed attribute. This attribute is then specified as the rank index for a file of available requirements, and the highest priority requirement is always the first in the file. This facilitated the evaluation of priority dispatch rules discussed in the Methodology section, Chapter III.

Selection of filler cargo follows the method of initial selection process, but with two major differences. First, the search routine (subroutine GTFILL) checks for other cargo that is part of the same unit requirement as the initial requirement. Secondly, if no more unit cargo is found, GTFILL attempts to match the origin--destination pair of the initial requirement. If neither search is fruitful, the mission entity is closed.

With this background, the sequence of events in the scheduler is now presented. When called (every two hours), the scheduler first checks aircraft availability. If aircraft are available, cargo availability is checked. A mission entity is set up with attributes for origin and destination. If the aircraft type and cargo type available are compatible under the existing allocation scheme, the cargo requirement is loaded (within aircraft capacity).

The scheduler attempts to fill partially loaded aircraft based on the cargo requirements in the master file at the time SCHD EVENT calls subroutine SCHED. When an aircraft is fully loaded, or if no cargo match is found to fill it, an attribute is set for the aircraft type and the entity is entered in the operations network at the appropriate ENTER node.

The process continues until either (1) the number of missions generated equals the number of available aircraft, or (2) all compatible cargo requirements are scheduled. Program control is then returned to the SLAM executive until EVENT 2 is again activated.

Since the verification and validation phases are critical in building a model, a discussion of each has been provided.

Verification

Fishman and Kiviat defined verification as follows:

> "Verification determines whether a model with a particular mathematical structure and data base actually behaves as the experimenter assumes it does." (Ref 21:70)

Questions normally asked during this phase include: (1) Are the parameters, the statistical distributions, and other data used in the model as intended by the modeler? and (2) Are the model's input-output transformations as intended? Verification, however, includes insuring that both the structure and data base are behaving properly. For structural verification, the computer program was written and debugged in modules. The simulation model's program and key program units were written and debugged first. Additional subprograms were added successively until the model was developed. A structural "walk through" was conducted on several occasions. The SLAM trace routine was utilized to assist in the verification process. Discrete data (constant times rather than distributions) was also used during the initial stages of development. The SLAM model was run under these simplifying conditions for which the results could be hand calculated. In addition, print statements were routinely placed in the program to print the values of key variables and insure the program was functioning properly.

Data base verification implies that the various specified distributions in the model are in fact producing the desired distributions. A special SLAM program was developed in the early phase of model development to determine if the SLAM random number generator was functioning properly (Appendix E). This program generates 1000 random numbers from exponential, normal, and uniform distributions for user specified SLAM random number streams. The user is permitted to specify any of nine possible streams. Chi-Square and Kolmogorov-Smirnov (K-S) statistical tests were run using the Statistical Program for the Social Sciences (SPSS) (Ref 28).

The first distribution tested was an exponential distribution with a specified mean of 6.0 hours from random stream four. A Chi-Square test was used to test the null hypothesis (H_0) that there was no significant difference between observed data and those which would be expected in an exponential distribution with a mean of 6.0 hours. For the 500 random numbers tested, the calculated Chi-Square test statistic obtained was 3.007. For a 95% confidence level with three degrees of freedom, the tabulated Chi-Square statistic was 7.81. Since the calculated statistic was less than the tabulated value, the null hypothesis could not be rejected.

Kolmogorov-Smirnov tests were used to evaluate four other distributions (two normal and two uniform). The first normal distribution tested was specified to have a mean of 6.0 and a standard deviation of 1.0 with random numbers coming from stream four. The computed mean and standard deviation for 500 random numbers were 5.984 and 1.087 respectively. The critical difference at a 95% confidence level was calculated using the following formula:

$$D = \frac{1.36}{\sqrt{n}} = \frac{1.36}{\sqrt{500}} = 0.0608$$

Since the calculated maximum difference obtained from SPSS was 0.0409, which was less than the critical value above, the authors failed to reject the null hypothesis. (The null hypothesis stated there was no significant difference between the observed data and that data which would be given by a normal distribution with a mean of 6.0 and standard deviation of 1.0.) A second normal distribution tested was based on a mean of 0.05 and a standard deviation of 0.017, but from stream three. Similar results were obtained for this test and the null hypothesis was not rejected.

Two uniform distributions were also evaluated using the Kolmogorov-Smirnov test. The first of these was specified to have a minimum value of 2.0 and a maximum value 5.0. The random stream chosen for this test was number nine. A mean value of 3.514 was obtained for the 500 random numbers generated. The calculated maximum difference obtained was 0.0320 and the critical difference at a 95% confidence level was again 0.0608. Since the calculated value was less than the critical value, the null hypothesis that there was no difference in population parameters could not be rejected. A second test was run for a uniform (2.5, 3.0) distribution from stream one with similar results. These five tests were completed for distributions initially included in the model.

The uniform distribution (2.0, 14.0) utilized in the completed model was also subjected to a K-S test and the null hypothesis rejected. The authors concluded from these tests that the SLAM random number generator was producing desired distributions. With the verification phase completed, attention was directed toward the model validation process.

Validation

> "The validation task consists of determining that the simulation model is a reasonable representation of the system. Validation of simulation models, although difficult, is a significantly easier task than validating other types of models, for example, validating a linear programming formulation. In making validation studies, the comparison yardstick should be both past system outputs and experimental knowledge of system performance behavior" (Ref 30:12-13).

As indicated by Pritsker (above), the objective of a simulation experiment is to represent a real system for the purpose of producing a specific output. The model output may then be compared with past system outputs (if available), or the "reasonableness" of the model output may be evaluated by personnel who have experience in the real system.

Shannon (Ref 32) presents a "utilitarian" approach to model validation, consisting of three stages which occur in an iterative manner throughout the development and implementation process:

> 1. Seek face validity of the internal structure of the model based upon a priori knowledge, past research, and existing theory. This first stage of validation entails looking at each of the simple processes modeled to ensure that the building blocks, so to speak, are the best possible. Are the hypotheses reasonable? Do the assumptions make sense?
>
> 2. Wherever possible, empirically test the hypotheses used. Test the assumptions, parameters, and distributions used in the model. (This step merges with the verification process.)
>
> 3. Compare the input-output transformations generated by the model with those generated by the

> real world system. Attempt to establish the usefulness of the model--for example, a model that predicts successfully is usually considered more valid than one that merely reproduces past observations.

The validation of this model proceeded in a manner similar to Shannon's approach, with heavy reliance on face validity. As with many models of military activities, real system output data for the scenario of interest were not available. Substantial reliance upon the expertise of personnel in HQ MAC/XP was, therefore, essential. Additionally, the authors' own airlift experience figured significantly in the validation process. Interviews with individuals possessing intimate knowledge of the strategic airlift system were also sought to assist in the validation effort. Major Charles Dillard and Captain Wayne Stanberry assisted in evaluating the assumptions used to scale the problem and aggregate several elements of the real world system. Their experience comes from observing the output of the MAC airlift system, both as analysts and as operators of airlift resources.

Major Dillard is the Chief of Simulation Applications at HQ MAC/XPSR (Systems Research). He has held this position for the past three years. Prior to this assignment, he accumulated five years of airlift experience as a MAC pilot. Major Dillard was able to provide particularly valuable assistance because of the combination of his airlift experience and his seniority in his present position.

As the Chief of Simulation Applications, Major Dillard is directly involved in all major simulations of the MAC airlift system. He has validated numerous estimates of MAC system parameters through the Simulated Wartime Advisory Group, which includes highly experienced personnel from all functional elements of HQ MAC.

Captain Wayne Stanberry (HQ MAC/XPSR) is also an experienced airlifter who is a graduate of the Air Force Institute of Technology, Graduate Strategic and Tactical Sciences program (class GST 82M). Capt. Stanberry used the SLAM language to simulate MAC base level maintenance for his thesis.

Both Maj Dillard and Capt Stanberry have reviewed the methods and the model used in this research project. Both have validated the "reasonableness" of the major assumptions via telephone, personal meeting, or both. Maj Dillard and Capt Stanberry both assert that the model and methodology developed in this research effort are adequate for the scope and purpose of this study.

In addition to the expertise they provided, the MAC analysts provided regulations and data which were used in establishing most of the model parameters. The list of cargo requirements used as input data for this thesis project is the same list used by their office for unclassified studies (Ref 7). Numerous discussions were conducted with MAC personnel by telephone and a meeting was held in January to assist in the validation process. The agenda items discussed were as follows:

1. A line-by-line review of the SLAM network and its program statements.
2. A line-by-line review of the FORTRAN program inserts.
3. Validation of stated assumptions.

Based upon this meeting, modifications to the then existing model were made. A second meeting is planned with these individuals prior to the thesis defense for final validation purposes.

In seeking additional data points for face validation of the model, the authors contacted Captain George G. London, Jr., an Aircraft Project Manager in the Foreign Technology Division (FTD/SDNS) at Wright-Patterson Air Force Base. Capt London served as a C-141 Aircraft Commander and Instructor Pilot from 1977-1981 at McGuire Air Force Base. During this time, he was involved in mission planning and coordination while assigned to the Wing Command Post at McGuire; as well as being involved in surge airlift deployments to Yemen, Zaire, and Jonestown, Guyana. His experience and knowledge of the strategic airlift system warranted his inclusion in the validation process.

A two-hour presentation and discussion of the research project resulted in positive validation of the model and methodology. Particular attention was given to the aggregation of airfields by type of activity, the concept of "setup" time as used herein, and the ground time distributions used in the SLAM network. Capt London expressed an instinctive feeling that the C-141 upload and download ground times were overstated, but allowed the possibility that a sustained surge operation could result in the ground times used here. The assumption of aircraft availability, as well as the runway/ramp representation of airfields, was considered quite acceptable (Ref 23).

Summary

This chapter has included a comprehensive review of the SLAM network and the structural model used in its development. In addition, the FORTRAN program inserts and discrete event inserts were also explained. The operation of the Scheduler routine was covered indepth since this area represented the central justification for the thesis effort. Furthermore, the verification and validation of the SLAM model phases of the model development were addressed. The results of the experiment and analysis are presented in Chapter V.

V. Experimental Results

Introduction

This chapter reports the results obtained by controlling aircraft allocation and cargo dispatching rules in the simulation of a contingency airlift to Southwest Asia. As reported in the section on experimental design, fifteen policies were evaluated and thirty replications were run for each policy (Table II). The output data were scaled and used to compute a scalar scoring function (SSF) for each policy, which was then used to make relative comparisons among the policies. Finally, a sensitivity analysis was performed to evaluate the effects of changes in the weight (importance) assigned to the response variables.

Calculations

In an effort to improve the readability of the data and analysis, a summary of the calculations used in data reduction is provided here. Scaling of response variables is considered first. (Complete explanations are available in Chapter 3 under Methodology.)

$$\text{Unit Closures} = \frac{\text{Total unit closures}}{\text{Total unit requirements}}$$

$$\text{Cargo Tardiness} = \frac{(\text{ Tardiness of delivered cargo} + \text{Tardiness of undelivered cargo})}{\text{Maximum possible tardiness}}$$

TABLE II

SCHEDULING POLICIES

		Aircraft Allocation Rules		
		C-5 Outsize	C-5 Outsize Oversize	C-5 Outsize oversize Bulk
Cargo Priority Rules	Aircraft Preference	1	2	3
	EDD	4	5	6
	LWT	7	8	9
	SWT	10	11	12
	Slack Operation	13	14	15

Note: Although only C-5s are specified above in allocation rules, C-141s are launched for oversize, bulk, or passenger airlift requirements for each rule.

$$\text{Cargo Deliveries} = \frac{\text{Total cargo delivered}}{\text{Total cargo available}}$$

The total unit closures are computed by tallying the sum of {0,1} indicator variables. The tardiness of each late shipment (ton-days) is summed with the tardiness of undelivered cargo at the end of the measurement period. Total cargo delivered is the sum of cargo delivered on time and cargo delivered late (tons). Each of the ratios calculated above is dimensionless, and lies between zero and one (inclusive).

Customer needs were represented by the importance placed on each response variable. Each variable was rated separately on a scale of zero (least important) to one (most important). The initial ratings were: closures (0.8), tardiness (0.6), and deliveries (0.2). Relative weights were calculated so that the sum of the weights was one.

Variable	Relative Weight
Closures (C)	$WT1 = \frac{0.8}{0.8 + 0.6 + 0.2} = 0.5$
Tardiness (T)	$WT2 = \frac{0.6}{0.8 + 0.6 + 0.2} = 0.375$
Deliveries (D)	$WT3 = \frac{0.2}{0.8 + 0.6 + 0.2} = 0.125$

The ratios calculated for the closure(C) and delivery(D) variables are multiplied by their relative weights in the SSF. In order to make zero the minimum value of the function (and one the maximum value), the tardiness(T) ratio is subtracted from one before being multiplied by its relative weight. Then the scoring function:

$$SSF = (WT1)(C)+(WT2)(1-T)+(WT3)(D)$$

is a scalar value in the range [0,1]. The actual and scaled values of the response variables for each policy are tabulated in Table III.

The recalculation of relative weights for sensitivity analysis was accomplished by arbitrarily assigning a relative weight to one variable and computing a relative weight for the other two variables in proportion to the importance initially assigned to them.

Example: Let $WT2 = 0.5$

$$\text{then} \quad WT1 = 0.5 \; \frac{0.8}{(0.8 + 0.2)} = 0.4$$

$$WT3 = 0.5 \; \frac{0.2}{(0.8 + 0.2)} = 0.1$$

This simulates a situational change in the relative importance of the response variables, and the new score of a particular policy may be calculated from the same dimensionless ratios used with the original weights.

The scaled response values were used in the objective function to compute an SSF for each scheduling policy. For the purpose of sensitivity analysis, it was presumed that situations could occur in which the weights assigned to the response variables would be different from those obtained in this study. Since the unit closures variable had the highest

TABLE III

ACTUAL AND SCALED OUTPUT DATA

POLICY	MEAN CLOSURES	CLOSURE RATIO	MEAN TARDINESS	TARDINESS RATIO	MEAN DELIVERIES	DELIVERY RATIO
1	18.77	0.1104	53,635	0.3570	22,590	0.6008
2	20.09	0.1182	49,007	0.3262	23,714	0.6307
3	23.70	0.1394	47,114	0.3136	23,782	0.6325
4	49.33	0.2902	66,345	0.4416	18,856	0.5015
5	95.80	0.5635	21,244	0.1414	30,663	0.8155
6	97.00	0.5706	21,424	0.1426	30,678	0.8159
7	0.24	0.0014	91,960	0.6121	13,893	0.3695
8	4.27	0.0251	59,764	0.3978	20,631	0.5487
9	10.47	0.0616	60,065	0.3998	20,627	0.5486
10	106.44	0.6261	71,573	0.4764	18,823	0.5006
11	123.86	0.7286	61,838	0.4116	22,804	0.6065
12	126.60	0.7447	60,786	0.4046	22,067	0.5869
13	47.57	0.2798	69,440	0.4622	18,917	0.5031
14	88.50	0.5206	30,543	0.2033	29,727	0.7906
15	91.27	0.5369	31,099	0.2070	29,430	0.7827

relative weight of 0.5, it was decided to arbitrarily assign a relative weight of 0.5 to each of the variables in turn. The remaining variables were then assigned relative weights as shown above. A final case used equal weights for each response variable. The resulting SSF values obtained for each policy are listed in Table IV according to the following schema:

SSF	WT1	WT2	WT3
1	0.5	0.375	0.125
2	0.4	0.5	0.1
3	0.286	0.214	0.5
4	0.333	0.333	0.333

Mean scoring functions were computed for each policy under each set of weights listed above. The next step was to determine whether there were significant differences among the scores resulting from various policies. For each set of scores, one-way analysis of variance was used to test the following hypotheses at a 95% confidence level:

H_0: Scalar scoring functions are the same for all policies.

H_A: Scalar scoring functions are not the same for all policies.

Standard procedures available in the Statistical Package for the Social Sciences (SPSS) were used for this test. While differences in mean SSF for various polices were indicated, the ratio of maximum SSF variance to minimum SSF variance (over all policies), as well as the ratio of maximum variance to the sum

TABLE IV

SCALAR SCORING FUNCTIONS (SSF)

POLICY	SSF1	SSF2	SSF3	SSF4
1	0.3162	0.4257	0.4696	0.4514
2	0.3315	0.4473	0.4933	0.4742
3	0.3365	0.4622	0.5030	0.4861
4	0.2721	0.4454	0.4533	0.4500
5	0.4239	0.7362	0.7526	0.7459
6	0.4235	0.7385	0.7546	0.7480
7	0.1916	0.2314	0.2681	0.2529
8	0.2944	0.3660	0.4104	0.3920
9	0.2937	0.3796	0.4203	0.4035
10	0.2589	0.5623	0.5414	0.5501
11	0.2965	0.6463	0.6375	0.6412
12	0.2967	0.6543	0.6339	0.6424
13	0.2646	0.4311	0.4467	0.4402
14	0.3976	0.6856	0.7147	0.7026
15	0.3952	0.6895	0.7146	0.7042

of variances, indicated that the variance of SSF values was not the same for all policies.

Since the tests used rely on equal group variances, the indication of unequal variances was cause for closer examination. A discussion on the effects of unequal variances was found in a standard text. It states that "...the Scheffe multiple comparison procedure ... is not affected to any substantial extent by unequal variances if the sample sizes are equal." (Ref 27:514)

To get a visual indication of the range of policy scores, the mean scores for the highest-ranking policies were plotted on a number line with bands enclosing three standard deviations from each mean (Figures 15 - 16). These graphs also indicated that the group differences shown by standard range tests have some validity. Given the above arguments, it was decided that the results of the standard range tests could be considered valid for the purposes of this project.

The stochastic elements of the model introduce some error in estimating the mean value of the SSF for each policy. The range tests that were used assume that the error terms for each factor level (i.e., policy) are normally distributed; however, the tests are not sensitive to nonnormality of error terms in a fixed effects model when the sample size is not extremely small (Ref 27). The experimental model uses fixed effects (factor levels are controlled). Additionally, the use of 30 sample

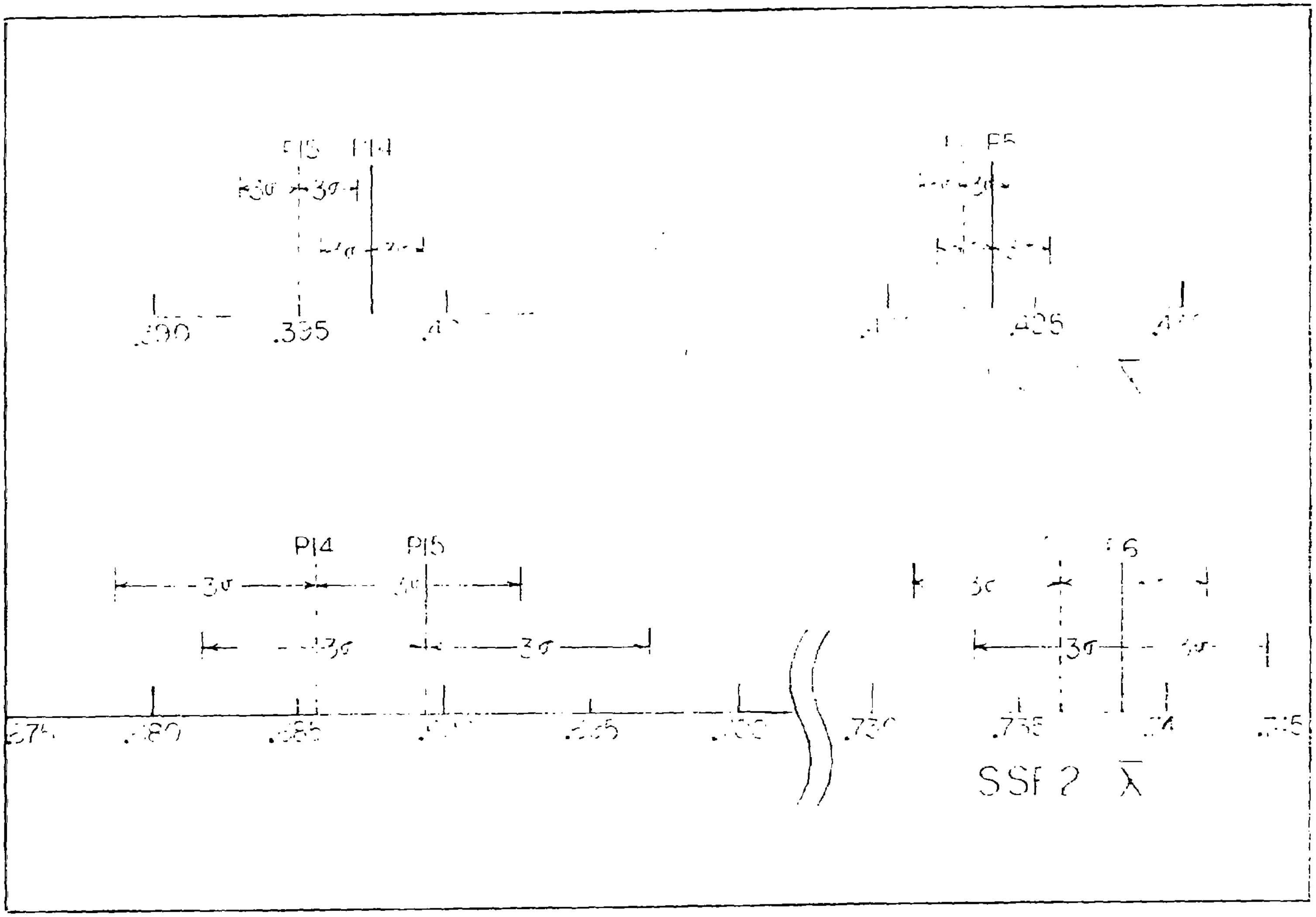


Figure 15 SSF Sample Mean Graphs

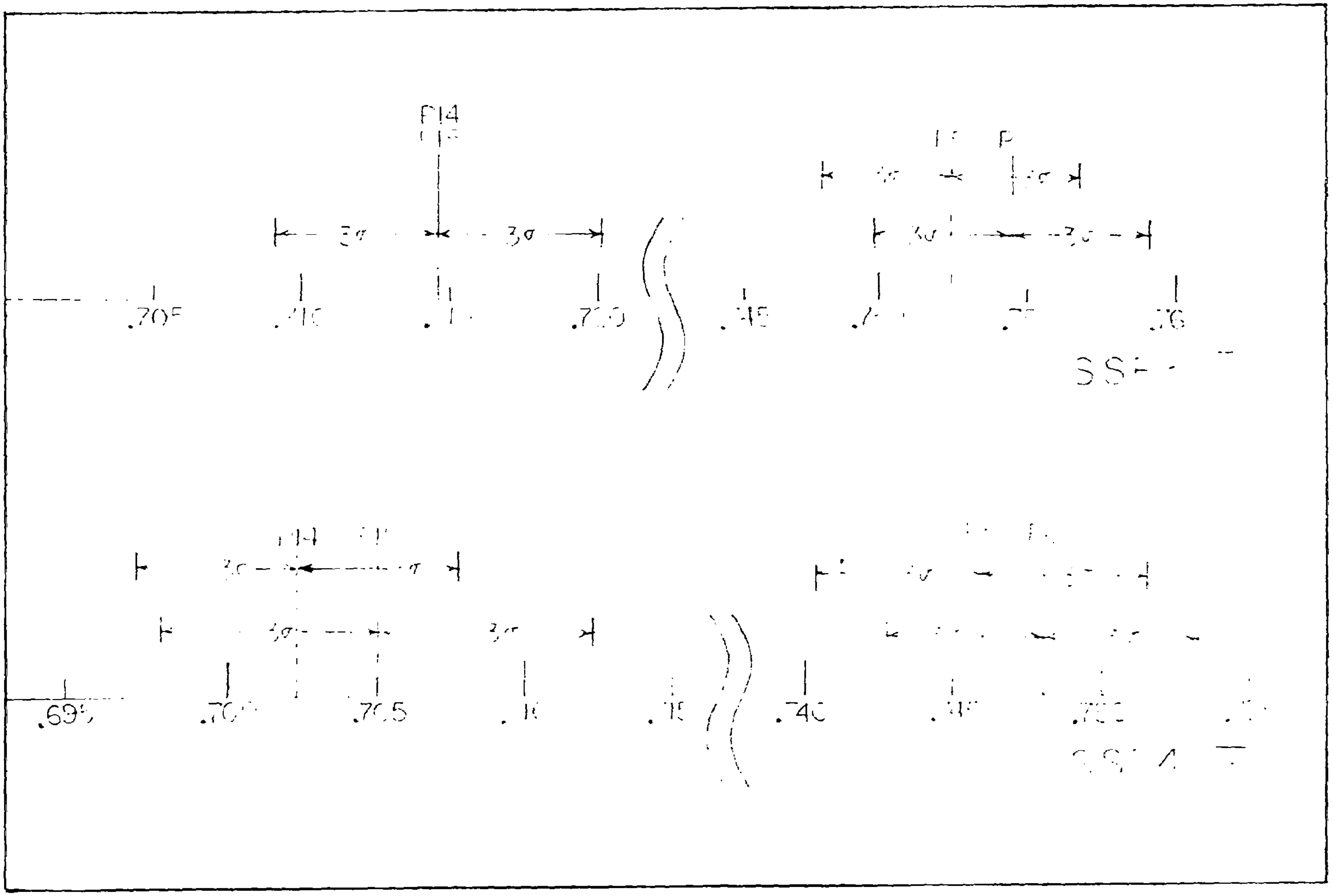


Figure 16 SSF Sample Mean Graphs

points for each factor level is usually considered a reasonable sample size.

Statistical Tests

With the evidence presented above, it is considered reasonable to apply the results of the standard tests used for the purpose of this study. These results indicate that distinctions among the fifteen policies tested may be made based on certain levels of both experimental factors.

Regarding the allocation of C-5 aircraft, for example, three levels of this factor were combined with one dispatching rule (one level of the second factor) used to prioritize cargo requirements. These three combinations are referred to as three separate policies (see Table II). Multiple range tests were used to statistically compare the policies based on the mean SSF value for each. Results of the multiple range tests used in the one-way analysis of variance are shown in Tables V-VIII. Each subset represents a group of policies whose mean SSF values are not significantly different (with confidence coefficient .95). In each column, the groups are ranked from highest mean to lowest mean according to the results of a particular test. The reader is referred to the SPSS manual (Ref 28) and to Neter and Wasserman (Ref 27) for further discussion of the multiple range tests. Each separate table represents a different combination of weights in the scalar scoring function.

TABLE V

SSF1 MULTIPLE RANGE TEST BY SUBSET RANK

SUBSET	LSD	S-N-K	TUKEY-HSD	SCHEFFE
1	5,6	5,6	5,6	5,6
2	14,15	14,15	14,15	14,15
3	3	3	2,3	2,3
4	2	2	1	1
5	1	1	8,9,11,12	8,9,11,12
6	8,11,12	8,9,11,12	4	4,13
7	8,9	4	13	10,13
8	4	13	10	7
9	13	10	7	--
10	10	7	--	--
11	7	--	--	--

TABLE VI

SSF2 MULTIPLE RANGE TEST BY SUBSET RANK

SUBSET	LSD	S-N-K	TUKEY-HSD	SCHEFFE
1	5,6	5,6	5,6	5,6
2	14,15	14,15	14,15	14,15
3	12	12	11,12	11,12
4	11	11	10	10
5	10	10	3	3
6	3	3	2,4	2,4
7	2,4	2,4	1,13	1,13
8	13	13	9	9
9	1	1	8	8
10	9	9	7	7
11	8	8	--	--
12	7	7	--	--

TABLE VII

SSF3 MULTIPLE RANGE TEST BY SUBSET RANK

SUBSET	LSD	S-N-K	TUKEY-HSD	SCHEFFE
1	5,6	5,6	5,6	5,6
2	14,15	14,15	14,15	14,15
3	11,12	11,12	11,12	11,12
4	10	10	10	10
5	3	3	2,3	2,3
6	2	2	1	1
7	1	1	4,13	4,13
8	4	4	8,9	8,9
9	13	13	7	7
10	9	9	--	--
11	8	8	--	--
12	7	7	--	--

TABLE VIII

SSF4 MULTIPLE RANGE TEST BY SUBSET RANK

SUBSET	LSD	S-N-K	TUKEY-HSD	SCHEFFE
1	5,6	5,6	5,6	5,6
2	14,15	14,15	14,15	14,15
3	11,12	11,12	11,12	11,12
4	10	10	10	10
5	3	3	3	2,3
6	2	2	2	1,4,13
7	1,4	1,4	1,4	8,9
8	13	13	13	7
9	9	9	9	--
10	8	8	8	--
11	7	7	7	--

In general terms, a policy reserving C-5 aircraft for outsize cargo scored lower than the other two C-5 allocation policies when combined with the same cargo selection (priority) rule. Eliminating this set of policies (Policies 1,4,7,10,13) from consideration, then, the Earliest Due Date rule ranked highest (Policies 5 & 6), followed by the Slack per Operation rule (Policies 14 & 15). The range tests fail to distinguish between policies five and six, which have the highest mean values for each set of scoring functions calculated. Policies fourteen and fifteen form another group which consistently ranks second for each set of scoring functions. Below this group the rankings tend to vary from one scoring function to the next, except for policy seven, which has the lowest mean in each case. These results indicate that policies five and six dominate all other policies under the conditions of this experiment. The term "dominate", as used here, means that the scores of these two policies are never less than those of other policies--and they are sometimes (or always) better. Similarly, policies fourteen and fifteen dominate all policies except five and six. Policy seven is dominated by the other fourteen policies.

It should be restated here that the manner in which the scalar scoring functions were calculated reflects an implicit assumption that the value curves for the experimental response variables are linear. This means, for example, that the value of getting half of the cargo delivered is 0.5. While that assumption may be valid, the point to be made is that the scalar scoring functions calculated in this study are useful for the relative comparisons made above, but may not be an accurate indication of how well a certain policy would perform when actual values (rather than linear scaling) are used to calculate the scalar scoring function.

This concludes the presentation of experimental results. Conclusions drawn from these results and some recommendations for further research are presented in Chapter VI.

VI. Conclusions and Recommendations

Conclusions

This research project was undertaken to develop and test a method for using classical scheduling techniques in a strategic airlift simulation. The specific issues selected were: (1) allocation of C-5 aircraft to various types of cargo, and (2) adaptation of priority dispatching rules from job shop problems to strategic airlift. The general conclusion of the authors is that a useful method has been developed in this project, and the method indicates that certain combinations of priority dispatching and C-5 allocation rules offer improvement over procedures currently in use.

The use of adapted job shop scheduling rules to prioritize cargo requirements could increase cargo throughput and decrease cargo tardiness, compared with the results achieved by prioritizing requirements according to cargo type. In the course of this project, cargo priorities were established by other predetermined attributes (e.g., due date, weight), as well as a computed attribute (slack per operation) which was assigned when the cargo entered the airlift system. Both the due date and slack per operation rules consistently performed better than priorities assigned by cargo type.

Policies that reserve C-5 aircraft for missions with outsize cargo produce lower system performance levels (as measured by the scalar scoring function) than policies that release the C-5 for missions with oversize and/or bulk cargo, but no outsize cargo. In most cases, little difference was observed between the scores for policies that released C-5s for both oversize and bulk cargo under a single cargo priority rule. On the other hand, the score of the policy using that same cargo priority rule, but reserving C-5s for missions without outsize cargo, was always significantly lower.

Cargo priority rules have a greater effect on policy scores than do C-5 allocation rules. This conclusion is based on the observation that equal rankings of two policies which used the same priority rule occurred frequently, while there were no equal rankings of policies using the same aircraft allocation rule.

Assigning highest priority to cargo requirements having the largest cargo weight resulted in lower performance scores than any of the other priority rules, in most cases. Policy 7, which combined "largest weight" priority with "reserved for outsize" C-5 allocation, ranked lowest in every case tested.

The ranking of scheduling policies was rather robust (not sensitive) to changes in the weights assigned to the response variables. Robustness was tested in a scenario-oriented sensitivity analysis. The policies ranked first and second were ranked the same in all cases, as was the policy ranked last. While there were shifts in other rankings among subgroups, there was almost no inversion of policy rank.

In a more gener . case, the conclusions to be drawn from the project are that: (1) job shop priority dispatching rules are applicable to cargo selection procedures in strategic airlift; (2) the use of a scalar scoring function permits straightforward comparisons among airlift scheduling policies; and, (3) an aggregate simulation model is adequate for comparing scheduling policies with regard to the response variables under consideration.

Recommendations

The following recommendations are submitted as a result of this research:

1. The experimental model could be expanded from its pipeline configuration to permit alternate routings. This would suggest working with maximal flow and line balancing techniques to improve system performance.

2. Heuristic combinations of priority dispatching rules should be attempted with the model used in this experiment. For example, slack per operation or smallest weight criteria could be used as the basic rule for selecting cargo; but the basic rule could be suspended, when cargo requirements in the queue reached the maximum waiting time, until those requirements were serviced. Other priority rules and heuristic combinations that were not tested in this study may yield even greater improvements in system performance.

3. Multiattribute utility theory should be applied in a more rigorous manner to the problem of airlift performance measurement. Adequate investigation of measurable attributes, and determination of the true utility curves for those attributes, would permit formulation of a utility function which would yield a true measure of merit (rather than a scalar scoring of policies).

4. Combinations of cargo requirements from different onload points (or destined for offload points) may improve the overall performance of the system or the performance of certain policies. This could be studied by adding a new dimension to the search for filler cargo and modifying a user function to account for an increase in the loading (or unloading) time.

5. The design of the experimental model should permit some attempts at optimizing a schedule for the entire simulation period, or for smaller periods with a reasonably large quantity of cargo. Analytical work on such problems as bin packing and set partitioning could be helpful in this regard.

6. Finally, priority dispatching rules for cargo selection should be tested in a more detailed strategic airlift model for a scenario the same as, or similar to, that simulated in this experiment. Aggregation of certain airlift system parameters, while considered reasonable in making the desired

comparisons, may mask interaction which would deny achievement of the high performance scores observed in this experiment.

Bibliography

1. AFM 1-1. Functions and Basic Doctrine of the United States Air Force. Washington, D.C.: Headquarters U.S. Air Force, 14 February, 1979.

2. AFR 76-2. Military Airlift Planning Factors. Washington, D.C.: Headquarters U.S. Air Force, 11 June 1979.

3. Armstrong, Gerald R. "The Single and Multiple Vehicle Pickup and Delivery Problem: Exact and Heuristic Algorithms." Unpublished Doctoral Dissertation. University of Tennessee, Knoxville, Tennessee, June 1981.

4. Baker, Kenneth R. Introduction to Sequencing and Scheduling. New York: John Wiley & Sons, Inc., 1974.

5. Brown, Harold. Department of Defense Annual Report, Fiscal Year 1981. Washington, D.C.: U.S. Government Printing Office, 1980.

6. Carpenter, Maj Billie R. "Bright Star, MAC-First In, Last Out, "Airlift Operations Review, 4 (2): 14-16 (April-June 1982).

7. Computer Generated Test Data. HQ MAC/XPSR, Scott AFB, IL, November 1982.

8. Conway, Richard W., William L. Maxwell and Louis W. Miller. Theory of Scheduling. Reading, Massachusetts: Addison-Wesley Publishing Company, 1967.

9. Conway, Richard W. "Priority Dispatching and Job Lateness in a Job Shop," Journal of Industrial Engineering, 16 (4): 228-237 (July-August 1965).

10. -----. "Priority Dispatching and Work-in-Process Inventory in a Job Shop, "Journal of Industrial Engineering, 16 (2): 123-130 (March-April 1965).

11. Day, James E.and Michael P. Hottenstein. "Review of Sequencing Research," Naval Research Logistics Quarterly, 17 (1): 11-39 (March 1970).

12. Department of Defense. Single Manager Operating Agency for Airlift Service. DOD Directive 5160.2. Washington D.C.: U.S. Government Printing Office, 17 October, 1973.

13. Dewispelare, Aaron R. and Andrew P. Sage. "On Combined Multiple Objective Optimization Theory and Multiple Attribute Utility Theory For Evaluation and Choicemaking," Journal of Large Scale Systems, 2: 1-19 (1981).

14. Digby, James. The Emerging American Strategy; Application to Southwest Asia. Report on research in international security and arms control. Santa Monica, California: Rand Corp., May 1981, (N-1700-FF).

15. Dillard, Maj Charles. Chief, Simulation Applications, Operations Research Division, HQ MAC, Scott AFB, IL. (conference). 8 January 1983.

16. -----. Operations Research Division, HQ MAC, Scott AFB, IL. (telephone conversation). 3 November 1982.

17. Edwards, Ward. "How To Use Multiattribute Utility Measurement For Social Decisionmaking," IEEE Transactions on Systems, Man, and Cypernetics, SMC-7 (5): 326-340 (May 1977).

18. Elmaghraby, Salah E. "The Sequencing of 'Related' Jobs," Naval Research Logistics Quarterly, 15 (1): 205-232 (March 1968).

19. Fabyanic, Thomas A. "Conceptual Planning and the Rapid Deployment Joint Task Force," Armed Forces and Society, 7 (3): 343-365 (Spring 1981).

20. Gere, William S. "Heuristics in Job Shop Scheduling," Management Science, 13 (3): 167-190 (November 1966).

21. Greenberger, Martin, Matthew A.Crenson and Brian L. Crissey. Models In The Policy Process. New York: Russell SACE Foundation, 1976.

22. JDA Staff. "Joint Deployment Agency, A Deployment Broker," Airlift Operations Review, 3 (1): 8-13 (Jan-Mar 1981).

23. London, Capt George G. Aircraft Project Manager, Foreign Technology Wright Patterson AFB, OH. (personal interview) 19 February 1983.

24. Moorer, Adm Thomas H. and Alvin J. Cottrell. "The Search For U.S. Bases in the Indian Ocean: A Last Chance," Strategic Review, 8 (2): 30-38 (Spring 1980).

25. Montgomery, Robin N. "The Questionable U.S. Strategy for the Middle East, "National Defense, 65 (368): 64-68+ (May-June 1981).

26. Ness, L., Zologa S. et al. Rapid Deployment Force. Special study. Greenwich, Connecticut: DMS Inc., 1980.

27. Neter, John and William Wasserman. Applied Linear Statistical Models. Homewood, Illinois: Richard D. Irwin, Inc., 1974.

28. Nie, Norman H. et al. SPSS: Statistical Package for the Social Sciences. Second Edition. New York: McGraw-Hill Book Company, 1975.

29. Pilcsh, Lt Col Thomas D. "The C-X Requirement: Perspective on Airlift," Airlift Operations Review, 3 (1): 8-17 (Jan-Mar 1981).

30. Pritsker, A. Alan B. and Claude D. Pegden. Introduction to Simulation and SLAM. New York: John Wiley and Sons, 1979.

31. Rowe A.J. "Toward a Theory of Scheduling," Journal of Industrial Engineering, 11 (2): 125-136 (Mar-Apr 1960). 1960).

32. Shannon, Robert E. Systems Simulation, The Art and Science. Englewoods Cliffs, N.J: Prentice-Hall Inc., 1975.

33. Trilling, Donald R. "Job Shop Simulation of Orders that are Networks," Journal of Industrial Engineering, 17 (2) (February 1966).

34. Ulsamer, Edgar. "The Pentagon Is Gearing Up for the '80s," Air Force Magazine, 64 (5): 50-56 (May 1981).

Appendix A

SLAM Description

SLAM (Simulation Language for Alternative Modeling) is a FORTRAN-based simulation language which allows simulation models to be created in three world views:

1. Network
2. Discrete
3. Continuous

A SLAM model consists of a set of interconnected symbols that describe the operation under study. SLAM provides network symbols (see Figures A-1 to A-12) which can be used to build models and which can be translated into input statements for computer processing. SLAM symbols and input statements used for the SLAM Simulation Model described in Chapter IV are explained here. The following symbols, statement formats, and definitions are taken from Introduction to Simulation and SLAM by A. Alan B. Pritsker and Claude D. Pegden (1979), pp. 435-551.

Network Element	Figure
1. ALTER NODE	A-1
2. ASSIGN NODE	A-2
3. AWAIT NODE	A-3
4. CREATE NODE	A-4
5. ENTER NODE	A-5
6. EVENT NODE	A-6
7. FREE NODE	A-7
8. GOON NODE	A-8

NODE TYPE: ALTER

FUNCTION: The ALTER node changes the capacity of resource RLBL by CC units. In the case where the capacity is decreased below current utilization, the excess capacity is destroyed as it becomes freed. The capacity can be reduced to a minimum of zero with additional reductions having no effect. At each release, a maximum of M emanating activities are initiated.

INPUT FORMAT: ALTER,RLBL/CC,M;

SPECIFICATIONS:

ENTRY	OPTIONS
RLBL	maximum of 8 characters beginning with an alphabetical character.
CC	SLAM variable, SLAM random variable, or a constant.
M	positive integer.

SYMBOL:

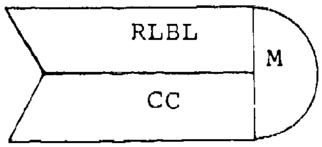


Fig. A-1. ALTER Node Desciption Summary

NODE TYPE: ASSIGN

FUNCTION: The ASSIGN Node is used to assign values to SLAM variables (VAR) at each arrival of an entity to the node. A maximum of M emanating activities are initiated.

INPUT FORMAT: ASSIGN, VAR=value, VAR=value,...,M;

SPECIFICATIONS:

ENTRY	OPTIONS
VAR	ATRIB (INDEX),XX(INDEX),II, where INDEX is a positive integer or SLAM variable II.
value	an expression containing constants, SLAM variables, or SLAM random variables.
M	positive integer.

SYMBOL:

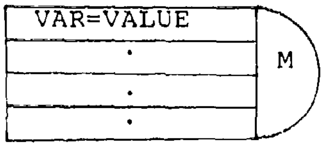


Fig. A-2. ASSIGN Node Description Summary

NODE TYPE:	AWAIT
FUNCTION:	In the RESOURCE node, the AWAIT node delays an entity in file IFL until UR units of resource RLBL are available. The entity then seizes UR units of RLBL. The ALLOC(IAC) option provides user written resource allocation capability ("and" and "or" resource allocation). Subroutine ALLOC(IAC,IFAG) is called when an entity arrives at the node. Subroutine ALLOC will also be called when a resource required by this AWAIT node becomes available and there is an entity at the AWAIT node that might use the resource.
INPUT FORMAT:	AWAIT (IFL,QC), RLBL/UR or ALLOC(IAC),,M;

SPECIFICATIONS:

ENTRY	OPTIONS
IFL	Integer between 1 and MFIL (maximum file number).
QC	AWAIT queue capacity
RLBL	Resource label, maximum of 8 characters beginning with an alphabetic character.
UR	positive SLAM variable, SLAM random variable, or constant.
ALLOC(IAC)	calls subroutine ALLOC with argument number IAC.
M	positive integer.

SYMBOL:

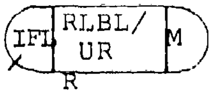


Fig A-3. AWAIT Node Description Summary

NODE TYPE: CREATE

FUNCTION: The CREATE node is used to generate entities within the network. The node is released initially at time TF and thereafter according to the specified time between creations, TBC, up to a maximum of MC releases. At each release a maximum of M emanating activities are initiated.

INPUT FORMAT: CREATE,TBC,TF,MA,MC,M;

SPECIFICATIONS:

ENTRY	OPTIONS
TBC	constant, SLAM variable, or SLAM random variable.
TF	constant.
MA	positive integer.
MC	positive integer.
M	positive integer.

SYMBOL:

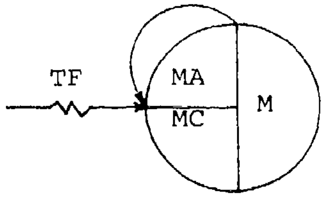


Fig. A-4. CREATE Node Description Summary

NODE TYPE: ENTER

FUNCTION: The ENTER node is provided to permit the user to enter an entity into the network from a user-written event routine. The node is released at each entity arrival and at each user call to subroutine ENTER(NUM). A maximum of M emanating activities are initiated at each release.

INPUT FORMAT: ENTER, NUM,M;

SPECIFICATIONS:

ENTRY	OPTIONS
NUM	positive integer.
M	positive integer.

SYMBOL:

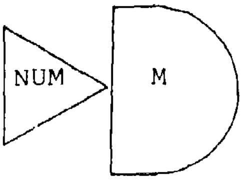


Fig A-5. ENTER Node Description Summary

NODE TYPE: EVENT

FUNCTION: The EVENT node causes subroutine EVENT to be called with event code JEVNT at each entity arrival. This allows the user to model functions for which a standard node is not provided. A maximum of M emanating activities are initiated.

INPUT FORMAT: EVENT, JEVNT, M;

SPECIFICATIONS:

ENTRY	OPTIONS
JEVNT	positive integer.
M	positive integer.

SYMBOL:

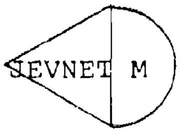


Fig. A-6. EVENT Node Description Summary

NODE TYPE: FREE

FUNCTION: The FREE node releases UF units of resource RLBL. The resource is made available to waiting entities according to the order of the wait files specified in the resource statement. A maximum of M emanating activities are initiated.

INPUT FORMAT: FREE, RLBL/UF, M;

SPECIFICATIONS:

ENTRY	OPTIONS
RLBL	maximum of 8 characters beginning with an alphabetic character.
UF	positive SLAM variable, SLAM random variable, or constant.
M	positive integer.

SYMBOL:

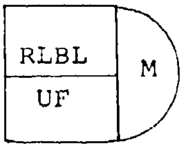


Fig. A-7. FREE Node Description Summary

NODE TYPE: GOON

FUNCTION: The GOON node provides a continuation node where every entering entity passes directly through the node.

INPUT FORMAT: GOON,M;

SYMBOL:

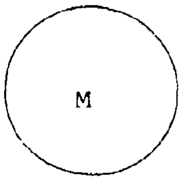


Fig. A-8. GOON Node Description Summary

NODE TYPE: TERMINATE

FUNCTION: The TERMINATE node is used to destroy entities and/or terminate the simulation. All incoming entities to a TERMINATE node are destroyed. The arrival of the TCth entity causes a simulation run to be terminated.

INPUT FORMAT: TERMINATE,TC;

SPECIFICATIONS:

ENTRY	OPTIONS
TC	postive integer.

SYMBOL:

Fig. A-9. TERMINATE Node Description Summary

ACTIVITY TYPE: REGULAR

FUNCTION: A REGULAR activity is any activity enamating from a node other than a QUEUE node. The REGULAR activity is used to delay entities by a specified duration, perform conditional/probabilistic testing, and to route entities to non-sequential nodes.

INPUT FORMAT: ACTIVITY/A,duration,PROB or COND,NLBL;

SPECIFICATIONS:

ENTRY	OPTIONS
A	positive integer.
duration	constant, SLAM variable, SLAM random variable.
PROB or	probability: constant between 0 and 1.
COND	condition: value .OPERATOR. value where value is a constant, SLAM variable, or SLAM random variable and OPERATOR is LT, LE, EQ, GE, GT, or NE.
NLBL	the label of a labeled node which is at the end of the activity.

SYMBOL:

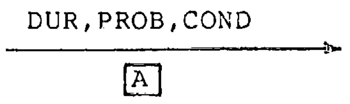


Fig. A-10. REGULAR Activity Description Summary

ACTIVITY TYPE: SERVICE

FUNCTION: The SERVICE activity is any activity emanating from a QUEUE node. The SERVICE activity is used in conjunction with the QUEUE node.

INPUT FORMAT: ACTIVITY(N)/A,duration,PROB,NLBL;

SPECIFICATIONS:

ENTRY	OPTIONS
N	positive integer.
A	positive integer between 1 and 50.
duration	constant, SLAM variable, SLAM random variable.
probability-constant between 0 and 1.	
NLBL	label of a labeled node.

SYMBOL:

Fig. A-11. SERVICE Activity Description Summary

BLOCK TYPE: RESOURCE

FUNCTION: A RESOURCE block defines a resource by its label RLBL and its initial capacity or availability IRC. The file numbers, IFLs, associated with AWAIT nodes are where entities requesting units of the resources are queued. The IFLs are listed in the order in which it is desired to allocate the units of the RESOURCE where they are made available.

INPUT FORMAT: RESOURCE/RLBL(IRC),IFLs;

SPECIFICATIONS:

ENTRY	OPTIONS
RLBL	maximum of 8 characters beginning with an alphabetic character.
IRC	positive integer.
IFLs	integers between 1 and MFIL.

SYMBOL:

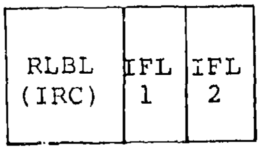


Fig. A-12. RESOURCE Block Description Summary.

Appendix B

Network Program Statements

```
GEN,HAMLTN & POE,NET1,2/19/83,30,,N,,,,5;
LIMITS,21,15,1500;
PRIORITY/1,LVF(5)/2,LVF(5)/3,LVF(5)/4,LVF(5);
NETWORK;
      RESOURCE/C5(10),5;
      RESOURCE/C141(18),7;
      RESOURCE/RW88(6),6,19;
      RESOURCE/RP67(459),9;
      RESOURCE/RW67(1),10,9;
      RESOURCE/RP89(528),11;
      RESOURCE/RW89(2),12,11;
      RESOURCE/RW14(6),8,20;
      RESOURCE/RP90(273),17;
      RESOURCE/RW90(1),18,17;
      RESOURCE/RP71(817),15;
      RESOURCE/RW71(2),16,15;
      RESOURCE/RPXX(45),13;
      RESOURCE/RWXX(4),14,13;
;C5 AND C141 ARRIVALS
;-----------
GENA  CREATE,1,0,,48,1;
      ALTER,C5/1,1;
      ALTER,C141/4,1;
      TERMINATE;
;------------------------------
TIMR  CREATE,24,0,,11,1;
      ACT,0.01;
CGEN  EVENT,1,1;
      TERMINATE;
;      -------
      CREATE,2,,,132,1;
      ACT,0.02;
SCHD  EVENT,2,1;
      TERMINATE;
;------------------------------
      ENTER,1,1;
      ASSIGN,ATRIB(12)=1.0,1;
      ACT,,,BIG;
      ENTER,2,1;
      ASSIGN,ATRIB(12)=2.0,1;
      ACT,,,SMAL;
BIG   AWAIT(5),C5/1,1;
      ACT,UNFRM(2.0,14.0),,HBC5;
;C5 DEPARTS HOME STATION
;----------------------
HBC5  AWAIT(6),RW88/1,1;
      ACT,0.0833;
      FREE,RW88/1,1;
      ACT,USERF(1);
      GOON,1;
;C5 FLY TO LAJES
;---------------
      ACT,4.,,LD67;
;CARGO AWAITS C141
;-----------------
```

```
SMAL  AWAIT(7),C141/1,1;
      ACT,UNFRM(2.0,14.0),,HB14;
HB14  AWAIT(8),RW14/1,1;
      ACT,0.0833;
      FREE,RW14/1,1;
      ACT,USERF(1);
      GOON,1;
;C141 FLY TO LAJES
;---------------
      ACT,5.,,LD67;
LD67  AWAIT(9),ALLOC(1),,1;
      ACT,0.167;
      FREE,RW67/1,1;
      ACT,USERF(2),ATRIB(12).EQ.1,B67R;
      ACT,USERF(2),ATRIB(12).EQ.2,S67R;
B67R  FREE,RP67/17,1;
      ACT,,,T067;
S67R  FREE,RP67/9,1;
      ACT,,,T067;
T067  AWAIT(10),RW67/1,1;
      ACT,0.0833;
      FREE,RW67/1,1;
;C5 AND C141 FLY TO CAIRO
;----------------------
      ACT,7.23,ATRIB(12).EQ.1,LD89;
      ACT,7.6,ATRIB(12).EQ.2,LD89;
LD89  AWAIT(11),ALLOC(3),,1;
      ACT,0.167;
      FREE,RW89/1,1;
      ACT,USERF(2),ATRIB(12).EQ.1,B89R;
      ACT,USERF(2),ATRIB(12).EQ.2,S89R;
B89R  FREE,RP89/33,1;
      ACT,,,T089;
S89R  FREE,RP89/16,1;
      ACT,,,T089;
T089  AWAIT(12),RW89/1,1;
      ACT,0.0833;
      FREE,RW89/1,1;
;C5 AND C141 FLY TO DESTINATION
;----------------------------
      ACT,3.,ATRIB(12).EQ.1,LDXX;
      ACT,3.,ATRIB(12).EQ.2,LDXX;
LDXX  AWAIT(13),RWXX/1,1;
      ACT,0.167;
      FREE,RWXX/1,1;
      ACT,USERF(3),ATRIB(12).EQ.1,TOXX;
      ACT,USERF(3),ATRIB(12).EQ.2,TOXX;
TOXX  AWAIT(14),RWXX/1,1;
DLVR  EVENT,3,1;
      ACT,0.0833;
      FREE,RWXX/1,1;
;C5 AND C141 FLY TO JEDDAH
;----------------------
      ACT,2.,ATRIB(12).EQ.1,LD71;
      ACT,2.,ATRIB(12).EQ.2,LD71;
```

```
LD71  AWAIT(15),ALLOC(2),,1;
      ACT,0.167;
      FREE,RW71/1,1;
      ACT,USERF(2),ATRIB(12).EQ.1,B71R;
      ACT,USERF(2),ATRIB(12).EQ.2,S71R;
B71R  FREE,RP71/43,1;
      ACT,,,T071;
S71R  FREE,RP71/19,1;
      ACT,,,T071;
T071  AWAIT(16),RW71/1,1;
      ACT,0.0833;
      FREE,RW71/1,1;
;C5 AND C141 FLY TO PRESTWICK
;---------------------------
      ACT,7.3,ATRIB(12).EQ.1,LD90;
      ACT,7.7,ATRIB(12).EQ.2,LD90;
LD90  AWAIT(17),ALLOC(4),,1;
      ACT,0.167;
      FREE,RW90/1,1;
      ACT,USERF(2),ATRIB(12).EQ.1,B90R;
      ACT,USERF(2),ATRIB(12).EQ.2.,S90R;
B90R  FREE,RP90/21,1;
      ACT,,,T090;
S90R  FREE,RP90/13,1;
      ACT,,,T090;
T090  AWAIT(18),RW90/1,1;
      ACT,0.0833;
      FREE,RW90/1,1;
;C5 AND C141 FLY TO HOME BASES
;----------------------------
      ACT,7.,ATRIB(12).EQ.1,LD88;
      ACT,8.,ATRIB(12).EQ.2,LD14;
LD88  AWAIT(19),RW88/1,1;
      ACT,0.167;
      FREE,RW88/1,1;
      ACT,USERF(4);
      FREE,C5/1,1;
      TERMINATE;
LD14  AWAIT(20),RW14/1,1;
      ACT,0.167;
      FREE,RW14/1,1;
      ACT,USERF(4);
      FREE,C141/1,1;
      TERMINATE;
      ENDNETWORK;
INIT,0,350;
SIMULATE;
FIN;
XEOR
```

Appendix C

Definitions of User Support and Callable Subprograms of SLAM

SUBROUTINE COPY (NRANK, IFILE,A): copies the values of the attributes of an entry into the vector A. If NRANK is negative, then the entry with pointer-NRANK is to be copied.

SUBROUTINE ALLOC(IAC,IFLAG): user written allocation option specified for an AWAIT node. It is called when an entity arrives at a node and when a resource required by this AWAIT node becomes available and there is an entity at the AWAIT node that might use the resource. The user specifies which resources should prompt this call by the file specifications in the resource block. File operations are not permitted. The arguments to subroutine ALLOC are the user code specified on the network, IAC, and a flag, IFLAG, to inform the SLAM processor whether or not an allocation has been made.

FUNCTION DPROB (CPROB, VALUE, NVAL,IS): returns a sample from a user fefined discrete probability function with cumulative probabilities and associates values specified in arrays CPROB and VALUE with NVAL values using random stream IS.

FUNCTION DRAND (IS): returns a pseudo-random number obtained from random number stream IS.

SUBROUTINE ENTER (IN,A): releases ENTER node whose number is IN with an entity whose attribute values are in a vector A.

SUBROUTINE FILEM (IFILE,A): files an entry with attributes stored in A into file IFILE.

FUNCTION LOCATE (NRANK,IFILE): returns the pointer to the location of the entry whose rank is NRANK in file IFILE.

FUNCTION MMFE (IFILE): returns the pointer to first entry (rank 1) in file IFILE.

FUNCTION NFIND (NRANK,IFILE,JATT, MCODE, XVAL, TOL): locates an entry with rank >= NRANK in file IFILE whose JATT attribute is related to the value XVAL according to the specification given by MCODE as shown below:

MCODE=2: maximum value but greater than XVAL
MCODE=1: minimum value but greater than XVAL
MCODE=0: value within XVAL =TOL
MCODE=-1: minimum value but less than XVAL
MCODE=-2: maximum value but less than XVAL

FUNCTION NNRSC (NRES): current number of resource type NRES available.

FUNCTION NSUCR (NTRY): returns pointer to the successor entry of the entry who pointer is NTRY.

SUBROUTINE RMOVE (NRANK, IFILE,A): removes an entry defined by the variable NRANK from a file defined by the variable IFILE. If NRANK is positive, it defines the rank of the entry to be removed. If NRANK is negative, it points to the negative of the location where the entry to be removed is stored. REMOVE loads the vector A with the attributes of the entry removed. The value of MFA is reset to the pointer of the entry removed.

SUBROUTINE SEIZE (IR,N): resources may be seized by the user through a call to subroutine SEIZE (IR,N) where IR is the numeric code for the resource type and N is the number of units to be seized.

Appendix D

FORTRAN Program

Inserts

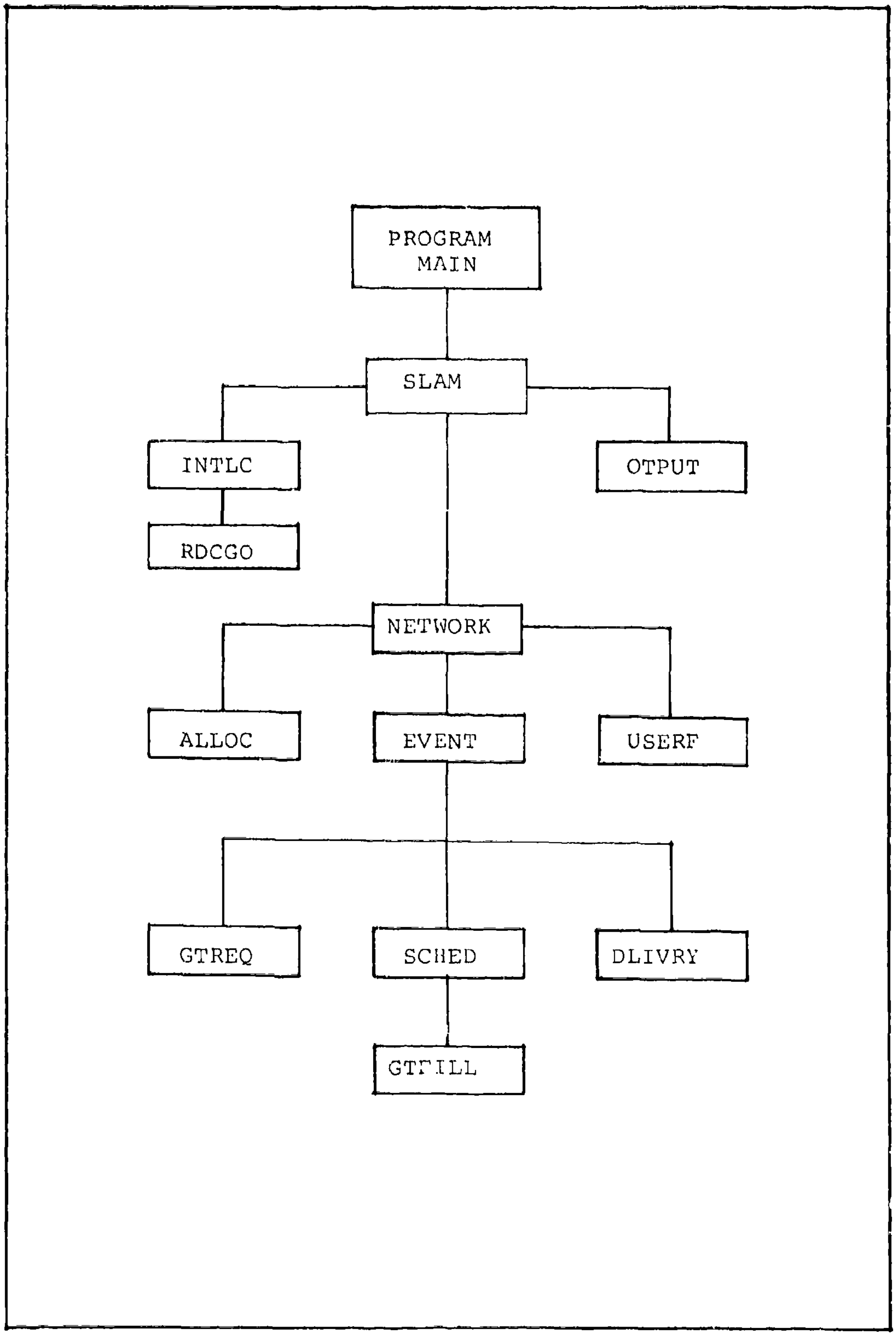


Figure D-1 Hierarchy of Program Units

```
      PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,TAPE50)
C
**********************************************************************
**********************************************************************
**                                                                  **
**    IN PROGRAM MAIN, THE DIMENSION OF THE SLAM NSET/QSET ARRAY    **
**    IS INCREASED FROM THE DEFAULT VALUE AND THE VARIABLE NNSET    **
**    IS SET EQUAL TO THE DIMENSION OF THE ARRAY.  USER COMMON      **
**    BLOCKS ARE DEFINED HERE SO THEY WILL ALWAYS BE DEFINED.       **
**    ONCE THE SLAM EXECUTIVE PROGRAM IS CALLED, IT CONTROLS ALL    **
**    PROGRAM FUNCTIONS UNTIL SIMULATION IS COMPLETE.               **
**                                                                  **
**********************************************************************
C
      DIMENSION NSET(30000)
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON QSET(30000)
C
      COMMON/UCOM1/CGOAT(15),BUFFER(15),NEWJOB(15)
      COMMON/UCOM2/OUTDEL(13),OVSDEL(13),BLKDEL(13),PAXDEL(13),
     +OUTTDY(13),OVSTDY(13),BLKTDY(13),PAXTDY(13),
     +TOTDEL(13),TOTTDY(13),SRTIES(13)
      COMMON/UCOM3/MSNNUM
C
      EQUIVALENCE(NSET(1),QSET(1))
C
      NNSET=30000
      NCRDR=5
      NPRNT=6
      NTAPE=7
C
      CALL SLAM
C
      STOP
      END
C
```

```
*********************************************************************
*********************************************************************
**                                                                 **
**    THE BLOCK DATA PROGRAM UNIT IS SOLELY FOR THE PURPOSE OF     **
**    INITIALIZING VARIABLES AND ARRAYS IN COMMON BLOCKS.  ONLY    **
**    USER COMMON IS INITIALIZED HERE.                             **
**                                                                 **
*********************************************************************
C
      BLOCK DATA USER
C
      INTEGER SRTIES
      REAL NEWJOB
      COMMON/UCOM1/CGOAT(15),BUFFER(15),NEWJOB(15)
C
      COMMON/UCOM2/OUTDEL(13),OVSDEL(13),BLKDEL(13),PAXDEL(13),
     +OUTTDY(13),OVSTDY(13),BLKTDY(13),PAXTDY(13),
     +TOTDEL(13),TOTTDY(13),SRTIES(13)
C
      DATA CGOAT,BUFFER,NEWJOB/45*0.0/
C
      DATA OUTDEL,OVSDEL,BLKDEL,PAXDEL,TOTDEL/65*0.0/
      DATA OUTTDY,OVSTDY,BLKTDY,PAXTDY,TOTTDY/65*0.0/
      DATA SRTIES/13*0/
C
      END
C
```

```
********************************************************************
********************************************************************
**                                                                **
**    SUBROUTINE INTLC IS AN OPTIONAL USER SUBROUTINE WHICH IS    **
**    CALLED BY SLAM AT THE BEGINNING OF EACH SIMULATION RUN TO   **
**    ESTABLISH INITIAL CONDITIONS FOR THE RUN.  THIS SUBROUTINE  **
**    CALLS ANOTHER FORTRAN INSERT (SUBROUTINE) TO ESTABLISH A    **
**    MASTER FILE OF CARGO REQUIREMENTS FOR USE THROUGHOUT THE    **
**    SIMULATION RUN.                                             **
**                                                                **
********************************************************************
C
      SUBROUTINE INTLC
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON QSET(30000)
C
      INTEGER SRTIES,MSNNUM
      COMMON/UCOM1/CGOAT(15),BUFFER(15),NEWJOB(15)
      COMMON/UCOM2/OUTDEL(13),OVSDEL(13),BLKDEL(13),PAXDEL(13),
     +OUTTDY(13),OVSTDY(13),BLKTDY(13),PAXTDY(13),
     +TOTDEL(13),TOTTDY(13),SRTIES(13)
      COMMON/UCOM3/MSNNUM
C
C **  READ CARGO REQUIREMENTS INTO SLAM MASTER FILE
C     *********************************************
C
      CALL RDCGO
C
C **  CLEAR ARRAYS AND VARIABLES FOR SUMMARY REPORT
C     *********************************************
C
      DO 9 II = 1,13
         OUTDEL(II) = OVSDEL(II) = BLKDEL(II) = PAXDEL(II) = 0.0
         OUTTDY(II) = OVSTDY(II) = BLKTDY(II) = PAXTDY(II) = 0.0
         TOTDEL(II) = TOTTDY(II) = 0.0
         SRTIES(II) = 0
    9 CONTINUE
C
      SUMOUT = SUMOVS = SUMBLK = SUMPAX = SUMTOT = 0.0
      MSNNUM = 0
C
      RETURN
      END
C
```

```
*********************************************************************
*********************************************************************
**                                                                 **
**     SUBROUTINE EVENT IS AN OPTIONAL SLAM INSERT WHICH ALLOWS    **
**     INTERFACE OF A SLAM NETWORK WITH USER-WRITTEN DISCRETE      **
**     EVENT CODE.  BY THIS MEANS PROCESSING OF NORMAL TRANS-      **
**     ACTIONS IS HALTED WHILE STATISTICS ARE COLLECTED OR THE     **
**     STATE OF THE MODEL IS ALTERED BY DIRECT ACCESS TO FILES     **
**     AND ACTIVITIES.                                             **
**                                                                 **
**     EVENT 1 CALLS A SUBROUTINE WHICH COPIES CARGO REQUIRE-      **
**     MENTS FOR A GIVEN DAY FROM THE MASTER FILE AND PLACES       **
**     THEM IN FILES ACCORDING TO THE TYPE OF CARGO.               **
**                                                                 **
**     EVENT 2 CALLS A SUBROUTINE WHICH SELECTS CARGO REQUIRE-     **
**     MENTS AND SCHEDULES MISSIONS TO TRANSPORT THE CARGO.        **
**                                                                 **
**     EVENT 3 CALLS A SUBROUTINE WHICH COLLECTS STATISTICS        **
**     ON CARGO DELIVERY WHEN A MISSION IS COMPLETED.              **
**                                                                 **
*********************************************************************
C
      SUBROUTINE EVENT(NEV)
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNK
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
C
      INTEGER TODAY
C
      GO TO (1,2,3),NEV
C
C **  EVENT 1 BRINGS TODAY'S REQUIREMENTS INTO SCHEDULING FILES
C     *********************************************************
    1 TODAY = INT(TNOW/24.0) + 1
      CALL GTREQ(TODAY)
      RETURN
C
C **  EVENT 2 SCHEDULES MISSIONS ON AVAILABLE AIRCRAFT
C     ************************************************
    2 CALL SCHED
      RETURN
C
C **  EVENT 3 COLLECTS STATISTICS ON CARGO DELIVERIES
C     ***********************************************
    3 CALL DLIVRY
      RETURN
      END
C
```

```
**********************************************************************
**********************************************************************
**                                                                  **
**    SUBROUTINE RDCGO IS CALLED BY SUBROUTINE INTLC AT THE         **
**    BEGINNING OF EACH SIMULATION RUN TO READ A LIST OF CARGO      **
**    REQUIREMENTS FROM AN EXTERNAL FILE INTO A MASTER SLAM FILE    **
**    TO BE USED THROUGHOUT THE RUN.  UNIT CARGO REQUIREMENTS ARE   **
**    READ FROM THE EXTERNAL FILE, AND ARE BROKEN DOWN INTO         **
**    SEPARATE REQUIREMENTS FOR EACH TYPE CARGO.  THE IDENTITY OF   **
**    UNIT REQUIREMENT IS RETAINED AS AN ATTRIBUTE OF EACH OF THE   **
**    NEW REQUIREMENTS, AS ARE THE ORIGIN, DESTINATION, DATE        **
**    AVAILABLE, AND DATE DUE.                                      **
**                                                                  **
**********************************************************************
C
      SUBROUTINE RDCGO
C
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
C
      COMMON QSET(30000)
C
      REAL OUTSZ,OVRSZ,BULK,PAX,CGOAT
      INTEGER SIZ1,SIZ2,SIZ3,SIZ4,BIGFIL,TAPENO,JJ
      INTEGER REQNO,ORIG,DEST,DAVBL,DDUE,MAXATR,UNITNO
      COMMON/UCOM1/CGOAT(15),BUFFER(15),NEWJOB(15)
C
C **  SET VALUES OF PARAMETERS AND VARIABLES
C     ************************************
      BIGFIL = 21
      MAXATR = 15
      SIZ1 = 1
      SIZ2 = 2
      SIZ3 = 3
      SIZ4 = 4
      TAPENO = 50
      REWIND TAPENO
      REQNO = 0
      UNITNO = 0
C
C **  LOOP UNTIL END OF FILE
C     **********************
C
   22 CONTINUE
C
C **  READ A REQUIREME    AND CHECK FOR END OF FILE
      READ(TAPENO,*) O,    DEST,DAVBL,DDUE,OUTSZ,OVRSZ,BULK,PAX
      IF (ORIG.LE.0) RETURN
C
C **  CLEAR ARRAY FOR NEW REQUIREMENT ATTRIBUTES
      DO 33 JJ = 1,MAXATR
         CGOAT(JJ) = 0.0
   33 CONTINUE
C
```

```
C **  ASSIGN ATTRIBUTES TO IDENTIFY REQUIREMENTS
      UNITNO = UNITNO + 1
      CGOAT(2) = ORIG
      CGOAT(3) = DEST
      CGOAT(4) = DAVBL
      CGOAT(5) = DDUE
      CGOAT(11) = UNITNO
C
C **  ASSIGN CARGO TYPE AND QUANTITY (OUTSIZE)
      IF (OUTSZ.GT.0.0) THEN
         CGOAT(6) = SIZ1
         CGOAT(7) = OUTSZ
         REQNO = REQNO + 1
         CGOAT(1) = REQNO
         CALL FILEM(BIGFIL,CGOAT)
      ENDIF
C
C **  ASSIGN CARGO TYPE AND QUANTITY (OVERSIZE)
      IF (OVRSZ.GT.0.0) THEN
         CGOAT(6) = SIZ2
         CGOAT(7) = OVRSZ
         REQNO = REQNO + 1
         CGOAT(1) = REQNO
         CALL FILEM(BIGFIL,CGOAT)
      ENDIF
C
C **  ASSIGN CARGO TYPE AND QUANTITY (BULK)
      IF (BULK.GT.0.0) THEN
         CGOAT(6) = SIZ3
         CGOAT(7) = BULK
         REQNO = REQNO + 1
         CGOAT(1) = REQNO
         CALL FILEM(BIGFIL,CGOAT)
      ENDIF
C
C **  ASSIGN CARGO TYPE AND QUANTITY (PASSENGERS)
      IF (PAX.GT.0.0) THEN
         CGOAT(6) = SIZ4
C **     CONVERT NUMBER OF (300 #) PASSENGERS TO TONS
         CGOAT(7) = PAX * (300.0/2000.0)
         REQNO = REQNO + 1
         CGOAT(1)= REQNO
         CALL FILEM(BIGFIL,CGOAT)
      ENDIF
C
C **  CONTINUE LOOP
C     *************
      GO TO 22
C
      END
C
```

```
*********************************************************************
*********************************************************************
**                                                                 **
**      SUBROUTINE SCHED IS CALLED BY EVENT 2 TO SELECT CARGO      **
**      REQUIREMENTS FROM THE APPROPRIATE FILES AND TO ALLOCATE     **
**      AIRCRAFT AGAINST THOSE REQUIREMENTS ACCORDING TO A         **
**      PARTICULAR SCHEME.  THIS VERSION OF THE SCHEDULER USES     **
**      AN "AIRCRAFT PREFERENCE" FOR A CERTAIN TYPE OF CARGO TO    **
**      DETERMINE WHICH REQUIREMENTS WILL FIRST BE SATISFIED BY    **
**      THE TYPE OF AIRCRAFT BEING ALLOCATED.                      **
**                                                                 **
**      WORKING FIRST WITH C-5 AIRCRAFT RESOURCES, THEN WITH       **
**      THE C-141, THE SCHEDULER CHECKS THE NUMBER OF AIRCRAFT     **
**      AVAILABLE FOR SCHEDULING.  IF AIRCRAFT ARE AVAILABLE,      **
**      THE CARGO FILE OF THE TYPE PREFERRED BY THAT AIRCRAFT      **
**      IS CHECKED FOR REQUIREMENTS.  IF CARGO OF THAT TYPE IS     **
**      AVAILABLE, THEN MISSION ENTITY IS ESTABLISHED WITH AN      **
**      AMOUNT OF CARGO NOT GREATER THAN THE AIRCRAFT CAPACITY.    **
**      IF THE FIRST CARGO SELECTED DOES NOT FILL THE AIRCRAFT,    **
**      SCHED CALLS SUBROUTINE GTFILL TO SEARCH FOR OTHER CARGO    **
**      WITH THE SAME ORIGIN-DESTINATION PAIR.  THIS SEARCH MAY    **
**      PROCEED THROUGH ALL FILES OF AVAILABLE CARGO.              **
**                                                                 **
**      IF A MATCH IS FOUND, SCHED FILLS THE REMAINING CAPACITY    **
**      OF THE AIRCRAFT.  A THIRD OR FOURTH REQUIREMENT MAY BE     **
**      FOUND, AS LONG AS THE AIRCRAFT CAPACITY IS NOT EXCEEDED.   **
**      WHEN THE AIRCRAFT CAPACITY IS MATCHED, OR ALL FILES HAVE   **
**      BEEN SEARCHED, THE MISSION ENTITY IS ENTERED INTO THE      **
**      OPERATIONS NETWORK.  THIS PROCESS CONTINUES UNTIL ALL      **
**      AVAILABLE AIRCRAFT HAVE BEEN ALLOCATED, OR UNTIL ALL       **
**      AVAILABLE CARGO HAS BEEN ASSIGNED TO A MISSION ENTITY.     **
**                                                                 **
**      NOTE THAT A C-141 CANNOT CARRY OUTSIZE CARGO (TYPE 1),     **
**      AND THAT A C-5 WILL NOT BE LAUNCHED UNLESS OUTSIZE         **
**      OR OVERSIZE CARGO IS AVAILABLE.                            **
**                                                                 **
*********************************************************************
C
      SUBROUTINE SCHED
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
C
      COMMON QSET(30000)
C
      REAL AVGLD1,AVGLD2,REMCAP,UPLD,NEWQTY,NEWJOB,BUFFER,CGOAT
      INTEGER TYPAC,ACAVBL,PREFQ,FSTREQ,NUMLDS,MSNNUM,REQNO
      INTEGER QTYATT,ORIG,DEST,MARKER,MAXATR,UNITNO
      COMMON/UCOM1/CGOAT(15),BUFFER(15),NEWJOB(15)
      COMMON/UCOM3/MSNNUM
C
C **  SET VALUE OF PARAMETERS
C     ***********************
C
      MAXATR = 15
```

```
      TYPAC = 1
      AVGLD1 = 68.0
      AVGLD2 = 24.0
C
C XX  FIRST SCHEDULER WORKS ON C-5 MISSIONS
C     **********************************
C
C XX  FIND NUMBER OF C-5 AIRCRAFT AVAILABLE
      ACAVBL = NNRSC(TYPAC)
      PREFQ = 1
C
C XX  IF NO C-5'S AVAILABLE, TRY FOR C-141
  220 IF ((ACAVBL.LE.0).OR.(PREFQ.EQ.4)) THEN
         GO TO 260
      ELSE IF (ACAVBL.GT.0) THEN
C XX     CHECK PREFERRED FILE (TYPE 1) FOR AVAILABLE CARGO
         FSTREQ = MMFE(PREFQ)
         REMCAP = AVGLD1
         IF (FSTREQ.GT.0) THEN
            CALL RMOVE(-FSTREQ,PREFQ,BUFFER)
C
C XX     IF NO TYPE 1 CARGO AVAILABLE TRY FOR TYPE 2 OR TYPE 3
         ELSE IF ((FSTREQ.LE.0).AND.(PREFQ.LT.3)) THEN
            PREFQ = PREFQ + 1
            GO TO 220
C
C XX     IF NO TYPE 1, 2, OR 3 CARGO, GO TO C-141'S
         ELSE IF (FSTREQ.LE.0) THEN
            GO TO 260
         ENDIF
      ENDIF
C
C XX  SET UP A MISSION ENTITY FOR THE CARGO
C     **********************************
C
      QTYATT = 7
      NUMLDS = 1
      MSNNUM = MSNNUM + 1
      REQNO = BUFFER(1)
      ORIG = BUFFER(2)
      DEST = BUFFER(3)
      UPLD = BUFFER(QTYATT)
      UNITNO = INT(BUFFER(11))
C
C XX  CLEAR ARRAY FOR MISSION ATTRIBUTES
      DO 225  JJ = 1,MAXATR
         CGOAT(JJ) = 0.0
  225 CONTINUE
C
C XX  ASSIGN ATTRIBUTES TO IDENTIFY MISSION
      CGOAT(1) = NUMLDS
      CGOAT(12) = TYPAC
      CGOAT(13) = ORIG
      CGOAT(14) = DEST
      CGOAT(15) = MSNNUM
```

```
C
C **  ENTER FULL LOADS IN OPERATIONS NETWORK
  230 IF (UPLD.GE.AVGLD1) THEN
         CGOAT(2*NUMLDS) = REQNO
         CGOAT(2*NUMLDS + 1) = AVGLD1
         CALL ENTER(TYPAC,CGOAT)
         ACAVBL = ACAVBL - 1
         UPLD = UPLD - AVGLD1
         BUFFER(QTYATT) = UPLD
         IF ((UPLD.GT.0).AND.(ACAVBL.GT.0)) THEN
            MSNNUM = MSNNUM + 1
            CGOAT(15) = MSNNUM
            GO TO 230
C
C **     REPLACE CARGO IN FILE IF NO AIRCRAFT REMAIN
         ELSE IF ((UPLD.GT.0).AND.(ACAVBL.LE.0)) THEN
            CALL FILEM(PREFQ,BUFFER)
            GO TO 260
C
C **     WHEN THIS REQUIREMENT IS SCHEDULED, LOOK FOR MORE
         ELSE IF (UPLD.LE.0) THEN
            PREFQ = 1
            GO TO 220
         ENDIF
      ENDIF
C
C **  CARGO QUANTITY LESS THAN AIRCRAFT CAPACITY
         IF ((UPLD.GT.0).AND.(UPLD.LT.AVGLD1)) THEN
         CGOAT(2*NUMLDS) = REQNO
         CGOAT(2*NUMLDS + 1) = UPLD
         REMCAP = AVGLD1 - UPLD
C
C **  SEARCH FOR FILLER WITH SAME UNIT NUMBER OR ORIG AND DEST
C     *****************************************************
C
  240    CALL GTFILL(UNITNO,ORIG,DEST,PREFQ,MARKER)
         IF ((MARKER.EQ.0).AND.(PREFQ.LT.4)) THEN
            PREFQ = PREFQ + 1
            GO TO 240
C
C **     NO MATCH FOUND...SCHEDULE MISSION WITH PARTIAL LOAD
         ELSE IF ((MARKER.EQ.0).AND.(PREFQ.EQ.4)) THEN
            CALL ENTER(TYPAC,CGOAT)
            ACAVBL = ACAVBL - 1
            PREFQ = 1
            GO TO 220
         ENDIF
C
C **  ORIG-DEST MATCH FOUND...FILL  AIRCRAFT
         IF (MARKER.GT.0) THEN
            NUMLDS = NUMLDS + 1
            CGOAT(1) = NUMLDS
            QTYATT = 7
            NEWQTY = NEWJOB(QTYATT)
            REQNO = NEWJOB(1)
```

```
C
C **        LIMIT PAX UPLOAD TO 73 MAX
            IF (MARKER.EQ.4) THEN
               REMCAP = MIN(REMCAP,10.95)
            ENDIF
C
            IF (NEWQTY.LE.REMCAP) THEN
               CGOAT(2*NUMLDS) = REQNO
               CGOAT(2*NUMLDS + 1) = NEWQTY
               REMCAP = REMCAP - NEWQTY
               GO TO 240
            ELSE IF (NEWQTY.GT.REMCAP) THEN
               CGOAT(2*NUMLDS) = REQNO
               CGOAT(2*NUMLDS + 1) = REMCAP
               CALL ENTER(TYPAC,CGOAT)
               ACAVBL = ACAVBL - 1
               NEWQTY = NEWQTY - REMCAP
C
C **        REPLACE EXTRA CARGO IN FILE
               NEWJOB(QTYATT) = NEWQTY
               CALL FILEM(MARKER,NEWJOB)
               PREFQ = 1
               GO TO 220
            ENDIF
         ENDIF
      ENDIF
C
C **  NOW SCHEDULER WORKS ON C-141 MISSIONS
C     *************************************
C
  260 CONTINUE
      PREFQ = 2
      TYPAC = 2
      ACAVBL = NNRSC(TYPAC)
C
C **  IF NO C-141'S AVAILABLE, LEAVE SCHEDULER
  320 IF (ACAVBL.LE.0) THEN
         RETURN
      ELSE IF (ACAVBL.GT.0) THEN
C **     CHECK PREFERRED FILE (TYPE 2) FOR CARGO
         FSTREQ = MMFE(PREFQ)
         REMCAP = AVGLD2
         IF (FSTREQ.GT.0) THEN
            CALL RMOVE(-FSTREQ,PREFQ,BUFFER)
         ELSE IF (FSTREQ.LE.0) THEN
C **     CHECK OTHER FILES FOR CARGO
            IF (PREFQ.LT.4) THEN
               PREFQ = PREFQ + 1
               GO TO 320
C **  IF NO CARGO AVAILABLE, LEAVE SCHEDULER
            ELSE IF (PREFQ.GE.4) THEN
               RETURN
            ENDIF
         ENDIF
      ENDIF
```

```
C
C **  SET UP A MISSION ENTITY FOR THE CARGO
C     **********************************
      QTYATT = 7
      NUMLDS = 1
      MSNNUM = MSNNUM + 1
      REQNO = BUFFER(1)
      ORIG = BUFFER(2)
      DEST = BUFFER(3)
      UPLD = BUFFER(QTYATT)
      UNITNO = INT(BUFFER(11))
C
C **  CLEAR ARRAY FOR MISSION ATTRIBUTES
      DO 325  KK = 1,MAXATR
         CGOAT(KK) = 0.0
  325 CONTINUE
C
C **  ASSIGN ATTRIBUTES TO IDENTIFY MISSION
      CGOAT(1) = NUMLDS
      CGOAT(12) = TYPAC
      CGOAT(13) = ORIG
      CGOAT(14) = DEST
      CGOAT(15) = MSNNUM
C
  330 IF (UPLD.GE.AVGLD2) THEN
         CGOAT(2*NUMLDS) = REQNO
C
C **     LIMIT NUMBER OF (300 #) PASSENGERS TO 122 MAX
         IF (PREFQ.LT.4) THEN
            CGOAT(2*NUMLDS + 1) = AVGLD2
         ELSE IF (PREFQ.EQ.4) THEN
            CGOAT(2*NUMLDS + 1) = 18.3
         ENDIF
C
C **  ENTER MISSION IN OPERATIONS NETWORK
         CALL ENTER(TYPAC,CGOAT)
         ACAVBL = ACAVBL - 1
C
C **  REDUCE CARGO QUANTITY FOR THIS REQUIREMENT
         IF (PREFQ.LT.4) THEN
            UPLD = UPLD - AVGLD2
         ELSE IF (PREFQ.EQ.4) THEN
            UPLD = UPLD - 18.3
         ENDIF
C
         BUFFER(QTYATT) = UPLD
         IF ((UPLD.GT.0).AND.(ACAVBL.GT.0)) THEN
            MSNNUM = MSNNUM + 1
            CGOAT(15) = MSNNUM
            GO TO 330
         ELSE IF ((UPLD.GT.0).AND.(ACAVBL.LE.0)) THEN
C **     RETURN EXTRA CARGO TO FILE
            CALL FILEM(PREFQ,BUFFER)
            RETURN
         ELSE IF (UPLD.LE.0) THEN
```

```
            GO TO 320
         ENDIF
      ENDIF
C
      IF ((UPLD.GT.0).AND.(UPLD.LT.AVGLD2)) THEN
C **     ASSIGN CARGO TO MISSION ENTITY
      CGOAT(2*NUMLDS + 1) = UPLD
         REMCAP = AVGLD2 - UPLD
C
C **  SEARCH FOR FILLER WITH SAME UNIT NUMBER OR ORIG AND DEST
C     ***************************************************
C
  340    CALL GTFILL(UNITNO,ORIG,DEST,PREFQ,MARKER)
         IF ((MARKER.EQ.0).AND.(PREFQ.LT.4)) THEN
            PREFQ = PREFQ + 1
            GO TO 340
C **  IF NO FILLER FOUND, ENTER MISSION WITH PARTIAL LOAD
         ELSE IF ((MARKER.EQ.0).AND.(PREFQ.GE.4)) THEN
            CALL ENTER(TYPAC,CGOAT)
            ACAVBL = ACAVBL - 1
            PREFQ = 2
            GO TO 320
         ENDIF
C
C **  IF FILLER IS FOUND, FILL UP TO AIRCRAFT CAPACITY
         IF (MARKER.GT.0) THEN
            NUMLDS = NUMLDS + 1
            CGOAT(1) = NUMLDS
            QTYATT = 7
            NEWQTY = NEWJOB(QTYATT)
            REQNO = NEWJOB(1)
C
C **        LIMIT PAX LOAD TO 122 MAX
            IF (MARKER.EQ.4) THEN
               REMCAP = MIN(REMCAP,18.3)
            ENDIF
C
            IF (NEWQTY.LE.REMCAP) THEN
               CGOAT(2*NUMLDS) = REQNO
               CGOAT(2*NUMLDS + 1) = NEWQTY
               REMCAP = REMCAP - NEWQTY
               GO TO 340
            ELSE IF (NEWQTY.GT.REMCAP) THEN
               CGOAT(2*NUMLDS) = REQNO
               CGOAT(2*NUMLDS + 1) = REMCAP
               NEWQTY = NEWQTY - REMCAP
               CALL ENTER(TYPAC,CGOAT)
               ACAVBL = ACAVBL - 1
               NEWJOB(QTYATT) = NEWQTY
               CALL FILEM(MARKER,NEWJOB)
               PREFQ = 2
               GO TO 320
            ENDIF
         ENDIF
      ENDIF
```

```
C
C **  IF MORE AIRCRAFT AVAILABLE, SCHEDULE ANOTHER MISSION
      IF (ACAVBL.GT.0) THEN
         PREFQ = 2
         GO TO 320
      ENDIF
      RETURN
      END
C
```

```
************************************************************************
************************************************************************
**                                                                    **
**    SUBROUTINE GTFILL IS CALLED BY THE SCHEDULER TO FIND FILLER     **
**    CARGO WITH THE SAME ORIGIN-DESTINATION PAIR AS A PREVIOUSLY     **
**    SELECTED REQUIREMENT.  THE SEARCH BEGINS IN A FILE              **
**    DESIGNATED BY THE THIRD ARGUMENT, AND PROCEEDS SEQUENTIALLY     **
**    THROUGH FILE NUMBER FOUR, OR UNTIL A MATCH IS FOUND.            **
**                                                                    **
**    IF A MATCH IS FOUND, THE CARGO ENTITY IS REMOVED FROM ITS       **
**    FILE AND PLACED IN A BUFFER ARRAY FOR PROCESSING BY THE         **
**    SCHEDULER.  THE FOURTH ARGUMENT RETURNS THE FILE NUMBER         **
**    OF THE MATCHING REQUIREMENT SO THAT EXCESS CARGO MAY BE         **
**    RETURNED TO ITS PROPER FILE.  A ZERO IS RETURNED IF NO          **
**    MATCH IS FOUND.                                                 **
**                                                                    **
************************************************************************
C
      SUBROUTINE GTFILL(UNITNO,CGORIG,CGDEST,IFILE,MARKER)
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON QSET(30000)
C
      REAL NEWJOB
      INTEGER MARKER,NEXT,CGORIG,CGDEST,MAXATR,UNITNO,FOUND
      COMMON/UCOM1/CGOAT(15),BUFFER(15),NEWJOB(15)
C
C **  SET VALUE OF PARAMETERS
C     ***********************
      XUNIT = REAL(UNITNO)
      FOUND = 0
      MARKER = 0
      MAXATR = 15
C
C **  TRY TO FILL WITH CARGO FROM SAME UNIT REQUIREMENT
C     *************************************************
      DO 390  MD = IFILE,4
         FOUND = NFIND(1,MD,11,0,XUNIT,0.0)
         IF (FOUND.GT.0) THEN
            MARKER = MD
            CALL RMOVE(FOUND,MD,NEWJOB)
            GO TO 410
         ENDIF
  390 CONTINUE
C
C **  IF NO CARGO WITH THE SAME UNIT NUMBER IS FOUND,
C **  TRY TO MATCH CARGO ORIGIN AND DESTINATION
C     ************************************************
C
C **  FIND THE FIRST ENTRY IN THE PREFERRED FILE
      NEXT = MMFE(IFILE)
C
C **  IF THE PREFERRED FILE IS EMPTY, RETURN
  400 IF (NEXT.EQ.0) THEN
```

```
            RETURN
         ELSE
C **        CLEAR THE BUFFER ARRAY
            DO 405 LM = 1,MAXATR
               NEWJOB(LM) = 0
  405       CONTINUE
C **        COPY FILE ENTRY INTO BUFFER ARRAY
            CALL COPY(-NEXT,IFILE,NEWJOB)
         ENDIF
C
C **     CHECK FOR ORIGIN-DESTINATION MATCH
         IF ((NEWJOB(2).EQ.CGORIG).AND.(NEWJOB(3).EQ.CGDEST)) THEN
C **        IF MATCH FOUND, REMOVE FROM FILE & SET MARKER
            CALL RMOVE(-NEXT,IFILE,NEWJOB)
            MARKER = IFILE
            GO TO 410
         ELSE
C **        IF NO MATCH, FIND NEXT ENTRY AND CHECK IT
            NEXT = NSUCR(NEXT)
            GO TO 400
         ENDIF
C
  410    CONTINUE
         RETURN
         END
C
```

```
*********************************************************************
*********************************************************************
**                                                                 **
**    SUBROUTINE ALLOC IS CALLED FROM CERTAIN ALLOCATE NODES IN    **
**    THE SLAM NETWORK.  IT SUBSTITUTES A USER-WRITTEN CODE FOR   **
**    THE NORMAL ALLOCATION LOGIC OF THE SLAM LANGUAGE.  RUNWAY    **
**    AND RAMP RESOURCES AT ENROUTE LANDING BASES ARE THUS         **
**    ALLOCATED ACCORDING TO A MORE COMPLEX SCHEME THAN WOULD BE   **
**    POSSIBLE WITH ONLY THE SLAM NETWORK.                         **
**                                                                 **
**    THIS SUBROUTINE CONTAINS A SEPARATE MODULE FOR EACH BASE     **
**    REPRESENTED.  RAMP SPACE IS ALLOCATED ACCORDING TO THE       **
**    TYPE OF AIRCRAFT ASSIGNED A MISSION ENTITY, AND A LANDING    **
**    RUNWAY IS ALLOCATED ONLY WHEN RAMP SPACE IS AVAILABLE.       **
**                                                                 **
**    THE FIRST ARGUMENT GIVES THE SUBROUTINE A NUMERICAL CODE     **
**    FOR THE BASE CALLING FOR ALLOCATION, AND THE SECOND ARGU-    **
**    IS A FLAG TELLING THE SLAM EXECUTIVE WHETHER RESOURCES       **
**    HAVE BEEN ALLOCATED.                                         **
**                                                                 **
*********************************************************************
C
      SUBROUTINE ALLOC(BASE,IFLAG)
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET.NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON QSET(10000)
C
      INTEGER PARKB,PARKS,RWNO,RPNO,TYPAC,BASE
C
C **  SET VALUE OF PARAMETERS
C     ***********************
      TYPAC = ATRIB(12)
      IFLAG = 0
C
      IF (BASE.EQ.1) GO TO 67
      IF (BASE.EQ.2) GO TO 71
      IF (BASE.EQ.3) GO TO 89
      IF (BASE.EQ.4) GO TO 90
C
C **  SET REQUIRED RAMP SPACE AND RESOURCE ID NUMBERS
C     ***************************************
   67 PARKB = 17
      PARKS = 9
      RWNO = 5
      RPNO = 4
C
C **  CHECK IF RAMP SPACE AVAILABLE FOR BIG AIRCRAFT
      IF ((TYPAC.EQ.1).AND.(NNRSC(RPNO).GE.PARKB)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
C **        ALLOCATE RUNWAY AND RAMP IF BOTH AVAILABLE
            CALL SEIZE(RPNO,PARKB)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
            RETURN
```

```
         ENDIF
C XX  CHECK IF RAMP SPACE AVAILABLE FOR SMALL AIRCRAFT
      ELSE IF ((TYPAC.EQ.2).AND.(NNRSC(RPNO).GE.PARKS)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
C XX        ALLOCATE RUNWAY AND RAMP IF BOTH AVAILABLE
            CALL SEIZE(RPNO,PARKS)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
            RETURN
         ENDIF
      ENDIF
      RETURN
C
C XX  SET REQUIRED RAMP SPACE AND RESOURCE ID NUMBERS
C     ******************************************
   71 PARKB = 43
      PARKS = 19
      RWNO = 12
      RPNO = 11
C
C XX  CHECK IF RAMP SPACE AVAILABLE FOR BIG AIRCRAFT
      IF ((TYPAC.EQ.1).AND.(NNRSC(RPNO).GE.PARKB)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
C XX     ALLOCATE RUNWAY AND RAMP IF BOTH AVAILABLE
            CALL SEIZE(RPNO,PARKB)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
            RETURN
         ENDIF
C XX  CHECK IF RAMP SPACE AVAILABLE FOR SMALL AIRCRAFT
      ELSE IF ((TYPAC.EQ.2).AND.(NNRSC(RPNO).GE.PARKS)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
C XX     ALLOCATE RUNWAY AND RAMP IF BOTH AVAILABLE
            CALL SEIZE(RPNO,PARKS)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
            RETURN
         ENDIF
      ENDIF
      RETURN
C
C XX  SET REQUIRED RAMP SPACE AND RESOURCE ID NUMBERS
C     ******************************************
   89 PARKB = 33
      PARKS = 16
      RWNO = 7
      RPNO = 6
C
C XX  CHECK IF RAMP SPACE AVAILABLE FOR BIG AIRCRAFT
      IF ((TYPAC.EQ.1).AND.(NNRSC(RPNO).GE.PARKB)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
C XX     ALLOCATE RUNWAY AND RAMP IF BOTH AVAILABLE
            CALL SEIZE(RPNO,PARKB)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
```

```
            RETURN
         ENDIF
C **  CHECK IF RAMP SPACE AVAILABLE FOR SMALL AIRCRAFT
      ELSE IF ((TYPAC.EQ.2).AND.(NNRSC(RPNO).GE.PARKS)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
C **     ALLOCATE RUNWAY AND RAMP IF BOTH AVAILABLE
            CALL SEIZE(RPNO,PARKS)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
            RETURN
         ENDIF
      ENDIF
      RETURN
C
C **  SET REQUIRED RAMP SPACE AND RESOURCE ID NUMBERS
C     ******************************************
   90 PARKB = 21
      PARKS = 13
      RWNO = 10
      RPNO = 9
C
      IF ((TYPAC.EQ.1).AND.(NNRSC(RPNO).GE.PARKB)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
            CALL SEIZE(RPNO,PARKB)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
            RETURN
         ENDIF
      ELSE IF ((TYPAC.EQ.2).AND.(NNRSC(RPNO).GE.PARKS)) THEN
         IF (NNRSC(RWNO).GE.1) THEN
            CALL SEIZE(RPNO,PARKS)
            CALL SEIZE(RWNO,1)
            IFLAG = 1
            RETURN
         ENDIF
      ENDIF
C
      RETURN
      END
C
```

```
************************************************************************
************************************************************************
**                                                                    **
**     SUBROUTINE GTREQ IS CALLED BY EVENT 1 TO SELECT, FROM THE      **
**     MASTER FILE, CARGO REQUIREMENTS AVAILABLE FOR SCHEDULING       **
**     DURING A GIVEN ACTIVITY PERIOD (ONE DAY).  BEGINNING FROM      **
**     THE FIRST LINE OF THE MASTER FILE ON DAY 1, GTREQ FINDS        **
**     THE FIRST REQUIREMENT FOR THE NEXT DAY AND COPIES ALL          **
**     CARGO ENTITIES UP TO THAT POINT INTO SEPARATE FILES            **
**     ACCORDING TO TYPE CARGO.  (FILES TO BE USED BY THE SCHED-      **
**     ULER.)                                                         **
**                                                                    **
************************************************************************
C
C
      SUBROUTINE GTREQ(IDATE)
C
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
C
      COMMON QSET(30000)
C
      INTEGER FSTLIN,LSTLIN,BIGFIL,TYPCGO,DTATR,MCODE,REQNO
      REAL BUFFER(15)
C
C **  SET VALUES OF PARAMETERS
C     ************************
      BIGFIL = 21
      DTATR = 4
      MCODE = 1
C
C **  FIND FIRST AND LAST REQUIREMENT FOR TODAY
C     *****************************************
C
      DATE = REAL(IDATE)
      YESTDY = DATE - 1.0
      FSTLIN = NFIND(1,BIGFIL,DTATR,MCODE,YESTDY,0.0)
      LSTLIN = NFIND(FSTLIN,BIGFIL,DTATR,MCODE,DATE,0.0)
      LSTLIN = LSTLIN - 1
C
C **  COPY TODAY'S REQUIREMENTS INTO SCHEDULING FILES
C     ***********************************************
C
      DO 60 REQNO = FSTLIN,LSTLIN
C
      CALL COPY(REQNO,BIGFIL,BUFFER)
      TYPCGO = NINT(BUFFER(6))
C **  FILE REQUIREMENT ACCORDING TO TYPE CARGO
      CALL FILEM(TYPCGO,BUFFER)
C
   60 CONTINUE
C
      RETURN
      END
```

```
**********************************************************************
**********************************************************************
**                                                                  **
**     SUBROUTINE DLIVRY IS CALLED BY EVENT 3 WHEN A MISSION        **
**     ENTITY REACHES A POINT IN THE NETWORK CORRESPONDING TO A     **
**     CARGO DESTINATION AIRFIELD.  STATISTICS ARE COLLECTED ON     **
**     THE QUANTITY OF CARGO DELIVERED BY THAT MISSION, AS WELL     **
**     AS THE TARDINESS OF THE CARGO.  THE QUANTITY DELIVERED IS    **
**     ALSO POSTED WITH THE ORIGINAL REQUIREMENT IN THE MASTER      **
**     FILE.  IF THAT SHIPMENT COMPLETES DELIVERY OF THE TOTAL      **
**     QUANTITY OF CARGO ASSOCIATED WITH THE REQUIREMENT, THE       **
**     DUE DATE FOR THE REQUIREMENT IS CHECKED AND A FLAG IS        **
**     SET TO INDICATE WHETHER CLOSURE WAS MET.                     **
**                                                                  **
**     THIS ROUTINE USES DIRECT ACCESS TO THE SLAM NSET/QSET         **
**     ARRAY BY WAY OF THE REQUIREMENT NUMBER, WHICH IS PRESET      **
**     TO MATCH THE RANK OF EACH REQUIREMENT IN THE MASTER FILE.    **
**     ONCE THE CORRECT LOCATION IS FOUND, THE VALUES ASSOCI-       **
**     TED WITH THAT REQUIREMENT MAY BE ACCESSED ACCORDING TO       **
**     THE ORDER IN WHICH THEY ARE FILED.                           **
**                                                                  **
**     CARGO DATA IS STORED IN ARRAYS, TO BE PRINTED AT THE END     **
**     OF THE SIMULATION RUN WHEN SUBROUTINE OTPUT IS CALLED.       **
**                                                                  **
**********************************************************************
C
      SUBROUTINE DLIVRY
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON QSET(30000)
C
      INTEGER NTRY,DUEMK,QTYMK,TYPEMK,ERLYMK,LATEMK,CLOSMK,LD,TARDY
      INTEGER BIGFIL,NUMLDS,REQNO,TYPCGO,DLIVDT,TODAY,SRTIES
      INTEGER MISSNO,TYPAC,MYTAP1,MYTAP2,CLSURE,POLICY,TRDYMK
      REAL DLVELY,DLVLAT
      REAL TOL,QTYDEL,TOTQTY,OUTDEL,TOTDEL,TOTTDY,DLVS
      REAL OVSDEL,BLKDEL,PAXDEL,OUTTDY,OVSTDY,BLKTDY,PAXTDY
C
      COMMON/UCOM2/OUTDEL(13),OVSDEL(13),BLKDEL(13),PAXDEL(13),
     +OUTTDY(13),OVSTDY(13),BLKTDY(13),PAXTDY(13),
     +TOTDEL(13),TOTTDY(13),SRTIES(13)
C
C **  SET VALUES OF PARAMETERS AND VARIABLES
C     **************************************
C
      BIGFIL = 21
      TODAY = INT(TNOW/24.0) + 1
      DLIVDT = TODAY
      XXDLIV = REAL(DLIVDT)
      TOL = 1.0
      NUMLDS = NINT(ATRIB(1))
C
C **  UNLOAD AIRCRAFT AND POST DATA ON CARGO DELIVERED
C     *********************************************
```

```
C
      DO 800 LD = 1,NUMLDS
C
      DLVELY = DLVLAT = CLSURE = 0
      REQNO = NINT(ATRIB(2*LD))
      QTYDEL = ATRIB(2*LD + 1)
C
C **  SET POINTERS FOR DIRECT ACCESS TO ATTRIBUTES IN
C **  THE MASTER REQUIREMENTS FILE
C
      NTRY = LOCAT(REQNO,BIGFIL)
      DUEMK = NTRY + 5
      TYPEMK = NTRY + 6
      QTYMK = NTRY + 7
      ERLYMK = NTRY + 8
      LATEMK = NTRY + 9
      CLOSMK = NTRY + 10
      TRDYMK = NTRY + 12
C
      DDUE = QSET(DUEMK)
      IDDUE = INT(DDUE)
C
C **  UPDATE THE DAILY SORTIE COUNT
      SRTIES(DLIVDT) = SRTIES(DLIVDT) + 1
C
C **  POST DELIVERY IN MASTER FILE AS ONTIME OR LATE
      IF (DLIVDT.LE.DDUE) THEN
         QSET(ERLYMK) = QSET(ERLYMK) + QTYDEL
         DLVELY = QTYDEL
         DLVS = QSET(ERLYMK)
         TOTQTY = QSET(QTYMK) - TOL
         IF (DLVS.GE.TOTQTY) THEN
            QSET(CLOSMK) = 1.0
            CLSURE = 1
         ENDIF
      ELSE IF (DLIVDT.GT.DDUE) THEN
         QSET(LATEMK) = QSET(LATEMK) + QTYDEL
         DLVLAT = QTYDEL
      ENDIF
C
C **  UPDATE DATA ON DAILY CARGO DELIVERIES AND TARDINESS
C     ***************************************************
C
      TOTDEL(DLIVDT) = TOTDEL(DLIVDT) + QTYDEL
      TYPCGO = NINT(QSET(TYPEMK))
C
C **  UPDATE DAILY DATA FOR OUTSIZE CARGO
      IF (TYPCGO.EQ.1) THEN
         OUTDEL(DLIVDT) = OUTDEL(DLIVDT) + QTYDEL
         TARDY = MAX(0.0,(XXDLIV-DDUE))
            OUTTDY(DLIVDT) = OUTTDY(DLIVDT) + QTYDEL * TARDY


            TOTTDY(DLIVDT) = TOTTDY(DLIVDT) + QTYDEL * TARDY
C **  UPDATE DAILY DATA FOR OVERSIZE CARGO
```

```
      ELSE IF (TYPCGO.EQ.2) THEN
         OVSDEL(DLIVDT) = OVSDEL(DLIVDT) + QTYDEL
         TARDY = MAX(0.0,(XXDLIV-DDUE))
            OVSTDY(DLIVDT) = OVSTDY(DLIVDT) + QTYDEL * TARDY
            TOTTDY(DLIVDT) = TOTTDY(DLIVDT) + QTYDEL * TARDY
C **  UPDATE DAILY DATA FOR BULK CARGO
      ELSE IF (TYPCGO.EQ.3) THEN
         BLKDEL(DLIVDT) = BLKDEL(DLIVDT) + QTYDEL
         TARDY = MAX(0.0,(XXDLIV-DDUE))
            BLKTDY(DLIVDT) = BLKTDY(DLIVDT) + QTYDEL * TARDY
            TOTTDY(DLIVDT) = TOTTDY(DLIVDT) + QTYDEL * TARDY
C **  UPDATE DAILY DATA FOR PASSENGERS
      ELSE IF (TYPCGO.EQ.4) THEN
         PAXDEL(DLIVDT) = PAXDEL(DLIVDT) + QTYDEL
         TARDY = MAX(0.0,(XXDLIV-DDUE))
            PAXTDY(DLIVDT) = PAXTDY(DLIVDT) + QTYDEL * TARDY
            TOTTDY(DLIVDT) = TOTTDY(DLIVDT) + QTYDEL * TARDY
      ENDIF
C
      QSET(TRDYMK) = QSET(TRDYMK) + QTYDEL * TARDY
C
  800 CONTINUE
C
      RETURN
      END
C
```

```
      SUMOUT = SUMOUT + OUTDEL(LL)
      SUMOVS = SUMOVS + OVSDEL(LL)
      SUMBLK = SUMBLK + BLKDEL(LL)
      SUMPAX = SUMPAX + PAXDEL(LL)
      SUMTOT = SUMTOT + TOTDEL(LL)
      SRTYCT = SRTYCT + SRTIES(LL)
C
 900  CONTINUE
C
      PRINT*,'  '
      WRITE(*,922) SUMOUT,SUMOVS,SUMBLK,SUMPAX,SUMTOT,SRTYCT
      PRINT*,'  '
C
C **  HEADING FOR TABLE ON CARGO TARDINESS
C     ************************************
      PRINT'(////T19,'' *****    TARDINESS (TON-DAYS)    *****'',//)'
      PRINT'(4X,'' DAY'',4X,'' OUTSZ'',4X,'' OVRSZ'',5X,'' BULK'',
     +      6X,'' PAX'',6X,'' SUM'',5X,'' ACCUM'')'
      PRINT*,'  '
C
C **  PRINT DAILY SUMMARY AND CUMULATIVE CARGO TARDINESS
      DO 910  MM = 1,DAYS
C
      CUMTDY = CUMTDY + TOTTDY(MM)
      WRITE(*,930) MM,OUTTDY(MM),OVSTDY(MM),BLKTDY(MM),PAXTDY(MM),
     +TOTTDY(MM),CUMTDY
C
 910  CONTINUE
C
 920  FORMAT(T6,I4,5F10.1,I10)
 922  FORMAT(T3,' TOTALS',T10,5F10.1,I10)
 930  FORMAT(T6,I4,5F10.2,F11.2)
C
C **  STORE DATA FROM MASTER FILE ON DISK FOR LATER ANALYSIS
C     ******************************************************
C
      MYTAP3 = 203
      LINES = NNQ(BIGFIL)
      DO 950  REQNO = 1,LINES
         CALL COPY(REQNO,BIGFIL,BUFFER)
         DDUE = INT(BUFFER(5))
         TYPCGO = INT(BUFFER(6))
         QNTY = BUFFER(7)
         ONTIME = BUFFER(8)
         LATE = BUFFER(9)
         CLOSED = INT(BUFFER(10))
         UNITNO = INT(BUFFER(11))
         TRDY = BUFFER(12)
C
         WRITE(MYTAP3,955) REQNO,TYPCGO,DDUE,QNTY,ONTIME,LATE,
     +                    CLOSED,TRDY,UNITNO,POLICY,NNRUN
C
 950  CONTINUE
C
 955  FORMAT(I4,I2,I3,3F7.1,I2,F7.1,I4,2I3)
```

```
**********************************************************************
**********************************************************************
**                                                                  **
**     SUBROUTINE OTPUT IS CALLED BY SLAM AT THE END OF EACH        **
**     SIMULATION RUN TO PRINT USER-COLLECTED STATISTICS OR         **
**     PERFORM ANY OTHER FUNCTION CODED IN THE FORTRAN SUBROU-      **
**     TINE.  IT IS USED HERE TO PRINT STATISTICS ON THE            **
**     QUANTITY OF CARGO DELIVERED DAILY, THE DAILY MISSION         **
**     COUNT (COUNTED AT DELIVERY POINT), AND THE TARDINESS OF      **
**     CARGO DELIVERED.  SELECTED DATA FROM THE MASTER FILE OF      **
**     REQUIREMENTS IS WRITTEN TO TAPE FOR LATER ANALYSIS.          **
                                 **
**********************************************************************
C
      SUBROUTINE OTPUT
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
C
      COMMON QSET(30000)
C
      INTEGER DAYS,LL,MM,SRTIES,SRTYCT,BIGFIL,POLICY
      INTEGER MYTAP3,LINES,REQNO,DDUE,TYPCGO,CLOSED,UNITNO
      REAL QNTY,ONTIME,LATE,TRDY
      REAL OUTDEL,OVSDEL,BLKDEL,PAXDEL,TOTDEL
      REAL OUTTDY,OVSTDY,BLKTDY,PAXTDY,TOTTDY,CUMTDY
      REAL SUMOUT,SUMOVS,SUMBLK,SUMPAX,SUMTOT
C
      COMMON/UCOM1/CGOAT(15),BUFFER(15),NEWJOB(15)
      COMMON/UCOM2/OUTDEL(13),OVSDEL(13),BLKDEL(13),PAXDEL(13),
     +OUTTDY(13),OVSTDY(13),BLKTDY(13),PAXTDY(13),
     +TOTDEL(13),TOTTDY(13),SRTIES(13)
C
C **  SET VALUES OF PARAMETERS AND VARIABLES
C     **************************************
C
      POLICY = 3
      SRTYCT = 0
      SUMOUT = SUMOVS = SUMBLK = SUMPAX = SUMTOT = 0.0
      CUMTDY = 0.0
      DAYS = 13
      BIGFIL = 21
C
C **  HEADING FOR TABLE ON CARGO DELIVERED
C     ************************************
      PRINT'(///T18,'' *****    CARGO DELIVERED (TONS)    *****'',//)'
      PRINT'(4X,'' DAY'',4X,'' OUTSZ'',4X,'' OVRSZ'',5X,'' BULK'',
     +      6X,'' PAX'',6X,'' SUM'',5X,'' MSNS'')'
      PRINT*,' '
C
C **  PRINT DAILY SUMMARY OF CARGO DELIVERED AND SORTIE COUNT
      DO 900  LL = 1,DAYS
C
      WRITE(*,920) LL,OUTDEL(LL),OVSDEL(LL),BLKDEL(LL),PAXDEL(LL),
     +        TOTDEL(LL),SRTIES(LL)
```

```
C
      RETURN
      END
C
```

```
*********************************************************************
*********************************************************************
**                                                                 **
**    FUNCTION USERF ASSIGNS AIRCRAFT GROUND TIMES ACCORDING TO    **
**    THE TYPE OF AIRCRAFT AND TYPE OF GROUND ACTIVITY.  AIRCRAFT  **
**    TYPES ARE C-5 AND C-141.  POSSIBLE ACTIVITIES ARE LOADING,   **
**    UNLOADING, ENROUTE SERVICE, AND HOME BASE MAINTENANCE.  THE  **
**    TIMES ARE ASSIGNED FROM DISCRETE PROBABILITY DISTRIBUTIONS.  **
**                                                                 **
*********************************************************************
C
      FUNCTION USERF(IFN)
C
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR,
     +NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
C
      INTEGER UGT1SZ,UGT2SZ,EGT1SZ,EGT2SZ,DGT1SZ,DGT2SZ,TYPAC
      REAL USERF,UGT1(6),UGT1D(6),UGT2(5),UGT2D(5),EGT1(6),EGT1D(6),
     +     EGT2(6),EGT2D(6),DGT1(6),DGT1D(6),DGT2(5),DGT2D(5)
C
      IS = 6
      USERF = 0
      TYPAC = ATRIB(12)
C
      GO TO (1,2,3,4),IFN
C
C **  GROUNDTIME FOR UPLOAD STATIONS
C     ******************************
C
C **  DISTRIBUTION FOR C-5
    1 UGT1(1) = 2.1
      UGT1(2) = 3.3
      UGT1(3) = 4.5
      UGT1(4) = 6.5
      UGT1(5) = 15.0
      UGT1(6) = 24.0
C
      UGT1D(1) = 0.389
      UGT1D(2) = 0.792
      UGT1D(3) = 0.955
      UGT1D(4) = 0.985
      UGT1D(5) = 0.990
      UGT1D(6) = 1.0
C
      UGT1SZ = 6
C
C **  DISTRIBUTION FOR C-141
      UGT2(1) = 1.10
      UGT2(2) = 2.15
      UGT2(3) = 3.5
      UGT2(4) = 5.5
      UGT2(5) = 15.0
C
      UGT2D(1) = 0.015
      UGT2D(2) = 0.678
```

```
      UGT2D(3) = 0.957
      UGT2D(4) = 0.990
      UGT2D(5) = 1.0
C
      UGT2SZ = 5
C
C **  ASSIGN GROUNDTIME
      IF (TYPAC.EQ.1) THEN
         USERF = DPROB(UGT1D,UGT1,UGT1SZ,IS) + 6.0
      ELSE IF (TYPAC.EQ.2) THEN
         USERF = DPROB(UGT2D,UGT2,UGT2SZ,IS) + 6.0
      ENDIF
      RETURN
C
C **  GROUNDTIME FOR ENROUTE STATIONS
C     ******************************
C
C **  DISTRIBUTION FOR C-5
    2 EGT1(1) = 1.2
      EGT1(2) = 2.5
      EGT1(3) = 3.5
      EGT1(4) = 5.0
      EGT1(5) = 8.0
      EGT1(6) = 14.0
C
      EGT1D(1) = 0.012
      EGT1D(2) = 0.464
      EGT1D(3) = 0.888
      EGT1D(4) = 0.978
      EGT1D(5) = 0.998
      EGT1D(6) = 1.0
C
      EGT1SZ = 6
C
C **  DISTRIBUTION FOR C-141
      EGT2(1) = 1.33
      EGT2(2) = 2.33
      EGT2(3) = 3.0
      EGT2(4) = 4.0
      EGT2(5) = 5.0
      EGT2(6) = 6.33
C
      EGT2D(1) = 0.230
      EGT2D(2) = 0.790
      EGT2D(3) = 0.850
      EGT2D(4) = 0.900
      EGT2D(5) = 0.915
      EGT2D(6) = 1.0
C
      EGT2SZ = 6
C
C **  ASSIGN GROUNDTIME
      IF (TYPAC.EQ.1) THEN
         USERF = DPROB(EGT1D,EGT1,EGT1SZ,IS)
      ELSE IF (TYPAC.EQ.2) THEN
```

```
           USERF = DPROB(EGT2D,EGT2,EGT2SZ,IS)
        ENDIF
        RETURN
C
C XX  GROUNDTIME FOR DOWNLOAD STATIONS
C     XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C
C XX  DISTRIBUTION FOR C-5
    3 DGT1(1) = 2.5
      DGT1(2) = 4.3
      DGT1(3) = 6.5
      DGT1(4) = 9.0
      DGT1(5) = 15.0
      DGT1(6) = 24.0
C
      DGT1D(1) = 0.444
      DGT1D(2) = 0.812
      DGT1D(3) = 0.962
      DGT1D(4) = 0.975
      DGT1D(5) = 0.988
      DGT1D(6) = 1.0
C
      DGT1SZ = 6
C
C XX  DISTRIBUTION FOR C-141
      DGT2(1) = 1.10
      DGT2(2) = 2.15
      DGT2(3) = 3.5
      DGT2(4) = 5.5
      DGT2(5) = 15.0
C
      DGT2D(1) = 0.015
      DGT2D(2) = 0.678
      DGT2D(3) = 0.957
      DGT2D(4) = 0.990
      DGT2D(5) = 1.0
C
      DGT2SZ = 5
C
C XX  ASSIGN GROUNDTIME
      IF (TYPAC.EQ.1) THEN
         USERF = DPROB(DGT1D,DGT1,DGT1SZ,IS)
      ELSE IF (TYPAC.EQ.2) THEN
         USERF = DPROB(DGT2D,DGT2,DGT2SZ,IS)
      ENDIF
      RETURN
C
C XX  GROUND TIME AT HOME BASE
C     XXXXXXXXXXXXXXXXXXXXXXXX
C
C XX  GET A RANDOM NUMBER FROM A UNIFORM(0,1) DISTRIBUTION
    4 RN = DRAND(IS)
C
C XX  DETERMINE WHETHER MAINTENANCE WILL BE PERFORMED AT
C XX  HOME BASE AND ASSIGN GROUND TIME
```

```
C
      IF (TYPAC.EQ.1) THEN
         IF (RN.LE.0.25) THEN
            USERF = 8.0
         ELSE IF (RN.GT.0.25) THEN
            GO TO 2
         ENDIF
      ELSE IF (TYPAC.EQ.2) THEN
         IF (RN.LE.0.20) THEN
            USERF = 8.0
         ELSE IF (RN.GT.0.20) THEN
            GO TO 2
         ENDIF
      ENDIF
      RETURN
C
      END
*
*********************************************************************
********************  END OF FORTRAN CODE  **************************
*********************************************************************
*EOR
```

80N

Appendix E

Verification Program

```
GEN,POE,VERIFY,1/3/83,1,Y,Y,Y,Y,Y/F;
LIMITS,1,2,500;
NETWORK;
      CREATE,.5,0,,1000,1;
      ASSIGN,ATRIB(1)=EXPON(110.0,8),ATRIB(2)=UNFRM(2.0,14.0,9);
      EVENT,1,1;
      TERMINATE;
      ENDNETWORK;
INIT,0,700;
SIMULATE;
FIN;
```

```
      SUBROUTINE EVENT(I)
      GO TO (1),I
1     CALL HOT
      RETURN
      END
C
C
C
      SUBROUTINE HOT
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP
     +,NCLNR,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT
     +,TNOW,XX(100)
      COMMON/UCOM1/X1,X2
      SAVE/UCOM1/
      X1=0.0
      X2=0.0
      X1=X1+ATRIB(1)
      X2=X2+ATRIB(2)
      CALL OUT
      RETURN
      END
C
C
C
      SUBROUTINE OUT
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP
     +,NCLNR,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT
     +,TNOW,XX(100)
      COMMON/UCOM1/ X1,X2
      SAVE/UCOM1/
      WRITE(10,'(2F11.5)') X1,X2
      RETURN
      END
```

VITAE

Vernon H. Hamilton was born on May 26, 1947, in Lake Charles, Louisiana, where he lived until graduating from high school in 1964. In 1968 he graduated from Southern University in Baton Rouge, Louisiana, receiving a Bachelor of Science degree in Chemistry.

He entered the Air Force in 1969 and was commissioned at Officer Training School. After attending Undergraduate Pilot Training at Vance AFB, Oklahoma, he flew the T-29 Flying Classroom at Mather AFB, California from 1970 to 1974. From 1973 to 1976 he served as an Equal Opportunity Officer at Mather AFB, at Udorn RTAFB, Thailand, and finally, in an Air Staff Training Assignment (ASTRA) at the Pentagon. While stationed at the Pentagon, he checked out in the T-39 at Andrews AFB, Maryland.

Following the ASTRA assignment, he was assigned to Rhein Main AFB, Germany, where he served as a C-9 aircraft commander and instructor pilot in both aeromedical evacuation and operational support airlift missions from 1976 to 1981. While in Germany he earned a Master of Science degree in Human Resources Management from the University of Utah. He entered the Graduate Strategic and Tactical Sciences program at the Air Force Institute of Technology in 1981.

He is married to the former Anna B. Porter of Atlanta, Georgia. They have one daughter, Erica.

Permanent Address: P.O. Box 574
Lake Charles, Louisiana 70602

Major Jerry W. Poe was born on 25 October 1946 in Winnsboro, Texas, the son of Clifton F. and Martha L. Poe. Upon graduation from high school in Gonzales, Texas, he enrolled at Southwest Texas State University, San Marcos, Texas, where he received the degree of Bachelor of Science in Mathematics in May 1968.

He entered the Air Force in 1968 and attended undergraduate Pilot Training at Del Rio, Texas and received his pilot wings in July 1969. He served as an aircraft commander, C-123K, while assigned to the 310th TAS, from December 1969 to October 1970. After completion of his tour in Vietnam, he returned to Air Training Command where he served until 1975 as an instructor pilot; Chief, Wing Flight Management; and, Chief, Life Support Branch. He was assigned as Commander, 3796th Student Squadron, Sheppard AFB from 1975 to 1978. He served as C-130H aircraft commander, and Chief of Flight Safety, 463rd TAW, Dyess, AFB, Texas. While at Dyess he earned a Master of Public Administration from the University of Oklahoma. In August 1981, he entered the School of Engineering, Air Force Institute of Technology.

He is married to the former Rebecca Susan Walcher of Denver, Colorado. They have a daughter, Deborah Marie and a son, Bryan Wayne.

Permanent Address: Route 2 Box 105AA
Gonzales, Texas 78629