Theses and Dissertations   1. Thesis and Dissertation Collection, all items

1993-06

A cost benefit analysis of two products of the Fleet Numerical Oceanography Center

Brown, James L., Jr.

Monterey, California. Naval Postgraduate School

https://hdl.handle.net/10945/24139

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE June 1993	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE

A COST BENEFIT ANALYSIS OF TWO PRODUCTS OF THE FLEET NUMERICAL OCEANOGRAPHY CENTER

5. FUNDING NUMBERS

6. AUTHOR(S)

BROWN, James L., Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release, distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

A cost benefit analysis of the Fleet Numerical Oceanography Center (FNOC) is conducted with specific attention to the Optimum Path Aircraft Routing System and the Optimum Track Ship Routing System. These two products out of the many produced by FNOC comprise the bulk of the savings realized by the U.S. Navy through FNOC's work. The Optimum Path Aircraft Routing System (OPARS) is evaluated using modified flight plans received by the system. These plans were resubmitted to OPARS to determine the range of fuel usage around the optimum provided to OPARS.

The optimum Track Ship Routing System (OTSR) is evaluated using an adaptation of Dijkstra's algorithm to determine the optimum routing if perfect wave height information were available compared to a purely greedy strategy capturing the shortest arc available enroute to the destination. The damage sustained is compared to actual damage reported to the Naval Safety Center to determine the savings to the U.S. Navy from the OTSR system.

14. SUBJECT TERMS

Cost Benefit Analysis, Optimum Path Aircraft Routing System (OPARS), Optimum Track Ship Routing System (OTSR), Shortest Path

15. NUMBER OF PAGES 70

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

Standard Form 298 (Rev 2-89)

A Cost Benefit Analysis
of Two Products of the
Fleet Numerical Oceanography Center

by

James L. Brown Jr.
Lieutenant, United States Navy
B.S., University of Pittsburgh, 1985

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 1993

ABSTRACT

A cost benefit analysis of the Fleet Numerical Oceanography Center (FNOC) is conducted with specific attention to the Optimum Path Aircraft Routing System and the Optimum Track Ship Routing System. These two products out of the many produced by FNOC comprise the bulk of the savings realized by the U. S. Navy through FNOC's work. The Optimum Path Aircraft Routing System (OPARS) is evaluated using modified flight plans received by the system. These plans were resubmitted to OPARS to determine the range of fuel usage around the optimum provided by OPARS.

The Optimum Track Ship Routing System (OTSR) is evaluated using an adaptation of Dijkstra's algorithm to determine the optimum routing if perfect wave height information were available compared to a purely greedy strategy capturing the shortest arc available enroute to the destination. The damage sustained is compared to actual damage reported to the Naval Safety Center to determine the savings to the U. S. Navy from the OTSR system.

TABLE OF CONTENTS

v

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

A. BACKGROUND

Fleet Numerical Oceanography Center (FNOC) provides numerical and oceanographic products for use by subordinate and individual commands. FNOC is the master computer center for the Naval Oceanography and Meteorological Support System (NOMSS). Operationally, FNOC falls under the Naval Oceanography Command. Data is received from around the world and is used to produce a wide variety of products designed to benefit the military, particularly the U. S. Navy. Several broad classifications of the types of products exist. These are atmospheric weather conditions, ocean weather conditions, radar propagation data, and underwater conditions.

Atmospheric weather condition data is used primarily by the Optimum Path Aircraft Routing System (OPARS). This system is a computer model that receives requests for flight plans directly from individual users. The program then processes the request and sends out an optimal flight plan based on actual or climatological weather conditions. The choice of actual or climatological conditions is based on the requested date for the flight plan. FNOC estimates that this computer model saves the military ten million dollars annually in fuel costs.

Ocean weather data is provided to two centers, one in Norfolk, Virginia and the other in Pearl Harbor, Hawaii. These centers use this data to provide USN, USNS, and contracted vessels with Optimum Track Ship Routes (OTSR). Currently, these routes are manually generated and distributed. In this case FNOC estimates that the annual savings in fuel and damage costs is seventeen million dollars.

Radar propagation data is computed using atmospheric weather conditions. The daily radar range estimates are generated by onsite computer models and distributed to requesting commands. Because of the diversity of radars in use by the military, this model incorporates parameters specific to the requesting command's radar. FNOC has no estimate of the benefit provided by these models.

Underwater conditions are also predicted by computer models. The output from these models is then used by other computer models to determine estimates of sonar ranges for various ship configurations. Ranges are predicted for both active and passive sonars and sonobouys. As with the radar propagation data, requesting commands provide the type of sonar or sonobouy along with other operating parameters.

Of the products discussed above, two are most beneficial during peacetime. The first is the Optimum Track Ship Routing (OTSR) and the second is the Optimum Path Aircraft Routing System (OPARS). As FNOC has estimated, these two products generate a savings to the military in excess of seven million

dollars over FNOC's operating budget. These estimates are based on conjecture and do not have an underlying model to support them.

B. PROBLEM STATEMENT

It is desired to produce a model that can be used to determine the benefit gained from the products generated by FNOC. Since OTSR and OPARS appear to provide the large bulk of the peacetime savings, they will be addressed in the following analysis.

OTSR provides a recommendation to ship captains and masters on the track that would save the most time and fuel and result in the least damage to the ship from weather. Without this system, ships would take routes based on historical climatological data. In fact, while the first few days of the initial OTSR uses actual weather forecasts, the remainder of the voyage is based solely on climatological data, just as any captain would plan his route. It is the routing updates generated by the centers using ongoing weather forecasts that generate the most cost savings.

Several studies have looked at this system, the latest being completed in 1976 by Lulejian & Associates, Inc. [Reference 1]. Although detailed, this study looked only at the costs associated with the weather centers that actually produced the OTSR and not the costs incurred by FNOC in providing the required information. Realizing that there is

3

also a cost incurred in gathering the information and providing it to the ships, aircraft, and personnel, this paper will address only those costs incurred by FNOC and the two centers. The reason for this is that FNOC is only one of several weather data collection agencies; the National Weather Service and the Air Force Weather Center receive the same information. Additionally, much of the data used by FNOC is collected in conjunction with routine military operations, with the exception of hurricane/typhoon locator flights. These flights, in the absence of FNOC, would be conducted for the National Weather Service to provide early warning to coastal regions that may be affected by the storm. It is therefore concluded that data collection is not unique to FNOC and will not be considered as a cost.

In order to determine the benefit received from the OTSR system, it will be necessary to determine how ships would be routed in the absence of OTSR and how they would be routed with perfect information. Routings can be made in three different ways. The great circle route is the shortest distance that can be travelled, and is also the easiest and least costly to calculate. This choice, however, could result in severe weather encounters, thereby negating any fuel savings with damage costs. This route would provide an upper bound on the cost in fuel and damage, because even without OTSR, a better route could be chosen.

The second alternative is a route based solely on climatology. Routings of this sort have been conducted for centuries. Although more costly than a great circle route to compute, nearly any captain with access to pilot charts can compute a climatological course. Again, however, the danger of encountering severe weather still exists since climatological routes are only an expectation of future weather in a region.

The most desirable choice is a route based on perfect information. If exact weather conditions could be predicted, the optimum route in terms of fuel savings and damage avoidance could be chosen. The OTSR system provides a route that represents the cost of expected damage and fuel consumption that lies somewhere between the cost incurred by climatological routes and that of a route based on perfect weather information. It is not the aim of this thesis to perfectly predict weather conditions, but to determine the savings of the current system over the use of climatological routes. Climatological routes in this case provide a worst case situation in determining the cost of expected damage and fuel consumption. Routes costing more in damage could be chosen, but this is unlikely. Traditionally, ships have followed established climatological routes to obtain the least cost due to damage in the long run. By determining the expected cost when using climatological routes, an estimation of the benefit gained by OTSR can be determined. The major

5

benefit of OTSR over climatological routes is that OTSR takes into account current and forecasted weather in order to determine a route. Additionally, as forecasting continues during the route, adjustments can be made to take advantage of unexpected fair weather in a region that would otherwise be avoided by climatological routes. It is the dynamic nature of OTSR that allows it to make great gains in damage avoidance and fuel savings.

OPARS is a computer program that provides direct access to users on optimum paths for aircraft based on the following:

- aircraft performance parameters
- weather conditions
- minimum fuel consumption or least time enroute for the flight requested.

The majority of the flight plans are generated for Navy and Coast Guard units, with the Air Force and Army making up about 20 percent of the requests.

The major cost savings associated with OPARS are fuel, damage avoidance, flight time, and flight planners' time. As with the OTSR, these flight plans are only recommendations. Other operational considerations may preclude the use of the optimal flight plan.

Flight plans can be calculated in the same manner as the ship routes. That is, by great circle, climatological route, or perfect information. Currently flight plans produced by

FNOC are better than climatological routes but fall short of the optimum that could be obtained with perfect information. As with the OTSR, the problem is to develop a model that will simulate the route that would be chosen if the OPARS model were not available. This will be accomplished by modifying actual flight plans to determine the range of fuel consumption around the optimum flight plan chosen by OPARS.

The unaided flight planner would be required to sift through all applicable weather information to determine the optimum route by hand. This would be the same information that is currently provided to OPARS. It can be expected that an experienced flight planner would choose a route that is close to the optimum chosen by OPARS. The modified flight plans will provide the range of fuel usage around the optimum. Therefore, the amount of fuel and the amount of flight planning time that is saved by OPARS is a significant measure of its effectiveness and worth.

II. METHODOLOGY

A. ASSUMPTIONS

This section is a brief description of the assumptions necessary for the model formulation. A more detailed description of the assumptions made here follows in later sections.

1. Optimum Track Ship Routing System

The following assumptions are necessary regarding the analysis of the OTSR system.

- Perfect information results in minimal transit cost due to damage and fuel consumption.

- Climatological routing gives an upper bound on cost due to damage.

- Ocean grid gives rise to a sparse graph since all points are not directly accessible from a given point.

- Spruance class destroyer as a representative ship for model.

- Wave height is the only significant parameter involved in ship damage.

- Wave heights at grid points are independent.

- The conditional probability of damage given sea height is known.

- 95 percent of routings are accepted by Commanding Officers.

Further explanation of these assumptions is in the following section.

8

2. Optimum Path Aircraft Routing System

The assumptions necessary for OPARS are as follows:

- Fuel and flight planner's time provide the savings.

- OPARS route is optimum.

- The bulk of the fuel savings is realized by a small number of aircraft that fly the majority of the flight plans.

- The unaided flight planner would choose a flight plan within 4000 feet of the optimum altitude, normally distributed about the optimum altitude.

- OPARS is capable of calculating the fuel required for alternate non-optimal flight plans.

- All flights are flown using an OPARS route.

- The FNOC weather model is accurate.

Further explanation of these assumptions is in the following section.

B. DISCUSSION OF ASSUMPTIONS AND MODELING

1. Optimum Track Ship Routing System

The difficulties in routing ships optimally on long voyages are numerous. Forecasting techniques are only accurate for up to a few days from the forecast date. Since ships move relatively slowly, great care must be taken to avoid placing a ship in a situation from which it cannot easily escape. Routings must be closely monitored and updated continuously as weather conditions change. In this way, OTSR uses a somewhat greedy strategy in that the initial three to four days transit is based on forecasted weather and the

remainder of the initial routing is based on climatological data. It is easy to see that this will not always lead to the optimum routing in terms of fuel and damage avoidance as could be expected with perfect weather information. By the choice of the initial days of the route, future options to take advantage of fair weather may not be available. In other words, some damage can be expected even on an OTSR route. This leads to the first assumption, that perfect information would result in minimal cost due to damage and fuel consumption and that climatological routing would result in an upper bound on cost since this is the worst routing that could be chosen using all available information with the exception of OTSR. Prior to OTSR, climatological routing was the best available choice.

A second assumption that is necessary for the forthcoming model is that not all points in the ocean are accessible from the ship's current position. The model used to calculate the cost of a route uses a grid in which each point is separated by five degrees of latitude and longitude. From the ship's current point, three points are accessible to it, namely five degrees of longitude further along its track and its current latitude plus or minus five degrees. This gives rise to a sparse graph that can be used in the shortest path algorithm to be described later.

The model will also use a Spruance class destroyer as the platform to calculate fuel usage for the route. This is

soon to be the most prevalent engineering plant in the fleet, and its fuel usage is representative of the fleet.

The only parameter to be used in calculating the damage that a vessel encounters on the route will be wave height. Previous studies [Reference 1 and 2] have shown that wave height has the most significant effect on the damage to a vessel. Wave period and direction also play an important role, especially if the period is such to cause resonance at the current speed. This problem can easily be solved by an adjustment of ship speed. Nagel [Reference 2] has shown that the effect of this speed decrease is small between the optimum route and the climatological route. That is to say, it is felt that the benefit of optimum routing is greater in terms of damage avoidance than in terms of time saved.

The probability of a particular wave height at a given point on the grid described earlier is derived from climatological charts in the Defense Mapping Agency's Sailing Directions [Reference 3]. The probability of sea height in these charts is based on observed wave height during a specific month in the case of the North Atlantic Ocean or a specific season in the case of the North Pacific Ocean. The use of a five degree grid was chosen to gain independence of the wave heights from one grid point to the next. Allowing roughly 300 nautical miles between points, creates a large enough fetch for seas to fully develop in that region and not necessarily be influenced by an adjacent region. That is, the

11

area of ocean covered by each grid point is large enough so as to maintain its own sea height without regard to adjacent conditions.

Finally, values for the probability of damage given sea heights for particular ship types is not known, nor is it necessary in order to develop a relative cost for climatological routes over routes based on perfect information. Aggregate values for the conditional probability of damage given sea height have been determined [Reference 1]. These conditional probabilities are based on historical data from July 1969 to June 1975 from records of the Naval Safety Center, and are recreated in Table I.

Table I   CONDITIONAL PROBABILITY OF DAMAGE GIVEN SEA HEIGHT WITH AVERAGE COST OF DAMAGE [REFERENCE 1]

Sea Height, x (in feet)	Conditional Probability of Damage Given Sea Height	Average Damage per Incident (dollars)
0<x<4	0.0000	0
4<x<8	0.0001	48427
8<x<12	0.0008	
12<x<16	0.0009	129969
16<x<20	0.0118	
20<x<24	0.029	312196
24<x<28	0.0700	
x>28	0.2860	340771

All costs have been converted to 1992 dollars using six percent inflation. The dollar amounts are estimates based on the Commanding Officer's assessment of the damage and are therefore not actual cost to repair the damage. For this reason, it is felt that these values are lower than the actual cost of repairs. Data for the average damage per incident was available only for eight foot increments from 4 to 28 feet. The value in each eight foot increment will be used with the conditional probability of damage given sea height values within this increment. The probability of damage given here is without regard to ship type or class, but is pooled from available data from the Naval Safety Center.

As mentioned earlier, no follow-up by the Naval Safety Center is conducted to determine actual costs of damage by unfavorable weather conditions. These figures represent an estimate of the cost to repair damage. They do not include the cost in loss of availability of the ship's services. Loss of availability may or may not be applicable. Much of the damage caused by adverse weather is not of a serious enough nature to require the ship to be taken out of action to repair. The bulk of the damage can be repaired during scheduled maintenance periods and would therefore not impinge on ship operations. Table II is a summary of damage sustained by USN and USNS vessels for the period from January 1982 to May 1992.

Table II SUMMARY OF ACTUAL DAMAGE SUSTAINED JANUARY 1982 TO MAY 1992, FROM NAVAL SAFETY CENTER DATA.

	Totals	Ave. per Year
Damage Cost	38.659 million	3.741 million
Ship Days Lost	172	16.6
Lost Work Days	617	59.7
Fatalities	14	1.4
No. of Incidents	279	27

Figure 1 shows the monthly mean of the damage for the same time period. The means follow closely what would be expected during the winter and summer seasons, with the exception of April and October. This was due to a single unusually high cost in each of these months. If this value is eliminated in each month, the means are as depicted in Figure 2.

Table II and Figures 1 and 2 are introduced as indicators of the damage sustained by vessels even while operating under the OTSR system. Although OTSR will be shown to be very beneficial, we cannot assume that the system is perfect and not without limitations. Even under a routing system like OTSR, some damage will occur.

OTSR routings are advisory in nature. There currently are no requirements for Commanders and Commanding Officers to follow these routes. In the case of OTSR, routings are viewed

14
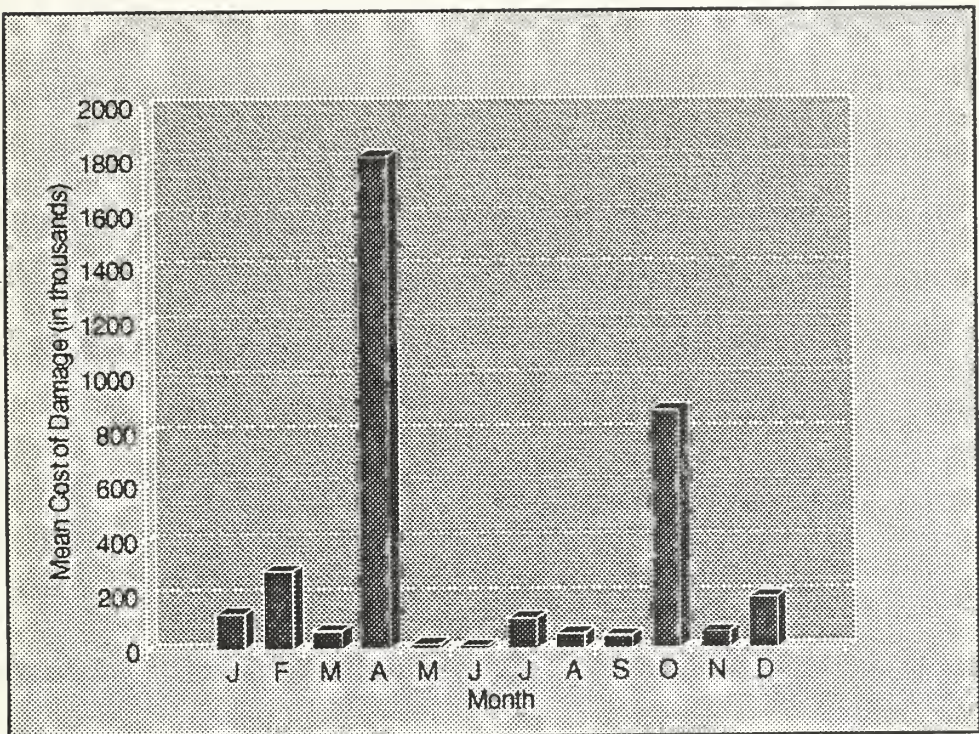
Figure 1   Mean Damage Cost by Month
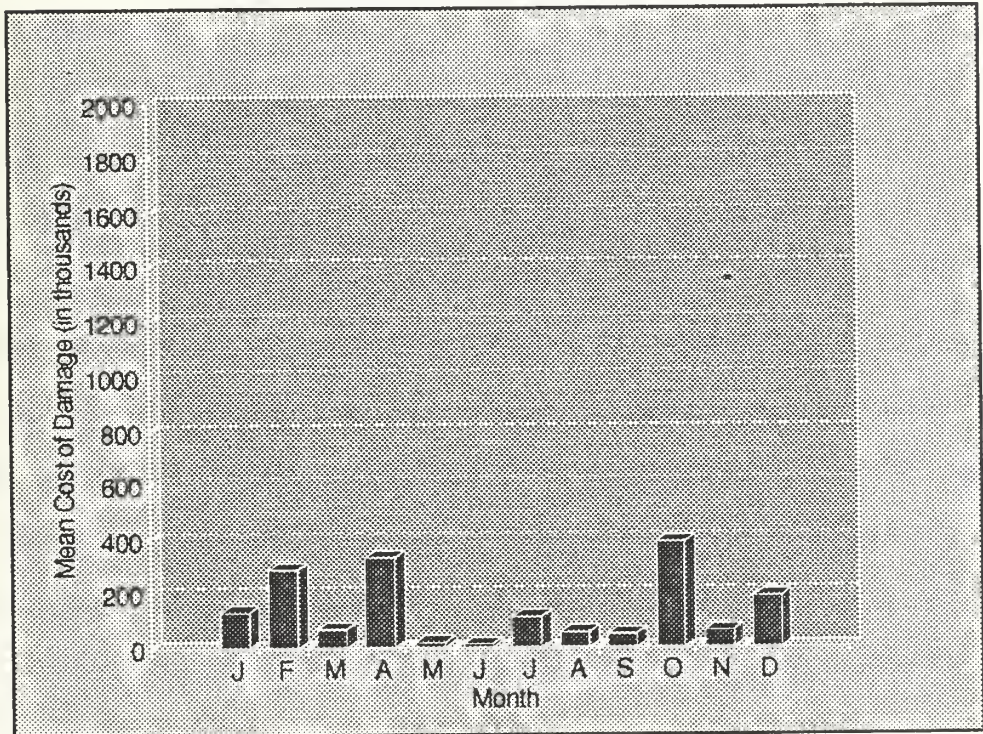
Figure 2   Mean Damage Cost by Month, Abnormal Values Removed

15

as extremely beneficial by Commanding Officers. This is evident in a high acceptance rate of the recommendations. An acceptance rate of 95 percent has been shown [Reference 1]. The routings are not perfect though. In this same study, it was shown that approximately 11 percent of the routed ships received routing changes during their voyages. These changes were to:

- avoid adverse weather, and
- take advantage of unexpected favorable weather.

The results of these route changes are shown in Table III.

Table III   SEA   STATE   ALONG
DIVERTED ROUTE

Better	82 percent
Equal	12 percent
Worse	6 percent

As shown, in Table III, upon analysis of the weather conditions of the route taken and the recommended route, 94 percent of the ships that were rerouted experienced seas of equal or lesser severity. Only six percent encountered more damaging seas. As the above study [Reference 1] went on to show, four of those ships rerouted chose not to follow the

recommended course changes. Of these, two encountered rough seas and one ship suffered damage.

An acceptance rate of 0.95 will be used throughout this analysis. Additionally, ships that do not accept OTSR recommendations will encounter heavier seas at the rate of 0.5 from observations [Reference 1] of ships that chose not to accept rerouting. Those ships that do follow OTSR rerouting directions will experience heavier seas at the lower rate of 0.06, reflecting the error rate in OTSR rerouting.

The following discussion will describe a method to determine the savings by optimally routing ships vice routing by climatological data.

In the absence of OTSR, ships would be routed with climatological and short range weather predictions. Historical wave height information is available from climatological charts [Reference 3]. In this model, each five degrees of latitude and longitude in the North Atlantic and North Pacific is assigned a probability of a specific wave height. In the North Atlantic this is the probability of seas greater than 12 feet and greater than 20 feet. For the North Pacific it is the probability of seas greater than 8 feet and greater than 12 feet. Data for the month of January in the North Atlantic will be used to estimate the savings gained from using OPARS. The region used extended from 080°W to 005°W and from 20°N to 65°N. Data points were numbered from one at 080°W, 20°N, north to ten and then starting again at

17

075°W, 20°N with eleven and continuing in that fashion to 160 at 005°W, 65°N.

As described earlier the graph that is developed is sparse with only a limited number of points accessible from the parent node. All arcs in the graph are directed, and movement is allowed only in one direction. Additionally, all costs of moving from one node to the next are positive so that no negative cycles can develop. To determine the shortest path, i.e., the optimal route through this directed acyclic graph, Dijkstra's algorithm [Reference 4] is used. The complete algorithm is contained in Appendix A. At each node, a random number is drawn from a uniform distribution to determine the wave height at that node. The climatological charts provide the probability of wave height for all ocean areas. In this way, a situation is produced in which perfect weather information is known for the entire region. A second random number will be drawn and compared to the probability of damage given sea height from Table I. The damage and the fuel used to arrive at that node provide the cost of transit to that node. The distance between nodes is computed using the following formula:

$$Dist = 60 \arccos[\sin L_s \sin L_d + \cos L_s \cos L_d \cos(\lambda_d - \lambda_s)]$$

where:

- L_s is the latitude of start

18

- L_d is the latitude of destination
- λ_s is the longitude of start
- λ_d is the longitude of destination, and
- Dist is the great circle distance between start and destination. [Reference 5]

Dijkstra's algorithm adds each node to a set of completed nodes one at a time. As the node is added, all arcs leaving the node are examined and adjacent nodes are updated if an improvement in the distance to that node is found. The algorithm computes the distance from the source to all other nodes. By choosing the destination and tracing back through the parents, the shortest path is defined.

In order to determine the cost of routing ships in the absence of OTSR, a second algorithm is used. Utilizing the same random number seed as above, wave heights and the probability of damage conditions were duplicated. This time though, instead of determining the optimum route, a greedy strategy was employed. The complete algorithm is contained in Appendix B. Prior to the start of the algorithm, all nodes that can lead to the destination are marked. From the start node then, the least cost route to the next accessible marked node is chosen. From this node then, the next least cost arc is chosen until the destination is reached. In this way, a cost can be determined for a route in which a ship has placed itself in a situation where high seas must be encountered to reach the destination.

After multiple runs of each algorithm, the cost of the damage incurred and the fuel used on the shortest path is compared to the cost of the non-optimal route determined in the second algorithm. A single route will be considered. To determine an annual savings it will be necessary to multiply the mean savings from the algorithm by 99 routings per month times twelve months, for the North Atlantic, and 126 routings per month times twelve months times 2.5 to compensate for the greater distance traveled in the North Pacific. This will provide a mean savings if perfect information were available. From Table II it is known that 3.741 million dollars in damage is sustained each year under OTSR. A yearly mean was determined from the 124 months of data obtained. The difference between this value and the cost of damage incurred on the optimal routes will be subtracted from the final value determined for the savings. This will provide an estimate for the savings possible under OTSR.

2. Optimum Path Aircraft Routing System

In creating a model to establish the benefit gained from using a computer model to optimally route aircraft over routing each aircraft by hand, it must be established what it is that provides the greatest gain. Overwhelmingly, the answer is fuel savings. Unlike ships, aircraft are capable of quickly changing course to avoid adverse weather conditions.

Additionally, routings usually are much shorter in duration and are therefore able to take full advantage of short range weather forecasts.

The Naval Safety Center does not classify weather as a cause for aircraft damage. This is because the aircraft is either all-weather or it does not fly when forecasts show that the aircraft would encounter adverse weather. It is concluded that fuel savings from the optimal routing of the aircraft is by far a greater indication of its benefit than is damage from adverse weather.

Secondly, it is assumed that the route provided by OPARS is indeed the optimum route, since it is not feasible to verify this by hindsight routing of the aircraft. In any case, OPARS is the closest routing system available to the true optimum. The desire is to determine how much this system benefits the military when compared to the next best alternative, i.e., manually computed routes, and not to determine how much could be saved with some other system.

The OPARS database is capable of providing routings for over ninety aircraft types. A limited number of these aircraft types were chosen for study. Eleven aircraft and their variants were chosen for use in the analysis. These eleven aircraft cover twenty-one of the variants for which OPARS is able to provide flight plans. The selected aircraft also comprise over 70 percent of the legs computed by OPARS. The significance of the chosen aircraft is also apparent when

considering the savings that FNOC has determined from using OPARS. The selected aircraft account for over 80 percent of the fuel savings as calculated by FNOC. The FNOC formula used in calculating this savings will be discussed later. Table IV shows the aircraft that are used in this analysis.

Table IV   AIRCRAFT USED AS BASIS FOR ANALYSIS

T43	P3C
C9D	C20D
UC12	C9B
P3A	P3B
HC130	DC9
KC130	

Ā means of determining the route that the unaided flight planner could reasonably be expected to choose in lieu of the optimum route must be determined. If it is valid to conclude that the flight planner would not be far off in his estimate of the optimum altitude at which to fly, then an altitude of 4000 feet on either side of the optimum should include even the most uneconomical of plans that the flight planner would choose. This is reasonable due to extensive training that pilots receive in flight planning and their intimate knowledge of their aircraft. It has been indicated, that for the S-3 aircraft [Reference 6] pilots typically

22

select an altitude as much as 10,000 feet below the optimum altitude. Since information of this type is not available for all aircraft, the value of 4000 feet was chosen here. This will tend to under estimate the savings by OPARS if indeed the pilot's range of error is greater than 4000 feet. For the purposes of this analysis it will be assumed that the manual flight planner will choose a flight plan that is the optimum route for the altitude chosen. The overall route, as far as way points chosen, will be the same, but the route between way points will be allowed to vary in order to optimize the route at each altitude. OPARS will select the optimum jet route between user way points. These jet routes may differ at different altitudes.

Finally, OPARS is capable of calculating the fuel required to fly the alternate flight plans that could be chosen by the flight planner. This is necessary for the comparisons to be conducted in the model that will be discussed later. The fuel that OPARS calculated to be used on the alternate routes will be optimum for that route. It is unlikely that the flight planner would be this accurate in his/her calculations. Therefore, this will be a lower bound on the percentage of additional fuel that the manual planner would require for the flight.

Information is not available on the rate of acceptance of OPARS flight plan recommendations. Independent studies from two aircraft communities [References 6 and 7] have shown

23

faith in the system. Additionally, all communities have submitted estimates of the fuel savings that they feel are afforded by use of OPARS. Their estimates are used by FNOC in its calculation of the benefit of OPARS from fuel savings. It will be assumed for this analysis that all flights are flown using an OPARS recommendation. The approach used in the analysis of the OPARS system is to determine what could be saved by the model. Because of the short duration of the flights and the ability to obtain a routing just minutes prior to the actual flight time, it can be assumed that perfect information is available. The main factor that would decrease the realized savings is the accuracy of the weather information itself. For OPARS, it will be assumed that the weather model used by FNOC is accurate.

Currently, FNOC uses the following formula to calculate the savings from OPARS:

$$L \times 0.7M \times \theta = F$$

where:

- L is the number of legs flown,
- M is the maximum internal fuel load,
- θ is the percent of fuel estimated to be saved by OPARS, and
- F is the total fuel saved, in pounds.

This formula has three major faults. The first is that θ is an estimate provided by the squadrons and has no

underlying analysis. Secondly, not all flight plans are operational plans, as is assumed by the above equation. A portion of the plans submitted to OPARS are duplicates or are for more than 72 hours in the future. The third fault is that the formula assumes that all legs are loaded to 70 percent of maximum internal load. With increased pilot awareness of fuel conservation, it is felt that this is too high. It will be shown that a lower figure should be used.

The proposed solution to this formula is,

$$L \times \psi \times M \times \theta^* \times P = F$$

where in addition to the FNOC formula:

- ψ is the new value for percent fuel load,
- θ^* is the new estimate of savings and,
- P is the probability that the plan is an operational plan.

Without OPARS, flight plans would have to be manually planned. In order to determine what the unaided flight planner would choose for a flight plan, a range of altitudes must be decided upon. As earlier described, a value of 4000 feet on either side of the optimum was chosen.

Actual flight plan requests were captured for a forty-eight hour period. These plans were then modified to force OPARS to compute the fuel necessary to fly at specific altitudes. In this way, the amount of fuel, over the optimum, necessary to fly at the various altitudes can be computed.

Once the minimum amount over optimum is determined, a mean and standard deviation from all aircraft types can be found.

The choices that the unaided flight planner will make will be normally distributed about this mean out to 4000 feet on either side. The normal distribution function combined with the fuel use curve to be developed and the number of flight plans generated will provide an estimate of the savings from OPARS over manually generated flight plans.

III. RESULTS

A. OPTIMUM TRACK SHIP ROUTING SYSTEM

To determine the benefit from OTSR, the shortest path algorithm and the greedy algorithm described earlier were used. The probability of wave heights greater than twelve feet were obtained for the month of January from climatological charts for the North Atlantic [Reference 3]. The month of January was chosen because this provided the worst case for sea conditions. This would provide an upper bound on the damage avoidance estimate for OTSR.

For each algorithm, optimal and non-optimal, a route from node 14 (075°W, 35°N) to node 157 (005°N, 50°W) was used. Once the difference between the mean values for the damage sustained on the optimal and non-optimal routes is determined, it will be multiplied by the mean number of routings conducted per month. It is not possible to accurately determine a fuel savings from comparison of the fuel used on each of the routing techniques due to the general nature of the fuel calculations. By using the fuel required by the general calculation, it was possible to determine the most economical path under each of the routing schemes used. If the only criterion for determining the route had been damage cost, the algorithms would have chosen a path to avoid damage even at

27

the expense of much higher fuel consumption. Clearly this would not provide the optimum route. As discussed earlier, it has been shown by other studies that the savings in fuel from optimum routing of ships is far outweighed by that of damage avoidance.

For the North Atlantic, 99 routings are processed per month. As discussed earlier, five percent (5) of these vessels will not follow the OTSR recommended track, and of those not following OTSR, fifty percent (2.5) will encounter heavier seas and possibly sustain damage. Another eleven percent (10) will receive rerouting instructions and of these, eighteen percent (2) will encounter equal or heavier seas. In all, 4.5 ships per month will not be helped by the OTSR system. Therefore the estimate of the savings of the optimal over non-optimal routes should be multiplied by 94.5 vice 99, since the vessels not helped by OTSR cannot be counted as a benefit to the OTSR system.

A similar procedure can be used to calculate the actual number of vessels aided by OTSR in the North Pacific. In this case the number of vessels should be 126 vice the 132 actually routed by the Pearl Harbor Center. Additionally, since the length of the routes are typically 2.5 times longer in the North Pacific than in the North Atlantic, the cost of the climatological route found for the North Atlantic will be multiplied by 2.5 to estimate the cost of a climatological route in the North Pacific. Once a yearly savings in damage

28

avoidance is determined, the difference between the damage known to occur under OTSR and that incurred under optimal routing in known weather conditions can be determined. The results of this analysis using the route given above is shown in Table V.

Table V   COSTS ASSOCIATED WITH OPTIMAL AND NON-OPTIMAL ROUTES.

	Optimal Atl/Pac	Non-Opt Atl/Pac
Damage Cost/Voyage	433/1082	2166/5415
Ship Days Lost	30.6	152.7
Number of Incidents/Year	49.7	248.4
Damage Cost/Year Total	2,127,006	10,643,724
Cost of Ship Days Lost/Year	958,147	4,781,342
Total Cost	3,085,153	15,425,066

The number of incidents per year in Table V was determined using the rate of incidents from the two algorithms: 0.01 for the optimal, and 0.05 for the non-optimal. Ship days lost were determined using the ratio ship days lost to number of incidents from Table II. The cost of ship days lost per year is computed using the cost of MSC vessels per day at 31,312 dollars [Reference 8].

As shown in Table II and Table V, the cost of damage to ships with OTSR recommendations and the cost of damage on optimally routed ships is very close. Once the cost of damage

29

from Table II is deducted from the total cost of non-optimal routing, since this amount of damage will occur with or without OTSR, what remains is the savings attributed to OTSR. That savings is 11,684,066 dollars.

B. OPTIMUM PATH AIRCRAFT ROUTING SYSTEM

1. Discussion of Procedure

To determine the savings in fuel costs afforded by OPARS, it is necessary to determine the cost of non-optimal routes. As discussed earlier, this is accomplished using modified flight plans and resubmitting them to OPARS. Original flight plans were obtained as they were submitted by users to the OPARS model. In all, 364 flight plans were collected. After review, it was determined that 223 of these were in fact unique flight plans. The remainder, upon close examination, were found to be in one of the following categories:

- Duplicates,
- Slight modifications of a basic plan, or
- Requested for more than 72 hours in the future.

Duplicate plans could be readily eliminated. It is unknown why they were submitted, but it is assumed to be due to user impatience: at times, the queue of flight plans submitted may become long, and the user may feel that his flight plan was not properly submitted, so he resubmits it.

30

Flight plans that are slight modifications of another plan are much more difficult to eliminate. It is felt that these plans are an attempt by the user to experiment and provide a range of options available for the requested flight.

· The third category was eliminated because it is felt that these plans would be resubmitted at a future date that is closer to the actual flight time. In this way, actual rather than climatological weather would be used. Flight plans submitted for a flight time greater than 72 hours in the future can only be used for planning purposes and cannot be considered as operational plans.

A summary of these results is shown in Figure 3. The labeled aircraft are those that were used in the analysis of OPARS.

Of the flight plans considered operational, 57 where chosen at random from the eleven aircraft types chosen for the study. These flight plans were then modified to force OPARS to calculate the fuel required to fly at each of four different altitudes evenly spaced over the range of altitudes available to the aircraft. In this way, the fuel required for non-operational flight plans could be determined.

The altitude restrictions that were used are also available to the users. Twenty-one of the original flight plans contained constraints on the altitude, either as an upper altitude or lower altitude restriction. These constraints were removed and the flight plans were resubmitted
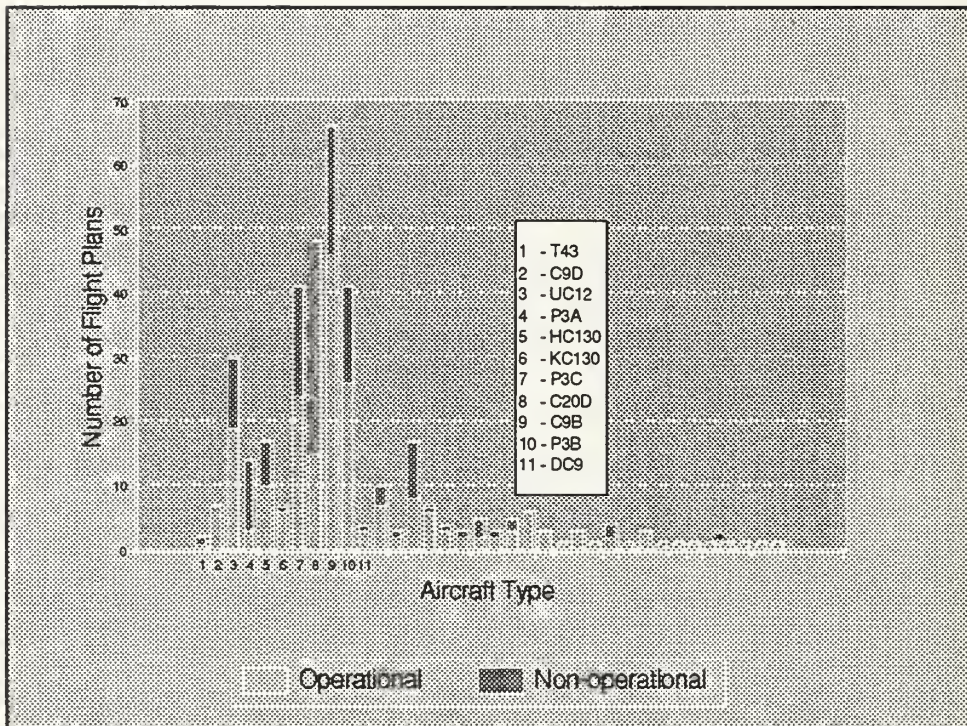
Figure 3   Operational and Non-operational Flight
Plans from the Sample Taken of User Inputs

along with the original and modified plans to provide an
alternate base line for the optimum. In all, 306 flight plans
were resubmitted to OPARS. Because of weight and climb
restrictions, not all flight plans could be processed. Of
those submitted 184 were successfully processed and provided
355 individual flight legs for analysis. Each flight plan
could contain up to six legs.

When OPARS provides the completed flight plan, three
alternate altitudes and the fuel required for that altitude
are also provided for each leg of the flight. These non-
optimal fuel requirements were combined with the fuel
requirements from the modified flight plans.

The optimum fuel required for each leg was compared to all available non-optimal fuel requirements for that same leg. A percentage of additional fuel required for each altitude was determined. All altitudes and the percentage of additional fuel required for that altitude within 4000 feet of the optimum altitude were retained. The altitude values were then coded to their distance from the optimum and are displayed in Figure 4. By coding the data, it was possible to compare different legs for different aircraft and altitudes on a common ground.

As expected, the additional fuel required for each leg increases as the distance from the optimal altitude increases. Also apparent from Figure 4 is that the minimum lies to the right of, i.e., at a higher altitude than, the optimum computed by OPARS. On further inspection, it was found that this was due to the altitude restrictions imposed by the user. These constraints inhibited OPARS from selecting the optimum altitude. Figure 5 shows the same information as Figure 4, but in this case, the fuel comparisons were made against the unconstrained flight plans vice the original as entered by the user.

The minimum to the right of the optimum still persists in Figure 5, but to a lesser extent. This appears to be due to OPARS reluctance to change altitude for a short leg if the preceding and succeeding legs are at the same altitude, so as to keep the flight plan at a level altitude.
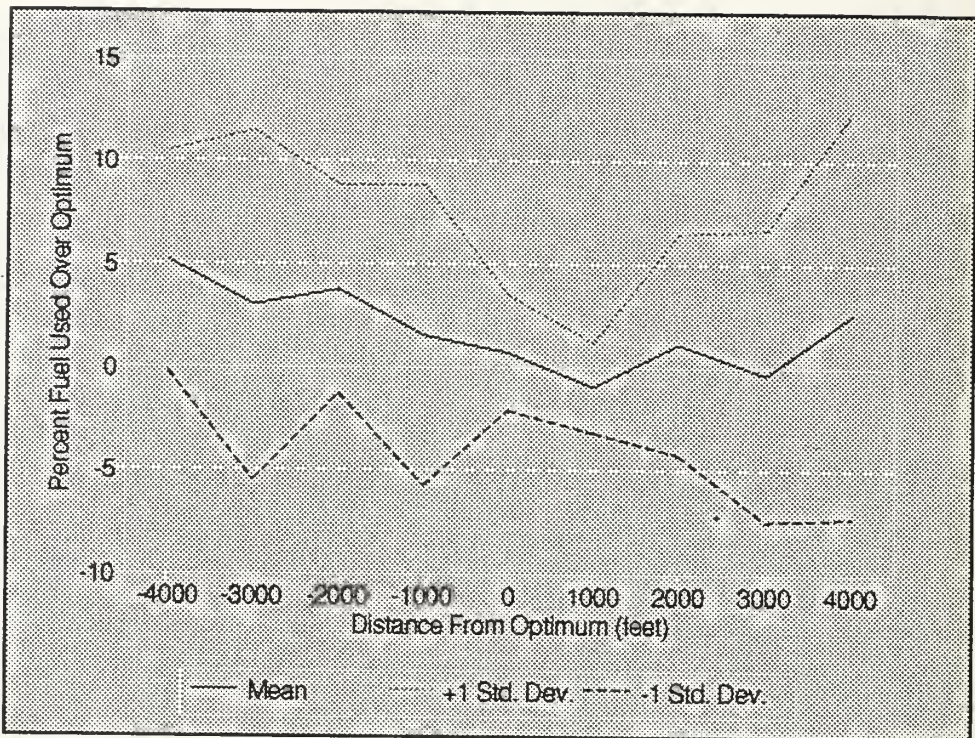
Figure 4   Percentage of Fuel Over Optimum as Compared to Original Flight Plans
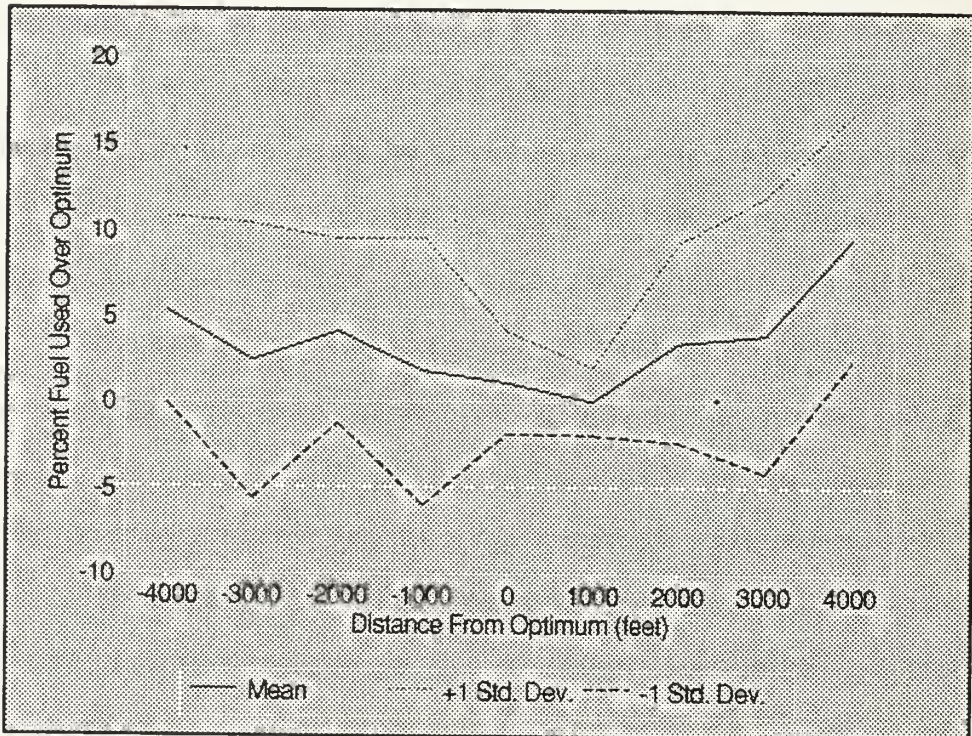
Figure 5   Percentage of Fuel Over Optimum as compared to Unconstrained Flight Plans

34

For the purpose of flight planning and navigation, altitude is not continuous, but discrete in thousand foot increments. For example an aircraft is assumed to be at 15000 feet if his actual altitude is between 14500 and 15500 feet. It is assumed, as discussed earlier, that the unaided flight planner would choose an altitude that is within 4000 feet of the optimum altitude. It is assumed that these choices will be normally distributed about the optimum with a mean of zero and a standard deviation of 4000. By converting the distance from the optimum altitude to a standard normal, and using the standard normal distribution function, the probability that the unaided flight planner will choose a specific incremental altitude can be determined. The probability of choosing a specific altitude and the mean percent fuel required above optimum at that altitude are shown in Table VI.

Case 1 compares the modified flight plans to the original flight plans. Case 2 compares the modified flight plans to original flight plans but, with the altitude constraints removed.

The expected savings in terms of percentage of fuel used over the optimum will be,

$$\Sigma P_a \times \Theta_a, \quad a = -4, -3, \ldots, 3, 4.$$

where,

- a is the altitude in thousands of feet from the optimum,

35

- P_a is the probability that a specific altitude is chosen, and

- Θ_a is the percentage of fuel used over optimum at that altitude.

Table VI   OPARS SAVINGS OVER UNAIDED FLIGHT PLANS

Alt, x from Opt. ft x100	Mean Percent Fuel Used Over Optimum		Prob. L<x≤U L=Lower U=Upper	Percent fuel Over Optimum Given Altitude	
	Case 1	Case 2		Case 1	Case 2
x≤-35	5.169	5.294	0.1908	0.986	1.010
-35<x ≤-25	3.022	2.366	0.0752	0.227	0.178
-25<x ≤-15	3.81	4.068	0.0879	0.335	0.358
-15<x ≤-5	1.547	1.746	0.0964	0.149	0.168
-5<x≤5	0.738	1.060	0.0995	0.073	0.105
5<x≤15	-0.950	-0.022	0.0964	-0.092	-0.002
15<x≤25	1.156	3.402	0.0879	0.102	0.299
25<x≤35	-0.400	3.832	0.0752	-0.003	0.288
x>35	2.608	9.545	0.1907	0.497	1.820
			Total	2.324	4.224

The value for $\Theta^{\cdot}$ in the following equation,

$$L \times \psi \times M \times \theta^{*} \times P = F,$$

where:

- ψ is the new value for percent fuel load,
- θ^{*} is the new estimate of savings and,
- P is the probability that the plan is an operational plan,

as computed from the previous equation, is then 4.224 for case 2 or 2.324 for case 1 as an alternate. The value of 4.224 is what would be saved by OPARS if the system were used without altitude constraints allowing OPARS to choose the optimum altitude without operator intervention.

As can be seen in Table VI, nearly 67 percent of the savings estimate comes from the tails of the altitude distribution. At each end of the 4000 foot range, the cumulative probability remaining in the tails is great, as is the percentage of fuel used over the optimum. As stated earlier, the choice of 4000 feet on either side of the optimum ensured that from available data, the true estimate of savings would be greater than the value determined here.

Currently, when computing OPARS fuel savings, FNOC uses 0.70 for Ψ, the percentage of maximum fuel load. Prior to the introduction of OPARS, it was routine to load aircraft to 100 percent of internal fuel load for every flight. When OPARS was introduced, the value of 0.70 was chosen to reflect

increased awareness of fuel conservation and to underestimate the savings of OPARS. With the further increases in fuel conservation by squadrons and the need to maximize the number of hours flown with the fuel at hand, it is felt that this value should be lowered to 0.40. To explain this further, it must be understood that in order to determine the savings of OPARS, it is necessary to analyze the savings at the level of the flight leg. The current method assumes that each leg is loaded to 70 percent of maximum internal load. For this to be the case, refueling would have to be conducted on each leg of the flight plan. This is not so. The mean loading by OPARS on an individual leg is 25 percent of internal capacity even taking refueling into account. Given today's concerns over fuel usage and conservation, the 40 percent chosen here for Ψ is felt to be an accurate estimate of fuel loading in the absence of OPARS.

Finally the probability, P, that a flight plan is an operational plan is determined from the number of operational plans observed in the sample. In the sample, 61 percent of the plans were operational. For this analysis, P will be set at 0.75 to eliminate the possibility of undercounting the number of operational plans.

2. The Savings in Fuel from the use of OPARS

For this analysis, the mean number of legs flown each month by aircraft type in 1991 was used to arrive at the savings in fuel by OPARS.

Using the revised estimation procedure for each aircraft type, OPARS is estimated to save 6.773 million dollars when using the preferred case 2 data, and 3.726 million dollars if the case 1 data is used. This is in comparison to FNOC's estimate of 8.348 million dollars.

IV.   SUMMARY AND RECOMMENDATIONS

A. OPTIMUM TRACK SHIP ROUTING SYSTEM

Calculations to determine the savings realized by the U.
S. Navy from OTSR are based on two related algorithms. The
first uses an adaptation of Dijkstra's algorithm to determine
the shortest path across the North Atlantic with all wave
heights known. The second algorithm uses a greedy strategy
and looks only at the next accessible nodes that can lead to
the desired destination and chooses the least cost of those
available.

Surprisingly, it was found that the annual damage costs
sustained under the first algorithm closely match the damage
costs that are experienced under the OTSR system. The
significance of this is not explored here, but it may be
possible to show that the OTSR route is quite close to the
true optimum route. The second algorithm was used to
determine damage costs in the absence of OTSR. In this case
Commanding Officers would be required to rely on
climatological or short range forecasts to choose their route.
As has been shown, this results in much greater damage costs.

To determine an estimate of the savings from OTSR, a
single route from the northeastern coast of the United States
to the southern tip of England was used. Random numbers

chosen from a uniform distribution determined wave heights and probability of damage at each node. The same wave heights and probabilities were used for each of the algorithms by using the same seed. Each algorithm was run one hundred times with different sea conditions and damage probabilities to determine a mean cost for the route under optimal and non-optimal routing.

This mean of the non-optimal routes was then multiplied by the number of routings per year in the North Atlantic and the North Pacific. In the North Pacific an additional scaling factor was used due to the length of voyages there. The annual cost of non-optimal routing once decreased for damage costs occurring even while under OTSR control was 11.7 million dollars.

B. OPTIMUM PATH AIRCRAFT ROUTING SYSTEM

This, the second product under study, was evaluated using the OPARS model itself. Modified flight plans were resubmitted to OPARS to determine the amount of fuel required for a non-optimal flight. The flight plans had originally been copied as the requests were received. They were then modified to require OPARS to determine fuel loading if the plan were flown at a specific altitude.

Eleven aircraft types were chosen and numerous flight plans from each were modified. Four altitudes were chosen for each aircraft, depending on its capabilities, in order to

bracket the optimum altitude computed for the original. For each flight plan then, five flight plans were resubmitted; the original and four at modified altitudes.

Once the fuel required for each altitude was determined, it was compared to the optimum. In this way, a percentage of fuel required over optimum could be computed. The distance from the optimum altitude was also determined. The percentage of fuel over optimum was plotted against the distance from the optimum altitude.

The percentage of fuel over optimum was multiplied by the probability of a manual flight planner choosing that altitude. The probability of choosing a particular altitude was based on a normally distributed random variable with mean zero and standard deviation 4000 feet. In doing this, an aggregate value for the estimate of fuel saved by OPARS was found.

Once this value was entered into the modified FNOC fuel equation, a fuel savings of 6.8 million dollars was estimated. The modified FNOC equation changes several of the parameters used by FNOC in their current calculation. First, the savings estimate described above is used instead of an estimate provided by each aircraft squadron. Next a new value for the percent of maximum fuel load that would be loaded in the absence of OPARS is used. Currently this value is at 70 percent. Actual loading by OPARS is 25 percent based on the observed flight plans. A value of 40 percent was used in the modified equation. Finally a parameter to indicate the

42

probability that the flight plan was actually an operational plan was added. A value of 0.75 was chosen for this parameter based on the observed flight plans.

C. RECOMMENDATIONS FOR FURTHER RESEARCH

As has been shown, FNOC does indeed provide a valuable service. The cost savings demonstrated here involved only two of the many products produced by FNOC. Further study should be devoted to quantifying the remaining products not covered here. Work must also be done to accurately determine the fuel savings that can be attributed to OTSR.

Additionally, work is being done to build a computer model that could be placed aboard ships to generate their own optimum routes. When this is accomplished, studies should be done to determine the added benefit from having this capability aboard ships.

In order to obtain more exact estimates of the savings from damage avoidance provided by OTSR, follow up to damage reports submitted to the Naval Safety Center must be conducted. The estimates provided here appear to be low and should be revised to obtain a more accurate benefit from OTSR.

The procedures used here for both OTSR and OPARS provide the framework for further study. In each case an estimation of the savings provided by the product is given. Further study should be given to sensitivity analysis of the parameters involved. Namely, in the case of OPARS, the

percentage of fuel used over optimum and the percentage of maximum internal load should be studied. Additionally, it will be necessary to more accurately determine the range of altitudes that would be chosen by manual flight planners. For OTSR, a more accurate method of calculating fuel use should be investigated to determine more accurately any benefit gained from fuel savings in optimum routing.

Additionally, in the case of OTSR, actual routings and weather conditions should be collected for use with the modified Dijkstra algorithm to determine possible future gains for OTSR savings should forecasting methods improve.

LIST OF REFERENCES

1. Lulejian & Associated, Inc., *An Analysis of Benefits Accuring to the Navy by using Weather Forecasts in the Trans-oceanic Routing of Surface Ships*, January 1976.

2. Environmental Prediction Research Facility, Technical Paper No. 10-72, *A Numerical Study in Optimum Track Ship Route Climatology*, by Frederick W. Nagle, September 1972.

3. *Sailing Directions (Planning Guide)*, 3rd ed., pub 140, Defense Mapping Agency Hydrographic/Topographic Center.

4. Cormen, T. H., Leiserson, C. E., and Rivest, R. L., *Introduction to Algorithms*, pp. 527-532, McGraw-Hill Book Co.,1989.

5. Bowditch, N., *American Practical Navigator, an Epitome of Navigation*, Vol. I, p. 1304, Defense Mapping Agency Hydrographic/Topographic Center, 1984.

6. Air Test and Evaluation Squadron One, Tactics Guide 600-1-78, *S-3A Fuel Managment*, 18 August 1978.

7. 323d Flying Training Wing, *Operational Test and Evaluation of United States Navy Optimum Path Aircraft Routing System (OPARS)*, by Kirk L. Brown, May 1985.

8. Military Sealift Command, COMSCINST 7600.3H ch 6, *Billing Rates*.

APPENDIX A

Optimal Path Algorithm

This program reads in a sparse graph and determines the shortest path through the graph.

```
CONST MAX=160;
      START=14;
      STOP=157;
      RANDSEED=7654321;

TYPE PADJ=^ADJNODES;
     PEDGE=^EDGES;
     EDGES=RECORD
                 REC:PADJ;          {POINTS TO THE TAIL}
                 NEXTNODE:PEDGE;    {NEXT ADJACENT ARC}
                 PARENT:PADJ;       {POINTS TO THE HEAD}
                 WT:REAL;           {WEIGHT OF ARC}
             END;
     ADJNODES=RECORD
                 NODE:INTEGER;      {NODE NUMBER}
                 NEXTNODE:PEDGE;    {POINTS TO NEXT ADJACENT
NODE}
                 DIST:REAL;         {DISTANCE FROM SOURCE}
                 PWAVE:REAL;        {PROB OF A WAVE HT.}
                 PDAM:REAL;         {PROB DAMAGE GIVEN WAVE HT.}
                 PRED:PADJ;         {POINTER TO PREDECESSOR}
                 DAM:REAL;
              END;
     HEADNODE=ARRAY[1..MAX] OF PADJ;
     QUEUE=ARRAY[1..MAX] OF PADJ;
     AY=ARRAY[1..MAX] OF REAL;
     PC=ARRAY[1..7] OF REAL;
     G=RECORD
                 GRAPH:HEADNODE;    {THE GRAPH}
                 LENGTH:INTEGER;    {THE LENGTH OF THE GRAPH}

         END;
     Q=RECORD
                 PRIQ:QUEUE;
                 SIZE:INTEGER
         END;

VAR I,J:INTEGER;
    DATAOUT:TEXT;
    GRAPH1:G;
    P1,P2,PNTLAT,PNTLON:AY;
    Q1:Q;
    PDAMAGE,CDAMAGE:PC;
```

46

```pascal
      CPNT:PADJ;
      CP:PEDGE;
      TOTAL:REAL;

PROCEDURE RPROB(VAR FIRST:AY;VAR SEC:AY);
   {READS PROBABILITY OF WAVE HEIGHT FROM INPUT FILE}

   VAR I:INTEGER;
       DATAIN:TEXT;

   BEGIN
     ASSIGN(DATAIN,'C:\PASCAL\WAVES.TXT');
     RESET(DATAIN);
     FOR I:=1 TO MAX DO
       READLN(DATAIN,FIRST[I],SEC[I]);
   END;

PROCEDURE RLATLON(VAR FIRST:AY;VAR SEC:AY);
   {READS LAT AND LONG FROM INPUT FILE}

   VAR I:INTEGER;
       DATAIN:TEXT;

   BEGIN
     ASSIGN(DATAIN,'C:\PASCAL\LATLON.TXT');
     RESET(DATAIN);
     FOR I:=1 TO MAX DO
       READLN(DATAIN,FIRST[I],SEC[I]);
   END;

FUNCTION FUEL(BEG:INTEGER;EN:INTEGER):REAL;
   {COMPUTES FUEL REQUIRED FOR AN ARC}

   VAR LATS,LATD,LONS,LOND,COSDIST,J,TEMP:REAL;

   BEGIN
     LATS:=PNTLAT[BEG]*(PI/180);
     LONS:=PNTLON[BEG]*(PI/180);
     LATD:=PNTLAT[EN]*(PI/180);
     LOND:=PNTLON[EN]*(PI/180);
    COSDIST:=(SIN(LATS)*SIN(LATD))+(COS(LATS)*COS(LATD)*
            (COS(LOND-LONS)));
     J:=0;
     TEMP:=1.0;
     REPEAT
       BEGIN
         J:=J+0.01;
         TEMP:=COS(J);
       END;
     UNTIL((COSDIST>=TEMP)OR(J=3.14));
     FUEL:=(((J*180/PI)*60)/15)*(650/24);
```

47

```
        END;

    FUNCTION BUILDWTS(VAR DEST:PEDGE):REAL;
       {BUILD THE WEIGHT OF THE ARC FROM FUEL AND DAMAGE}

        VAR I:INTEGER;
            COST,DIST:REAL;

        BEGIN
          COST:=0;
          IF((DEST^.REC^.PWAVE<=P2[DEST^.REC^.NODE])AND
                (P2[DEST^.REC^.NODE]<1))THEN
             FOR I:=1 TO 3 DO
               IF(DEST^.REC^.PDAM<=PDAMAGE[I])THEN
                 COST:=CDAMAGE[I];
          IF((COST=0)AND(DEST^.REC^.PDAM<=P1[DEST^.REC^.NODE])AND
                (P1[DEST^.REC^.NODE]<1))THEN
             FOR I:=4 TO 5 DO
               IF (DEST^.REC^.PDAM<=PDAMAGE[I])THEN
                 COST:=CDAMAGE[I];
          IF(COST=0)THEN
             FOR I:=6 TO 7 DO
               IF (DEST^.REC^.PDAM<=PDAMAGE[I])THEN
                 COST:= CDAMAGE[I];
          BUILDWTS:=COST+(FUEL(DEST^.PARENT^.NODE,
                     DEST^.REC^.NODE)*45.86);
        END;

    PROCEDURE HEAPIFY(VAR NQ:Q;START:INTEGER);

        VAR SMALLEST,L,R:INTEGER;
            TEMP:PADJ;

        BEGIN
          L:=2*START;
          R:=(2*START)+1;
          IF ((L<NQ.SIZE) AND
                (NQ.PRIQ[L]^.DIST<NQ.PRIQ[START]^.DIST)) THEN
             SMALLEST:=L
          ELSE
             SMALLEST:=START;
          IF ((R<NQ.SIZE)  AND
                (NQ.PRIQ[R]^.DIST<NQ.PRIQ[SMALLEST]^.DIST)) THEN
             SMALLEST:=R;
          IF (SMALLEST<>START) THEN
             BEGIN
               TEMP:=NQ.PRIQ[START];
               NQ.PRIQ[START]:=NQ.PRIQ[SMALLEST];
               NQ.PRIQ[SMALLEST]:=TEMP;
               HEAPIFY(NQ,SMALLEST);
             END;    {IF SWAPPED}
```

48

```
     END;    {PROCEDURE HEAPIFY}


PROCEDURE INSERTPQ(PNTR:PADJ; VAR NQ:Q);

   VAR I:INTEGER;

   BEGIN
     NQ.SIZE:=NQ.SIZE+1;
     I:=NQ.SIZE;
     WHILE ((I>1) AND (NQ.PRIQ[I DIV 2]^.DIST>PNTR^.DIST))
DO
       BEGIN
         NQ.PRIQ[I]:=NQ.PRIQ[I DIV 2];
         I:=I DIV 2;
       END;    {WHILE}
     NQ.PRIQ[I]:=PNTR;
   END;    {PROCEDURE INSERTPQ}

PROCEDURE BUILDPQ(THISGRAPH:G;VAR PQ:Q);

   VAR I:INTEGER;

   BEGIN
     FOR I:=1 TO THISGRAPH.LENGTH DO
       BEGIN
         INSERTPQ(THISGRAPH.GRAPH[I],PQ);
       END;    {FOR}
   END;    {PROCEDURE BUILDPQ}

FUNCTION EXTRACTMIN(VAR PQ:Q):PADJ;

   BEGIN       {FUNCTION EXTRACTMIN}
     EXTRACTMIN:=PQ.PRIQ[1];
     PQ.PRIQ[1]:=PQ.PRIQ[PQ.SIZE];
     PQ.SIZE:=PQ.SIZE-1;
     HEAPIFY(PQ,1);
   END;    {FUNCTION EXTRACTMIN}

FUNCTION EMPTYPQ(VAR PQ:Q):BOOLEAN;

   BEGIN
     IF (PQ.SIZE=0) THEN
       EMPTYPQ:=TRUE
     ELSE
       EMPTYPQ:=FALSE;
   END;    {FUNCTION EMPTYPQ}


PROCEDURE MAKEGRAPH(VAR THISGRAPH:G);
```

```pascal
{ THIS PROCEDURE GENERATES THE GRAPH.}

  TYPE ARY=ARRAY[1..2] OF INTEGER;

  VAR NEWREC:PEDGE;
      CP,LP:PEDGE;
      DUPE,FOUND:BOOLEAN;
      DATAIN:TEXT;
      NEWNODE:ARY;
      I,J,TEMP:INTEGER;
      NEWWT:INTEGER;

  BEGIN
    ASSIGN(DATAIN,'C:\PASCAL\G.TXT');
    RESET(DATAIN);
    NEWWT:=0;
    READLN(DATAIN,THISGRAPH.LENGTH);
    WHILE(NOT EOF(DATAIN)) DO
      BEGIN
        FOR I:=1 TO 2 DO
          READ(DATAIN,NEWNODE[I]);
        READLN(DATAIN);
        IF (NEWNODE[1]<>NEWNODE[2]) THEN           {IF NOT A
              SELF LOOP}
          BEGIN
            DUPE:=FALSE;
            FOUND:=FALSE;
            IF (THISGRAPH.GRAPH[NEWNODE[1]]^.NEXTNODE<>NIL)
                  THEN {FIRST NODE}
              BEGIN
                LP:=NIL;
                CP:=THISGRAPH.GRAPH[NEWNODE[1]]^.NEXTNODE;
                REPEAT
                  DUPE:=(CP^.REC^.NODE=NEWNODE[2]);
                  FOUND:=(CP^.REC^.NODE>NEWNODE[2]);
                  IF (NOT(FOUND) AND NOT(DUPE)) THEN
                    BEGIN
                      LP:=CP;
                      CP:=CP^.NEXTNODE;  ·
                    END;
                  IF (CP=NIL) THEN
                    FOUND:=TRUE
                UNTIL (FOUND OR DUPE OR (CP=NIL));
                IF (NOT(DUPE)) THEN
                  BEGIN
                    NEW(NEWREC);
                    NEWREC^.NEXTNODE:=CP;

NEWREC^.REC:=THISGRAPH.GRAPH[NEWNODE[2]];
                    NEWREC^.PARENT:=
                          THISGRAPH.GRAPH[NEWNODE[1]];
```

```
                            NEWREC^.WT:=0;
                            IF FOUND THEN
                              IF (LP=NIL) THEN
                                THISGRAPH.GRAPH[NEWNODE[1]]^.
                                    NEXTNODE:=NEWREC
                            ELSE
                              LP^.NEXTNODE:=NEWREC;
                          END;   {IF NOT DUPE}
                        IF DUPE THEN          {ADDS ONLY SMALLEST
ARC}
                            IF (NEWWT<CP^.WT) THEN     {MULTIPLE ARCS}
                              CP^.WT:=NEWWT;
                    END
                  ELSE          {IF FIRST EDGE}
                      BEGIN
                        NEW(NEWREC);
                        NEWREC^.NEXTNODE:=NIL;
                        NEWREC^.REC:=THISGRAPH.GRAPH[NEWNODE[2]];

NEWREC^.PARENT:=THISGRAPH.GRAPH[NEWNODE[1]];
                        NEWREC^.WT:=NEWWT;
                        THISGRAPH.GRAPH[NEWNODE[1]]^.
                            NEXTNODE:=NEWREC;
                    END;
              END;        {IF NO SELF LOOP}
          END;       {WHILE}
      END;         {PROCEDURE MAKEGRAPH}


PROCEDURE INITGRAPH(VAR G1:G);

    BEGIN
      FOR I:=1 TO MAX DO
        BEGIN
          NEW(G1.GRAPH[I]);
          WITH G1.GRAPH[I]^ DO
            BEGIN
              NODE:=I;
              DIST:=10000000;
              PWAVE:=RANDOM;
              PDAM:=RANDOM;
              PRED:=NIL;
              NEXTNODE:=NIL;
              DAM:=0.0;
            END;
        END;
      G1.LENGTH:=0;
    END;      {PROCEDURE INITGRAPH}

PROCEDURE INITQ(VAR THISQ:Q);
```

```
          VAR I:INTEGER;

      BEGIN
        FOR I:=1 TO MAX DO
          BEGIN
            NEW(THISQ.PRIQ[I]);
            THISQ.PRIQ[I]:=NIL;
          END;
        THISQ.SIZE:=0;
      END;


  PROCEDURE DIJKSTRA(VAR THISGRAPH:G;SOURCE:INTEGER);

      VAR I:INTEGER;
          WT:REAL;
          HERE:PEDGE;
          THISNODE:PADJ;

      BEGIN
        THISGRAPH.GRAPH[SOURCE]^.PRED:=THISGRAPH.GRAPH[SOURCE];
        THISGRAPH.GRAPH[SOURCE]^.DIST:=0;
        BUILDPQ(THISGRAPH,Q1);
        WHILE (NOT(EMPTYPQ(Q1))) DO
          BEGIN
            THISNODE:=EXTRACTMIN(Q1);
            HERE:=THISNODE^.NEXTNODE;
            WHILE(HERE<>NIL) DO
              BEGIN
                WT:=BUILDWTS(HERE);
                IF(HERE^.REC^.DIST>THISNODE^.DIST+WT) THEN
                  BEGIN
                    HERE^.REC^.DIST:=THISNODE^.DIST+WT;
                    HERE^.REC^.PRED:=THISNODE;
                    FOR I:=((Q1.SIZE+1) DIV 2) DOWNTO 1 DO
                      HEAPIFY(Q1,I);
                  END;
                HERE:=HERE^.NEXTNODE;
              END;
          END;
      END;      {PROCEDURE DIJKSTRA}


      BEGIN          {MAIN PROGRAM}
        ASSIGN(DATAOUT,'C:\PASCAL\OUTPUT3.TXT');
        REWRITE(DATAOUT);
        RPROB(P1,P2);
        RLATLON(PNTLAT,PNTLON);
        GRAPH1.GRAPH[START]^.DIST:=0;
        PDAMAGE[1]:=0.858;
        PDAMAGE[2]:=0.21;
```

```
   PDAMAGE[3]:=0.087;
   PDAMAGE[4]:=0.0354;
   PDAMAGE[5]:=0.0027;
   PDAMAGE[6]:=0.0024;
   PDAMAGE[7]:=0.0003;
   CDAMAGE[1]:=340771;
   CDAMAGE[2]:=312196;
   CDAMAGE[3]:=312196;
   CDAMAGE[4]:=129969;
   CDAMAGE[5]:=129969;
   CDAMAGE[6]:=48427;
   CDAMAGE[7]:=48427;
   INITGRAPH(GRAPH1);
   MAKEGRAPH(GRAPH1);
   INITQ(Q1);
FOR I:=1 TO 100 DO
BEGIN
   FOR J:=1 TO MAX DO
      BEGIN
         GRAPH1.GRAPH[J]^.PRED:=NIL;
         GRAPH1.GRAPH[J]^.DIST:=10000000;
         Q1.PRIQ[J]:=NIL;
      END;
   Q1.SIZE:=0;
   DIJKSTRA(GRAPH1,START);
   CPNT:=GRAPH1.GRAPH[STOP]^.PRED;
   TOTAL:=FUEL(CPNT^.NODE,GRAPH1.GRAPH[STOP]^.NODE)*45.86;
   WRITE(DATAOUT,STOP:4);
   WHILE(CPNT^.PRED^.NODE<>CPNT^.NODE) DO
      BEGIN
       ⁻ TOTAL:=TOTAL+
            (FUEL(CPNT^.PRED^.NODE,CPNT^.NODE)*45.86);
         WRITE(DATAOUT,CPNT^.NODE:4);
         CPNT:=CPNT^.PRED;
      END;
   WRITE(DATAOUT,START:4);
   WRITE(DATAOUT,GRAPH1.GRAPH[STOP]^.DIST:10:2);
   WRITELN(DATAOUT,GRAPH1.GRAPH[STOP]^.DIST-TOTAL:10:2);
   FOR J:=1 TO MAX DO
      BEGIN
         GRAPH1.GRAPH[J]^.PWAVE:=RANDOM;
         GRAPH1.GRAPH[J]^.PDAM:=RANDOM;
      END;
END;
   CLOSE(DATAOUT);
END.
```

APPENDIX B

Non-optimal Routing Algorithm

This program reads in a sparse graph and determines a
non-optimal path through the graph based on only the next
immediately available nodes.

```
CONST MAX=160;
      START=14;
      STOP=157;
      RANDSEED=7654321;

TYPE PADJ=^ADJNODES;
     PEDGE=^EDGES;
     EDGES=RECORD
                REC:PADJ;                {POINTS TO THE TAIL}
                NEXTNODE:PEDGE;          {NEXT ADJACENT ARC}
                PARENT:PADJ;             {POINTS TO THE HEAD}
                WT:REAL;                 {WEIGHT OF ARC}
            END;
     ADJNODES=RECORD
                NODE:INTEGER;    {NODE NUMBER}
                NEXTNODE:PEDGE;  {POINTER TO NEXT NODE}
                DIST:REAL;       {DISTANCE FROM SOURCE}
                PWAVE:REAL;      {PROB OF WAVE HEIGHT}
                PDAM:REAL;       {PROB OF DAMAGE GIVEN HT.}
                PRED:PADJ;       {POINTER TO THE PREDECESSOR}
                DAM:REAL;        {DAMAGE ENCOUNTERED}
                QIN:BOOLEAN;     {IS NODE ON PATH TO DEST.}
              END;
     HEADNODE=ARRAY[1..MAX] OF PADJ;
     AY=ARRAY[1..MAX] OF REAL;
     PC=ARRAY[1..7] OF REAL;
     G=RECORD
                GRAPH:HEADNODE;   {THE GRAPH}
                LENGTH:INTEGER;   {THE LENGTH OF THE GRAPH}
          END;

VAR I,J:INTEGER;
    DATAOUT:TEXT;
    GRAPH1:G;
    P1,P2,PNTLAT,PNTLON:AY;
    PDAMAGE,CDAMAGE:PC;
    CPNT:PADJ;
    CP:PEDGE;
    TOTAL:REAL;

PROCEDURE RPROB(VAR FIRST:AY;VAR SEC:AY);

    VAR I:INTEGER;
```

54

```
          DATAIN:TEXT;

    BEGIN
      ASSIGN(DATAIN,'C:\PASCAL\WAVES.TXT');
      RESET(DATAIN);
      FOR I:=1 TO MAX DO
        READLN(DATAIN,FIRST[I],SEC[I]);
    END;

PROCEDURE RLATLON(VAR FIRST:AY;VAR SEC:AY);

    VAR I:INTEGER;
        DATAIN:TEXT;

    BEGIN
      ASSIGN(DATAIN,'C:\PASCAL\LATLON.TXT');
      RESET(DATAIN);
      FOR I:=1 TO MAX DO
        READLN(DATAIN,FIRST[I],SEC[I]);
    END;

FUNCTION FUEL(BEG:INTEGER;EN:INTEGER):REAL;

    VAR LATS,LATD,LONS,LOND,COSDIST,J,TEMP:REAL;

    BEGIN
      LATS:=PNTLAT[BEG]*(PI/180);
      LONS:=PNTLON[BEG]*(PI/180);
      LATD:=PNTLAT[EN]*(PI/180);
      LOND:=PNTLON[EN]*(PI/180);
      COSDIST:=(SIN(LATS)*SIN(LATD))+
               (COS(LATS)*COS(LATD)*(COS(LOND-LONS)));
      J:=0;
      TEMP:=1.0;
      REPEAT
        BEGIN
          J:=J+0.01;
          TEMP:=COS(J);
        END;
      UNTIL((COSDIST>=TEMP)OR(J=3.14));
      FUEL:=(((J*180/PI)*60)/15)*(650/24);
    END;

FUNCTION BUILDWTS(VAR DEST:PEDGE):REAL;

    VAR M:INTEGER;
        COST,DIST:REAL;

    BEGIN
      COST:=0;
      IF((DEST^.REC^.PWAVE<=P2[DEST^.REC^.NODE])
```

```pascal
                    AND(P2[DEST^.REC^.NODE]<1)) THEN
            FOR M:=1 TO 3 DO
              IF(DEST^.REC^.PDAM<=PDAMAGE[M]) THEN
                COST:=CDAMAGE[M];
        IF((COST=0)AND(DEST^.REC^.PDAM<=P1[DEST^.REC^.NODE])
                    AND(P1[DEST^.REC^.NODE]<1)) THEN
            FOR M:=4 TO 5 DO
              IF (DEST^.REC^.PDAM<=PDAMAGE[M]) THEN
                COST:=CDAMAGE[M];
        IF(COST=0) THEN
            FOR M:=6 TO 7 DO
              IF (DEST^.REC^.PDAM<=PDAMAGE[M]) THEN
                COST:= CDAMAGE[M];
        BUILDWTS:=COST+
                (FUEL(DEST^.PARENT^.NODE,DEST^.REC^.NODE)*45.86);
    END;


PROCEDURE MAKEGRAPH(VAR THISGRAPH:G);

{ THIS PROCEDURE GENERATES THE GRAPH.}

    TYPE ARY=ARRAY[1..2] OF INTEGER;

    VAR NEWREC:PEDGE;
        CP,LP:PEDGE;
        DUPE,FOUND:BOOLEAN;
        DATAIN:TEXT;
        NEWNODE:ARY;
        I,J,TEMP:INTEGER;
        NEWWT:INTEGER;

    BEGIN
      ASSIGN(DATAIN,'C:\PASCAL\G.TXT');
      RESET(DATAIN);
      NEWWT:=0;
      READLN(DATAIN,THISGRAPH.LENGTH);
      WHILE(NOT EOF(DATAIN)) DO
        BEGIN
          FOR I:=1 TO 2 DO
            READ(DATAIN,NEWNODE[I]);
          READLN(DATAIN);
          IF (NEWNODE[1]<>NEWNODE[2]) THEN {IF NOT A SELF
LOOP}
            BEGIN
              DUPE:=FALSE;
              FOUND:=FALSE;
              IF (THISGRAPH.GRAPH[NEWNODE[1]]^.NEXTNODE<>NIL)
                    THEN {FIRST NODE}
                BEGIN
                  LP:=NIL;
```

56

```
                    CP:=THISGRAPH.GRAPH[NEWNODE[1]]^.NEXTNODE;
                    REPEAT
                      DUPE:=(CP^.REC^.NODE=NEWNODE[2]);
                      FOUND:=(CP^.REC^.NODE>NEWNODE[2]);
                      IF (NOT(FOUND) AND NOT(DUPE)) THEN
                        BEGIN
                          LP:=CP;
                          CP:=CP^.NEXTNODE;
                        END;
                      IF (CP=NIL) THEN
                        FOUND:=TRUE
                    UNTIL (FOUND OR DUPE OR (CP=NIL));
                    IF (NOT(DUPE)) THEN
                      BEGIN
                        NEW(NEWREC);
                        NEWREC^.NEXTNODE:=CP;

NEWREC^.REC:=THISGRAPH.GRAPH[NEWNODE[2]];
                        NEWREC^.PARENT:=
                            THISGRAPH.GRAPH[NEWNODE[1]];
                        NEWREC^.WT:=0;
                        IF FOUND THEN
                          IF (LP=NIL) THEN

THISGRAPH.GRAPH[NEWNODE[1]]^.NEXTNODE
                              :=NEWREC
                          ELSE
                            LP^.NEXTNODE:=NEWREC;
                      END;   {IF NOT DUPE}
                    IF DUPE THEN         {ADDS ONLY SMALLEST ARC}
                      IF (NEWWT<CP^.WT) THEN    {MULTIPLE ARCS}
                        CP^.WT:=NEWWT;
                  END
                ELSE           {IF FIRST EDGE}
                  BEGIN
                    NEW(NEWREC);
                    NEWREC^.NEXTNODE:=NIL;
                    NEWREC^.REC:=THISGRAPH.GRAPH[NEWNODE[2]];

NEWREC^.PARENT:=THISGRAPH.GRAPH[NEWNODE[1]];
                    NEWREC^.WT:=NEWWT;
                    THISGRAPH.GRAPH[NEWNODE[1]]^.NEXTNODE
                        :=NEWREC;
                  END;
            END;      {IF NO SELF LOOP}
        END;        {WHILE}
    END;          {PROCEDURE MAKEGRAPH}


PROCEDURE INITGRAPH(VAR G1:G);
```

```
    BEGIN
      FOR I:=1 TO MAX DO
        BEGIN
          NEW(G1.GRAPH[I]);
          WITH G1.GRAPH[I]^ DO
            BEGIN
              NODE:=I;
              DIST:=10000000;
              PWAVE:=RANDOM;
              PDAM:=RANDOM;
              PRED:=NIL;
              NEXTNODE:=NIL;
              DAM:=0.0;
              QIN:=FALSE;
            END;
        END;
      G1.LENGTH:=0;
    END;      {PROCEDURE INITGRAPH}

PROCEDURE CHOICES(VAR G1:G;DEST:INTEGER);

    TYPE QUEUE=ARRAY[1..MAX] OF BOOLEAN;

    VAR M:INTEGER;
        Q2:QUEUE;

    BEGIN
      FOR M:=1 TO MAX DO
        Q2[M]:=FALSE;
      Q2[DEST]:=TRUE;
      FOR M:=MAX DOWNTO 1 DO
        BEGIN
          IF (Q2[M]=TRUE) THEN
            BEGIN
              IF ((M MOD 10)=1) THEN
                BEGIN
                  IF (M>9) THEN
                    Q2[M-9]:=TRUE;
                  IF (M>10) THEN
                    Q2[M-10]:=TRUE;
                END
              ELSE IF((M MOD 10)=0) THEN
                BEGIN
                  IF (M>10) THEN
                    Q2[M-10]:=TRUE;
                  IF (M>11) THEN
                    Q2[M-11]:=TRUE;
                END
              ELSE
                BEGIN
                  IF (M>9) THEN
```

58

```
                         Q2[M-9]:=TRUE;
                      IF (M>10) THEN
                         Q2[M-10]:=TRUE;
                      IF (M>11) THEN
                         Q2[M-11]:=TRUE;
                END;
            END;
        END;
     ·FOR M:=1 TO MAX DO
        G1.GRAPH[M]^.QIN:=Q2[M];
   END;

PROCEDURE FINDPATH(VAR G1:G;S:INTEGER;D:INTEGER);

   TYPE A=ARRAY[1..3] OF PEDGE;

   VAR C:PEDGE;
       THISNODE,Z,J,K:INTEGER;
       W,TEM:REAL;
       CP:A;

   BEGIN
     G1.GRAPH[S]^.PRED:=G1.GRAPH[S];
     G1.GRAPH[S]^.DIST:=0.0;
     THISNODE:=S;
     FOR Z:=1 TO ((D DIV 10)-(S DIV 10)) DO
        BEGIN
          CP[1]:=G1.GRAPH[THISNODE]^.NEXTNODE;
          FOR J:=2 TO 3 DO
            CP[J]:=CP[J-1]^.NEXTNODE;
         ¯TEM:=10000000;
          K:=1;
          WHILE ((CP[K]<>NIL) AND (K<=3)) DO
            BEGIN
              W:=BUILDWTS(CP[K]);
              IF ((CP[K]^.REC^.QIN=TRUE) AND (W<=TEM)) THEN
                BEGIN
                  C:=CP[K];
                  TEM:=W;
                END;
              K:=K+1;
            END.
          THISNODE:=C^.REC^.NODE;
          C^.REC^.DIST:=C^.PARENT^.DIST+TEM;
          C^.REC^.PRED:=C^.PARENT;
        END;
   END;

   BEGIN            {MAIN PROGRAM}
     ASSIGN(DATAOUT,'C:\PASCAL\OUTPUT5.TXT');
     REWRITE(DATAOUT);
```

59

```
      RPROB(P1,P2);
      RLATLON(PNTLAT,PNTLON);
      GRAPH1.GRAPH[START]^.DIST:=0;
      PDAMAGE[1]:=0.858;
      PDAMAGE[2]:=0.21;
      PDAMAGE[3]:=0.087;
      PDAMAGE[4]:=0.0354;
      PDAMAGE[5]:=0.0027;
      PDAMAGE[6]:=0.0024;
      PDAMAGE[7]:=0.0003;
      CDAMAGE[1]:=340771;
      CDAMAGE[2]:=312196;
      CDAMAGE[3]:=312196;
      CDAMAGE[4]:=129969;
      CDAMAGE[5]:=129969;
      CDAMAGE[6]:=48427;
      CDAMAGE[7]:=48427;
      INITGRAPH(GRAPH1);
      MAKEGRAPH(GRAPH1);
      CHOICES(GRAPH1,STOP);
   FOR I:=1 TO 100 DO
   BEGIN
      FOR J:=1 TO MAX DO
        BEGIN
          GRAPH1.GRAPH[J]^.PRED:=NIL;
          GRAPH1.GRAPH[J]^.DIST:=10000000;
        END;
      FINDPATH(GRAPH1,START,STOP);
      CPNT:=GRAPH1.GRAPH[STOP]^.PRED;
      TOTAL:=FUEL(CPNT^.NODE,GRAPH1.GRAPH[STOP]^.NODE)*45.86;
      WRITE(DATAOUT,STOP:4);
      WHILE(CPNT^.PRED^.NODE<>CPNT^.NODE) DO
        BEGIN
          TOTAL:=TOTAL+
                  (FUEL(CPNT^.PRED^.NODE,CPNT^.NODE)*45.86);
          WRITE(DATAOUT,CPNT^.NODE:4);
          CPNT:=CPNT^.PRED;
        END;
      WRITE(DATAOUT,START:4);
      WRITE(DATAOUT,GRAPH1.GRAPH[STOP]^.DIST:10:2);
      WRITELN(DATAOUT,GRAPH1.GRAPH[STOP]^.DIST-TOTAL:10:2);
      FOR J:=1 TO MAX DO
        BEGIN
          GRAPH1.GRAPH[J]^.PWAVE:=RANDOM;
          GRAPH1.GRAPH[J]^.PDAM:=RANDOM;
        END;
   END;
      CLOSE(DATAOUT);
   END.
```

60

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria VA 22304-6145	2
2.	Library, Code 052 Naval Postgraduate School Monterey CA 93943-5002	2
3.	Daniel C. Boger, Code OR/BO Naval Postgraduate School Monterey CA 93943-5002	2
4.	Michael Sovereign, Code OR/MS Naval Postgraduate School Monterey CA 93943-5002	2
5.	J. Nottenkamper ADP Hardware Coordinator Fleet Numerical Oceanography Center Monterey CA 93943	2
6.	Lt James L. Brown Jr. 150 Brown Lane Harmony PA 16037	2
7.	James L. Brown Sr. 150 Brown Lane Harmony PA 16037	1