

NASA Contractor Report 4063

Design and Analysis of Advanced Flight Planning Concepts

John A. Sorensen

CONTRACT NAS1-17345
MARCH 1987

(NASA-CR-4063) DESIGN AND ANALYSIS OF	16-78-100
ADVANCED FLIGHT PLANNING CONCEPTS	
REPORT (ANALYTICAL RESEARCH AND DEVELOPMENT)	
1987	
	41/03 4063/0



NASA Contractor Report 4063

Design and Analysis of Advanced Flight Planning Concepts

John A. Sorensen

*Analytical Mechanics Associates, Inc.
Mountain View, California*

Prepared for
Langley Research Center
under Contract NAS1-17345

NASA

National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

1987

FOREWORD

This continuing effort for development of concepts for generating near-optimum flight profiles that minimize fuel or direct operating costs was supported under NASA Contract No. NAS1-17345, by Langley Research Center, Hampton VA. The project Technical Monitor at Langley Research Center was Dan D. Vicroy. Technical discussion with and suggestions from Mr. Vicroy, David H. Williams, and Charles E. Knox of Langley Research Center are gratefully acknowledged.

The technical information concerning the Chicago-Phoenix flight plan used as an example throughout this study was provided by courtesy of United Airlines. The weather information used to exercise the experimental flight planning program EFPLAN developed in this study was provided by courtesy of Pacific Southwest Airlines.

At AMA, Inc., the project manager was John A. Sorensen. Engineering support was provided by Tsuyoshi Goka, Kioumars Najmabadi, and Mark H. Waters. Project programming support was provided by Susan Dorsky, Ann Blake, and Casimer Lesiak.

PRECEDING PAGE BLANK NOT FILMED

DESIGN AND ANALYSIS OF
ADVANCED FLIGHT PLANNING CONCEPTS

John A. Sorensen

Analytical Mechanics Associates, Inc.

SUMMARY

The objectives of this continuing effort are to develop and evaluate new algorithms and advanced concepts for flight management and flight planning. This includes the minimization of fuel or direct operating costs, the integration of the airborne flight management and ground-based flight planning processes, and the enhancement of future traffic management systems design. Flight management (FMS) concepts are for on-board profile computation and steering of transport aircraft in the vertical plane between a city pair and along a given horizontal path. Flight planning (FPS) concepts are for the pre-flight ground based computation of the three-dimensional reference trajectory that connects the city pair and specifies the horizontal path, fuel load, and weather profiles for initializing the FMS.

As part of these objectives, a new computer program called EFPLAN has been developed and utilized to study advanced flight planning concepts. EFPLAN represents an experimental version of an FPS. It has been developed to generate reference flight plans compatible as input to an FMS and to provide various options for flight planning research. This report describes EFPLAN and the associated research conducted in its development.

PRECEDING PAGE BLANK NOT FILMED

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. GENERATION OF NEAR-OPTIMUM FLIGHT PLANS	5
Evaluation of an Existing Flight Plan	5
The Flight Planning - Flight Management Connection	15
The Flight Planning Sequence	21
EFPLAN Options	26
EFPLAN Input	28
Reference Flight Path Computation	39
EFPLAN Output	53
Example Results	58
III. SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS	67
Summary and Conclusions	67
Recommendations	70
 APPENDICES	
A. TWO PARAMETER OPTIMIZATION METHOD	73
B. MODELING OF NON-STANDARD DAY	91
C. 727-200 MODELING DETAIL	103
D. WEATHER DATA BASE	109
E. AIRCRAFT PERFORMANCE MODELING FOR FLIGHT PLANNING	121
F. DYNAMIC PROGRAMMING FUNDAMENTALS	139
REFERENCES	147

PRECEDING PAGE BLANK NOT FILMED

LIST OF FIGURES

	Page
1. Chicago (ORD) to Phoenix (PHX) Route 5.	6
2. Cruise Speed Comparison as a Function of Cost of Time with UAL Results	8
3. Comparison of Climb and Descent Range Requirements as a Function of Cost of Time When Cruise is at 35000 ft . . .	9
4. Comparison of Altitude vs Range for Cruise-Climb Profiles with Varying Cost of Time.	10
5. Comparison of Cruise Mach Number vs Range for Cruise Climb Profiles with Varying Cost of Time.	12
6. Comparison of Individual Flight Segment Costs for the ORD-PHX Routes.	12
7. Comparison of Constant Cruise Altitude and Cruise-Climb Results in Terms of Trip Time and Fuel Burned.	13
8. Comparison of Three Step climb Altitude Profiles with Cruise-Climb.	14
9. Comparison of Trip Fuel and Trip Time as a Function of Step Climb Point	14
10. Tradeoff Between Trip Fuel and Trip Time for Fixed Cruise, Step Climb, and Free Cruise Altitude Profiles	16
11. Example Effect of Powerplant Losses on Cruise Cost [1]. . .	17
12. Example Windfield Error as a Function of Forecast Time and Potential for Improvement [3].. . . .	20
13. First Level Flow Chart for a Flight Planning Program . . .	22
14. Sample of Weather at 250 mb (34000 ft).	30
15. Artificially Generated United States Wind Field	31
16. Example SID and STAR for Chicago-Phoenix Route	32
17. Illustration of Pseudo-Waypoint that Ends the Descent Phase	34
18. Navaids for Phoenix-Chicago City Pair.	36
19. Example Direct Routes Between Chicago and Phoenix	37
20. Values of V_{IAS} and Mach for Climb and Descent as Functions s of Performance Index.	42

LIST OF FIGURES (Cont.)

21.	Illustration of Step Climb/Descent and Cruise Climb/Descent Altitude Change.	48
22.	Even Grid with Equal Number of Nodes per Stage.	49
23.	Even Grid with Unequal Number of Nodes per Stage.	50
24a.	Horizontal grid Structure for ORD-PHX Route.	55
24b.	Selected Horizontal Path from Flight Plan for ORD-PHX. Enroute Wind is Indicated.	56
25.	Optimum Direct Route 6 for ORD-PHX.	57
26.	Norfolk-Charlotte Leg of Norfolk Loop	59
27.	Charlotte-Chicago Leg of Norfolk Loop	60
28.	Chicago-Dallas Leg of Norfolk Loop	61
29.	Dallas-Atlanta Leg of Norfolk Loop	62
30.	Atlanta-Norfolk Leg of Norfolk Loop	63
A-1.	Stages of Two-Parameter Climb Profile	75
A-2.	Illustration of Different Points with Different Costs	77
A-3.	Computation Flow Chart for the Complex Algorithm	79
A-4.	Two Points with Two Costs	80
A-5.	Three Points with Three Costs Used to Compute P	80
A-6.	The Problem Created by a Boundary	81
A-7.	Definition of Parameter Constraints for Climb-Descent Optimization	84
A-8.	Cost Contours for No Wind Climb	88
A-9.	Fuel Contours for No Wind Climb	89
A-10.	Time Contours for No Wind Climb	90
B-1.	Rawinsonde Data for Albuquerque, September 1980	94
B-2.	Range of Systematic Variability of Temperature Around the U.S. Standard Atmosphere, 1976.	95
B-3.	Example Comparing Temperature-Altitude Profile on a Given Non-standard Day with Standard Day Profile.	99

LIST OF FIGURES (Cont.)

C-1.	Drag Coefficient Polynomial Fit vs C and Mach.	104
C-2.	Lift Coefficient vs Angle-of-Attack	105
C-3.	Thrust Polynomial Fit vs Mach and EPR for Center Engine . .	107
C-4.	Example of Fuel Flow Plots.	108
D-1.	Sketch of Latitude-Longitude Locations for GRIB Code Data Arrays for Continental U. S.	112
D-2.	January 14, 1985 Wind Forecast.	114
E-1.	Thrust Minus Drag Margin as a Function of Airspeed at 33000 ft.	122
E-2.	Mach Limits to Define Fuel Rate Table Boundaries.	125
E-3.	Trimmed Cruise Fuel Rate as a Function of Mach, Altitude, Weight and Temperature.	127
E-4.	Step Climb Fuel and Time from 31000 to 35000 ft	129
E-5.	Step Descent Fuel and Time from 35000 to 31000 ft	130
E-6.	Scale Factor on Fuel Flow for Cruise-Climb.	132
E-7.	Holding Speed Fuel Rate at 25000 ft	133
E-8.	Payload Limit as a Function of Range.	137
F-1.	Simple Multistage Network	139
F-2.	Example Multistage Network.	141
F-3.	Multistage Network to Illustrate Computation Load.	144

LIST OF TABLES

	Page
1. Weather Data for Route 5, April 6, 1982	7
2. Effect of Wind Model Error.	18
3. Example of a Tabular Flight Plan.	52
4. Comparison of Flight Plan Costs and Run Times	64
5. Comparison of EFPLAN and OPTIM-7S Flight Profile Cost . . .	65
A-1: Comparison of Two-Parameter Search Results with Varying .	87
C-1. Maximum Cruise EPR vs Altitude and Temperature.	106
D-1. Definition of First Index I in Array A(I,L)	110
D-2. Definition of Second Index L in Array A(I,L).	111
E-1. 737 Holding Pattern Fuel Rate Requirements.	135
E-2. Weights and Limits for the TSRV (B737-100) Aircraft	136

PRECEDING PAGE BLANK NOT FILMED

INTRODUCTION

Flight management research began in the mid-1970s as a response of the NASA and the aviation industry to the fuel crisis. At that time, jet fuel became scarce, and the price per gallon rose more than 500% [1]. This rise in cost made fuel the dominant factor in an airline's operating cost - far outdistancing labor and equipment costs. Today, jet fuel prices have stabilized and even gone down a bit, as there is a temporary surplus in supply. Furthermore, new fuel-efficient aircraft have been developed with more efficient propulsion systems, lighter weight structures through use of composites, and lower drag through aerodynamic innovations such as winglets and control configured design. Boeing promises the "7J7" in 1992 which will feature laminar flow control, prop-fan engines, all-electronic cockpits, and a variety of new features that are to lower the cost per seat mile to about 50% of the average transport's cost today.

Part of the collection of improved aircraft design features has been the evolution of the airborne performance management (PMS) and flight management systems (FMS). These avionics systems are designed to help the pilot select more cost effective flight profiles to lower operating cost. Most new transport aircraft come today with FMS, and most airlines are systematically retrofitting their fleets with either PMS or FMS avionics. Thus, flight management technology to lower single aircraft operating costs has become widely accepted, and the research focus has shifted to lowering the operating costs on a fleetwide basis.

It can be said that flight management research is paying positive dividends - that the fuel cost per seat mile is coming down and that the system is being expanded to service a growing flying population. But we cannot relax because, despite a temporary surplus, the world's fuel supply is finite, and it will continue to decrease. There will be other future fuel crises, and it is best that the aviation community be well prepared for them.

To lower fleetwide operating costs requires continued research in three related areas:

1. Improvement of the reference paths followed by individual aircraft and their FMS by way of using time-controlled profiles (4-D guidance) and optimized flight planning techniques;
2. Incorporation of optimized flight plan concepts into the automated traffic management and flow control system with the intent that this new system will facilitate optimized flight while significantly removing the costs due to airspace saturation and delay; and
3. Upgrade of the ability to forecast and correctly model weather phenomena for use in flight planning and traffic management.

These research areas have different perspectives - the airframe, engine, and avionics manufacturers; the traffic managers and suppliers of weather information; the air transport operator; and the flying, tax-paying public. These perspectives must be integrated, and the research must take a systems approach. Important to the development of improved system capability is the ability to test new approaches in a realistic flight environment.

This effort addresses aspects of item (1) above. It develops a new program, called EFPLAN, for conducting flight planning research. EFPLAN is complementary to the OPTIM series of programs for conducting flight management research developed earlier[1].

The next chapter summarizes the effort to design the EFPLAN program. EFPLAN stands for Experimental Flight PLANning program. As the name implies, EFPLAN generates an optimized flight plan for the NASA TSRV aircraft between a given city pair. This program is designed so that its flight plan output is compatible as input to the OPTIM-7S FMS simulation program.

First, an existing airline flight plan is evaluated to show that even with only vertical optimization, reductions in cost can be made to this flight plan. This provides motivation for this continuing research. Then, a case is made for why flight planning and flight management research are strongly related and should be conducted together. The steps of generating a flight plan are next explained. This is followed with a description of the details of the EFPLAN program. In particular, attention is paid to the use of dynamic programming as the technique for generating the three-dimensional flight path during the cruise phase. Examples are given of EFPLAN's output and utility.

Chapter III summarizes this study and makes recommendations regarding continuing research. Appendices A through F provide technical detail of the new algorithms and modeling used to develop EFPLAN. In particular, Appendix A develops a new method of optimizing the climb and descent profiles through choice of the two speed parameters of the "Mach/CAS schedule." Appendix B presents the details of how non-standard day temperature profiles should be modeled in both FMS and FPS algorithms. Appendix C summarizes the modeling detail of the 727-200 performance data used to evaluate the existing flight plan in the next chapter. Appendix D describes the weather data base and its format as obtainable from the National Weather Service for use in flight planning. Appendix E summarizes some of the aircraft performance modeling techniques required for efficient flight planning. Finally, Appendix F outlines the fundamentals of the dynamic programming concept used to determine the near-optimal 3-D reference path of the EFPLAN program.

GENERATION OF NEAR OPTIMUM FLIGHT PLANS

Evaluation of an Existing Flight Plan

A generic model of a 727-200 aircraft was developed for the OPTIM-7S FMS simulation program, and this was used to evaluate an existing flight plan provided by United Airlines. The 727-200 is currently the most popular transport aircraft flying. The details of the modeling of this aircraft are summarized in Appendix C. The United Airlines (UAL) flight plan was for a flight between Chicago (ORD) and Phoenix (PHX) flown on April 6, 1982. This included all the additional information required to analyze the cost of that flight.

The route for the ORD-PHX trip is shown in Fig. 1. This route is designated as Route 5 by UAL for this city pair. This route is one of eleven consistently evaluated by UAL for this city pair. Distances between VOR stations in nautical miles are noted on the figure. It is the practice of UAL to add cruise distance to the flight plan to account for takeoff and descent maneuvers. This procedure was used for this example; 12 nmi were added to the takeoff leg, and 10 nmi were added to the landing leg. The takeoff gross weight of the aircraft for this trip was 171300 lb. Trip distance was 1294 nmi. In Fig. 1, the waypoints are indicated by the 3-letter call name of the associated waypoints - IOW, LNK, GLD, PUB, FMN, and INW. Each is numbered (1-6) for reference in the later plots. There are seven segments in this route.

Weather profiles were also provided by UAL for this route, and they are presented in Table 1. The wind velocity (magnitude and direction) is given over the range of altitudes from 18000 ft to 39000 ft at the six waypoints of Fig. 1. Wind velocity is assumed constant below 18000 ft. Ambient temperature is given in terms of deviation from standard day in degrees centigrade. Origin and destination weather is assumed to be the same as the nearest waypoints.

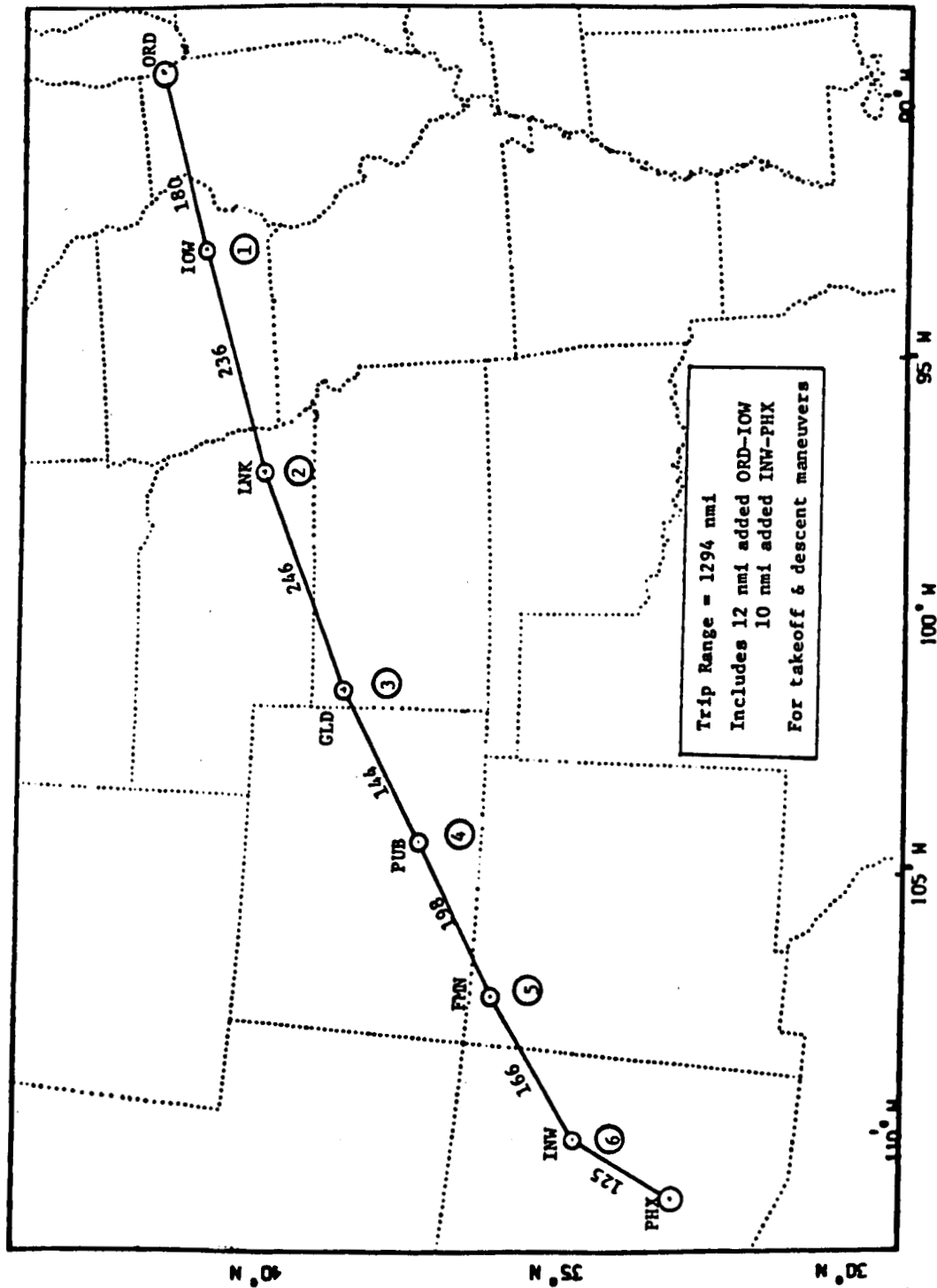


Figure 1. Chicago (ORD) to Phoenix (PHX) Route 5.

Table 1. Weather Data for Route 5, April 6, 1982.

TOTAL RANGE 1294. IN 7 SEGMENTS.			
WIND SEGMENT 1 STARTING INDEX 1 RANGE 180.			
HEIGHT BEARING VELOCITY TEMPERATURE*			
	DEG	KNOTS	C°
18000.00	260.00	49.00	-1.30 0
24000.00	260.00	73.00	0.50 0
30000.00	260.00	123.00	-1.60 0
34000.00	270.00	136.00	-0.60 0
37000.00	270.00	118.00	1.50 1
WIND SEGMENT 2 STARTING INDEX 6 RANGE 416.			
HEIGHT BEARING VELOCITY TEMPERATURE			
	DEG	KNOTS	C°
18000.00	250.00	50.00	-1.30 0
24000.00	250.00	77.00	0.50 0
30000.00	250.00	105.00	0.40 0
34000.00	260.00	115.00	1.40 0
37000.00	260.00	104.00	2.50 1
WIND SEGMENT 3 STARTING INDEX 11 RANGE 662.			
HEIGHT BEARING VELOCITY TEMPERATURE			
	DEG	KNOTS	C°
18000.00	250.00	58.00	-2.30 0
24000.00	250.00	81.00	-0.50 0
30000.00	250.00	100.00	1.40 0
34000.00	250.00	110.00	3.40 0
37000.00	250.00	103.00	3.50 1
WIND SEGMENT 4 STARTING INDEX 16 RANGE 806.			
HEIGHT BEARING VELOCITY TEMPERATURE			
	DEG	KNOTS	C°
18000.00	250.00	61.00	-2.30 0
24000.00	250.00	82.00	0.50 0
30000.00	250.00	107.00	2.40 0
34000.00	250.00	119.00	4.40 0
37000.00	250.00	105.00	3.50 1
WIND SEGMENT 5 STARTING INDEX 21 RANGE 1004.			
HEIGHT BEARING VELOCITY TEMPERATURE			
	DEG	KNOTS	C°
18000.00	260.00	63.00	-2.30 0
24000.00	260.00	85.00	1.50 0
30000.00	260.00	106.00	3.40 0
34000.00	260.00	113.00	4.40 0
37000.00	260.00	102.00	3.50 1
WIND SEGMENT 6 STARTING INDEX 26 RANGE 1170.			
HEIGHT BEARING VELOCITY TEMPERATURE			
	DEG	KNOTS	C°
18000.00	260.00	67.00	-0.30 0
24000.00	260.00	92.00	3.50 0
30000.00	260.00	114.00	4.40 0
34000.00	260.00	113.00	4.40 0
37000.00	260.00	104.00	1.50-1

*Temperature deviation from standard day.

ORIGINAL PAGE IS
OF POOR QUALITY

The UAL flight cruised at an altitude of 35000 ft. The first step was to use the OPTIM program to generate reference profiles that cruised at this fixed altitude and to assign a range of values to the cost-of-time parameter C_t . This parameter was varied from zero to \$900/hr. As it increased, cruise speed increased to further reduce the trip time. Note that a value of zero for the cost of time represents a minimum fuel flight trajectory.

The comparison of cruise speeds for this parametric variation are given in Fig. 2. Cruise Mach number increases with cost of time as expected, but there is virtually no change above a time value of \$600/hr.

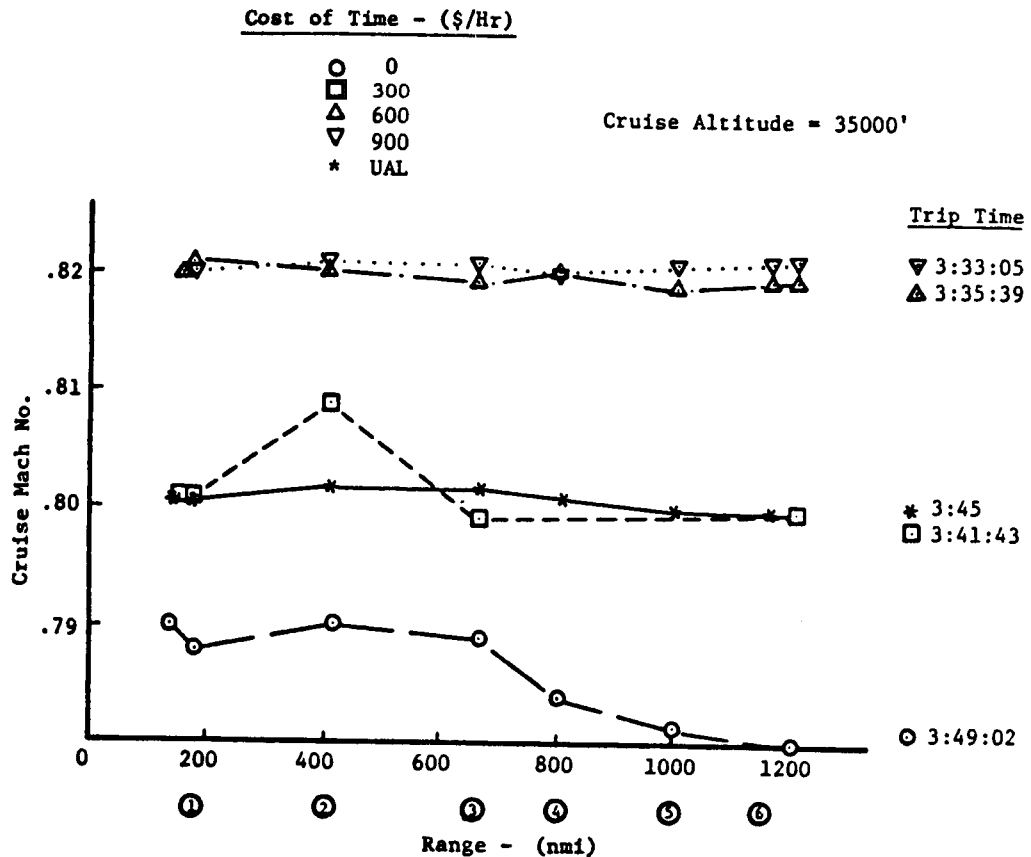


Figure 2. Cruise Speed Comparison as a Function of Cost of Time with UAL Results.

The UAL cruise Mach number for the actual flight is also plotted on the figure; it remains approximately constant at 0.80. The OPTIM data for a time cost of \$300/hr matches the UAL cruise speed fairly well. Note that the overall trip times are given on the figure. The OPTIM result for a time cost of \$300/hr matches the UAL flight plan (within 3 minutes and 17 sec). The comparisons of trip time vs fuel for this study are shown later.

Ranges for the climb and descent profiles are shown in Fig. 3 as a function of cost of time. As C_t increases, a longer climb profile and a

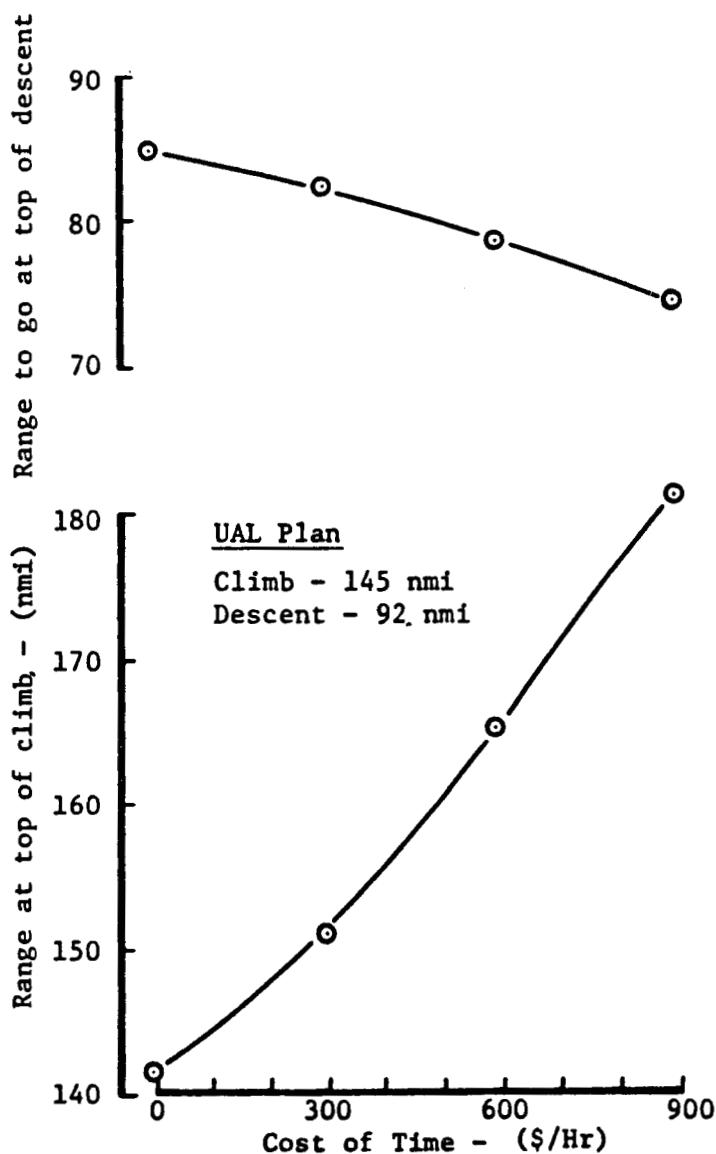


Figure 3. Comparison of Climb and Descent Range Requirements as a Function of Cost of Time When Cruise is at 35000 ft.

shorter descent profile result. By comparison, the UAL climb range is relatively short, and the UAL descent range is relatively long. Both compare to the OPTIM results with low values of C_t .

It is well known that a cruise-climb trajectory is more efficient than a constant cruise altitude trajectory. Cruise-climb is occasionally allowed in certain regions if traffic is very light, but generally it is not allowed by air traffic control (ATC). Nonetheless, it is useful to evaluate the costs of reference profiles that use the cruise-climb trajectory to determine ultimate economy. Altitude vs range profiles are shown for these trajectories in Fig. 4. Again, the range of cost of time varies from \$0/hr to \$900/hr. It is interesting to note that increasing time cost reduces the trajectory altitude over the complete trip. However, the top-of-descent point — 34200-35000 ft — is fairly close for all time costs.

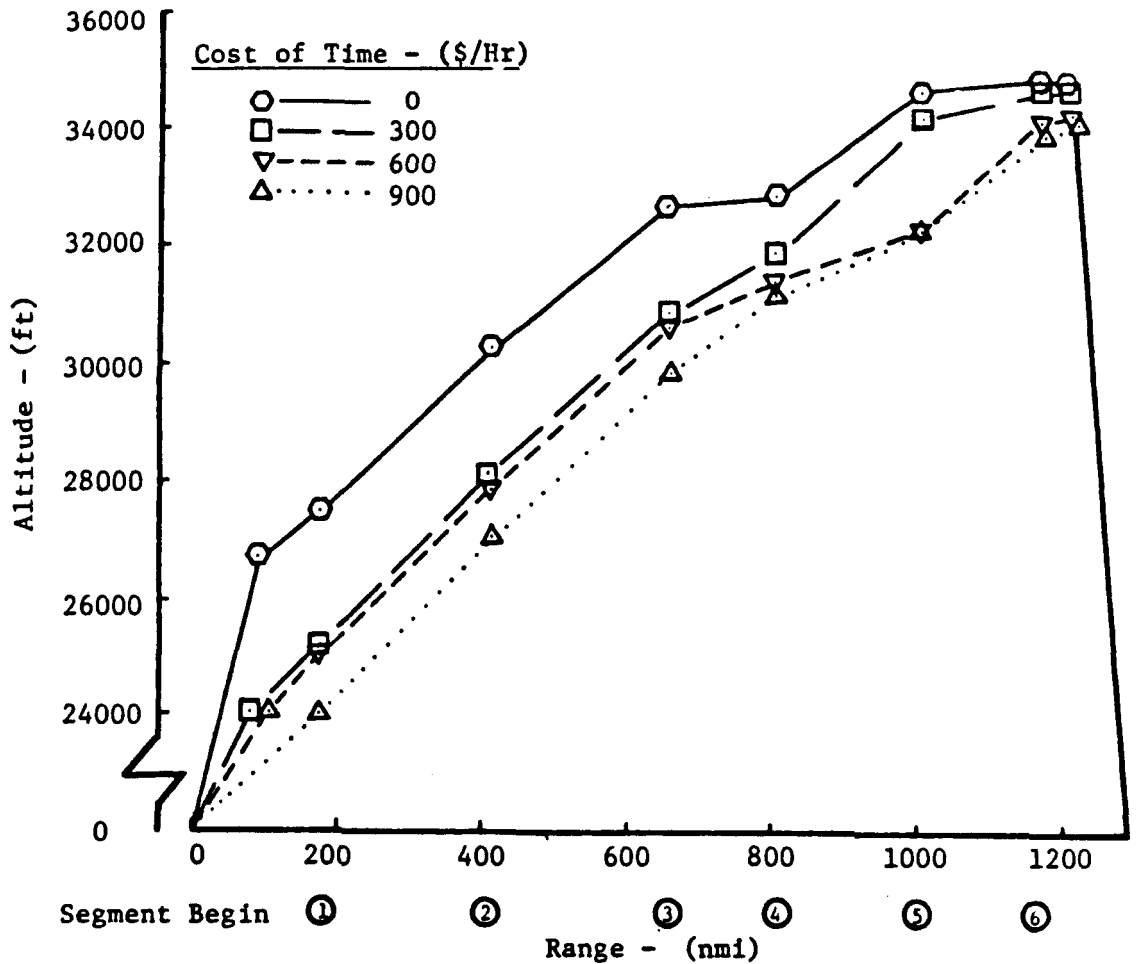


Figure 4. Comparison of Altitude vs Range for Cruise-Climb Profiles with Varying Cost of Time.

The cruise Mach numbers corresponding to these profiles are shown in Fig. 5. As expected, Mach number increases for increasing C_t . However, cruise Mach number is approximately constant at 0.82 at time cost above \$600/hr. This is a result of the steep rise in the drag coefficient at this speed. Note that for all these profiles, Mach number initially rises but levels off at 400-600 nmi into the trip.

The reference profile results for the flight with a constant cruise altitude of 35000 ft and cost of time of \$300/hr closely matches the actual UAL flight plan results, and it is useful to compare the fuel burn over each segment of this flight. This is done in Fig. 6. Also shown are cruise-climb fuel burn results for a time value of \$300/hr.

Because the OPTIM program computed near-optimum trajectories for climb and descent, one would expect differences from the UAL data. These differences exist as shown in the first and last segments plotted in Fig. 6. For both segments, the OPTIM results show a reduced fuel burn when compared to the UAL data. For the remaining segments, all at cruise, the fuel burn estimated by OPTIM at a cruise altitude of 35000 ft compares well with the UAL data.

For the cruise-climb trajectory, the fuel savings over the constant altitude cruise at 35000 ft occurs early in the trip. The altitude at the end of the first segment is 25000 ft so the reduced fuel burn is to be expected. There is also a reduced fuel burn for the cruise-climb trajectory in the second segment, but for the remaining segments there is no advantage.

An overall comparison of the trip time and fuel burn is shown in Fig. 7. The savings shown for both trip time and trip fuel for the cruise-climb trajectories is significant. As one would expect, the greater fuel saving is at the lowest cost of time (zero), and the greatest time savings is at the highest cost of time (\$900/hr).

However, the cruise-climb results represent only an ideal at this point when considering what can be done for flight planning. It may be

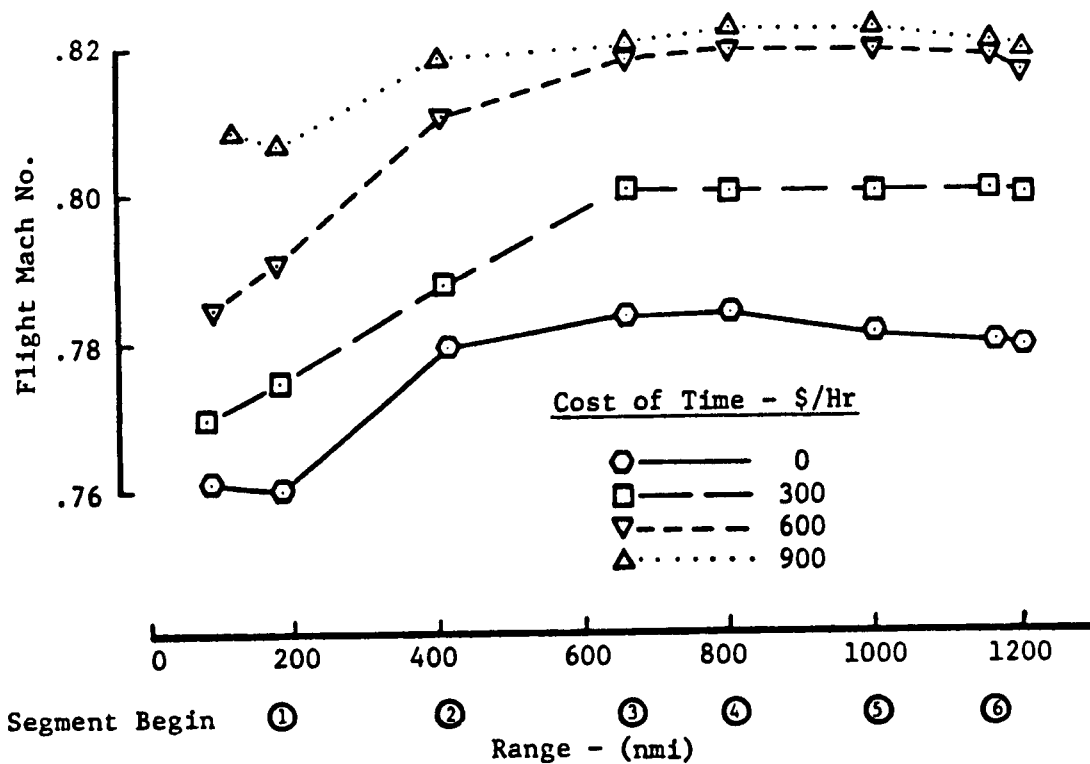


Figure 5. Comparison of Cruise Mach Number vs Range for Cruise Climb Profiles with Varying Cost of Time.

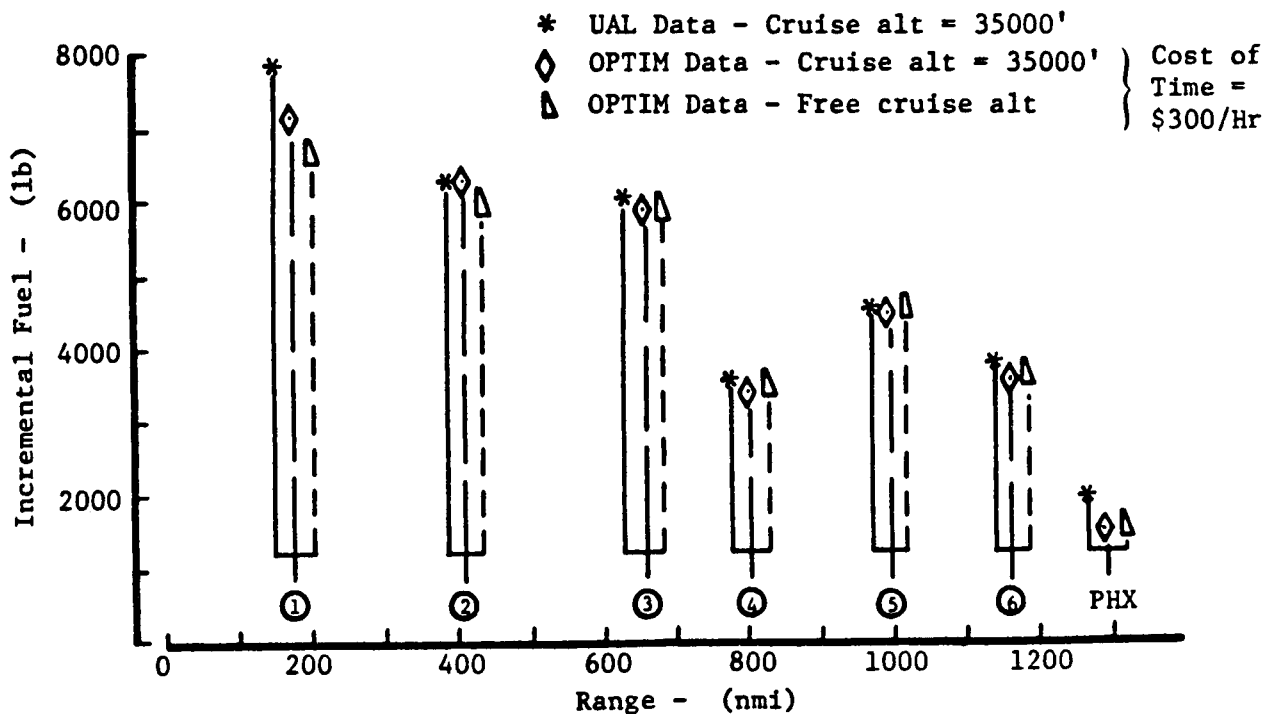


Figure 6. Comparison of Individual Flight Segment Costs for the ORD-PHX Route.

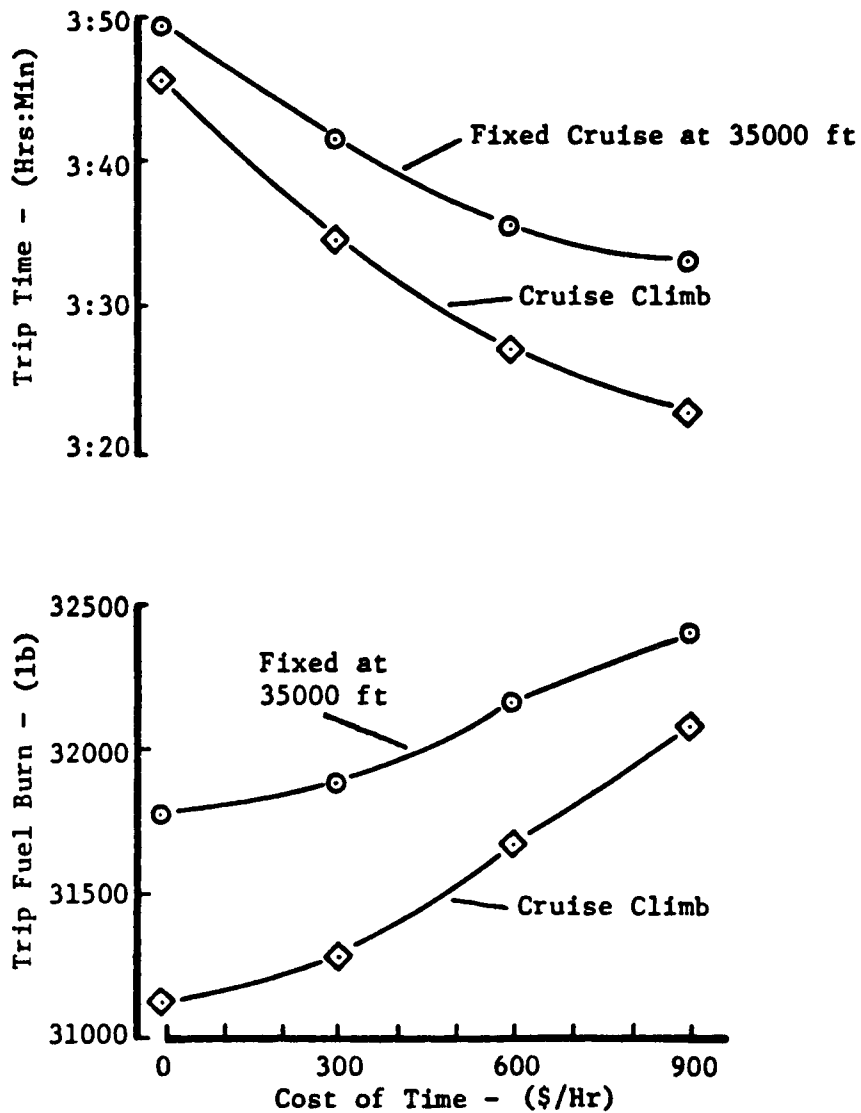


Figure 7. Comparison of Constant Cruise Altitude and Cruise-Climb Results in Terms of Trip Time and Fuel Burned.

years before ATC will allow general cruise-climb flight. However, this information can give us some clues with respect to use of the step climb.

A common practice for long range cruise is to have a step climb at some mid-range point. Three points were considered in stepping from 31000 ft to 35000 ft as shown in Fig. 8. Also shown is the cruise-climb trajectory given before. All trajectories are for C_t set at \$300/hr.

The trip time and fuel burn results for the three different step points are shown in Fig. 9. As expected, trip time is reduced for the

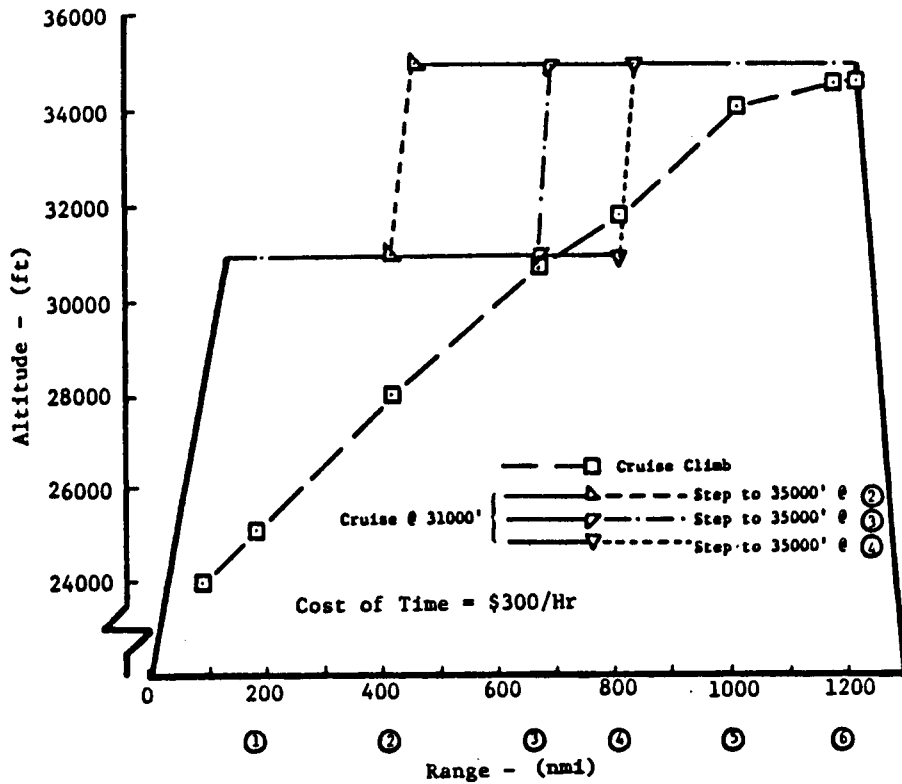


Figure 8. Comparison of Three Step Climb Altitude Profiles with Cruise Climb.

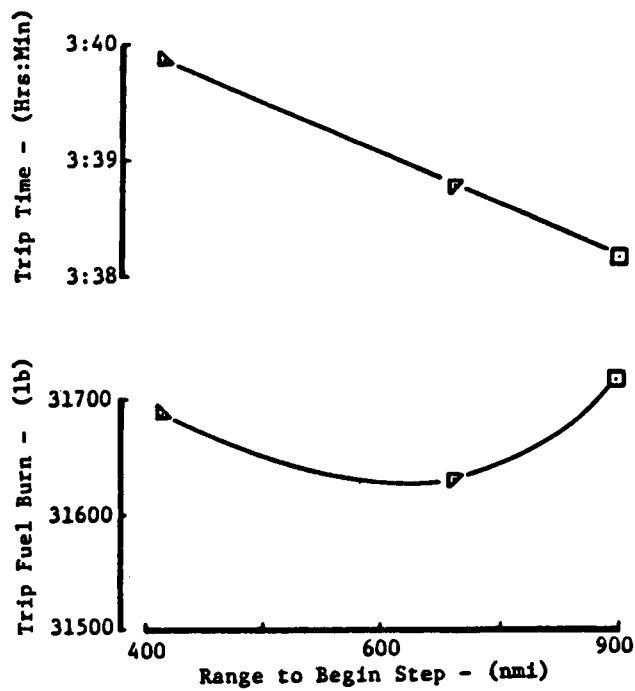


Figure 9. Comparison of Trip Fuel and Trip Time as Functions of Step Climb Point

longest range before the step (i.e., true airspeed is higher at 31000 ft). It is interesting to note that the minimum fuel step occurs at a point where 31000 ft intersects the cruise-climb trajectory.

The summary of this preliminary evaluation is shown in Fig. 10 which clearly demonstrates the tradeoff in trip time versus trip fuel burn. Also shown is the significant advantage that would result if this flight would have had a cruise-climb trajectory rather than cruise at a constant altitude of 35000 ft. A good part of this advantage can be attained by using the step climb. Finally, the UAL data point (trip time and fuel) for this flight is shown. Most of the fuel increment above the OPTIM result is attributed to the climb trajectory segment. Also, OPTIM does not account for takeoff or landing fuel requirements; adding these would shift the curves in Fig. 10 to the right.

All of the results shown must be considered preliminary. But, from this evaluation one can conclude that use of optimization techniques offers the potential for fuel/cost savings and versatile flight planning and flight management. In particular for the UAL flight plan, 400 or more lb of fuel (1.2% of the trip total) could have been saved by judicious choice of a step climb. This saving could have more than doubled with use of a cruise-climb. This provides the motivation for developing better flight planning methods to complement the on-board flight management process.

The Flight Planning - Flight Management Connection

The objective of the airborne flight management system is to compute a minimum-cost vertical flight profile connecting the origin and destination airports over some predefined horizontal route. The profile is subject to Air Traffic Control (ATC) and aircraft operational constraints. ATC constraints include terminal speed limits, enroute altitude constraints and potentially, time-of-arrival windows to facilitate flow control during some future era. Operational constraints include aircraft speed limits due to buffeting or structural limits, altitude ceilings, and possible cabin pressure limits on climb/descent rates.

- * UAL Cruise Alt = 35000 ft
- Cruise Altitude = 35000 ft
- ◇ Free Cruise Altitude
- Step Climb -- 31000 to 35000 ft

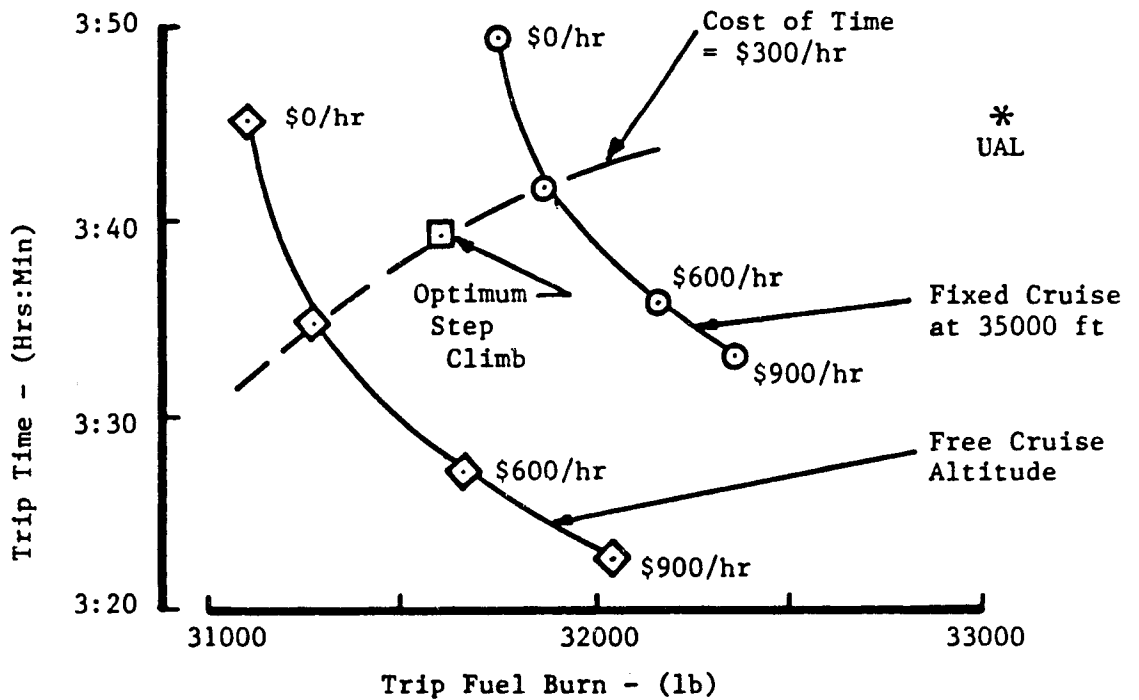


Figure 10. Tradeoff Between Trip Fuel and Trip Time for Fixed Cruise, Step Climb, and Free Cruise Altitude Profiles.

To compute an optimum profile, several types of data must be stored in the FMS computer [4,5]. First, consider the aircraft mathematical model itself. Either curve fit or tabular data of lift coefficient C_L , drag coefficient C_D , thrust, and fuel flow over the entire flight regime must be present as functions of Mach number, altitude, temperature, and angle-of-attack. In addition, aircraft performance limits must be included in these data. An incorrect model can significantly affect computation of the optimum cruise condition, and thus, the achievable savings from the FMS. For example, Fig. 11 shows the effect of not including normal installation and bleed losses on engine thrust and fuel flow models [1]. Optimum cruise speed can change by more than 10 kt (0.02 Mach) due to these losses.

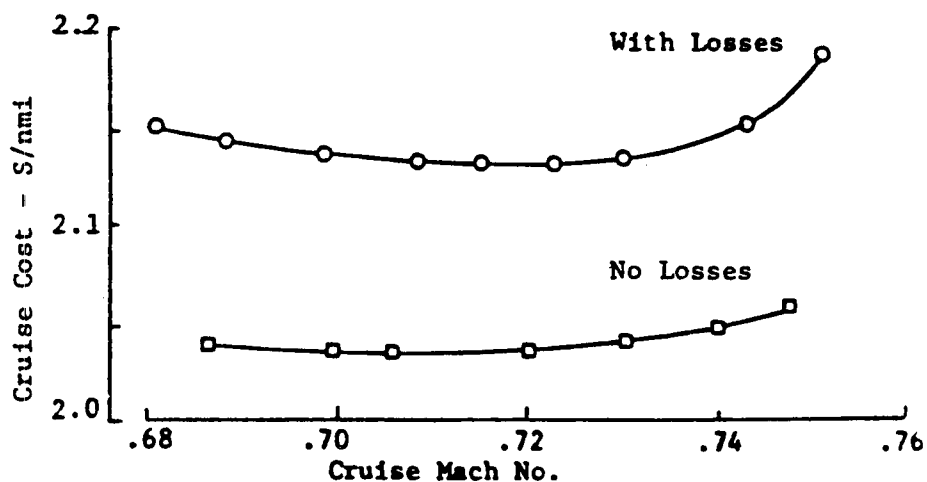


Figure 11. Example Effect of Powerplant Losses on Cruise Cost [1].

Two points need to be noted here. First, it is important that the ground-based flight planning system (FPS) and the airborne FMS use aircraft performance models that are reasonably similar so that the FPS will produce a pre-flight profile which agrees with the one produced by the on-board FMS. Second, there must be a systematic method of collecting aircraft performance data during flight and using these to tune the aircraft/engine models. Variable installation losses, engine and aerodynamic wear, and overhaul all affect individual aircraft performance and need to be tracked so that the FMS/FPS models produce the best operating reference. This need

for a parametric identification method (either on-board or as part of the flight planning process) is a corollary research area.

The second type of data required in the FMS is the vertical weather model. This assumes that the horizontal path is fixed so that lateral weather variations are not required. The weather model consists of wind (speed and direction) and temperature as functions of altitude and range. These are specified at waypoints along the route and at specific pressure altitudes. The weather model and computed aircraft weight are used by the FMS optimization algorithm to choose the best altitude and speed setting during cruise and to choose where to make step altitude changes.

The lack of knowledge of the true wind and temperature fields is the chief source of error and loss of performance produced by the FMS/FPS combination. Table 2 shows an example of wind inaccuracy on flight performance when the profiles are computed by an FMS [1]. Here, it is assumed that a wind profile varying from 0 kt at 0 ft to 100 kt at 40000 ft is constant over a 1000 nmi trip. The FMS computes four profiles based on the assumed wind conditions. In the first two cases, the actual wind is a head and tail wind, but the FMS assumes no wind. In the second two cases, there is no actual wind, but the FMS assumes a head and a tail wind. Costs are based on fuel at \$.15/lb and time at \$600/hr. As can be seen, cost losses range from \$5.26 to \$115.13 because of the wind model errors.

Table 2. Effect of Wind Model Error [1].

Model	Actual	Fuel	Time	Cost
Wind	Wind	Error	Error	Error
		(lb)	(sec)	(\$)*
None	Head	-378	1031	115.13
None	Tail	194	-143	5.26
Head	None	695	-330	49.24
Tail	None	-136	283	26.76

* Based on \$.15/lb and \$600./hr.

The FMS vertical weather model is input from the FPS. Here again, the horizontal FPS flight plans are limited by the goodness of the weather data base. Both NASA and FAA have recognized the shortcomings of the current weather models that flight planners obtain every 12 hr from the National Weather Service (NWS) [3,6].

With a ground-air data link, such as provided by the ARINC Communications and Reporting System (ACARS), it is also possible for the FPS to update the vertical weather model of the FMS while the flight is in progress [7]. Thus, with the airborne system also updating its own weather model by using its own measurements, the loss of performance due to weather knowledge inaccuracy can largely be removed. In addition, the data link can provide weather measurements back to the ground for inclusion in the weather data base.

On-board measurements would be data-linked to a central computer and blended with rawinsonde data so that 3-hr updates of weather forecasts could be available for flight planning. In addition, the forecast data could be available over a finer grid and every 2000 ft of altitude between 15000 ft and 43000 ft instead of the standard millibar levels. Thus, greatly improved predictions of weather fronts and jet stream locations are possible. Figure 12 shows the improved rms accuracy that is available by using the NASA design. Steinberg estimates that up to more than 4% fuel improvement is possible by the airline flight planning process utilizing this improved weather model [3].

The third type of data required by the FMS is the horizontal path to be followed as computed by the FPS. It consists of a sequence of either latitude/longitude waypoints, range/bearing segments, ground navigation aids, or published airways, depending upon the FMS input format. This flight plan also includes the suggested range points to make step climbs (or sometimes step descents) and the required fuel load. The data link can be used to transfer this information directly from the FPS to the FMS computers. As with the vertical weather model, the horizontal path can be changed by the ground-air data link while the flight is underway.

ORIGINAL PAGE IS
OF POOR QUALITY

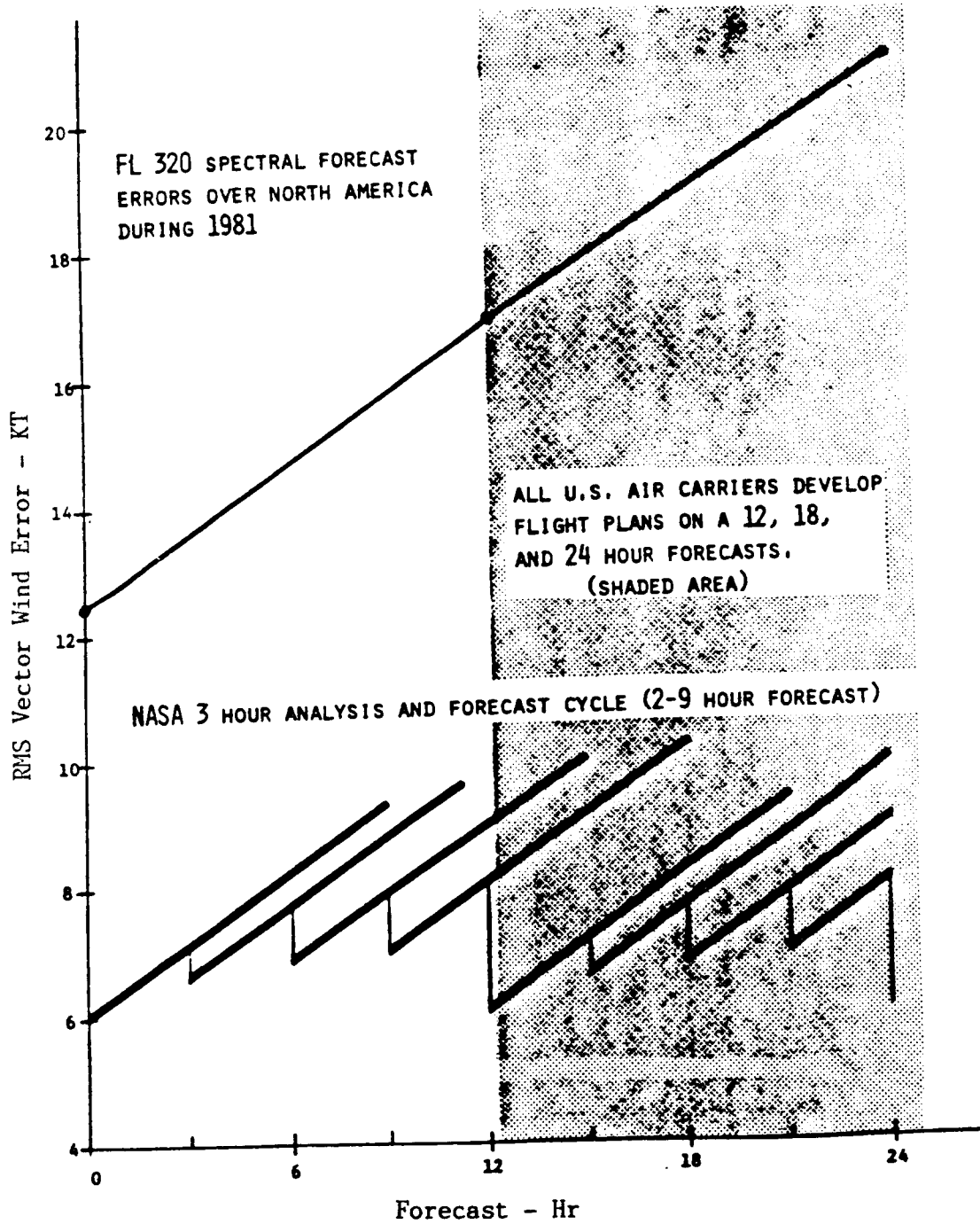


Figure 12. Example Windfield Error as a Function of Forecast Time and Potential for Improvement [3].

In summary, the airborne FMS and ground based FPS need to be tied together by data link. Weather and performance measurements are down-linked to improve the overall weather model and individual aircraft/engine performance models. Vertical weather models and the desired horizontal path are uplinked as input to the FMS.

The Flight Planning Sequence

EFPLAN is based on a variation of the process followed by an air carrier to generate a flight plan. The flight planning process specifies to the pilot what route to fly, how much fuel to carry, alternate (emergency) destination, and specific flight-dependent information such as radio frequencies, expected arrival times, and reporting requirements. The route description consists typically of the sequence of waypoints, the altitudes and Mach numbers for the connecting segments, and specific points such as top-of-climb, top-of-descent, and step climb beginning.

The flow of the flight planning computing sequence is shown in Fig. 13. The dispatcher enters program control variables before the flight, and this may be as simple as the flight number on a given day. The program is driven by an executive routine that refers to the operator's predefined requirements for a flight and procedures to be followed during the flight. This includes type of aircraft, scheduled departure times, city pairs, alternate destinations, fuel policy regarding time/fuel tradeoffs, source of fuel for multiple operation routes, and expected passenger/cargo payload from the company flight reservation system.

The program then queries three different data bases to obtain information necessary to generate the flight plan - aircraft performance, route alternatives, and weather. These are expanded versions of the data used by the airborne FMS.

The aircraft performance data base includes models for fleet average or individual airplane and engine combinations that are used. In the latter case, the serial numbers of the aircraft assigned to a flight on a

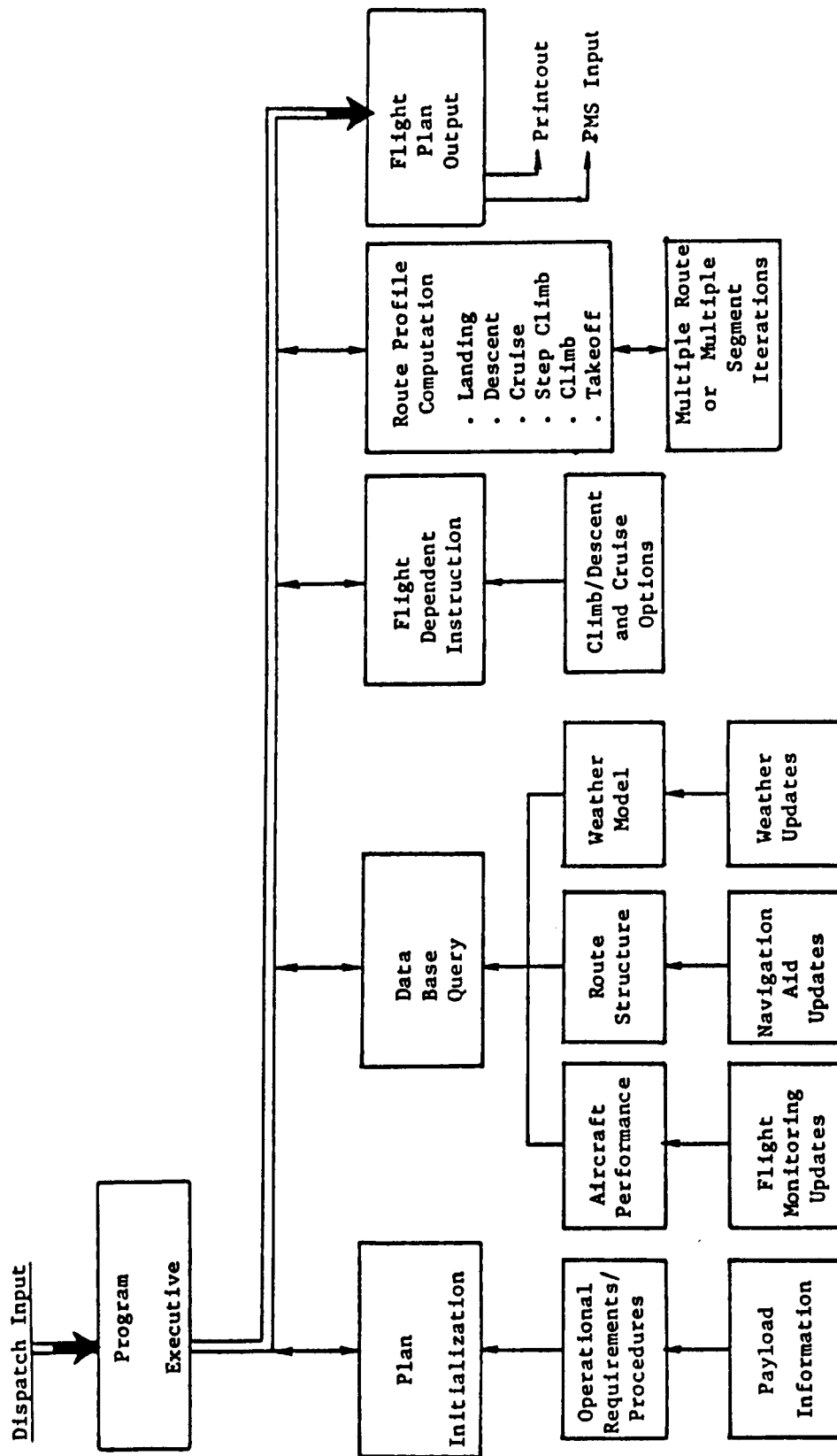


Figure 13. First Level Flow Chart for a Flight Planning Program.

given day must be known. This data base is regularly updated (e.g., monthly) based on results of monitoring aircraft performance during flight.

The route structure data base consists of all the possible route segments and navigation aid locations that support each given city pair. For short range flights, there is usually only one horizontal route considered. Continental and oceanic routes have different configurations. Area navigation (RNAV) capability on the aircraft governs whether "random routing" is possible. This route structure data base must also be updated regularly to account for changes in restricted areas, non-operating VORs, etc. The FAA provides this information.

The third data base is the global weather model. This is updated every 12 hours by the NWS. Research is underway to allow more frequent updates with more accurate data. Some airlines have their own meteorological staff or buy information from independent sources to improve their weather data.

With data from the above sources, the program extracts information specific to the given flight. This includes instructions on how to generate the 3D path for the flight plan and which of various options to use. Also, the desired landing weight is computed. The landing weight includes empty weight, payload, and reserve, tankered, and gage inaccuracy fuel.

The 3D route profile is usually computed in backwards time beginning with the desired landing weight. For explanatory simplification, we first discuss the steps to generate a vertical (2D) reference profile in backwards time.

Based on the landing weight, the first step is to compute the final approach and landing requirements at the destination. This is the phase from flap deployment to touchdown. Ground temperature, wind, runway altitude, and landing approach path are factors. The output of this step is the aircraft weight, time, bottom of descent point, speed, heading, and fuel burned including allowance for ATC approach vectoring.

The next step is to compute the descent profile. Descent begins at the final cruise horizontal waypoint and ends at the bottom of descent point. A nominal horizontal path must be pre-defined that connects these two points. This is typically a Standard Arrival Route (STAR), and allowance is made for ATC vectoring.

The descent from cruise to 10000 ft usually follows a Mach/CAS schedule where the pilot first follows a constant Mach speed and then transitions to a constant indicated airspeed. The two Mach/CAS parameters can be optimized to minimize descent cost, as discussed in Appendix A, or two preset values may be used. The descent speed choice is an option to the particular user.

The next step is to compute the cruise phase which may include altitude steps. For any given route, several choices exist for the cruise speed. This may be a fixed Mach number, a speed such as long range cruise (LRC) that is a function of altitude and weight, a speed that is based on a company fuel policy, or a speed that minimizes a cost function such as

$$J = \frac{C_f \dot{w} + C_t}{V_g} \cdot (C_f \dot{w} + C_t) / V_g \quad (1)$$

Here, C_f and C_t are cost of fuel (\$/lb) and time (\$/hr), \dot{w} is fuel rate (lb/hr), and V_g is ground speed.

Currently, cruise altitudes are fixed at certain levels specified by ATC. The optimum choice of altitude levels can be obtained from minimizing the function of Eq. (1) over both speed and altitude. Each altitude has a maximum weight/speed curve that must not be violated. Because weight is added as the cruise phase is computed backwards in time, the maximum weight point at each altitude must be monitored as computation proceeds. The point to begin a step climb between altitudes can be solved as part of the optimization process over the feasible range of cruise points [1].

Now, several possible horizontal route paths may be possible. For routes greater than 500 nmi in range, a sizable cost gain can often be achieved by considering routes other than the shortest ground distance. The simplest way to expand the choice is to generate vertical flight profiles for several pre-defined horizontal routes, and then pick the one with the smallest cost. Many airlines use this "direct search" technique.

The next level of sophistication is to use a dynamic programming approach to select the horizontal or 3D cruise route path [8]. In this case, an evenly spaced grid of potential waypoints is first defined around the great circle arc connecting the origin and destination. With a three dimensional grid, this automatically includes step climb and descent considerations. Dynamic programming requires computing flight costs backwards from point to point through this "even grid". Some airlines use this technique today for oceanic route selection [9]. EFPLAN is based on dynamic programming.

The next level of sophistication in route selection is to use a calculus of variations approach to find the optimum 3D path. This requires solution of the two-point boundary value problem which can be quite formidable when considering flight through a wind field with extensive variation. The classical problem of Zermelo gives an approach to steering control when the airspeed is fixed [10]. Another approach to make this a tractable problem is to solve for the approximate optimal solution using dynamic programming and then use calculus of variations to tune this by finding the neighboring optimal solution [11]. However, flight planning programs must have a practical constraint on running time in that they must produce solutions prior to takeoff time but with reasonable estimates of current weather and payloads. This running time constraint may eliminate the calculus of variations approach.

After computing the cruise phase, the last steps are to compute takeoff and climb. These are best done in forward time where an estimate is made for takeoff weight. The takeoff weight estimate is finally adjusted so that there is a weight matchup at the first cruise waypoint.

The stages of computing the climb profile are the same as those for descent. Again, there is a choice of methods to compute the exact profile.

The takeoff phase goes from brake release to flap extraction. The data required for this computation are similar to that required for the landing phase computation.

The flight plan output is typically a printed flight profile summary table that can be used by pilots and crew to prepare for and conduct the flight. This may require keyboard data entry to the on-board navigation and FMS. The logical improvement is direct data communication from the FPS to FMS computers via data link. The work advantage and removal of human entry error by using this approach are obvious.

The steps depicted in Fig. 13 and the above discussion were followed in designing EFPLAN. The following sections provide detail on program content.

EFPLAN Options

EFPLAN has been designed with several user options built into the reference trajectory generation process. These include the following:

1. The program currently provides the choice of two types of networks for designing the horizontal path followed during cruise. The first uses the "direct search" technique. Here, a series of direct routes connecting the Standard Instrument Departures (SIDs) and Standard Arrival Routes (STARs) are pre-defined for the given city pair. These routes consist of one or more sequences of waypoints defined by VOR/DME, Vortac, or Tacan navigation aid (navaid) waypoints, and they are representative of routes chosen and used by airlines today for their scheduled flights. The second type of network consists of a grid of evenly spaced nodes forming a matrix of three-dimensional waypoints in the airspace between the city pair. The choice of a cruise path connecting a sequence of these "even grid" nodes is representative of the "random route" of the

future with aircraft having RNAV equipment. The FAA has begun the process of determining how current ATC constraints may be removed to allow eventual random routing.

2. EFPLAN has a choice of cruise altitudes to use. These include:
 - a. A single fixed cruise altitude.
 - b. Multiple fixed cruise altitudes. These altitudes can either be today's constrained flight levels or the future flight levels where altitude separation above 29000 ft may be reduced to 1000 ft.
 - c. Multiple cruise altitudes with step climb or step descent allowed between the levels. The program chooses the best locations for these steps.
 - d. Cruise-climb and descent. Here, the aircraft can fly at any altitude, and the program selects the best position based on wind and aircraft weight. This is a future possibility tied to random routing.
3. EFPLAN can use either fixed Mach/CAS climb and descent schedules, or these schedules can be optimized, as discussed in Appendix A.
4. EFPLAN can simulate direct transfer of the flight plan to the on-board flight management system. It has been designed so that if this option is chosen, the weather profiles and sequence of flight segments required by the OPTIM-7S FMS program are collected as special output.

There are future options that can be added to EFPLAN such as a cruise network defined by existing navaid locations or high altitude (JET) routes, or oceanic routes consisting of the existing track systems for both

Atlantic and Pacific flight. EFPLAN has been designed to facilitate expansion to include additional options as future research requires.

EFPLAN Input

Input Flags EFPLAN is designed to be controlled by ten integer inputs referred to as "flags." These are:

- IFN- represents a three digit flight number, and every example city pair studied is given a unique number. Numbers less than 600 are for east-to-west flights, and those greater than 600 are for west-to-east flight. For example, 101 is the number designated for flight from Norfolk to Charlotte, and 601 designates flight from Charlotte to Norfolk. These numbers are used to read in the appropriate route structure data for the designated city pair.
- IAD- designates whether an alternate destination airport is to be used.
- ICALT- determines whether a fixed cruise altitude, multiple cruise altitudes, step climb or descent, or cruise climb should be used for choosing the cruise path.
- ICAC- determines what specific flight levels to examine if multiple cruise altitudes are used.
- ICD- determines whether to optimize the Mach/CAS schedule for climb and descent.
- IHOP- specifies whether to use the direct search or even grid method for defining the cruise paths.
- IWNTM- specifies whether to use weather input data or to assume a no-wind, standard day.
- ITOA- this is a flag for future control of time-of-arrival. It is not active at this point.
- IOPTS- controls whether to create output as input to OPTIM-7S.
- IPRT- controls print output.

More detail on these flags and other input is presented in the EFPLAN User's Guide [11].

Weather A description of the weather format and content used in EFPLAN is presented in Appendix D. Basically, a three-dimensional (3D) matrix of data is used to represent the weather field over the contiguous 48 states. The matrix has a data point every 2.5 deg in longitude, 1.25 deg in latitude, and at ten different pressure altitudes measured in millibars. This format is known as the GRIB Code (for Gridded Binary); it represents a recent advance in the NWS weather product for aviation.

At each weather grid point, the wind is specified in north and east components. Also, the temperature deviation from Standard Day, and the geopotential altitude are given for each pressure altitude point. The method that EFPLAN uses to process non-standard temperature is presented in Appendix B. Figure 14 is a plot of the typical forecast wind vectors at 250 mb.

Currently, EFPLAN only uses one representative weather forecast during a run. A future expansion would be to add the logic to interpolate on time between two forecasts.

Some studies were conducted with weather data that were artificially generated. Artificial data allow the study of specific weather patterns to check particular program logic such as step climb and step descent. The user can study any weather pattern he chooses if it is placed in the GRIB code format. Figure 15 is an example of an artificially generated wind field at 33000 ft.

For cruise, a weighted average interpolation is used to find the weather data at a particular route waypoint from the surrounding eight points in the appropriate weather cube. For a given cruise segment, the wind and temperature profiles are found as functions of altitude for the beginning, center and end of each segment which is typically 100 nmi long. Then, the beginning, center, and end wind vectors are interpolated from these profiles and used to compute the ground speed over the segment.

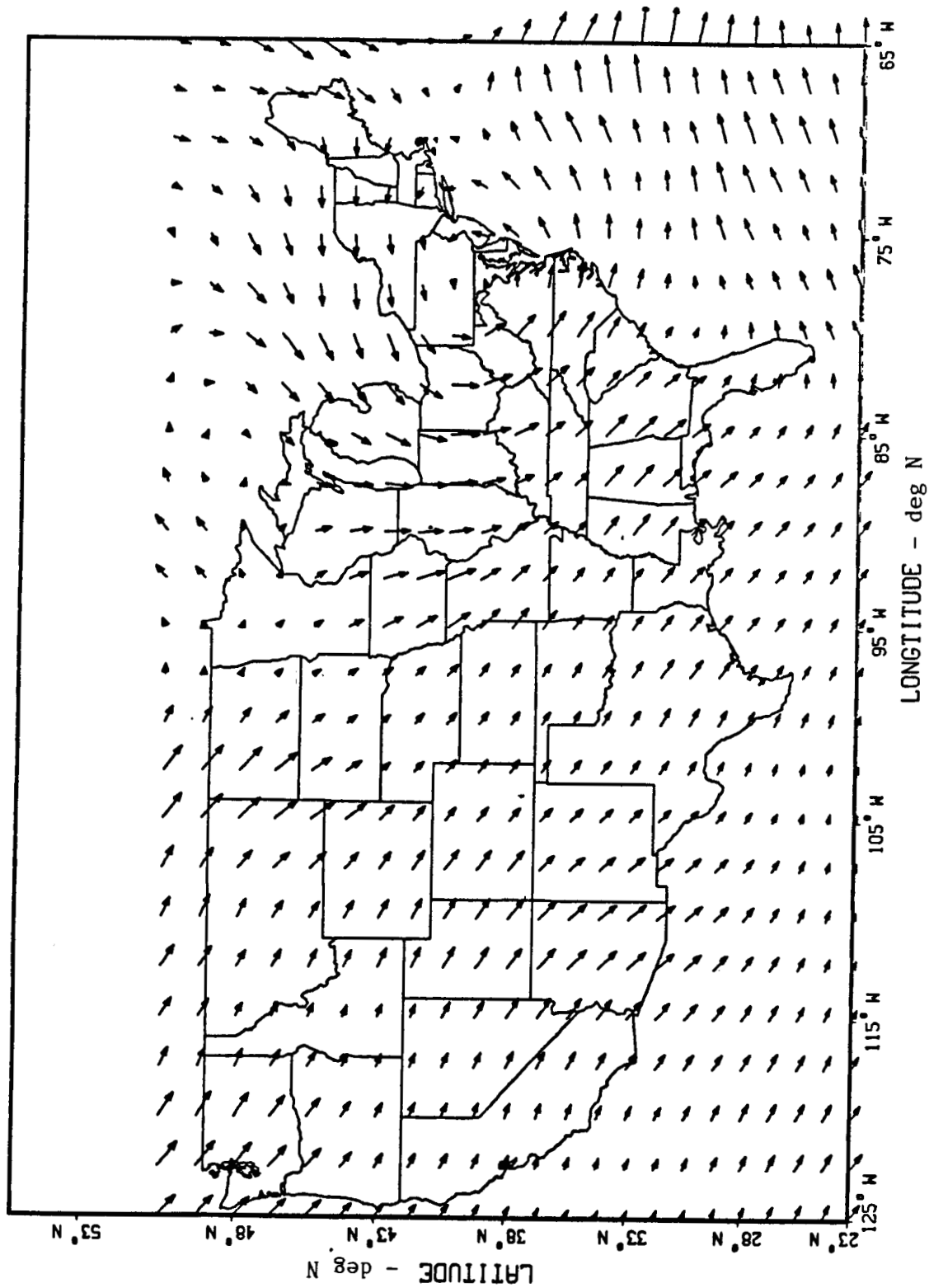


Figure 14. Sample of Weather at 250 mb (34000 ft).

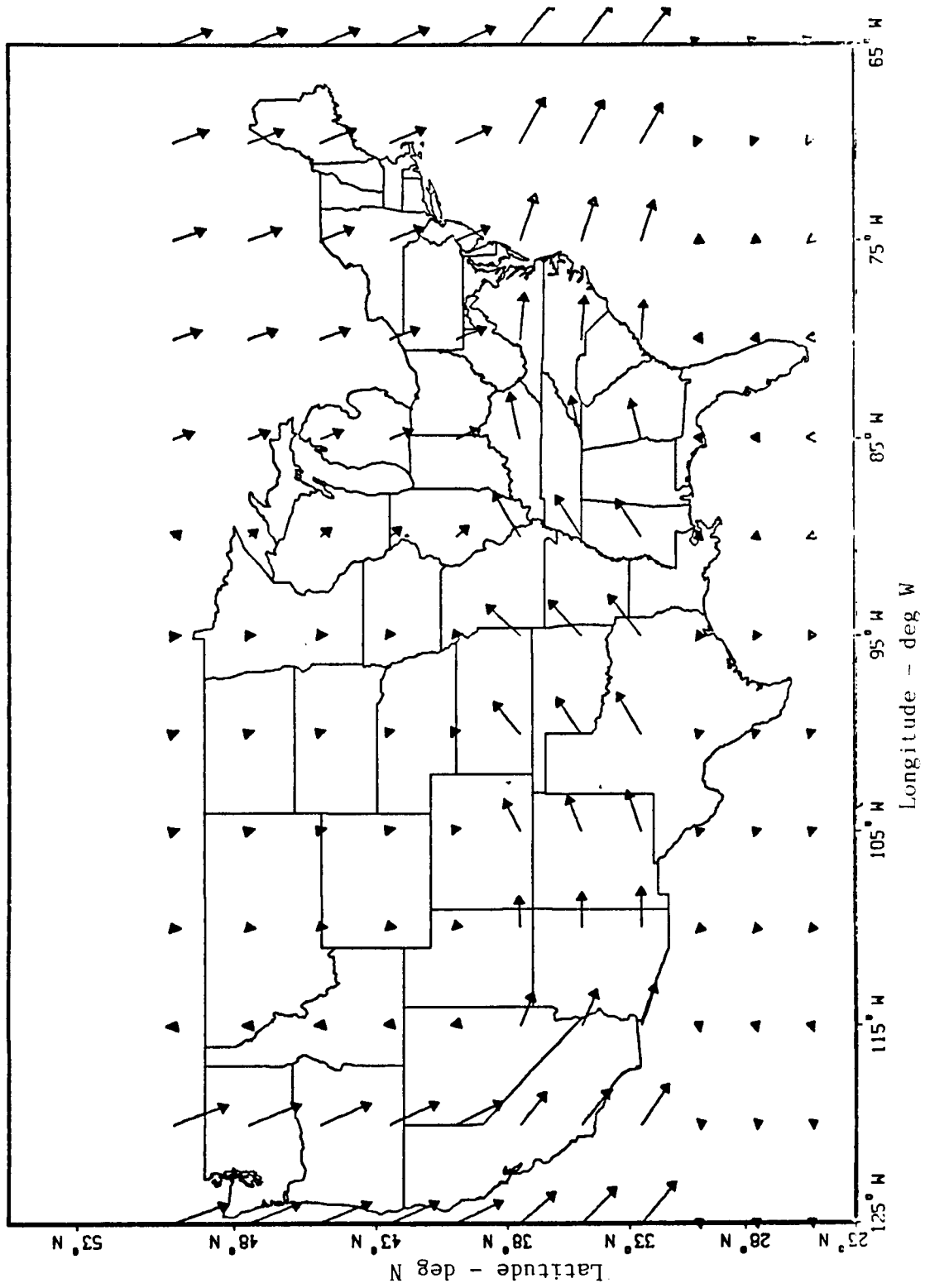
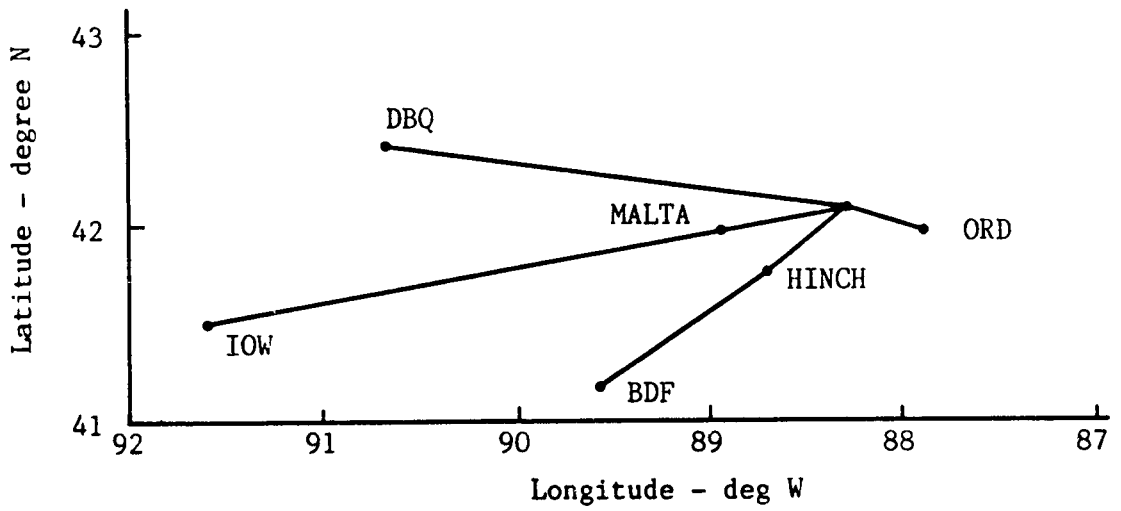
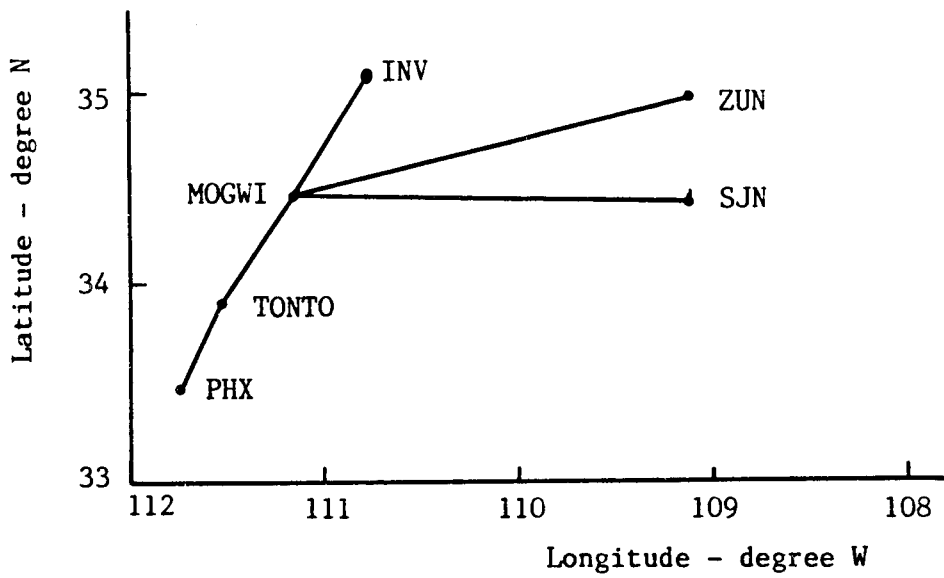


Figure 15. Artificially Generated United States Wind Field



a). Chicago O'Hare Westbound SID



b). Phoenix Northeast STAR

Figure 16. Example SID and STAR for Chicago - Phoenix Route.

For climb and descent, the SID and STAR paths are defined as sequences of range/bearing pairs. For each terminal segment, the altitude dependent weather profile is found at the center point. This profile is assumed to be independent of range over the segment.

Route Structure. Six city pairs have been set up as examples to exercise the EFPLAN program. They are Norfolk to Charlotte, Charlotte to Chicago, Chicago to Dallas-Ft.Worth, Dallas-Ft.Worth to Atlanta, Atlanta to Norfolk, and Chicago to Phoenix. We refer to the first five as the "Norfolk loop." Each city is specified by the latitude and longitude of the takeoff/touchdown point and the runway altitude. Each city pair is given a unique flight number IFN mentioned earlier. EFPLAN uses IFN to load in the city pair route structure plus the direction of flight.

EFPLAN is set up so that the user can specify arbitrary paths for the SID and STAR route structure in the terminal areas. For the city pairs used to test the program, some of the SID and STAR inputs are artificial, and some are those that are actually used today. Figure 16 shows the SID and STAR used for Chicago-Phoenix.

For the descent profiles, the end of the terminal phase is assumed to be a short cruise segment (about 10 nmi) past the top-of-descent for the upper altitude level $h(3)$ tested. Top-of-descent (TOD) is defined as where the aircraft begins to decelerate to initiate the descent. This is illustrated in Fig. 17. All terminal area paths that use this same STAR but are to lower altitudes $h(1)$ and $h(2)$ use the same horizontal stopping point. This point is referred to as a pseudo-waypoint, and it marks the beginning of the cruise phase of computations. Each STAR's horizontal path used for the arrival terminal area ends with a pseudo-waypoint defined by top-of-descent for the upper altitude of that path. Thus, multiple pseudo-waypoints define the beginning of the cruise phase.

For the climb profiles, when the even grid method of computing the cruise path is used, the upper altitude climb profile of each SID path is extended beyond the top-of-climb point out to a nominal range of RCLMB nmi.

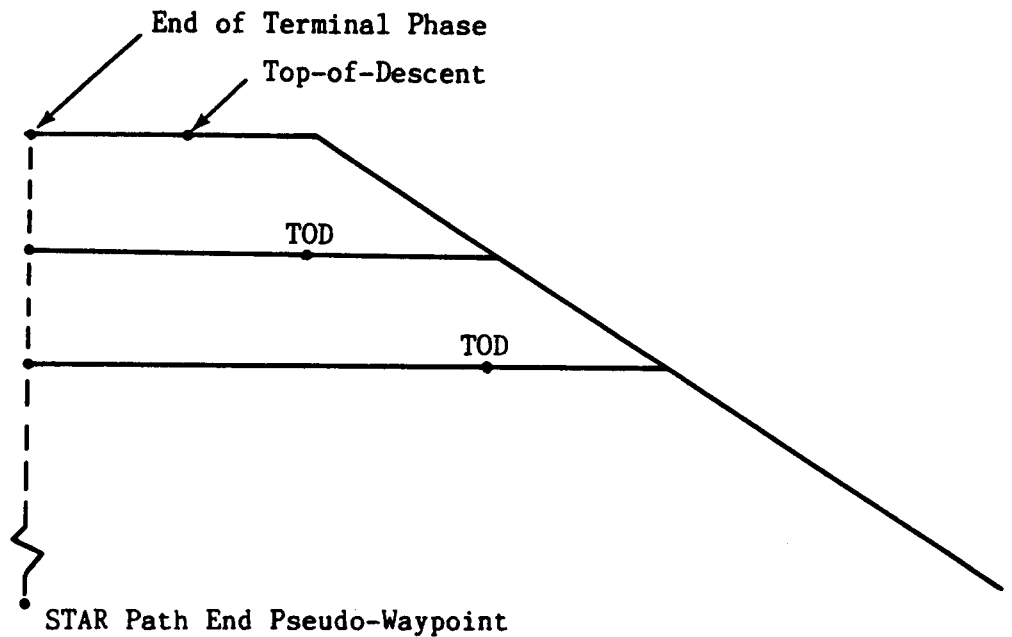


Figure 17. Illustration of Pseudo-Waypoint that Ends the Descent Phase

These points are also referred to as pseudo-waypoints. RCLMB is defined before the climb phase is computed to designate the range points where the cruise phase computations end. RCLMB must be set large enough so as to assure that the climb profile will reach the upper altitude level within the RCLMB range of the takeoff point. RCLMB is computed to be a function of performance index (C_t/C_f) and upper altitude level. For example, RCLMB is 150 nmi when upper altitude is 35000 ft and (C_t/C_f) is 2000 lb/hr.

For the even grid option, if the SID and STAR horizontal paths are not long enough to allow climbing to or descending from the upper altitude, the last segments of each path are artificially extended so that an appropriate distance is reached to define the pseudo-waypoints. For the direct search option, the cruise routes are specified by the navaid sequence and their latitude/longitude locations. For this case, the pseudo-waypoints which define the end of the climb phase are the first navaid locations on each specified route which are beyond the RCLMB range discussed above.

Direct routes can be defined by any sequence of waypoints that connect navaid locations. However, this produces a very large number of possibilities. For example, Fig. 18 illustrates the locations of all nav aids between Chicago and Phoenix. A judicious choice has to be made from these to produce a suitable selection of direct routes. Figure 19 shows one such set of eleven routes connecting Chicago and Phoenix that was used in this study. Figure 1 (shown earlier) is designated "Route 5" from the choices shown in Fig. 19.

Aircraft Performance. For climb and descent computations, EFPLAN uses the same aircraft performance data base at this point as is used by OPTIM. The data consist of lift coefficient as a function of angle-of-attack, drag coefficient as a function of lift coefficient and Mach number, and thrust and fuel flow as functions of engine pressure ratio (EPR), Mach number, temperature, and altitude. Future study may show that the descent and possibly the climb phase can be computed from tabular pre-computed results so that these data bases are not needed.

The predominant amount of computation for generating the flight plan consists of computing and searching over the sequence of segments which could be followed during the cruise phase. Because several flight plans could be required in a given day for an airline scenario, it is important to determine how the individual flight plans could be computed as rapidly as possible. Thus, we spent some effort in researching how to speed up the cruise phase computations. This led to the realization that what was required for optimizing the aircraft speed over a flight segment was a specification of the fuel rate over that segment; this is seen from Eq.(1). Thus, the first requirement for cruise performance was the generation of a trimmed aircraft fuel rate table. The characteristics of this fuel rate table are discussed in Appendix E. Briefly, this table provides trimmed cruise fuel rate as a function of temperature deviation from standard day, aircraft weight, pressure altitude, and Mach number.

Three other performance tables are used for the cruise phase of flight. These include:

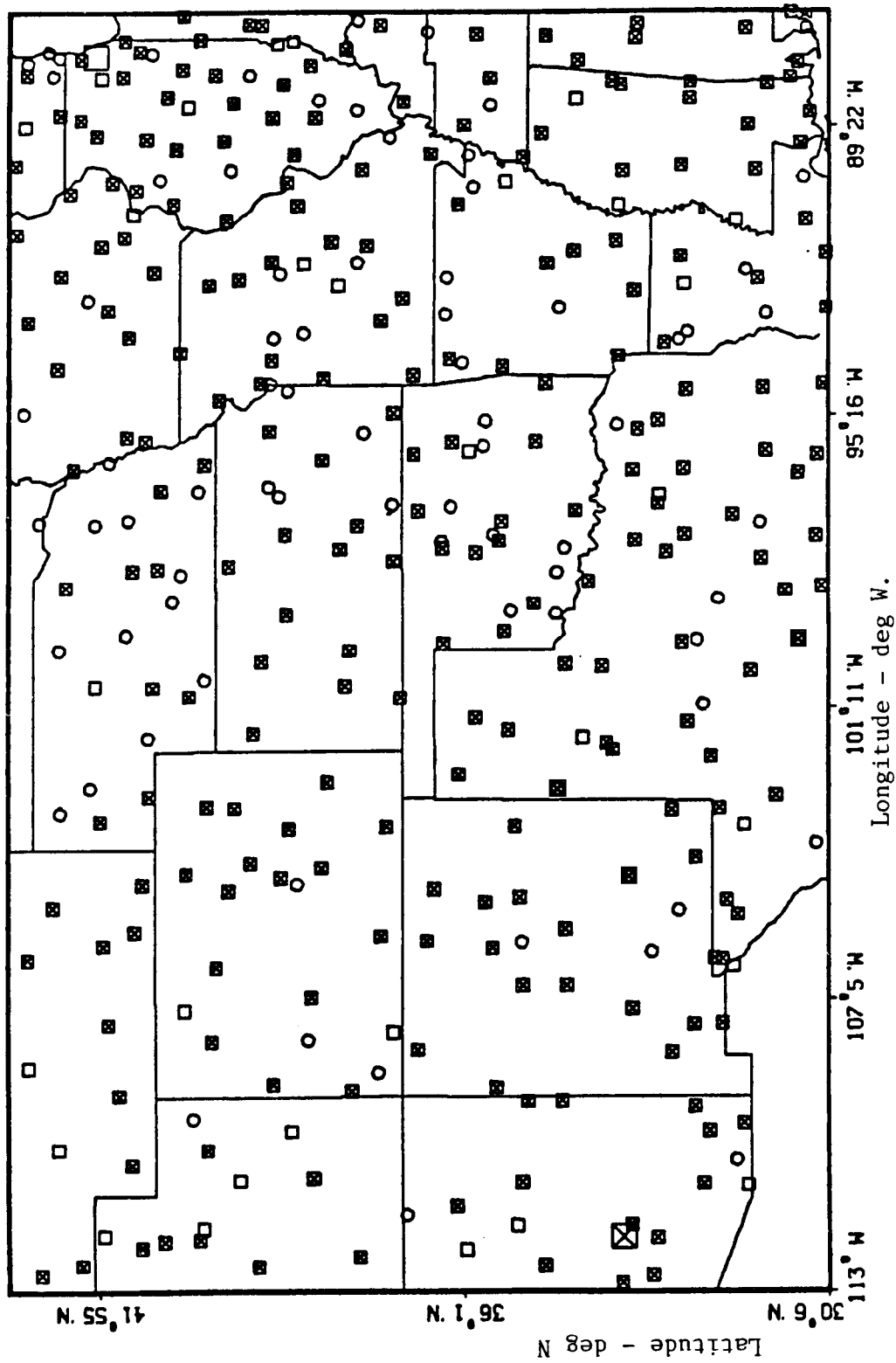


Figure 18. Navaid for Phoenix - Chicago City Pair.

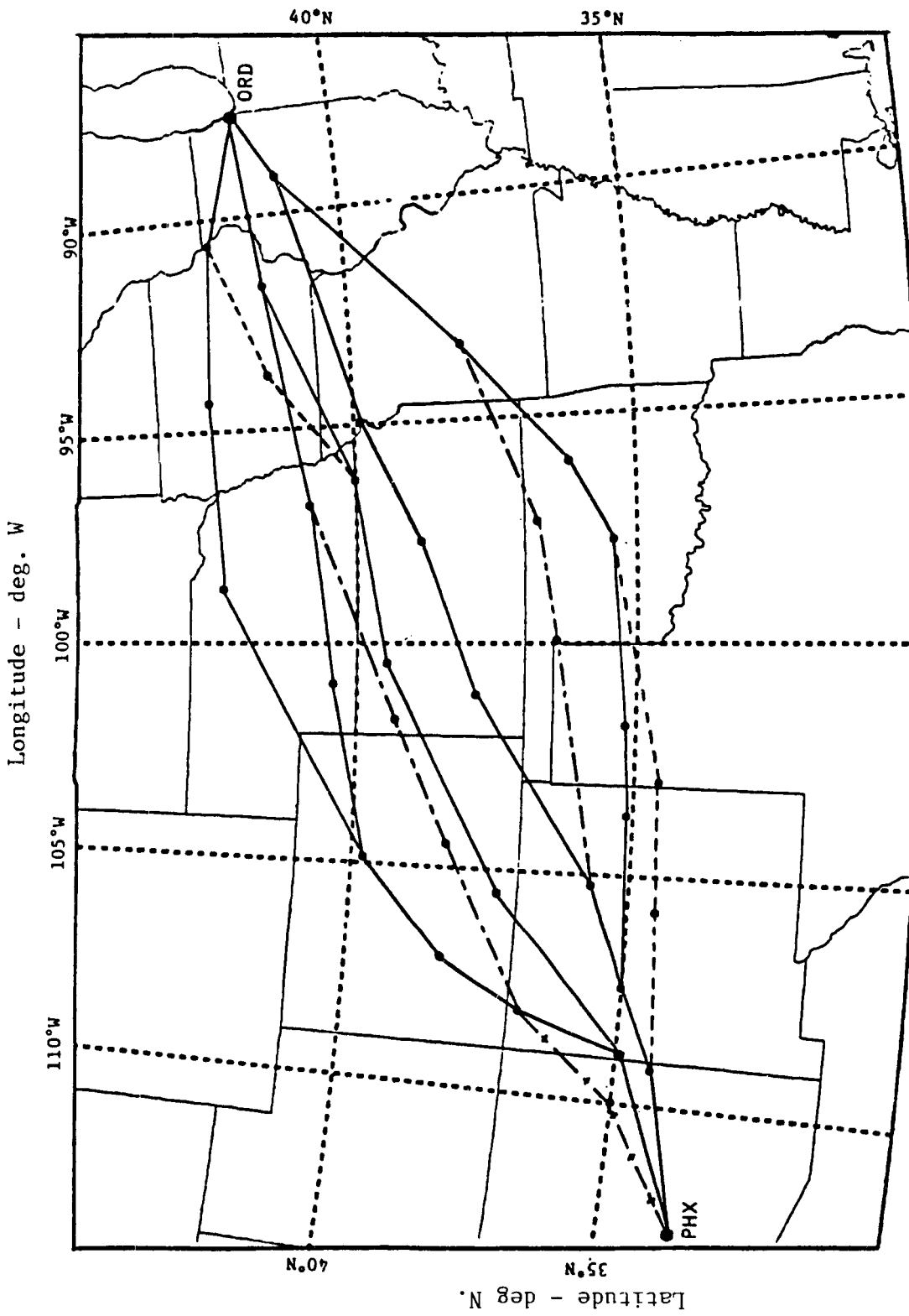


Figure 19. Example Direct Routes Between Chicago and Phoenix.

- a. The fuel and time required for a step climb. These are functions of temperature deviation, weight, altitude, and Mach number.
- b. The fuel and time required for a step descent. These are functions of weight, altitude, and Mach number.
- c. The percentage increase in fuel rate for cruise-climb as a function of weight, altitude, Mach number, and flight path angle.

These tables are also discussed in Appendix E.

Another table is used to compute the reserve fuel requirements for the aircraft. This table is computed based on the assumption that the aircraft would hold at 25000 ft for 45 minutes according to Federal Aviation Regulations with a speed consistent with trimming at the maximum lift coefficient. The reserve fuel table gives holding speed fuel rate as a function of aircraft weight and temperature. This is also presented in Appendix E.

As flight plan research continues, other aircraft performance tables or polynomials could be added. These include:

- a. Takeoff and landing requirements for fuel and time to produce the state change required for the aircraft to go from brake release to clean configuration to begin climb, and to go from initial flap deployment to touchdown.
- b. Fuel and time requirements for short range flight, especially for computing alternate destination fuel requirements.
- c. Tabular representation of descent requirements of fuel, time, and range to top-of-descent. This will have to account for the descent wind profile. It would be used to reduce computation time.
- d. Special performance tables or polynomials that reflect the TSRV aircraft performance for any phase of flight, based on measurements taken during flight test.

Reference Flight Path Computation

The computation sequence followed to generate the reference flight path begins with computing the desired landing conditions and proceeding backwards, as described before. Each of the important computation steps is now discussed.

Landing Weight. EFPLAN first computes the weather profile that exists over the destination airport. This is used later for computing the runway ground wind and the temperature at 25000 ft.

The payload limit is then computed based on range of the flight. This limit is a function of aircraft structural limits, maximum gross takeoff weight, and the maximum fuel capacity of the aircraft. The payload limit is depicted in Fig. E.8 of Appendix E.

Next, EFPLAN adds the input weight of the passengers and the cargo to compute dry payload. The dry payload is checked against the payload limit to make sure that it conforms. The payload is next added to the aircraft empty weight to obtain aircraft zero fuel weight. This sum is tested against the maximum zero fuel weight, again for compliance.

Next, the weights of the fuel required to go to an alternate destination, the reserve fuel, and any additional input tankered (ferried) fuel are added. The alternate destination fuel comes from a routine that computes a short range flight plan from the primary destination to the alternate destination. The reserve fuel is computed based on Federal Aviation Regulations which specify that enough reserve fuel must be on-board to maintain the aircraft in a holding pattern for 45 minutes over the destination runway. We have assumed this to be at 25000 ft. The computation process for obtaining reserve fuel is explained in Appendix E. It requires knowing the aircraft zero fuel weight plus the temperature deviation at 25000 ft.

The final summation of the above weights is the desired landing weight. A final check is made to see that this weight is less than the

maximum landing weight of the aircraft.

Final Approach. This is the phase of flight that begins with initial flap deployment and ends with touchdown. For flight planning purposes, we need to know how much fuel and time are taken in this phase and where will the aircraft be when this phase begins. This is not a significant part of the total cost, so the computations can be done from parametric tables that give the phase initial conditions as functions of airport altitude, ground wind, and aircraft landing weight. There may be some local considerations such as orientation of the runway with respect to the direction of the origin, local noise abatement constraints, local terrain constraints, and Air Traffic Control vectoring allowances.

In EFPLAN, the final approach requirements are simple, at this point. The initial altitude is set 4000 ft above the runway, and the initial indicated airspeed is set to 250 kt. The time required for this phase is set at 330 sec, and the fuel is assumed to be 470 lb. Setting the initial point above the runway adds an extra amount of range to account for ATC vectoring requirements. However, this point can later be set at a more realistic location for actual flight planning so that the top-of-descent point would be more accurately located.

Descent. The first step to compute the descent profiles is to determine the altitudes and speeds with which each profile would finish the cruise phase. The program computes the band of altitudes that act as initial conditions for the cruise phase, the target cruise speeds to end the descent phase and begin the cruise phase (as computed backwards), and then the locations of the pseudo-waypoints that mark the cruise/descent boundaries. Figure 17 illustrates the band of possible cruise altitudes.

Currently in EFPLAN, up to three candidate altitudes may be searched during cruise. They are set based on the input flags ICALT and ICAC, as discussed earlier. Assume for now that the flight plan process uses three candidate cruise altitudes. To compute the target cruise speeds, the program searches over a band of Mach numbers for each altitude to determine the value that minimizes the cruise cost function given by Eq. (1). This

requires an estimate of the aircraft weight at the top-of-descent and the weather conditions at that point. The resulting optimum Mach numbers become the target values to end the descent computation process.

If the cruise-climb option is chosen, the search process also determines the altitudes to end the cruise phase. Equation (1) is minimized over both altitude and speed to define the center altitude. Then for the descent profile computations, the altitudes 1000 ft above and below this "best" altitude are also used. It is important to note that the center altitude, although being the best for a particular location in a wind field, may not be the overall best altitude because of adjacent wind conditions. That is why the cruise search process must work with a band of altitudes. The current 1000 ft value between altitude bands is a parameter that may be varied by later study.

The descent profiles are computed, starting at the beginning of the final approach phase and working backwards to target pseudo-waypoints slightly beyond the potential tops-of-descent. Profiles are generated for each STAR path and to each altitude in the bands at the ends of these paths. The cumulative descent and landing cost to each of these endpoints is then stored for later use as initial conditions for the cruise phase computation. These costs are the direct operating costs, consisting of time and fuel costs, to go from cruise to landing.

As mentioned before, the user has the option of having EFPLAN choose either the best Mach/CAS schedule in computing the descent profile or a reasonably good set of these numbers that have been precomputed. Using the precomputed numbers has the obvious advantage of saving running time. Figure 20 shows a plot of these precomputed numbers that vary with the cost of time over cost of fuel ratio (C_t/C_f).

It is conceivable that the preferred descent profile that results from using the precomputed Mach/CAS schedule numbers could be different than the one that would be picked if the descent schedule were optimized. This could result in EFPLAN selecting a less than optimal STAR path and associated cruise path to end the flight plan. Thus, the option to optimize

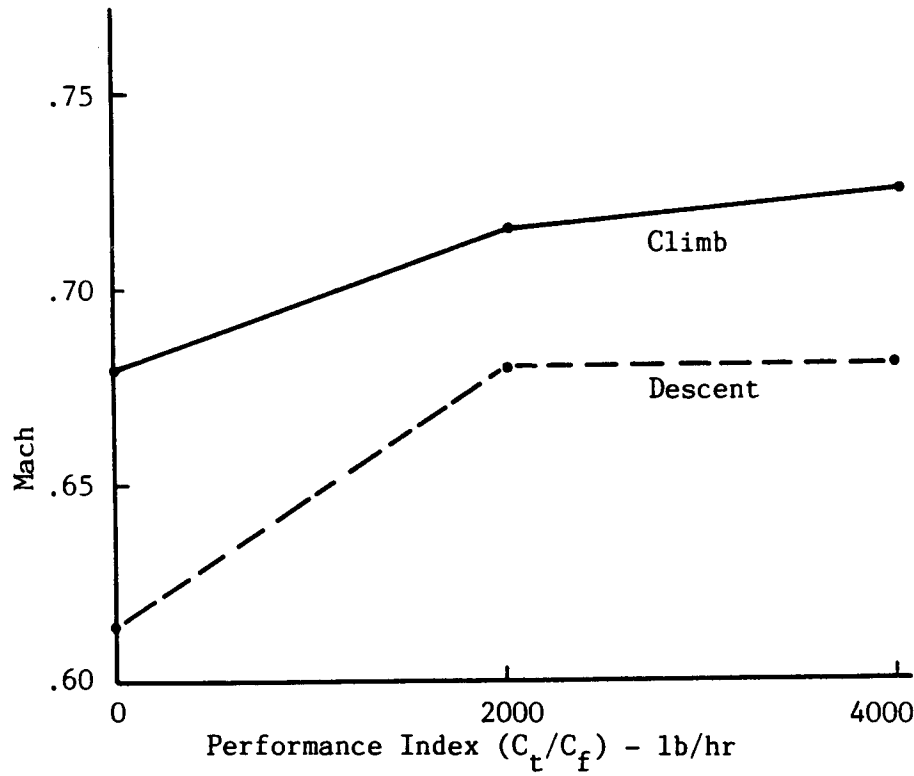
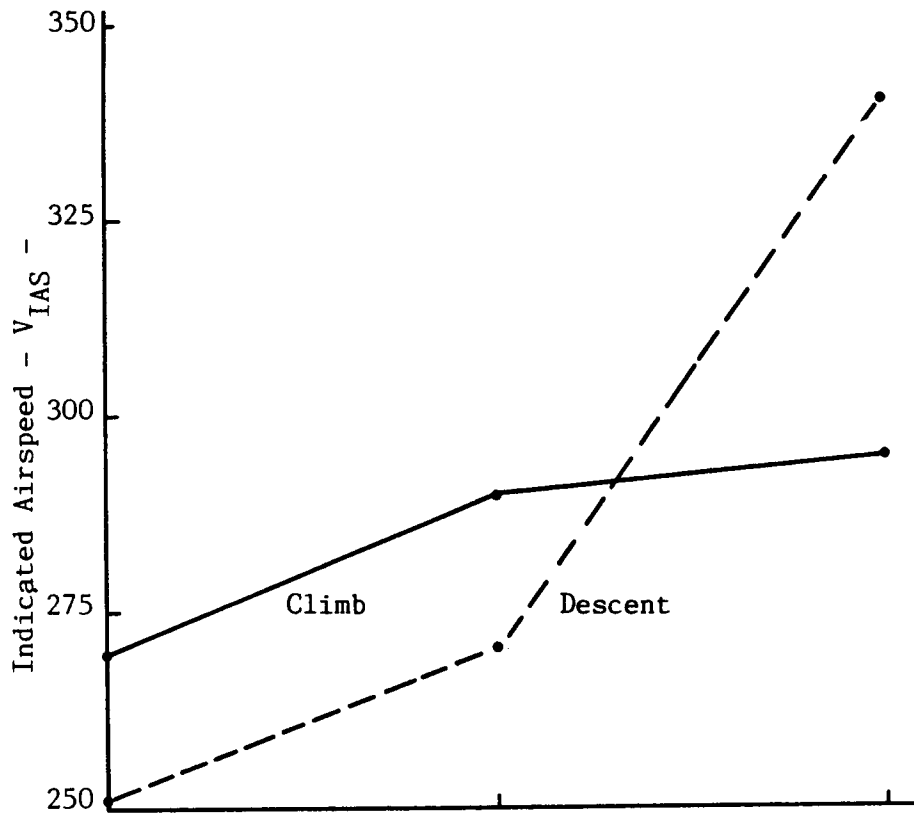


Figure 20. Values of V_{IAS} and Mach for Climb and Descent as Functions of Performance Index

Also required to compute segment cost is the fuel rate that corresponds to a given Mach number. Rather than computing the trim conditions that balance lift, drag, thrust, and weight of the aircraft, as part of the segment optimization, these computations were done beforehand for a range of aircraft weights, temperature deviations, altitudes, and Mach numbers. These values were placed in a fuel rate table that is described in Appendix E. Figure E-2 shows the envelope of Mach numbers for the table, and Fig. E-3 is a sample plot of fuel rate as a function of Mach number, altitude, weight, and temperature deviation. The table is interpolated for fuel rate given the trial Mach number to cross the segment. This greatly speeds the optimization process.

In computing the cost to cross a segment, the segment range is split into three sub-segments; the first one-fourth, the middle one-half, and the final one-fourth of the range. Then the cost across each sub-segment follows the sequence:

1. Assume a constant Mach number to travel the sub-segment.
2. Find the corresponding ground speed based on the wind of the sub-segment.
3. Compute crossing time by dividing range by ground speed.
4. Estimate the aircraft weight at the sub-segment center.
5. Use the center weight estimate, Mach number, temperature deviation, and altitude to look up the fuel rate.
6. Multiply fuel rate by crossing time to get fuel burned.

These six steps are repeated for each of the three sub-segments to obtain total fuel burned, time to cross, and cost of the segment as functions of the crossing Mach number. Wind vectors are from the beginning, middle, or end of the segment.

We tried several schemes to optimize the Mach number across the segment. Both linear interpolation on Mach, and quadratic fit of fuel rate vs Mach from the fuel rate tables had shortcomings. (This was an old lesson; when using numerical optimization it is best not to use tables.) We observed that there is little cost change in rounding cruise Mach number

the descent path to each target boundary point is retained. However, we have not seen any case of different paths chosen in the runs we have made up to this point. Also, the differences in cost would probably be quite small.

The descent portion of the flight plan uses a very small part of the overall fuel that is used during the flight. What is important is to account for the time required to descend and the range that locates the top-of-descent. The top-of-descent point will largely be a function of aircraft weight and the prevailing winds present during this phase. A future possibility for cutting compute time is to precompute tables or equivalent polynomials that give descent fuel, time, and range as functions of final approach weight and terminal wind profile.

Cruise. Just as we needed target points to end the descent profile computations, we also need end points for the cruise computations. From simulating climb several times for the TSRV aircraft, it was found that the range required to reach 35000 ft with a heavily loaded aircraft when the objective was to minimize fuel required almost 160 nmi of range. This dropped to 140 nmi when cost of time was \$600/hr. It dropped to 120 nmi when the upper altitude was 33000 ft, and it dropped to 105 nmi when the upper altitude was 31000 ft. Thus, EFPLAN sets this outer climb range RCLMB as a function of the cost of time over cost of fuel ratio (C_t/C_f) and upper cruise altitude. As mentioned before, RCLMB is used to define the boundary between the climb profiles (in the form of extended SID paths) and the beginning of the cruise paths. These climb boundary pseudo-waypoints include up to three different altitudes.

The fundamental computation in cruise is determining the speed to travel a segment between two node points. Again, this requires choosing the airspeed, or Mach number, to minimize the cost function given by Eq. (1). When true airspeed is combined with the local wind vector, the result is the local ground speed. Segment distance divided by ground speed yields the time to cross the segment.

to the nearest 0.01. This corresponds to the granularity of the fuel rate table, so it was not necessary to interpolate on values of Mach number between table values.

The conclusion was that EFPLAN should use three trial constant Mach numbers to compute cost across a segment. The three are 0.01 apart. If the middle value produces the minimum cost for the segment, this is the chosen Mach number. If one of the outer Mach numbers produces a lower cost, the three numbers are shifted in that direction until the middle value produces the lowest segment cost. This is a simple scheme which is computationally fast and produces near minimum segment costs.

Depending on the user option, the segment costs might be further modified to account for step climb/descent or cruise climb/descent. These two possibilities are illustrated in Fig. 21. EFPLAN works with up to three altitudes across a segment. It first computes the costs to cross these segments in level flight. The notation used is that the computation goes from the k th node to the $(k-1)$ st node.

As seen in Fig. 21.b, the step is assumed to occur right before the k th node. For the middle altitude $h(2)$, the question is: Is it cheaper to reach $h(2)$ at the $(k-1)$ st node by starting from $h(1)$, $h(2)$, or $h(3)$? These choices would require a step descent, level flight, or step climb, respectively. The solution is based on knowing the previous costs required to reach the three altitude levels at the k th node. Similarly, the aircraft can reach $h(1)$ at $(k-1)$ from $h(1)$ or $h(2)$ at k , and it can reach $h(3)$ at $(k-1)$ from $h(2)$ or $h(3)$ at k . To make these decisions, EFPLAN determines how the respective segment costs are affected by step climb and descent.

For computing step costs, we have developed two more tables, also explained in Appendix E. The cost of a step is measured in fuel and time. For step climb, maximum climb thrust was assumed, and the tables produce incremental fuel burned and time taken during the step as functions of temperature deviation, aircraft weight, pressure altitude, and Mach number at the beginning of the step. For step descent, fuel burned and time are

functions of aircraft weight, pressure altitude, and Mach number. Idle thrust was assumed in computing the step descent tabular values.

In computing the step effects, fuel and time increments ΔW_{sp} and Δt_{sp} are first interpolated from the tables. The ground speed is next computed for the mid-altitude of the step. The ground speed is multiplied by incremental time to compute the incremental range R_{stp} . This increment is divided by the total segment range R_{seg} to obtain a percentage factor PC. Then the segment time, fuel, and cost are computed as

$$\Delta t = (1-PC) \Delta t_{sg} + \Delta t_{sp} ; \quad (2)$$

$$\Delta W = (1-PC) \Delta W_{sg} + \Delta W_{sp} ; \quad (3)$$

$$\text{Cost} = C_f \Delta W + C_t \Delta t. \quad (4)$$

Here, Δt_{sg} and ΔW_{sg} are the original time and fuel increments for flying level across the segment, and C_f and C_t are cost of fuel and time.

In computing cruise climb/descent effects, we assume that the altitude changes occur very slowly, so the time to cross a given segment is not affected for a given Mach number. However, the fuel consumed will vary because of the small flight path angle effect. To compute fuel burn change due to a small flight path angle, we developed another table of fuel burn scale factors as functions of aircraft weight, altitude, Mach number, and flight path angle. This is also discussed in Appendix E. These factors are used to scale the fuel consumed by going level across the segment. It is assumed that the aircraft can descend from the upper altitude $h(3)$, it can climb or descend from $h(2)$, and it can climb from $h(1)$. This is illustrated in Fig. 21.c.

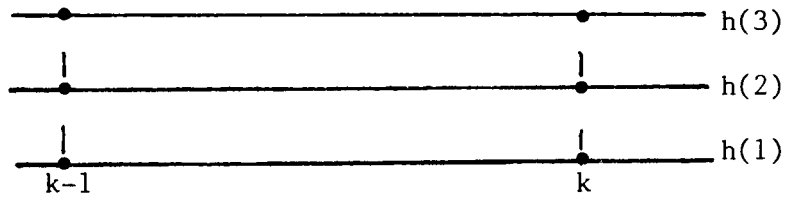
Fuel burn scale factors were computed for flight path angles as high as plus or minus 0.4 deg. This is equivalent to a 1000 ft altitude change in 25 nmi of range; this seems adequate at this time.

For the direct search option, the horizontal paths are pre-defined as a sequence of segments, and EFPLAN computes the altitude and airspeed over each of these. The objective for this case is to find the optimum altitude and airspeed for each segment. The dynamic programming concept is used to compute the profile backwards to each altitude node at each waypoint. Each waypoint is a navaid location, or if the navaids are more than 100 nmi apart, pseudo-waypoints are created on the arc connecting the navaids so that no resulting segment is more than 100 nmi in length.

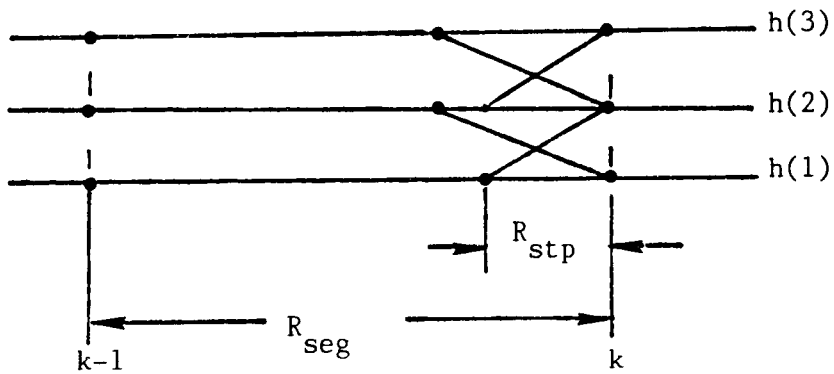
As the dynamic programming process computes the profile, it inserts step climbs or descents wherever appropriate. Each path is computed back to the pseudo-waypoint that marks the boundary with the climb profiles. Three costs associated with three altitudes over each climb pseudo-waypoint are found. Each cost corresponds to the downstream cruise-descent path leading from that waypoint.

Figure 19 is an example of eleven direct paths connecting Chicago and Phoenix. The climb pseudo-waypoints are at Dubuque, Iowa City, and Bradford. Four horizontal paths lead back from Phoenix to Bradford, for example. The computation process determines the best three altitude/speed profiles for each of the four paths. It then retains only the better three overall paths to the pseudo-waypoint's three altitudes. The climb profile computations will later determine which of these three is the overall best profile going through Bradford.

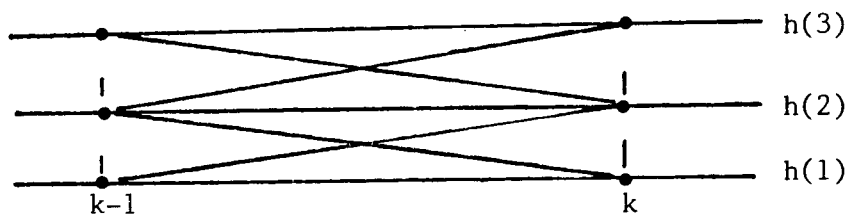
The even grid option is an expansion of the direct route option to search for the best lateral path. In this case, there is no pre-defined horizontal path. Instead, a horizontal grid is defined in reference to the great circle arc connecting the origin and destination city pair. Figures 22 and 23 are examples of possible grid structures. In Fig. 22 there are nine lateral nodes for every longitudinal stage of the network. Each stage is about 100 nmi apart. The distance to the outermost node is an exponential expansion which is a function of the longitudinal range along the central arc. In Fig. 23, there is a varying number of nodes as the stage number increases away from the origin or destination. Here, the intent is



a). No Altitude Change.



b). Step Climb and Step Descent.



c). Cruise-Climb and Cruise-Descent.

Figure 21. Illustration of Step Climb/Descent and Cruise-Climb/Descent Altitude Change.

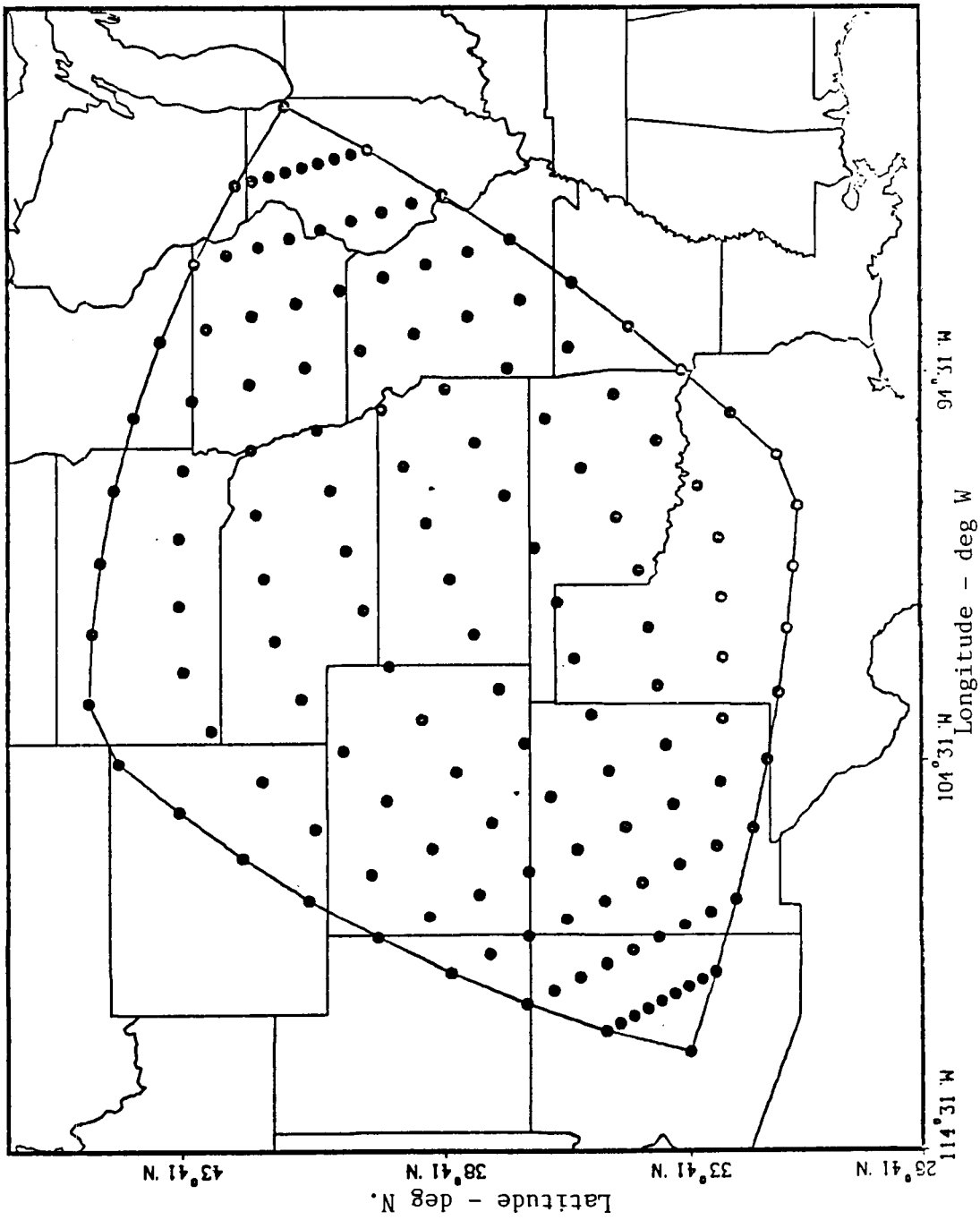


Figure 22. Even Grid with Equal Number of Nodes per Stage.

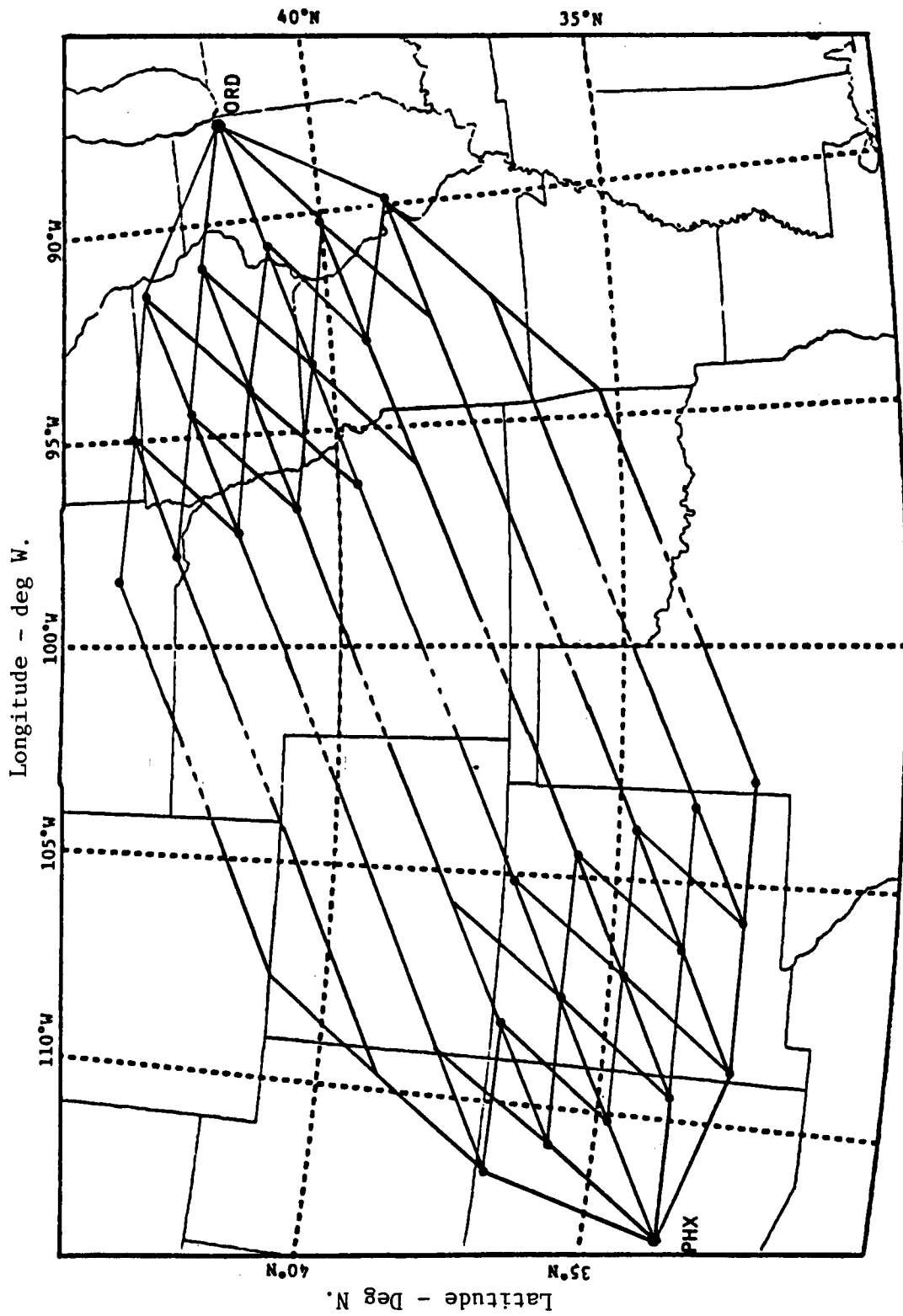


Figure 23. Even Grid with Unequal Number of Nodes per Stage.

to keep the lateral distances between nodes at any stage relatively constant.

The best network structure to use to define the horizontal path is a subject of continuing research. Currently, EFPLAN uses a structure similar to the one shown in Fig. 22, and up to two iterations are made about the previous best path to refine the selected cruise path.

After the three-dimensional network is defined, backwards dynamic programming is again used to compute the best sequence of down-stream nodes and connecting segments to reach each 3D node at each stage. This process again continues backwards until the climb pseudo-waypoints are reached.

One modification that has been added to the search through both the direct routes and the even grid is the ability to define restricted areas. These might be charted areas that restrict flight through them or severe weather areas that are temporarily blocked. The cruise search logic determines what segments penetrate these areas and then moves around them.

Climb. Before EFPLAN computes the climb profiles, it needs to know where the takeoff phase ends. Currently, this is assumed to be 2000 ft directly over the airport with indicated airspeed set at 250 kt. Like for the final approach phase, this adds extra distance to the overall range that can account for some ATC vectoring requirements. However, for an actual flight plan, the setting of the initial climb point should account for local departure requirements.

The next requirement is to estimate the aircraft weight at the beginning of the climb phase. This is done by use of an empirical equation that accounts for altitude difference between initial altitude, upper top-of-climb altitude, and minimum aircraft weight at the end of the cruise computations. We are interested in computing a close approximation of the time and fuel required to climb to each of the pseudo-waypoints, so this estimate can be off by 500 lb and still be adequate.

Table 3. Example of a Tabular Flight Plan.

FLIGHT PLAN FOR CHICAGO-PHOENIX

ORIGIN : LAT= N41.98 LNG= W 87.98 ALT= 667. FT
 DESTINATION: LAT= N33.43 LNG= W111.89 ALT= 1132. FT

TAKEOFF TIME 13:03:52 CST
 DURATION 3:26:07
 LANDING TIME 15:30 MST
 OPTIMUM 13:05:35 CST
 DIRECT 3:24:24
 15:30 MST

TAKEOFF WEIGHT 94850.
 FUEL 14901.
 FUEL 15350.
 COST 3265.79
 COST 3324.47

FL ST ND	LAT	LNG	FLVL	TRKANG	HG	MACH	VTA	VC	RANGE	TIME	WGHT	FUELUZ	VWV	VME	TEMP
1	1	42.8	-87.9	7.	-102.1	-101.6	0.401	446.	448.	0.0	0.0	0.0	0.0	-4.	4.2
3	2	41.6	-91.2	350.	-109.5	-102.8	0.714	417.	418.	150.0	0.32:40	3551.	-47.	13.	6.2
3	3	41.0	-93.3	350.	-105.4	-98.3	0.714	417.	398.	252.4	0.47:41	4637.	-45.	29.	6.5
3	4	40.6	-95.4	350.	-107.5	-101.3	0.717	420.	394.	351.8	1:02:48	89139.	-36.	36.	6.7
3	5	40.0	-97.5	350.	-109.4	-105.2	0.717	420.	399.	451.1	1:17:52	88074.	-22.	29.	6.9
3	6	39.5	-99.5	350.	-111.2	-106.9	0.717	420.	399.	550.4	1:32:49	87027.	-22.	30.	7.1
3	7	38.9	-101.5	350.	-118.0	-113.4	0.717	420.	401.	649.6	1:47:47	85990.	-21.	31.	7.3
3	8	38.1	-103.3	350.	-119.7	-114.9	0.717	420.	402.	749.4	2:02:43	84964.	-22.	32.	7.7
3	9	37.2	-105.2	350.	-121.4	-116.2	0.717	421.	404.	849.3	2:17:36	83950.	-24.	33.	8.1
3	10	36.3	-106.9	350.	-123.1	-117.3	0.717	421.	405.	949.2	2:32:28	82946.	-28.	35.	8.5
3	11	35.4	-108.6	350.	-116.5	-110.6	0.717	422.	402.	1049.3	2:47:15	81955.	-31.	35.	8.9
3	12	34.6	-110.5	350.	-105.9	-100.1	0.717	422.	392.	1152.9	3:02:44	80925.	-34.	38.	9.3
1	13	33.4	-111.9	11.	-105.9	-106.9	0.412	461.	473.	1259.4	3:26:07	79949.	5.	-14.	12.0

As for descent, the climb profiles are computed by using either precomputed good values of Mach/CAS numbers or values optimized for the particular cost coefficients, aircraft weight, and climb weather conditions. Profiles are computed for each SID path and to each ending cruise altitude.

The cost of each climb profile is computed and added to the corresponding cruise-descent cost previously computed for each pseudo-waypoint. From these overall cost sums, the best overall path is selected. The small discrepancy that exists between the aircraft weight at the end of the climb profile with the weight at the end of the cruise-descent profile has little effect on the overall solution.

Takeoff. It remains to compute the last increment of fuel and time to account for the takeoff phase of flight. This is from gate departure through brake release, takeoff, and climb through flap retraction. Currently, EFPLAN allows 500 lb of fuel and 600 sec of time for this phase.

A final test is made on the resulting initial weight of the aircraft to see if it exceeds the maximum takeoff weight. A further refinement could be to test this weight against local weather conditions and the length and slope of the takeoff runway.

EFPLAN computes the takeoff time with respect to the desired landing time. It is computed in local standard time.

EFPLAN Output

The primary flight plan output is a table such as the one shown in Table 3. The table gives the sequence of reference flight path waypoint locations in terms of their latitude, longitude, and altitude. Also given are the correct Mach number to hold on each segment and the cumulative time and fuel burned. The waypoint wind components provide added information. At the top of the table the takeoff and landing times are presented along with the required takeoff weight. The cost of the flight plan profile is

compared with that of the shortest distance route (nominally along the great circle path connecting origin and destination).

Additional output is the flight plan horizontal path in graphical form such as shown in Figs. 24 and 25. Figure 24a gives an example of the overall even-grid structure that was searched to find the horizontal path.. Note that this contains the SID and STAR routes shown in Fig. 16. Figure 24b is an example of the selected even-grid route from Chicago to Phoenix, modified to include the wind vectors expected on the route. Figure 25 is the selected direct route from Chicago to Phoenix. It is "Route 6" as shown in Fig. 19.

The other type of EFPLAN output is major portions of the flight plan used for input to OPTIM-7S. This simulates the direct transfer of the flight plan from the ground-based flight planning computer to the airborne flight management computer. The output includes the sequence of range and bearing segments from the beginning of the climb phase to the end of the descent phase, the aircraft weight at climb initiation, the terminal weather models, and the enroute weather models. The terminal weather models consist of altitude dependent wind and temperature profiles that are average conditions for the climb and descent regions. The enroute models consist of an altitude dependent wind and temperature profile for each enroute waypoint. The wind is converted to magnitude and heading of the origin which is consistent with aviation weather reporting.

Each of these three types of output can be eventually linked directly to the airborne FMS computer. The graphical output can be used as part of the cockpit HSI display to show flight progress. Such a display would be more convenient during flight than use of paper hard copy.

Part of the flight plan output could also be used for direct flight plan filing with the ATC system. This saves both dispatcher and flight crew time at the critical flight initiation point.

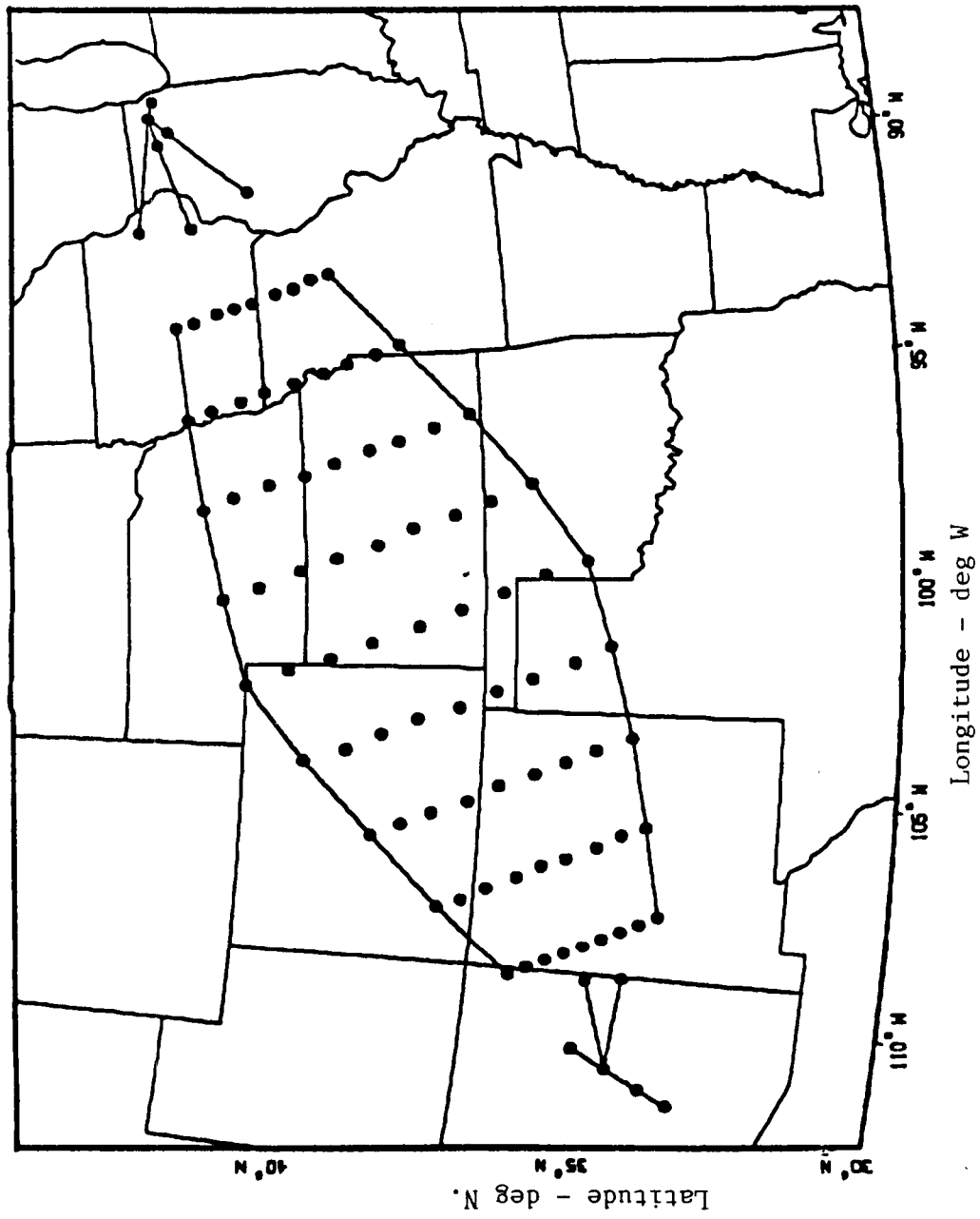


Figure 24a. Horizontal Grid Structure for ORD-PHX Route.

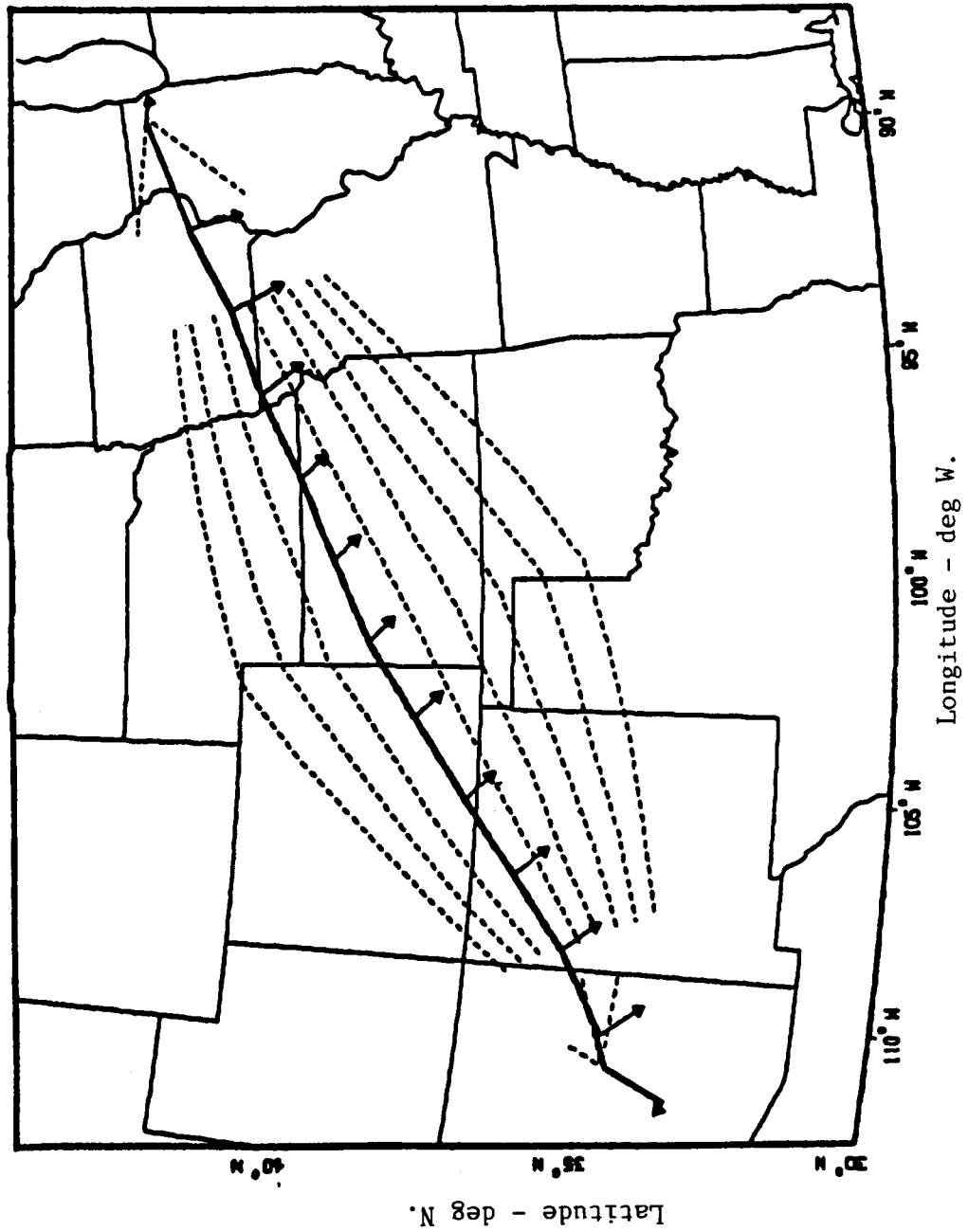


Figure 24b. Selected Horizontal Path from Flight Plan for ORD-PHX.
Enroute Wind is Indicated.

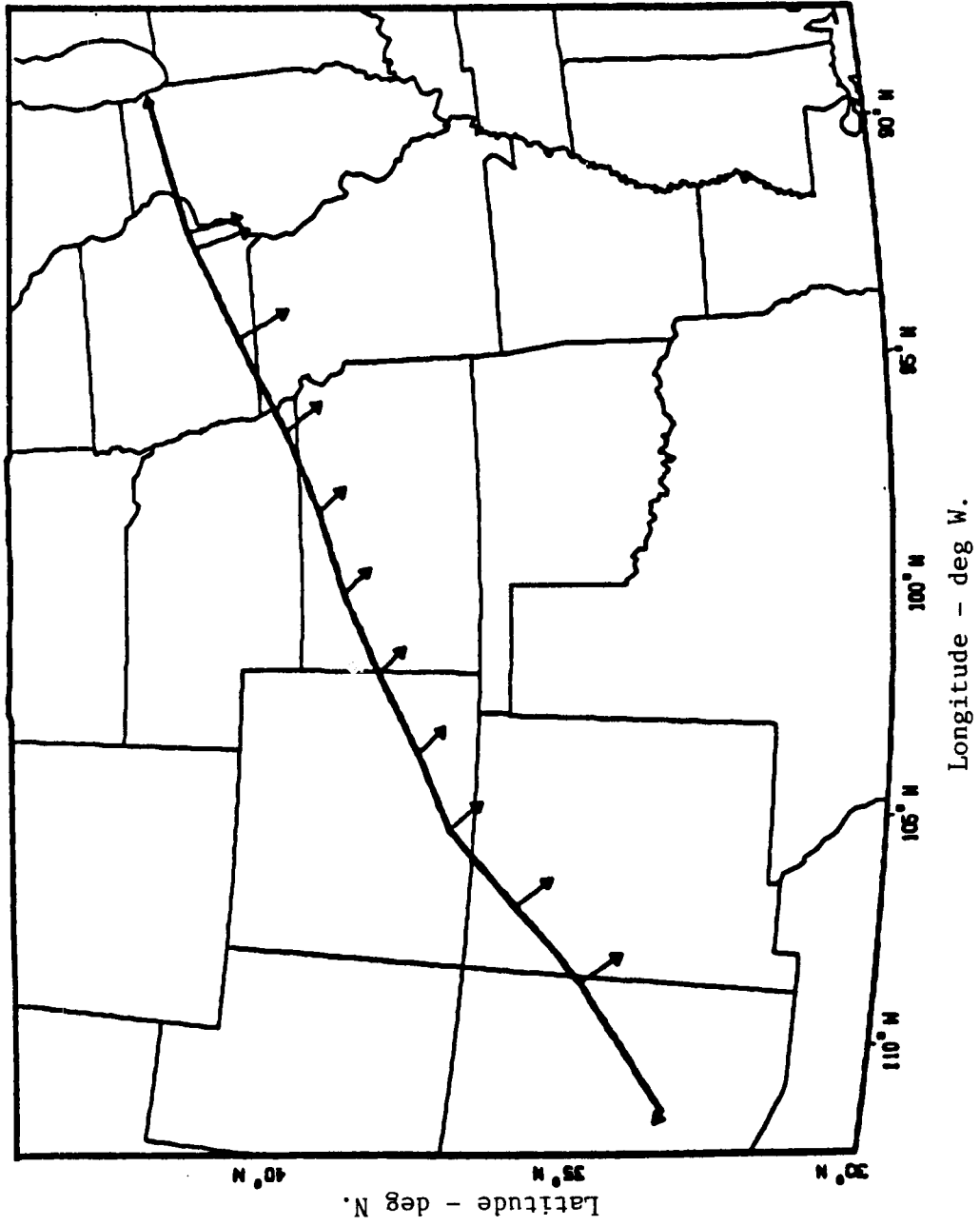


Figure 25. Optimum Direct Route 6 for ORD-PHX.

Example Results

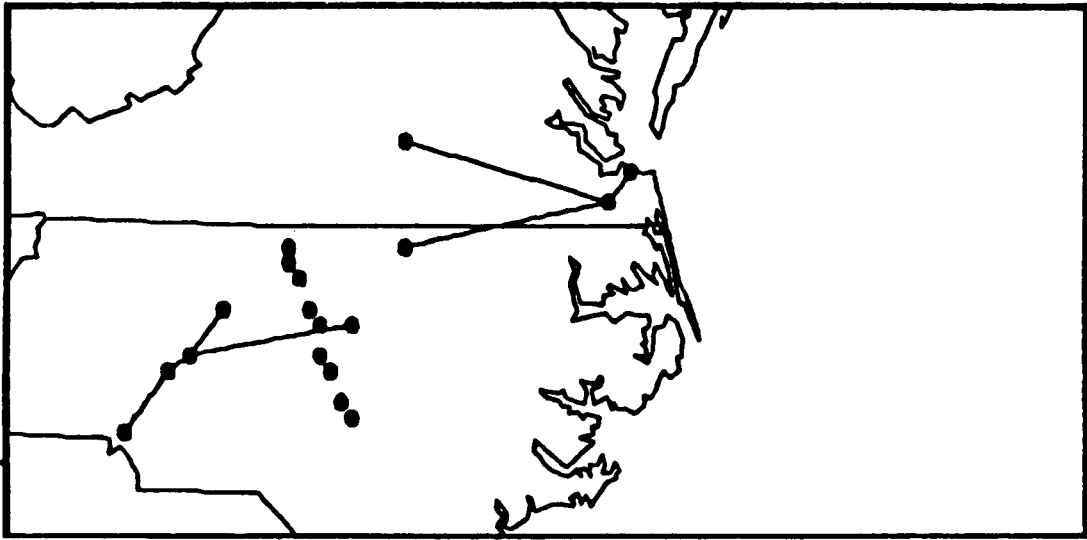
Figures 26-30a and b are plots of the route structure and routes chosen for example flights around the "Norfolk loop", as described earlier. The first (a) part of each figure shows the location of the even-grid node points plus the SIDs and STARs defined for each city pair. In some cases the SIDs and STARs are those published by the FAA. In other cases these departure and arrival paths were made up to test different features of the EFPLAN climb and descent logic. The second (b) part of each figure shows the selected horizontal path along with wind vectors at each route waypoint.

Figure 26 shows the Norfolk-Charlotte leg of the loop. This flight was interesting in that it was short range so that the upper cruise altitude had to be limited to 31000 ft, and the enroute waypoints were positioned with segment lengths of 50 nmi rather than the normal 100 nmi. These changes were made so that the short range flight plan could be derived from EFPLAN as it now is written. A future enhancement to the program will include special handling for short range (less than 300 nmi) flights.

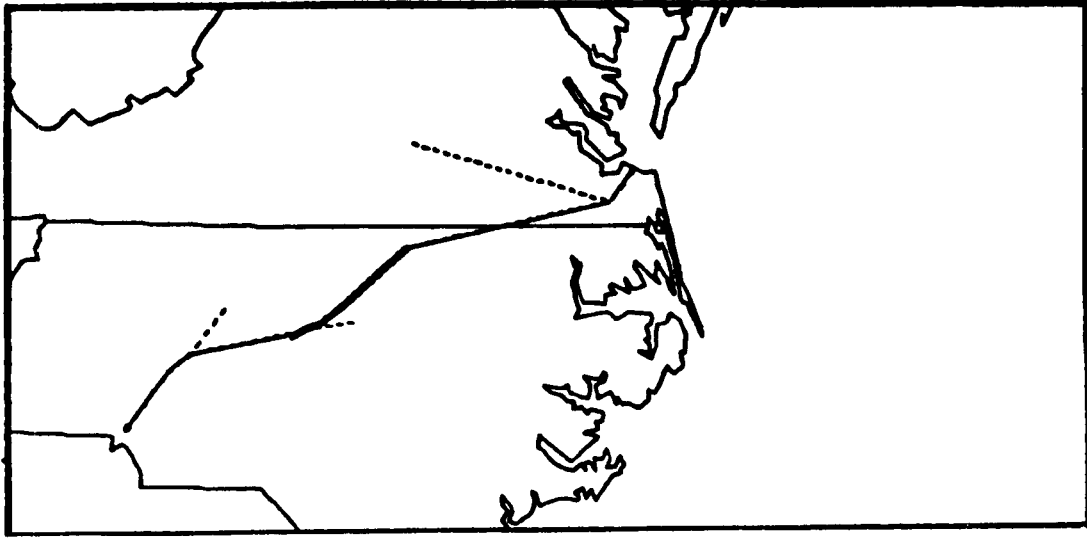
Figure 27 shows the Charlotte-Chicago leg of the flight. Of interest here is that there is only one long SID leaving Charlotte, and all horizontal cruise paths must fan out from its single endpoint.

Figures 28 and 29 show the Chicago-Dallas and Dallas-Atlanta legs. Both of these routes are fairly long with four enroute segments each. Figure 30 is the final Atlanta-Norfolk leg which is long enough for two enroute segments.

Each of these figures suggest that a future enhancement to the flight planning - flight management process might be the mapping of the flight plan horizontal path on a cockpit display. The maps could be stored in airborne memory, and the route waypoints and relevant weather could be superimposed after first being computed on the ground.

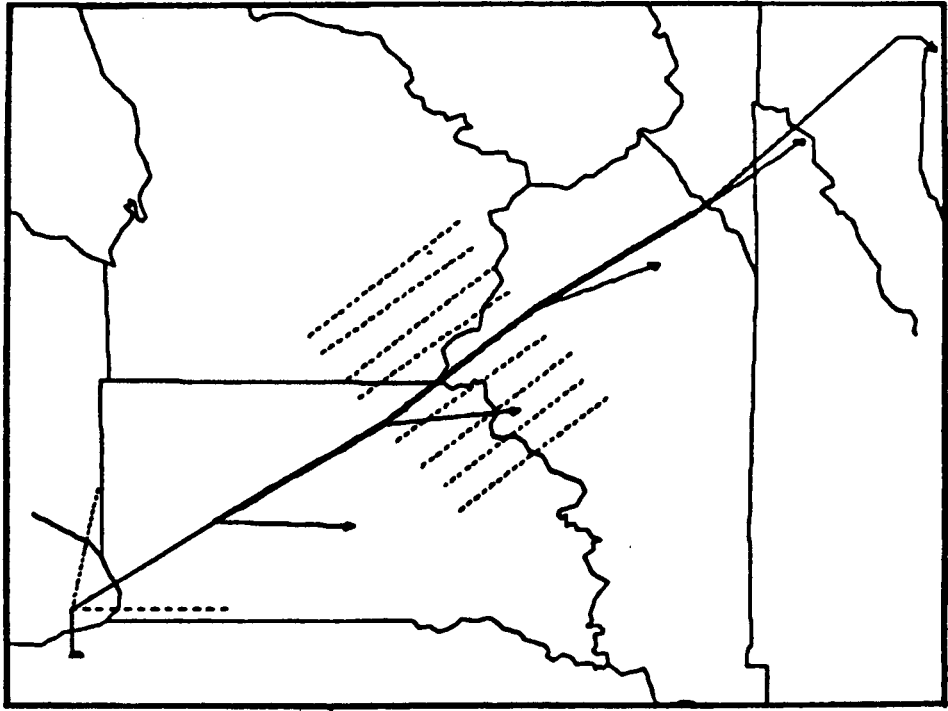


a. Network

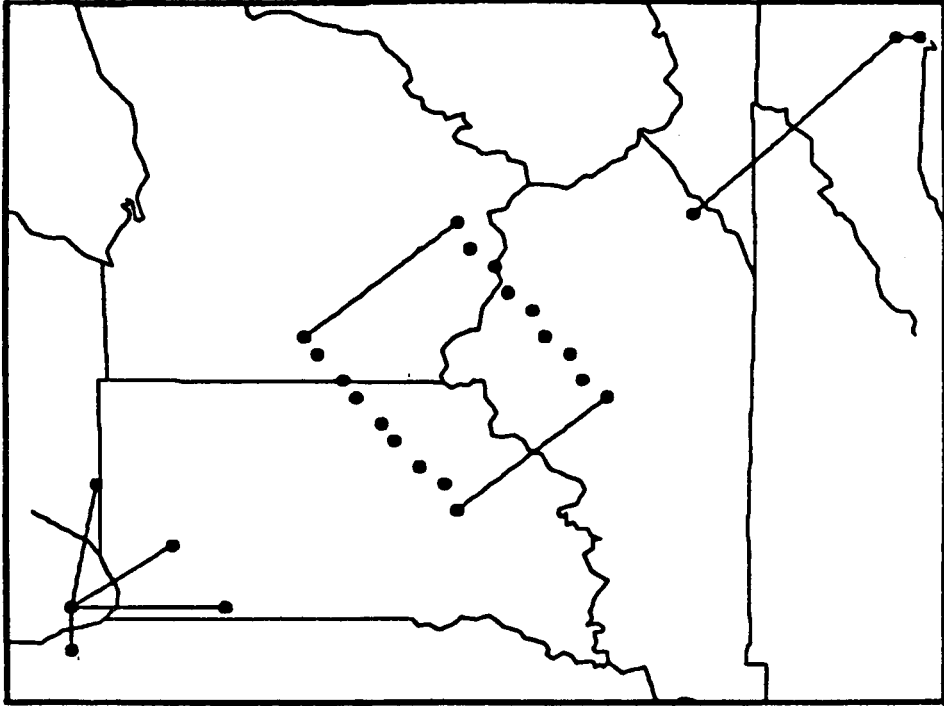


b. Selected Route.

Figure 26. Norfolk - Chicago Leg of Norfolk Loop.

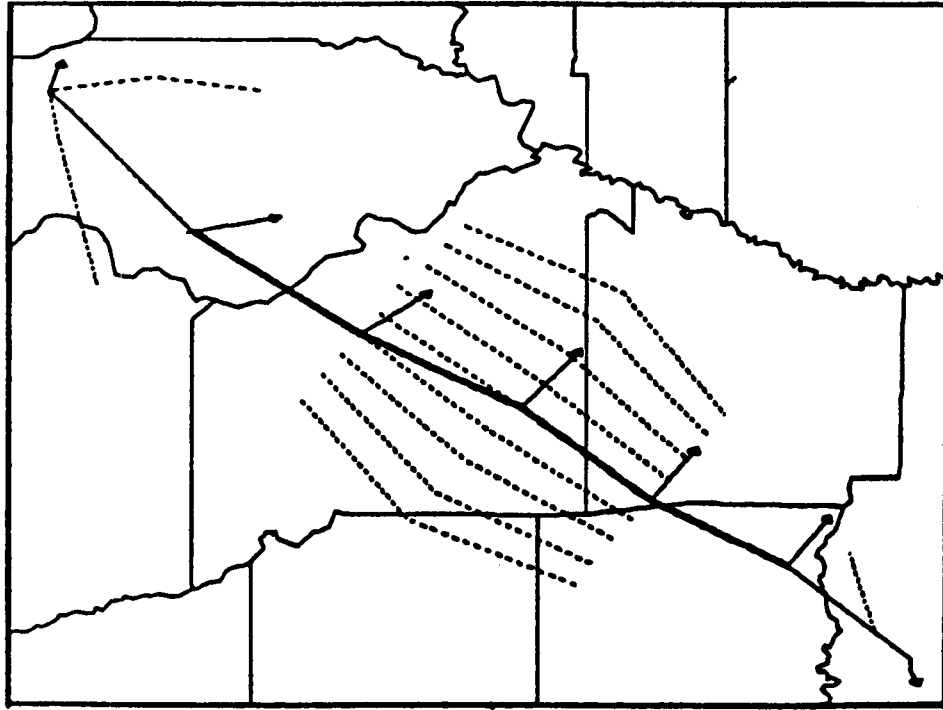


b. Selected Route.

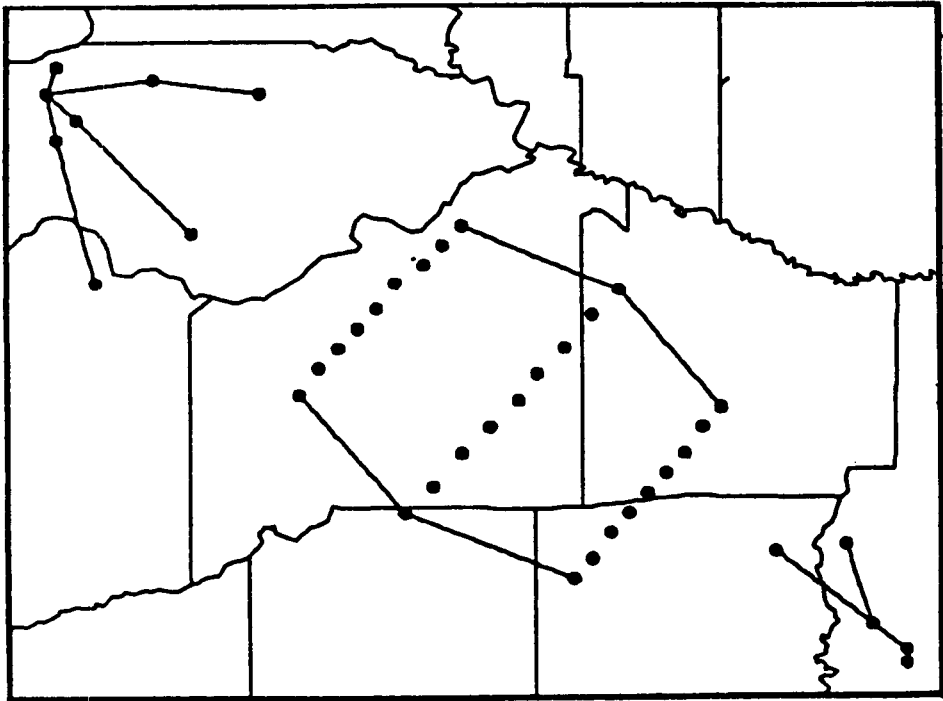


a. Network.

Figure 27. Charlotte - Chicago Leg of Norfolk Loop.

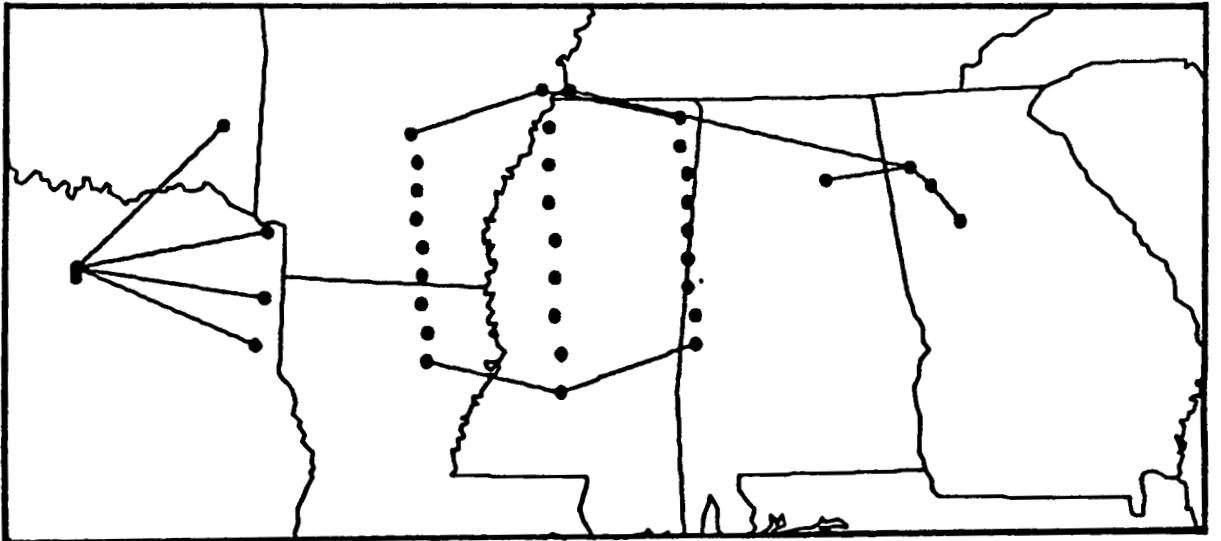


b. Selected Route

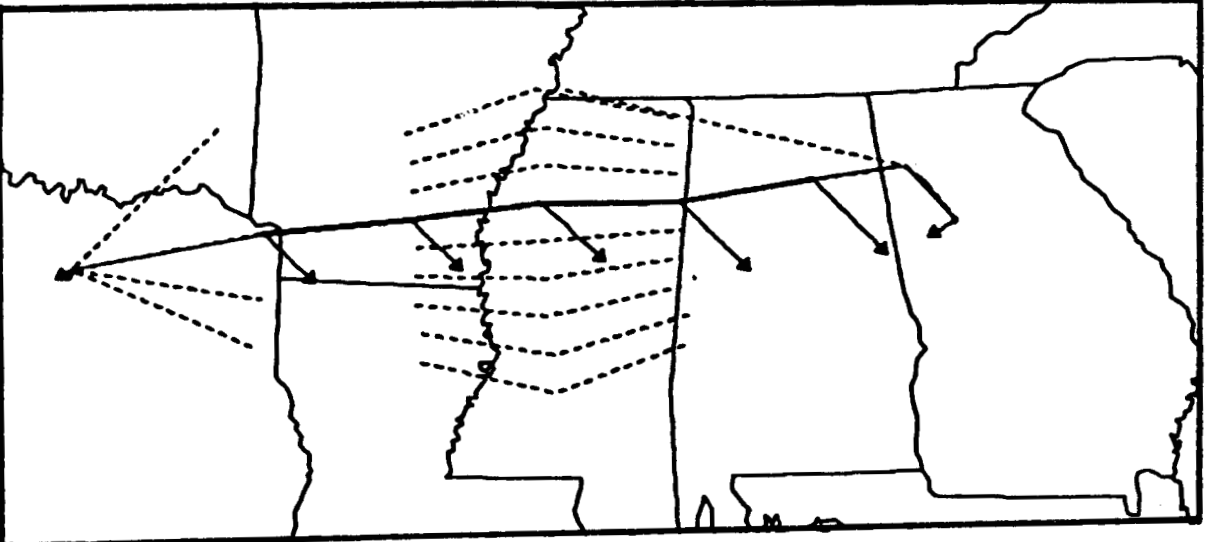


a. Network

Figure 28. Chicago - Dallas Leg of Norfolk Loop.

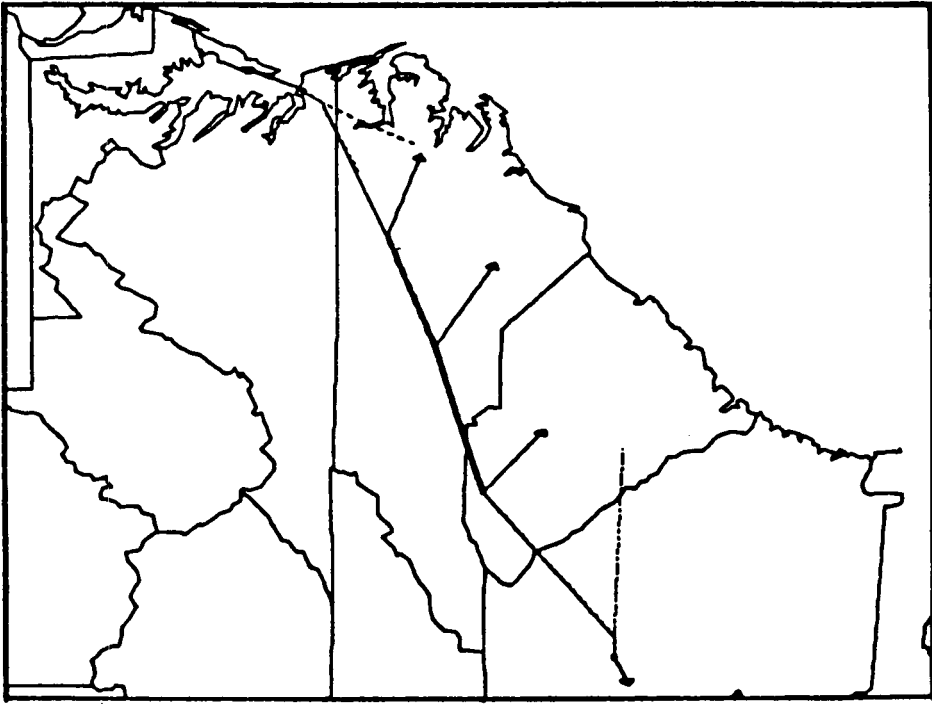


a. Network

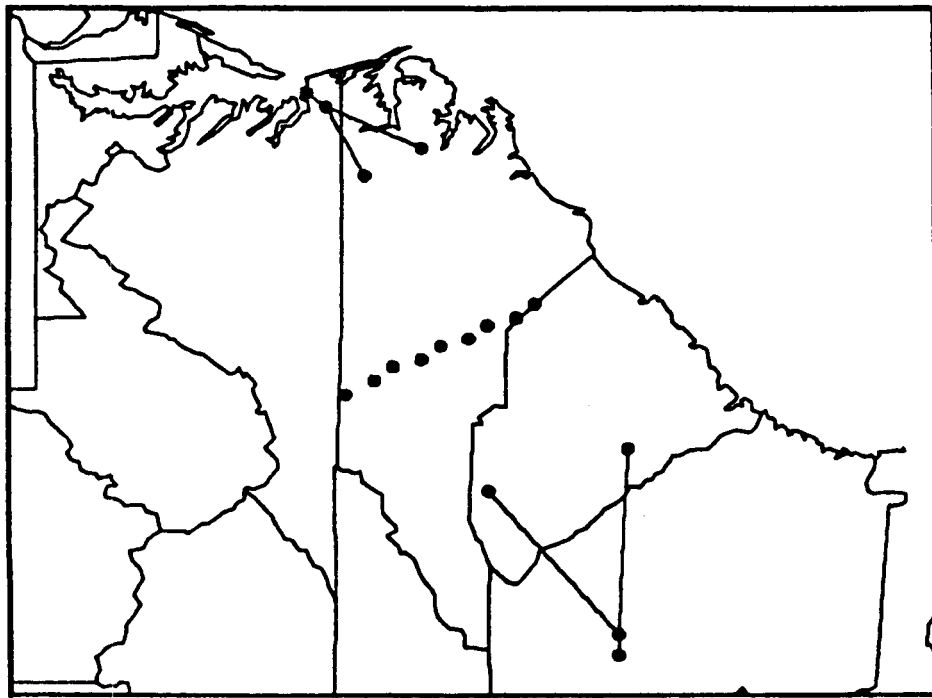


b. Selected Route

Figure 29. Dallas - Atlanta Leg of Norfolk Loop.



b. Selected Route



a. Network

Figure 30. Atlanta - Norfolk Leg of Norfolk Loop.

It is interesting to compare the relative cost of flight plans generated using the even grid and direct search methods. For an example comparison, we again used the ORD-PHX example. The direct search method uses eleven different routes between the city pair as shown in Fig. 24. The even grid method uses a network with nine lateral nodes as shown in Fig. 22. Cases where cruise altitude was fixed at flight level (FL) 350 are compared to where altitude was allowed to step between FL 280, 310, and 350. Also, cases where the Mach/CAS schedules were fixed or optimized are compared. The results are shown in Table 4.

Table 4. Comparison of Flight Plan Costs and Run Times.

Case	Mach/CAS Schedule	Search Method	Cruise Altitude	Flight * Cost (\$)	Run ** Time (sec)
1	Fixed	Even Grid	Fixed	3265.76	1.70
2	Fixed	Even Grid	Step	3265.76	2.70
3	Fixed	Direct Search	Fixed	3283.82	1.48
4	Fixed	Direct Search	Step	3283.82	1.94
5	Optimum	Even Grid	Fixed	3264.38	3.84
6	Optimum	Direct Search	Fixed	3278.30	5.06

* Based on \$300/hr and \$.15/lb;

** Ames Cray CPU Execution Time.

From Table 4 it is seen that the direct search plan (Case 3) would cost the operator \$18.06 more to fly (than Case 1) but with a savings of 15% less computation time (1.48/1.70 sec) for the constant cruise altitude case. For the particular wind field studied, there is no advantage of allowing step climbs in terms of flight cost. However, checking for this possibility costs 59% more computation time (2.70/1.70 sec) for the even grid method and 31% more computation time (1.94/1.48 sec) for the direct search method.

The flight cost results for the even grid case in Table 4 are based on only one iteration through the grid. Adding iterations to fine tune the grid will decrease the flight costs at the cost of extra computation time.

As seen in Table 4, optimizing the climb and descent Mach/CAS schedules can reduce flight costs somewhat. (\$1.38 for the even grid Case 5 and \$5.52 for the direct search Case 6). However, the improvement comes at a substantial increase in computation time (125% for the even grid case and 242 % for the direct search case). This flight cost/computation time tradeoff is questionable given that the on-board FMS can tune the Mach/CAS schedules for climb and descent after it receives the reference horizontal path from the flight planning program.

As a check of the EFPLAN trajectory computations against those of OPTIM-7S, Cases 1 and 5 of Table 4 were run on EFPLAN for a windless standard day. The resulting flight plans were input to initialize OPTIM-75. Comparisons are shown in Table 5.

Table 5. Comparison of EFPLAN and OPTIM-7S Flight Profile Cost.

Case	Program	Climb Speed Schedule	Descent Speed Schedule	Cruise Mach	Flight Time	Flight Fuel(lb)	Cost (\$)
1	EFPLAN	290/.715	270/.680	.710	3:08:14	13296	2935.57
	7S	294/.710	282/.626	.713-.715	3:07:29	13292	2931.30
2	EFPLAN	291/.711	282/.631	.710	3:08:22	13286	2934.73
	7S	294/.710	282/.626	.713-.715	3:07:28	13290	2930.90

As can be seen, the agreement is excellent. The 7S results have a slightly faster cruise time which accounts for the shorter flight times and savings of about \$4.00 per flight. This also points out that precomputing the Mach/CAS schedules for climb and descent in EFPLAN provides an adequate solution.

As mentioned before, flight planning computation time is of interest to a large airline where several flight plans have to be generated in a short time. For example, United Airlines generates over 1500 flight plans daily on a single computer. Thus, it is important to explore ways of reducing run time. A flow trace program was run on the Ames Cray to see where the EFPLAN run time was consumed for Case 2 in Table 4. Surprisingly, 74% of the run time was required just to read in weather, route, and aircraft data bases to initialize the program. Of the remaining 26% which represents actual computation and output time, 25% was used to interpolate the fuel rate tables during cruise, and 23% was used to convert true airspeed to ground speed during cruise. This flow trace utility program can be used for future study of ways to reduce EFPLAN execution time.

III

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Summary and Conclusions

This report summarizes the progress and results obtained during the study of methods for minimizing transport aircraft direct operating costs by generation of near-optimum flight paths between a given city pair. The primary end product of this effort has been the development of a new program called EFPLAN as a research tool for advanced flight planning studies. EFPLAN generates flight plans that are near-optimum reference paths; it serves to study the integration of flight planning and flight management as well as advanced flight planning techniques.

A generic 727-200 model was used to optimize a flight profile from Chicago to Phoenix over a route provided by United Airlines. The optimized results were compared to the actual UAL flight plan, and it was determined that 400 or more lb. of fuel (1.2% of the trip total) could have been saved by judicious choice of a step climb. This saving could have more than doubled with use of a cruise-climb. These results indicate the partial potential savings available by having better flight planning methods to complement the on-board flight management process.

It was established that there is a natural connection between airborne flight management and ground based flight planning and weather forecasting. The flight planning system provides the reference horizontal path, the required fuel load, and the weather profiles that are required input to the airborne flight management system. On the other hand, the airborne system has the potential of downlinking current weather measurements and actual aircraft performance data so that a better flight plan can be generated:

The flight planning process consists of five steps: initialization, data base query, flight dependent computation, route profile computation, and flight plan output. Three large data bases are required: aircraft performance, route structure, and weather.

The aircraft performance data can either model aircraft types or individual aircraft/engine combinations. These data are probably updated on a monthly basis for an airline situation. The route structure data consists of locations of available nav aids, defined airways, departure and arrival routes, and any airspace restrictions. These data are updated by the FAA every four weeks. The weather data come in grid form from the National Weather Service. It consists of six-hour forecasts that are currently updated every twelve hours. Thus, the flight planning program must be able to process data that are frequently changing.

In flight planning, the key step is the computation of the three-dimensional profile that the aircraft is to follow. The intent is to find the unique profile that provides minimum flight cost. The profile computation is done in backwards time starting with the desired landing weight. The computation phases are final approach, descent, cruise, climb, and takeoff. Cruise includes provision for step climb, step descent, or cruise-climb.

EFPLAN has been designed so that the climb and descent profiles can be optimized with the two-parameter technique presented in Appendix A. Alternatively, the climb and descent profiles can follow pre-computed Mach/CAS schedules.

Two methods have been provided to compute the cruise profile and the resulting horizontal reference path. The first, referred to as the "direct search" method, consists of searching over a series of pre-defined horizontal paths between a given city pair. A series of nav aid waypoints usually defines each path. The vertical profile is optimized over each path, and then the best path is selected. Dynamic programming methods are used to select the best cruise altitude for each route segment. A variation of this method is used today by some airlines that do their own flight planning.

The second method of generating the cruise path makes use of a grid of waypoints that are defined to be evenly spaced along and about the great circle arc that connects the city pair. Over each waypoint is a series of

vertical nodes. This "even grid" method uses dynamic programming to search for the best path through the network of grid points. The dynamic programming scheme finds the best altitude for each segment, the best segment for each stage of the network, and the points where step climbs or descents should be made between altitude levels. The resulting flight plan is one that would be suitable for future aircraft equipped with area navigation equipment and allowed to fly a "random route" by the air traffic control system.

EFPLAN has several options concerning what cruise altitude to use. These can be a fixed altitude, altitudes that are consistent with today's ATC constraints, altitude bands that are 2000 ft apart, or the cruise-climb situation where any cruise altitude can be used. The latter two cases are future possibilities in terms of actually being available for flight.

The primary EFPLAN output is a tabular flight plan that gives the sequential waypoints, altitudes, airspeeds, anticipated times and fuel burns as the flight progresses. It includes takeoff and landing time, initial fuel requirements, and the anticipated wind vectors along the route. An output option is to plot the horizontal route and the weather field between the city pair. These route and weather charts could eventually be displayed directly in the cockpit.

A key option of EFPLAN is the ability to simulate the transfer of the flight plan variables directly into a flight management system. That is, the EFPLAN output can be manipulated so that it provides compatible input to OPTIM-7S. Transferred variables includes initial aircraft weight, the waypoints that define the horizontal path, and the vertical weather profile defined over each cruise waypoint plus the terminal areas. This direct data transfer from ground-based flight planning computer to airborne flight management computer is possible with data links that are currently being purchased by the airlines.

EFPLAN is provided with six sample city pairs and their departure (SID) and arrival (STAR) routes, the necessary data base for the TSRV aircraft (B737-100), and forecast weather data for three sequential time

periods. Special emphasis was placed on design of EFPLAN so that it would be computationally efficient. Computation speed is very important to a large airline that may have over 1500 flight plans to generate in a given day.

In summary, EFPLAN has been designed as a flexible program suitable for conducting flight planning research. It is compatible with the OPTIM-7S program so that it allows future integration studies between airborne flight management and ground based flight planning.

Recommendations

Flight Planning. The EFPLAN program has been structured to allow continuing research in determining methods of generating superior flight plans. It is recommended that the following topics be pursued:

1. The even grid method assumes that the aircraft has RNAV equipment and is allowed by ATC to fly a random route. However, this is not the general case today. An alternative is to use the location of current navaid ground stations as node points to define an uneven grid. Research should be undertaken to develop a dynamic programming algorithm that finds the best path through this uneven grid.
2. Today's trans-oceanic travel is constrained somewhat by a route structure known as a track system (e.g., the North Atlantic Track system). EFPLAN should be extended to explore ways of generating better flight plans for trans-oceanic travel. This would include developing a cost performance model for a long haul aircraft such as a DC-10 or B-747.
3. Time-of-arrival control will be an important flight planning requirement for the future where 4-D guidance will be available. In this environment, airlines will be able to make flight plan contracts that allow time control from takeoff, through cruise, to meet an arrival time slot. Currently, in OPTIM-7, time-of-arrival control is done by iterating on the

cost-of-time parameter. This can be done as part of the flight planning process, but it adds an enormous computation time burden to the process of generating a single flight plan. Alternatives to this method exist. EF-PLAN could be used to generate the flight plan with some nominal cost of time, and then cruise speeds could be adjusted to meet the time requirements. Alternatively, three widespread cost-of-time parameters could be carried in the dynamic programming process to produce three different flight plans simultaneously. Then, interpolation of these solutions could be done to derive the flight plan with the required flight time. It is recommended that these alternatives be investigated.

4. The dynamic programming technique used to compute the cruise path lends itself well to use of parallel processing. Use of an advanced multi-processor architecture to host the flight planning program should be investigated.

5. To interface the flight planning system with the airborne system, there is a need to know what information must be transferred and what data link capability is required to make this transfer. This will especially affect generation of cockpit graphics to relay weather and route structure information to the pilot.

Weather. Key to good flight planning and flight management is having good weather information. Three topics should be investigated to determine what research and development are required with respect to the use of aviation weather information:

1. The differences between the cruise altitude winds and temperatures forecast and those actually measured in flight should be investigated and documented. There is much verbal referral to the fact that the weather forecasts are poor, especially over the oceans. Having actual recorded flight data compared to forecast data would quantify the degree of inaccuracy that actually exists.

2. Given that there are differences between forecast and actual winds aloft, a sensitivity study should be made to determine how detrimental these forecasting errors are to the generation of quality flight plans and to the cost of flight. The use of EFPLAN and OPTIM to make this study is a natural application.

3. Work should continue on defining how airborne in situ measurements of winds and temperature can be used to improve the forecasts. Questions include how to best measure and report the weather by an airborne system, and how best to integrate thousands of real-time weather measurements into an updated now-cast and disperse this information to various flight planning centers.

Traffic Management Interface. Both advanced flight management systems and flight planning systems should be designed with the thought that they will be integrated with each other and with the Traffic Management System emerging from the FAA's National Airspace System Plan. For effective central flow control and traffic management, it is important that the flow control computer have knowledge of the desired flight plan of each aircraft in the system.

To interface the flight planning system with the traffic management system, there is also a need to know what information to transfer. This includes specifying a description of the desired flight path to the traffic management and flow control facility. The feedback will modify this flight plan to make it compatible with other enroute crossing and approach merging flights. Conceivably, the central traffic management computer would be tied to each airline dispatch computer (the origin of the flight plan), airborne flight management computer, and traffic control center. This would be a formidable system design task, but the result could be an air traffic system that is much more fuel efficient than today's approach.

It is recommended that an investigation be made of how airline flight planning can be integrated with the evolving Traffic Management System.

APPENDIX A
TWO-PARAMETER OPTIMIZATION METHOD

Air transport crews typically follow speed profiles during climb and descent that are referred to as "Mach/CAS schedules." These speed schedules require the pilot to follow a constant indicated airspeed followed by a constant Mach number when climbing to cruise altitude and then to reverse this process when descending. This is an easy profile to follow relative to a speed profile that is continuously changing (such as the one generated by the energy state method [1,4]). The use of this simpler speed profile is adequate for flight planning purposes. A necessary task was to develop the technique to generate climb and descent profiles based on optimized Mach/CAS speed profiles.

In developing this technique, a program was first devised that could generate climb or descent profiles based on following the two-parameter Mach/CAS schedule. This program was used to generate a family of climb and descent profiles with different Mach/CAS combinations to see the effect on fuel used, time required, and direct operating cost to achieve a fixed down range distance. The results were plotted in a series of contour plots, with Mach and indicated airspeed as the variable parameters. These contour plots were then used as a reference in choosing a parameter optimization method that would find the "bottom of the bowl" - that is, the best choice of the Mach/CAS schedule for particular cost parameters, wind conditions, and aircraft weight. The choice was an algorithm called the "Complex" method. This technique was incorporated into both the flight management (OPTIM) and flight planning (EFPLAN) programs for two-parameter climb and descent optimization. The two-parameter optimization technique was then compared to the original energy state calculus of variations method.

Profile Computation

The steps in computing a climb profile that follows a Mach/CAS schedule are depicted in Fig. A.1. These are:

1. Accelerate from initial speed V_0 (210 kt) to V_{IAS1} . Hold constant altitude h_0 . V_{IAS1} is the speed limit of 250 kt indicated airspeed.
2. Climb to 10000 ft while holding V_{IAS1} .
3. Level off at 10000 ft and accelerate from V_{IAS1} to indicated airspeed V_{IAS2} . V_{IAS2} is the first variable parameter of the profile (i.e., the calibrated airspeed CAS).
- 3a. If there is no speed limit, Steps 2 and 3 are replaced by 2a/3a. Here, the aircraft first accelerates directly to V_{IAS2} while holding h_0 . Then it climbs right through 10000 ft while holding V_{IAS2} .
4. Climb while holding V_{IAS2} until intersection with Mach number M_3 . M_3 is the second variable parameter of the profile.
5. Climb while holding M_3 until reaching altitude h_f .
6. Level off at h_f and accelerate to V_f (e.g., $M = .73$). Maximum thrust is used up to this point.
7. Cruise at V_f until range r_f (e.g., 150 nmi) is reached.

The final cruise altitude and cruise speed are computed from the optimized cruise trim condition just as when using the energy state method. The final range r_f is set large enough so that there will be a small cruise portion for all Mach/CAS speed profiles attempted.

The cost of the profile is computed at the final cruise point according to

$$J = C_f (W_0 - W_f) + C_t (t_f - t_0). \quad (A.1)$$

The descent profile is computed backwards using the same steps as for climb. The difference is that except for cruise, idle thrust is used rather than maximum climb thrust.

The "Complex" Method

The currently utilized optimization technique in the OPTIM and EFPLAN programs is the Complex method [12]. This method searches for the minimum value of a function $F(x_1, x_2, \dots, x_n)$. Here, the set (x_1, x_2, \dots, x_n) are explicit values of the general x_1 . They are subject to constraints of the form

$$g_k \leq c_k(x_1, \dots, x_n) \leq h_k \quad (A.2)$$

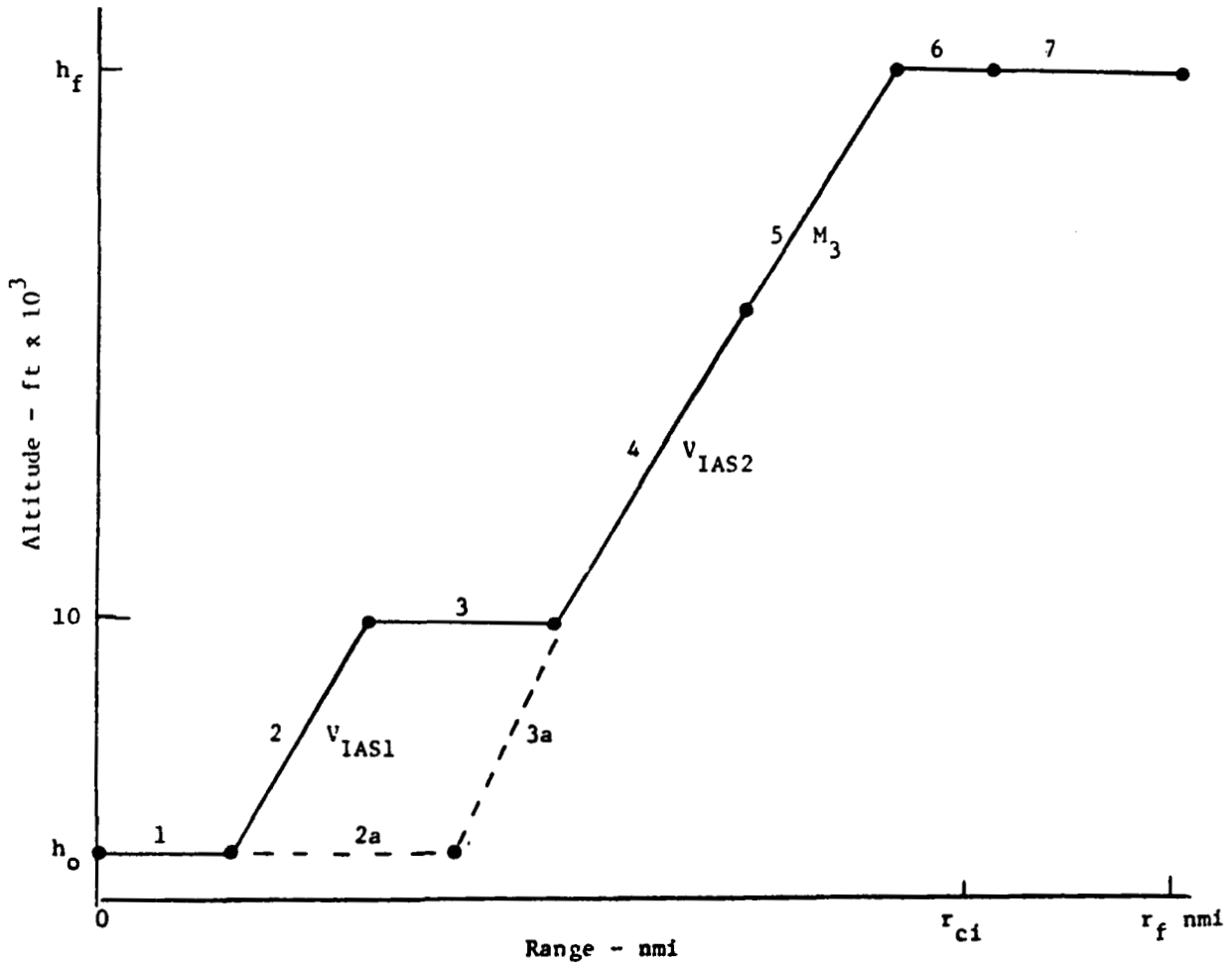


Figure A-1. Stages of Two-Parameter Climb Profile

where g_k and h_k are constants. If c_k is only a function of one variable, i.e., $c_k(x_k)$, then the constraint is said to be explicit, since the upper and lower limits on the independent variable x_k is explicitly stated. However, if c_k is a function of several variables, i.e., $c_k(x_1, x_2, \dots, x_n)$, then the constraint is said to be implicit. The complex method accepts both types of constraints. However, in our problem with V_{IAS2} and M_3 as the independent variables, we treat the problem as having no implicit constraint. Therefore, the present description is confined to explicit constraints only. We explain these constraints later.

The employed complex algorithm takes the following steps:

- 1) Select k acceptable points (points which span the space but do not violate the constraints);
- 2) Evaluate the cost function at the selected points;
- 3) Identify the two points (among the k points) at which the values of function are maximum and minimum.
- 4) Compute the difference between the maximum and minimum value of the function, and test for convergence; If convergence, exit.
- 5) Reject the worst point (point at which the functional value is maximum), and generate a new trial point where the functional value is less than those of the remaining $(k-1)$ points.
- 6) Continue the process by going back to Step 3 (next iteration) until either the convergence criteria is satisfied or the number of iterations exceeds an allowable number.

The above steps are discussed in the following paragraphs.

Point Selection

The criterion for initial point selection is discussed in the parameter selection section. At this point assume that k acceptable points have been chosen. Each point represents a choice of explicit values for the parameters (x_1, \dots, x_n) . They are acceptable if they fall within the

defined boundary constraints and span the space. By spanning the space, we mean they mathematically can be connected to describe each direction (or dimension) of the space.

Functional Evaluation

The functional value $F(x_i) \triangleq F_i$ at point i is obtained by evaluating the cost function Eq. (A.1) using the coordinate of point i . The obtained values of F_i at each i are used to identify the points of the set k at which the functional values are maximum or minimum. Let these points be P_{\max} and P_{\min} associated functional values of F_{\max} and F_{\min} .

Convergence Test

If the difference between the values of F_{\max} and F_{\min} remains less than a selected input β for a number of consecutive iterations γ (also an input), then convergence is assumed. P_{\min} and F_{\min} are then recognized as the optimal point and cost.

New Trial Point Generation

As mentioned in Step 5, the algorithm rejects the worst point and begins the search for a new trial point. Let us consider the following situation where there are three points P_i ($i = 1, 2, 3$), and their functional values F_i are shown.

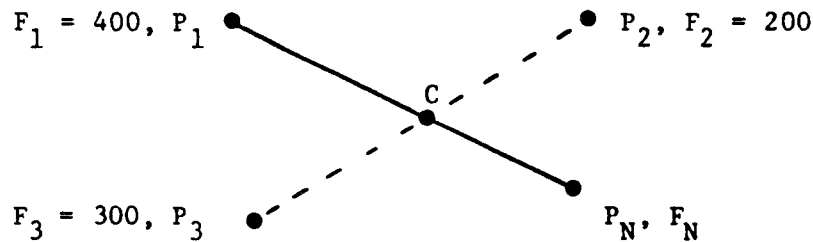


Figure A-2. Illustration of Different Points with Different Costs

The algorithm rejects P_1 because its value of 400 is greater than the function values at points P_2 and P_3 . The algorithm then finds the centroid (point C) of the remaining two ($k-1$) points. As the first guess, the algorithm assumes

that a better point can be obtained by moving away from the worst point along the line ($P_1 C$) connecting points P_1 and C . The new point P_N is found such that the length of $P_N C$ is α times that of $P_1 C$. The arbitrary factor (α) is called the reflection factor. The coordinates of the new trial point are first checked against the constraints. If a violation of a constraint occurs, the corresponding coordinate is moved by an amount (δ) inside the appropriate limit. If the functional value F_N at the new trial point is less than F_2 or F_3 , then the algorithm returns to Step 3 and starts a new iteration. If, on the other hand, F_N is not less than F_2 or F_3 , that is

$$F_N \geq F_2 \quad \text{and} \quad F_N \geq F_3 ,$$

then the subiteration process starts.

Subiteration Process

The intention of the algorithm is to move the centroid toward the optimal point in each iteration. However, if the new trial point is not an improved point, then the above purpose is not accomplished. Therefore, the subiteration process starts to provide an alternative trial point. The subiteration logic is based on the obtained functional value at the new trial point. Two possibilities that the subiteration process may follow are:

- 1) The obtained functional value is greater than the previously rejected point, i.e., $F_N > F_1$. In this case, the subiteration assumes that the new trial point (P_N) has passed a possible better trial point. Therefore, the new trial point must be moved toward the previously rejected point. In the COMPLX subroutine, the next trial point P_N is the centroid itself. If that fails, then the point half way between the centroid and P_1 is tried. If that fails, P_N is set on the line joining P_1 and the centroid but moving away from the centroid.
- 2) The obtained functional value is less than or equal to the previously rejected point, i.e., $F_N \leq F_1$. In this case, the subiteration assumes that the new trial point is not moved far enough from the previously rejected points and it moves the new trial point further away. Again, the term α is used to set this length.

The algorithm returns to next iteration upon success of the subiteration process. Success is when the new point has a functional value less than the worst of the $k-1$ functional values when F_1 is removed. The flow chart for the complex algorithm is presented in Fig. A-3.

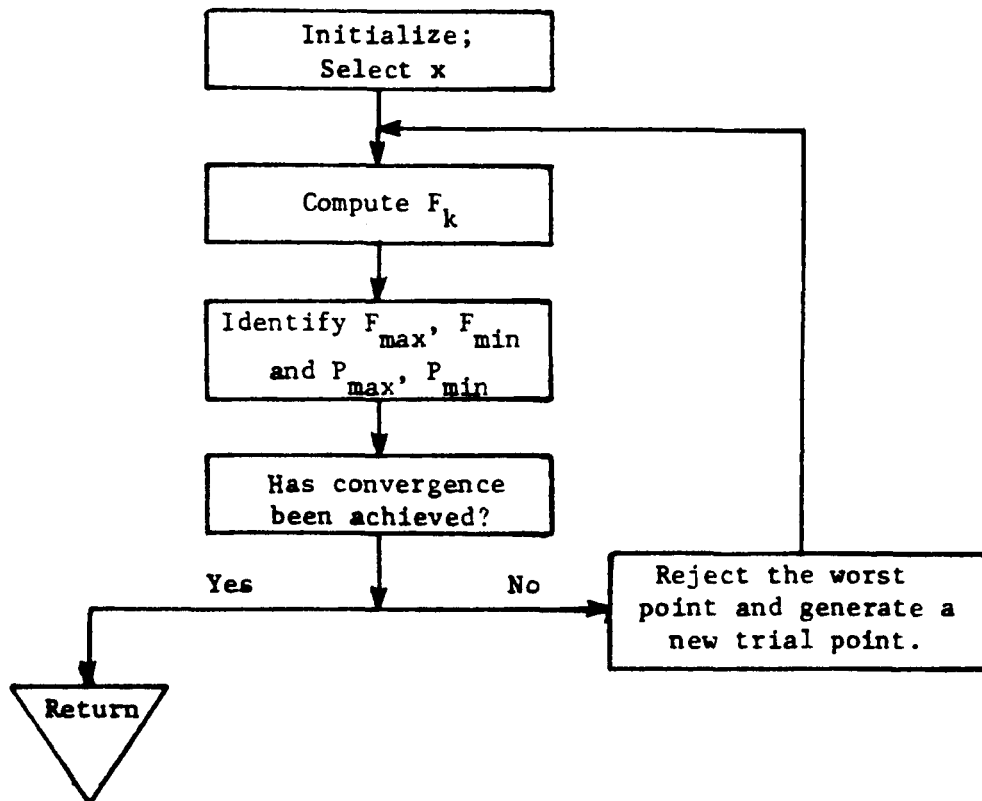


Figure A-3. Computation Flow Chart for the Complex Algorithm.

Selection of Parameters

This section presents a discussion to help in selecting the parameters in the Complex method.

Number of Points k

The number of points k must be larger than $(n+1)$, where n is the number of independent variables. Let's show this by means of the following examples:

Consider a case where $k = n$, or in our problem $k = 2$, and the following sketch shows P_i ($i = 1,2$) and F_i .

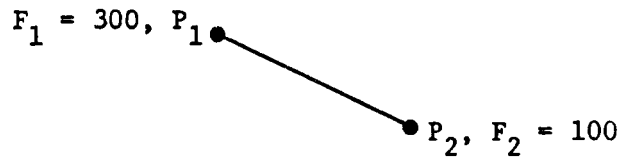


Figure A-4. Two Points with Two Costs

In this case, the new trial points for the next iterations will all lie along the line $P_1 P_2$ connecting P_1 to P_2 , whereas the optimal point P_{op} might be in another direction.

Let us show how the situation changes for $k = 3$:

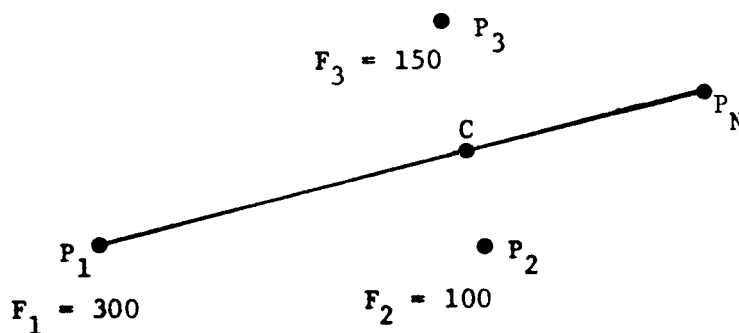


Figure A-5. Three Points with Three Costs Used to Compute P_N .

In the above figure, it is seen that the direction of search is no longer confined to one direction, and its direction is determined by the method explained earlier. However, situations can still arise where the

search loses its full dimensionality (incapable to search in all directions). This case is exemplified by considering the following points and their functional values:

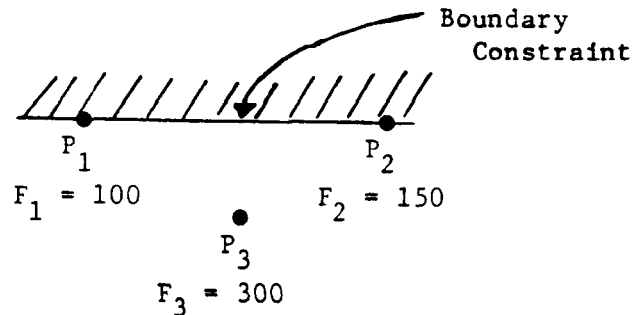


Figure A-6. The Problem Created by a Boundary

P_1 and P_2 lie on a boundary (upper or lower limit of constraint). Therefore, the new trial point P_N is found to be on the same line causing the search to lose its full dimensionality. That is, if the new trial point is on the boundary, all points thereafter will also be. To reduce this likelihood, k must be greater than $(n+1)$. However, increasing the number of points makes the numbers of search iterations greater, as it takes longer to bring all the points within the vicinity of the optimal point.

Location of the Initial Points

If the general area in which the optimal point is located is known, then selecting the initial points in that area reduces the search time. However, if this area is not known, or it changes from one case to another, then point selection must be made to cover as much area as possible (as opposed to clustering the selection).

Reflection Factor (α)

The choice of α is very important as it influences the selection of a new trial point. In the beginning, when the search is unlikely to be in the vicinity of the optimal point, rapid movement toward the optimal point can be made by large values of α ($\alpha > 1$). However, if the search is moving in the vicinity of the optimal point, then a large value of α

results in subiteration (i.e., the new trial point will jump over the optimal region). On the other hand, small values of α cause many small steps in the beginning, but reduce the subiteration time in the vicinity of the optimal point. Thus, the value of the parameter α is subject to tradeoff, and maybe it should vary in magnitude as the search progresses.

Convergence Parameter (β)

Beta (β) is defined as the allowable difference between the F_{\max} and F_{\min} if convergency is to be assumed. Beta is an indication of the required accuracy of the result.

Convergence Parameter (γ)

Each subsequent iteration must have functional values F_i with maximum allowable difference of β . There must be Gamma (γ) consecutive successful iterations to satisfy the convergence test. The recommended value for γ is from 3 to 5. This increases the probability that convergence is satisfied because it is possible that there can be k points on the same cost contour even though the function minimum is not reached.

Details for the Climb/Descent Problem

Function Evaluation

The method of generating a two-parameter climb or descent profile was outlined in Chapter II. The steps are indicated by the numbered segments in Fig. A-1. Both climb and descent are computed from the bottom up so that the cost is computed at the end of the cruise segment numbered "7" in Fig. A-1. The objective, of course, is to select V_{IAS2} and M_3 to minimize this cost. Thus, for this problem, the parameter n is two, and x_i is (V_{IAS2i}, M_{3i}) . When a trial point (V_{IAS2i}, M_{3i}) is selected, the climb or descent option of the program (called OPTTEST) is run to find the associated cost F_i .

Constraints

The constraints that must be accounted for in the optimization problem can be explained by reference to Fig. A-7. This shows the parameter relationship of altitude, true airspeed V_A , indicated airspeed V_{IAS} , and Mach number M . Assume here that the aircraft starts at 0 ft altitude, accelerates to 250 kt (V_{IAS}), climbs to 10000 ft at 250 kt, and then levels off to accelerate to V_{IAS2} . (Steps 1, 2, and 3 in Fig. A-1). To maintain the two-parameter characteristic of the profile, this value of V_{IAS2} could not be less than 250 kt. This is indicated by the constraint double line V_{min2} in Fig. A-7.

Likewise, V_{IAS2} cannot exceed 350 kt for a 737 aircraft for buffet reasons. This is indicated by the double line V_{max} . Thus, V_{IAS2} can be any value between V_{min2} and V_{max} . These values could change; V_{max} is a function of the aircraft type; V_{min2} is dependent upon the terminal area speed limit.

The upper limit for M_3 is some M_{max} , also defined by buffet constraints. In Fig. A-7, M_{max} is shown as being 0.80. However, this is aircraft dependent. Also, assume that the Mach number of the initial or final cruise speed is approximately known (say M_4); in the tests we have made,

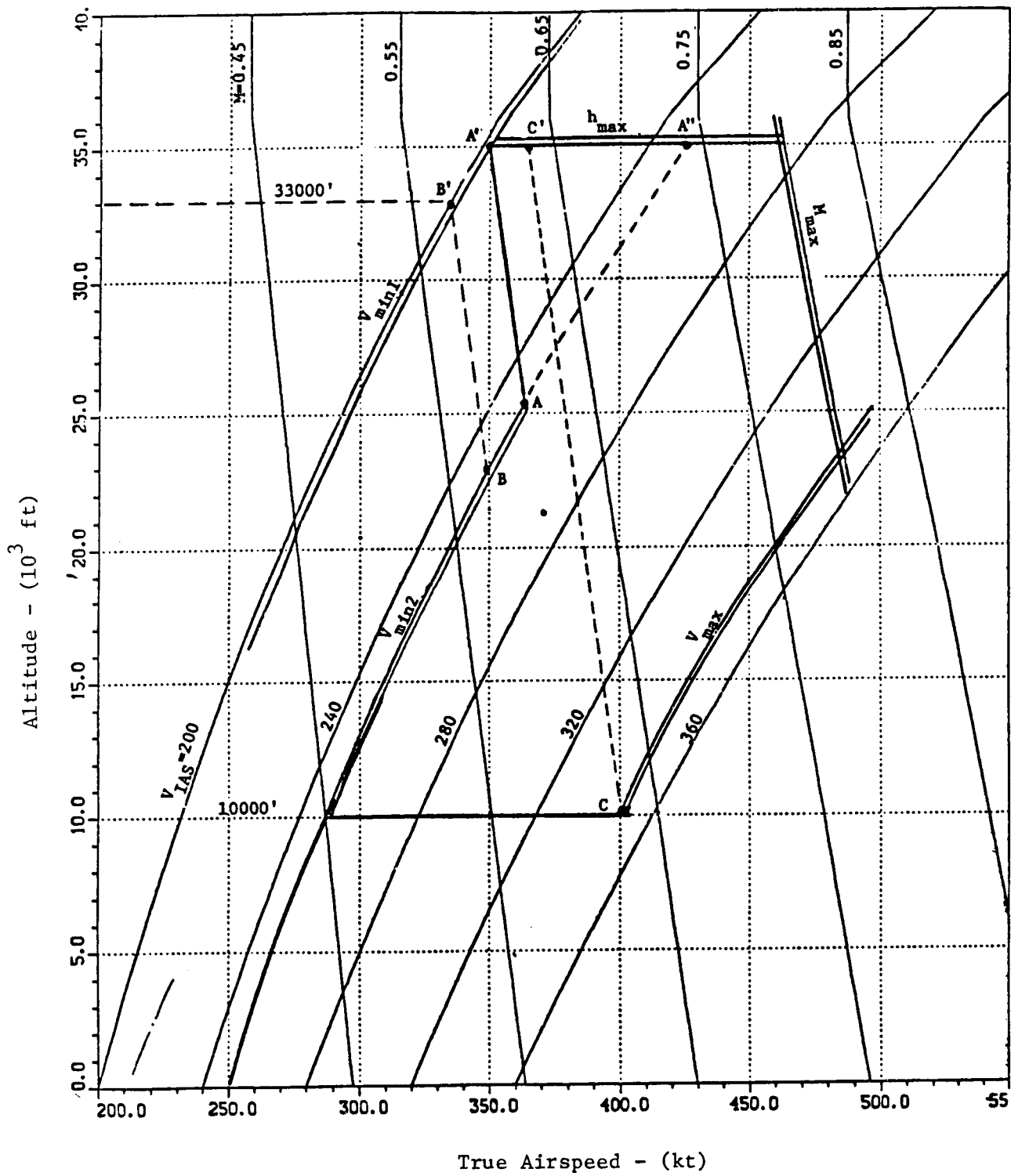


Figure A-7. Definition of Parameter Constraints for Climb-Descent Optimization

M_3 should not be greater than M_4 . That is, during climb, it is inefficient to accelerate to an M_3 greater than the M_4 of initial cruise (the value at the end of Segment 6, Fig. A-i. Thus, M_{\max} may be set lower than the value shown in Fig. A-7.

The minimum value of M_3 is dependent on the cruise altitude, the minimum cruise speed $V_{\min 1}$, and the parameter V_{IAS2} . Altitude is limited by h_{\max} , shown as 35000 ft for the 737 in Fig. A-7. However, the cruise altitude may be some fixed value less than h_{\max} (e.g., 33000 ft as shown by the dashed line in Fig. A-7.

The cruise altitude line intersects the $V_{\min 1}$ constant shown as the double line boundary in Fig. A-7. For 35000 ft, this is Point A' which is a Mach number of about 0.605. If V_{IAS} were along the 250 kt $V_{\min 2}$ line, transition to minimum M_3 of 0.605 would be at Point A. Thus, for $V_{IAS} = V_{\min 2}$, $M_{\min} = 0.605$; this is V_{IAS} of 200 kt converted to the equivalent Mach number at 35000 ft. For cruise at 33000 ft, the minimum Mach number M_{\min} is evaluated at Point B' - about 0.570. The transition to this minimum would occur at Point B.

Note that the choice of V_{IAS2} also affects M_{\max} . If for example, V_{IAS2} is $V_{\min 2}$, then the maximum Mach number that can be reached is the value when the climb at V_{IAS2} intersects the cruise altitude. This is shown in Fig. A-7 as Point A'', or M_{\max} is about 0.735.

If V_{IAS2} is set at V_{\max} , then the minimum M_{\min} is found at point C, or M_{\min} is about 0.630. For this case, M_3 can vary between 0.630 and the minimum of M_{\max} or cruise Mach number.

In the COMPLX algorithm, whenever a test pair (V_{IAS2} , M_3) is generated as a trial point, it is first tested against the boundaries depicted in Fig. A-7, before the function is evaluated. First, V_{IAS2} is tested to make certain it falls between $V_{\min 2}$ and V_{\max} . Then, M_3 is tested so that it falls between M_{\min} and M_{\max} . M_{\min} and M_{\max} are computed as functions of cruise altitude, cruise speed, $V_{\min 1}$ and V_{IAS2} .

Trial Points

For testing the COMPLX algorithm, we used k of 4 or 5. The program has an input pair xs_1 and xs_2 as typical values of V_{IAS2} and M_3 . For k of 4, the four initial points are

$$V_{IAS2} = xs_1 \pm VTA ,$$

$$M_3 = xs_2 \pm EMA .$$

VTA and EMA are set at 20 kt and 0.02 Mach, respectively. For k of 5, a fifth trial point is

$$V_{IAS2} = xs_1; \quad M_3 = xs_2 .$$

For the test cases with the 737, we used xs_1 of 310 kt and xs_2 of 0.680.

Results

In this section, sample results are shown for the climb with no wind case. We examined minimum cost ($C_f = \$0.15/\text{lb}$; $C_t = \$600./\text{hr}$), minimum fuel ($C_t = 0$), and minimum time ($C_f = 0$) results. It was assumed that takeoff weight was 90000 lb, cruise altitude was 33000 ft, and cruise Mach was 0.730. Down range distance was 150 nmi.

Several values of α , β , and γ were tried, and the intent was to insure convergence, but in the least number of iterations. This, of course, is also dependent upon the particular choice of initial guessed points. Reasonable results were found with β set at \$0.05 and γ set at 4. This means that there must be four consecutive solutions within \$0.05 of each other to declare convergence. Also, the value of k set to 4 was more efficient than k of 5.

The following Table A-1 indicates comparison of solution for minimum cost, time, and fuel with α set at 0.5, 0.7, and 0.8. Nine cases are shown, and the number of iterations to find a solution varies from 13 to 19. For cost, α of 0.7 and 0.8 give superior results. For fuel, α of

Table A-1. Comparison of Two-Parameter Search Results with Varying α .

Case	Criterion	α	Iter.	V_{IAS2}	M_3	Cost(\$)	Fuel(lb)	Time(sec)
1a	Cost	0.5	13	306.7	.7233	687.67	3064	1369
b		0.7	16	308.6	.7299	687.60	3068	1365
c		0.8	16	309.2	.7289	687.60	3068	1364
2a	Fuel ($C_t=0$)	0.5	17	283.7	.6861	457.61	3051	1401
b		0.7	15	286.4	.6764	457.66	3051	1403
c		0.8	17	284.3	.6837	457.62	3051	1401
3a	Time ($C_f=0$)	0.5	19	340.1	.7173	225.07	3099	1350
b		0.7	15	350.0	.7299	223.11	3123	1338.67
c		0.8	17	348.9	.7299	223.19	3120	1339.14

0.7 converges faster, but α of 0.5 has slightly less cost. All answers are within one lb. For time, α of 0.7 converges faster and gives the exact solution (i.e., V_{IAS2} and M_3 are maximum). Thus, we use $\alpha = 0.7$ in OPTIM7.

Figures A-8, A-9 and A-10 show the contour plots of the parameter search region for cost, fuel, and time, respectively. Also shown on these plots is the movement of the centroid during the search process of Cases 1, 2, 3b from Table A-1. As can be seen, the first two iterations move quickly to the desired region, and thereafter it is just fine tuning.

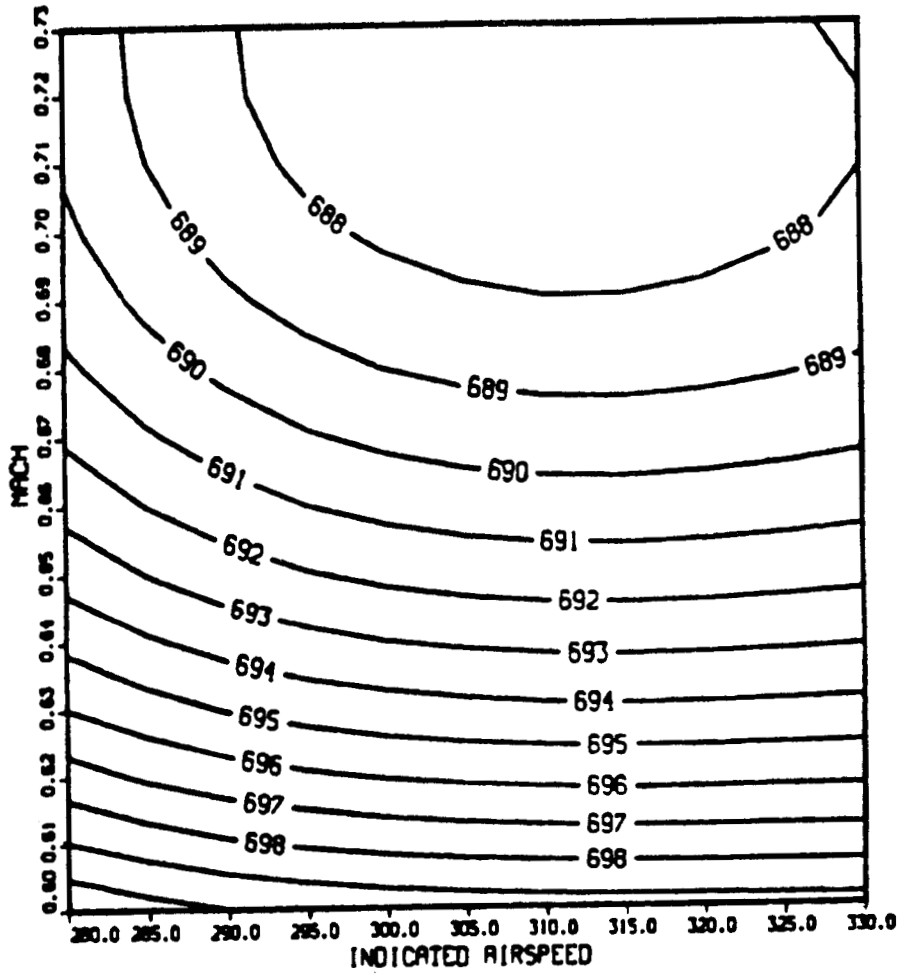


Figure A-8. Cost Contours (\$) for No Wind Climb

ORIGINAL COPY
OF POOR QUALITY

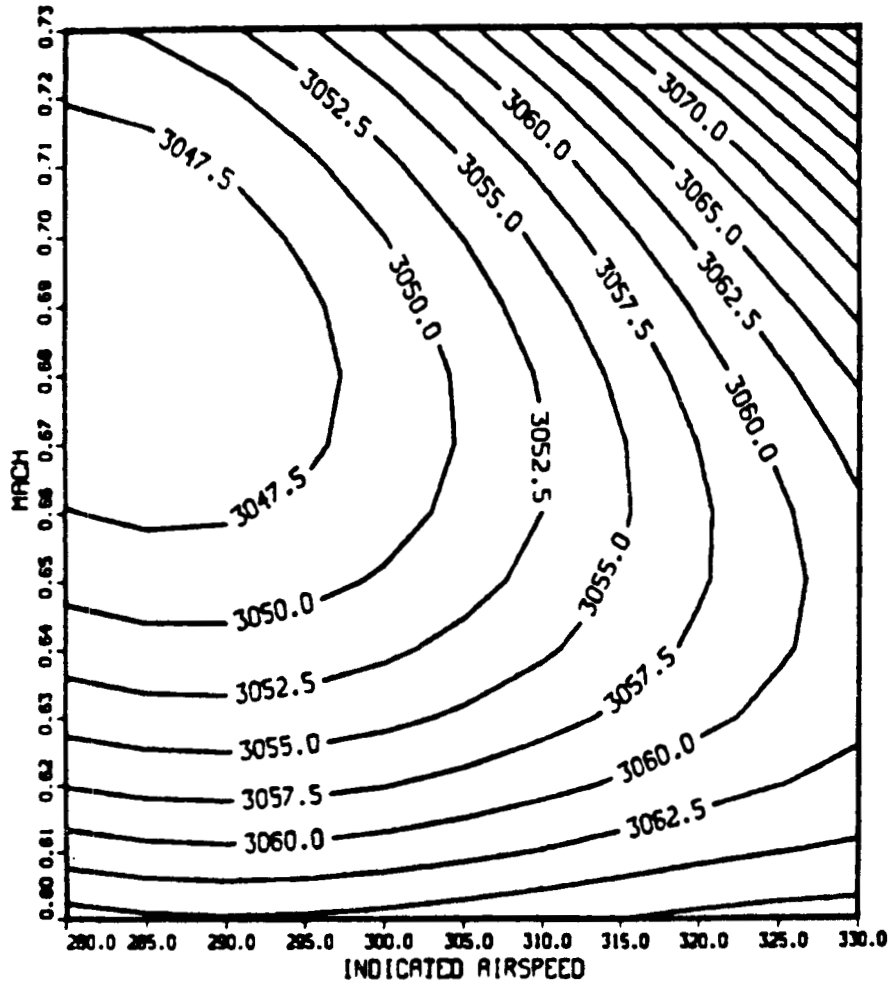


Figure A-9. Fuel Contours (Lb) for No Wind Climb

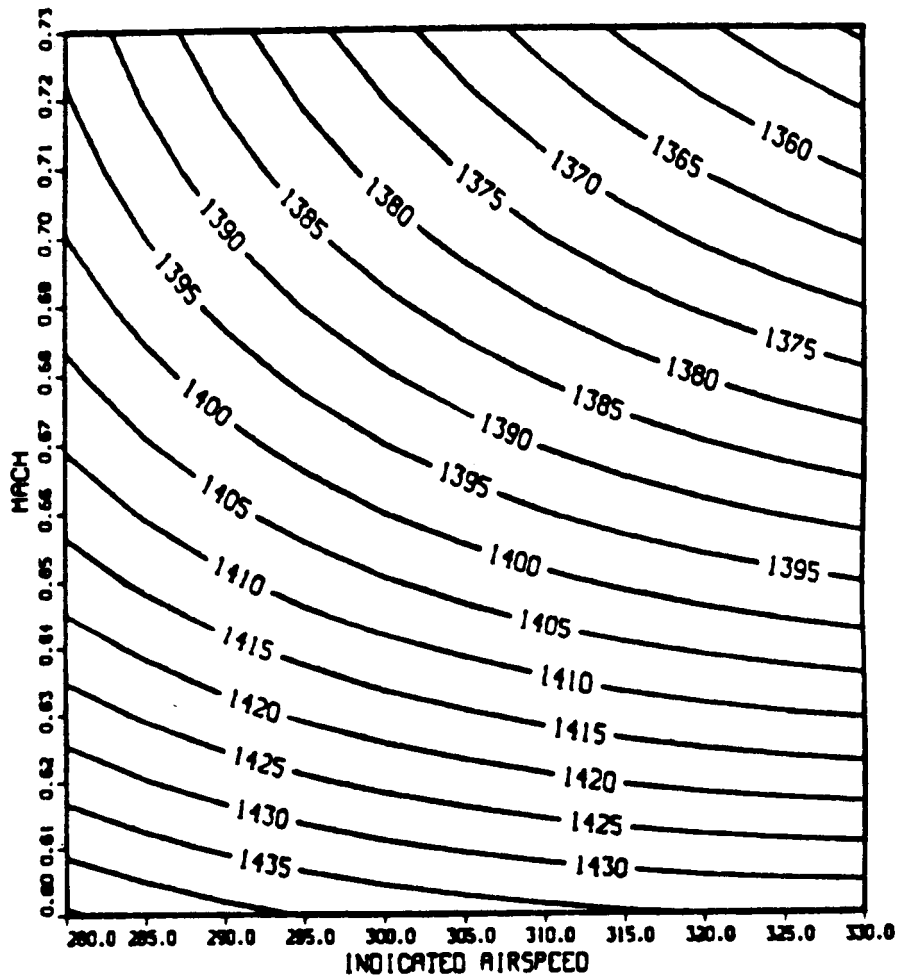


Figure A-10. Time Contours (Sec.) for No Wind Climb

APPENDIX B

MODELING OF NON-STANDARD DAY

The barometric altimeter determines altitude based on measured static pressure. If the ground pressure setting on the altimeter is set at 29.92 in Hg (standard sea level pressure), the altitude reading is referred to as pressure altitude. There are several sources of error for this measured altitude. These can be grouped into two categories:

1. Meteorological error due to the atmosphere being non-standard; and
2. Altimeter instrumentation errors.

With respect to generating optimum vertical profiles, we are primarily interested in the first error source. This is first described, followed by an explanation of how it can be modeled and compensated for in both a flight management system (FMS), and a flight planning system (FPS).

Meteorological Error

The baro-altimeter is calibrated to convert static pressure to altitude based on the Standard Atmosphere. The Standard Atmosphere [13] is a model of pressure, density, and temperature as functions of geopotential altitude. This model is based on the following assumptions:

1. Sea-level conditions are as follows:

$$\begin{aligned} \text{Pressure } p_o &= 29.921 \text{ in Hg} = 2116 \text{ lb/ft}^2 \\ &= 101325 \text{ N/m}^2 = 1013.25 \text{ mbars.} \end{aligned}$$

$$\text{Density } \rho_o = 0.0023769 \text{ slug/ft}^3$$

$$\text{Temperature } T_o = 59.0^\circ\text{F} = 288.16^\circ\text{K} = 15.0^\circ\text{C}$$

$$\text{Gravity } g_o = 32.17 \text{ ft/sec}^2 \text{ (assumed constant for geopotential altitude)}$$

2. The equation of state for dry air holds:

$$p/\rho = g_0 RT \quad (B.1)$$

where R is the gas constant (R=53.3 for air) and T is in °Rankine (°F + 459.4°). Also, p and ρ are pressure and density in lb/ft² and slug/ft³, respectively.

3. Temperature varies linearly with altitude up to the tropopause (at 36,089 ft = 11 km).

$$T (°F) = 59 - .003566°h \quad (B.2)$$

Thereafter, it is constant at -69.7°F up to 65,617 ft (20 km). Equation (B.2) is written as

$$T = T_0 - ah, \quad (B.3)$$

where a is known as the lapse rate.

4. The differential pressure over a small slice of air can be written as

$$\frac{dp}{p} = -\frac{dh}{RT}, \quad (B.4a)$$

or

$$dp = -\rho g_0 dh, \quad (B.4b)$$

where dh is the width of the slice.

Using Eqs. (B.3) and (B.4) gives

$$\int_{p_0}^p \frac{dp}{p} = \int_{T_0}^T \frac{dT}{aRT}, \quad (B.5)$$

or

$$\ln \left(\frac{p}{p_0} \right) = \frac{1}{aR} \ln \left(\frac{T}{T_0} \right), \quad (B.6)$$

From Eqs. (B.3) and (B.6), we can write

$$(p/p_0)^{aR} = (T_0 - ah)/T_0, \quad (B.7)$$

and

$$h = \frac{T_0}{a} \left(1 - \left(\frac{p}{p_0} \right)^{aR} \right). \quad (B.8)$$

Equation (B.8) is the expression of geopotential altitude as a function of lapse rate a , sea-level conditions p_0 and T_0 , and measured pressure p . It is the basic equation of the baro-altimeter.

Altimeters are designed so that they can be corrected for deviations in surface pressure. The pressure is measured on the ground, and an equivalent sea-level pressure p_{oc} is computed so that Eq. (B.8) holds for the known altitude h of the measurement site (usually the runway) and the measured pressure p . This corrected pressure setting p_{oc} is entered into the altimeter whenever the aircraft flies below 18000 ft. This term is then used by the altimeter so that it reads runway altitude upon landing. Above 18000 ft, the standard p_0 (29.92 in Hg) is used. There are no means for correcting for deviations in the lapse rate a or ground temperature T_0 in most altimeters. Thus, these variations cause an error between true altitude and the indicated baro-altitude.

Figure(B-1) compares standard atmospheric pressure and temperature vs geopotential altitude with the average values measured each day in Albuquerque as an example during September 1980. The deviation in temperature is most pronounced. However, if a new lapse rate a , reference temperature T_0 , and pressure p_0 are defined for each segment, then Eqs. (B.3) and (B.6) hold for this segment. The vertical distance between measured and standard pressure lines result in the average altitude error as a function of altitude. The maximum value of this error is about 600 m (2000 ft) at 10 km altitude.

For a given day, the temperature profile can be modeled as **piecewise** linear segments as functions of geopotential altitude. This would taking into account temperature inversions and movement of the tropopause height. As seen in Fig. B-1, the temperature above the tropopause may actually increase with altitude rather than be a constant as in the **Standard Day**.

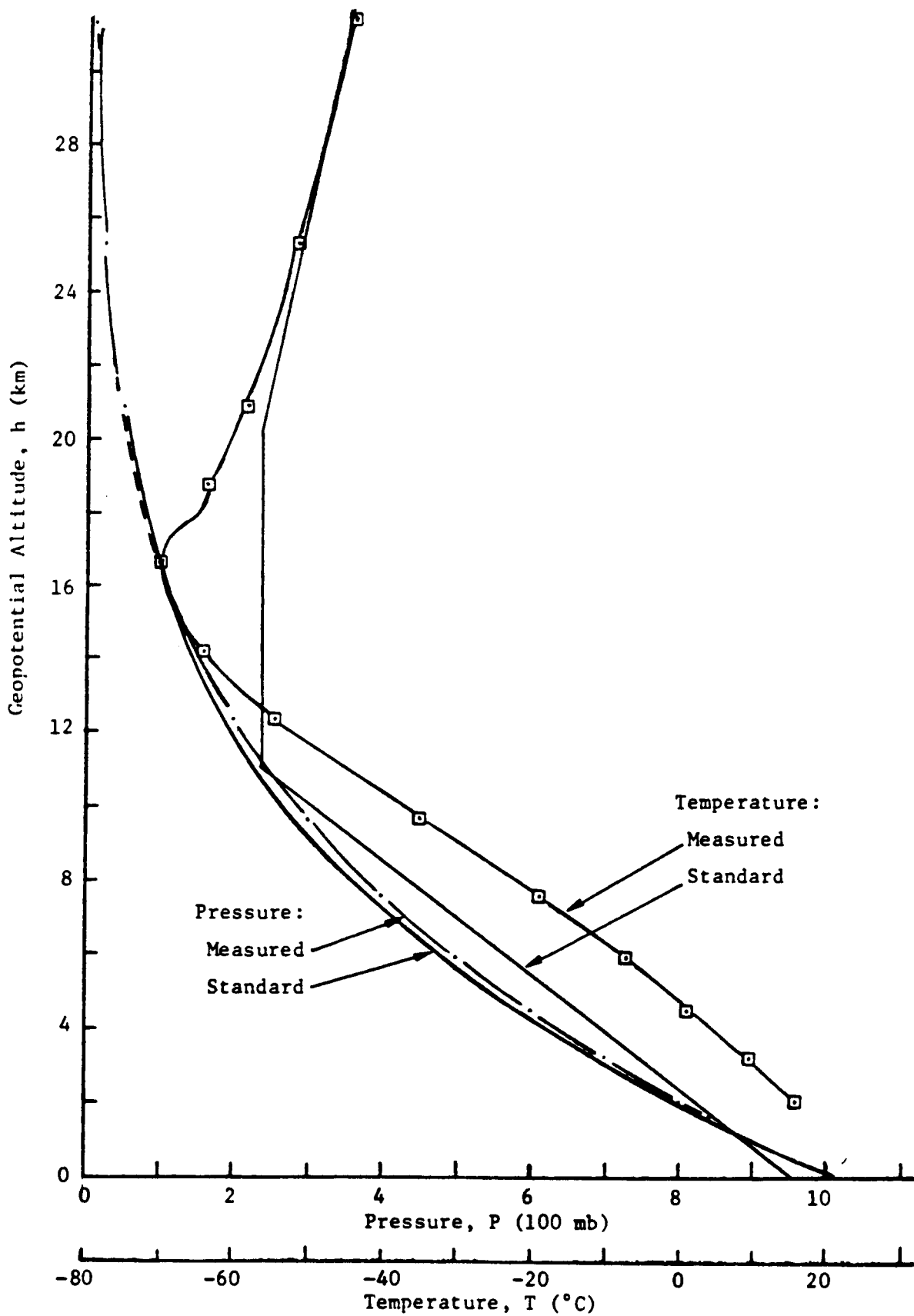


Figure B-1 Rawinsonde Data for Albuquerque, September 1980.

The arrows in Fig. B-2 show the lowest and highest mean monthly temperatures obtained for any location between the Equator and pole. Estimates of the one-percent maximum and minimum temperatures that occur during the warmest and coldest months, respectively, in the most extreme locations are shown by dashed lines. Values below 30 km are based on radiosonde observations.

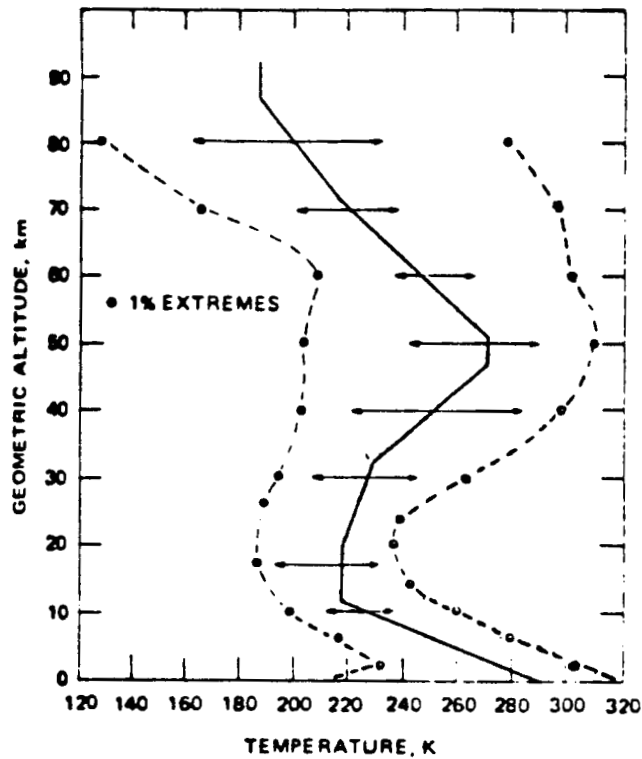


Figure B-2. Range of Systematic Variability of Temperature Around the U.S. Standard Atmosphere, 1976 [13]

Values shown for the various levels by envelope curves could not possibly be encountered at all altitudes at a given location and time. The warmest layers near the surface, for example, are associated with the coldest temperatures at the tropopause.

At locations between 30 and 90°N, maximum mean monthly temperatures at altitudes below 25 km usually occur in June or July, and minimum values occur in December or January.

If one can obtain the temperature profile and the surface pressure, then Eq. (B.8) can be used to compute the actual geopotential altitude for a given day from measured pressure p . This can be used with Eq. (B.8),

based on standard day values of (a, p_0, T_0) , to determine the altitude error due to meteorological conditions.

For higher altitudes, the assumption inherent in Eq. (B.4b) that the gravitational acceleration remains constant at the sea-level value begins to produce another minor source of error. Note that the altitude h of Eqs. (B.4) is referred to as "geopotential altitude". The Newtonian equation for gravitational potential is

$$g = g_0 \left(\frac{r_e}{r_e + z} \right)^2, \quad (\text{B.9})$$

where r_e is the earth's mean radius ($r_e = 3959 \text{ s.mi} = 6371 \text{ km}$), and z is the true, or "geometric altitude". The actual hydrostatic equation for atmospheric balance is

$$dp = -\rho g dz, \quad (\text{B.10})$$

which replaces Eq. (B.4b). From Eqs. (B.4b), (B.9), and (B.10), we get the relationship that

$$h = \frac{r_e z}{r_e + z}. \quad (\text{B.11})$$

At the tropopause ($h = 11 \text{ km}$), the difference in h and z is 19 m (62 ft). The geopotential altitude is used with the definition of the Standard Atmosphere as a matter of convenience.

Non-Standard Day Modeling Initialization

At any horizontal (x,y) location, we can define temperature as a function of pressure altitude. This is what is presently available from the National Weather Service as part of the Aviation Digital Forecast (ADF). These data also include the geopotential (or true) altitude data as included in the GRIB Code parameters (see Appendix D). Thus, our model is assumed to have temperature deviation from standard day values and true altitude as functions of pressure altitude. The standard day model is based on true altitude as the independent parameter, anyway, so it should be included.

The altitude/temperature profile table contains three variables:

- (1) Pressure altitude, h_p , (ft);
- (2) Temperature deviation, ΔT , ($^{\circ}K$); and
- (3) True geopotential altitude, h_t , (ft).

For climb, descent, and step climb, we work primarily in the true altitude frame. This is because energy steps are in terms of true altitude. Pressure altitude is needed for pilot reference during climb and descent. For cruise flight, we work in the pressure altitude frame because cruise is normally at a given pressure altitude. In cruise, true altitude is computed for reference.

The baro-altimeter is calibrated based on the standard day model of pressure vs. altitude. Thus, this model has to be cross-referenced to the alternate model representing altitude-temperature relationships that prevail on the given day of flight and over a given (x,y) location. The differences might be as shown in Figure B.3. Note in this figure that temperature is assumed to be a piecewise linear function of true altitude. The slope for each piece, or segment, is the "lapse rate" a . For standard day, lapse rate is designated a_0 , and temperature begins at T_0 at sea level and goes to T_T at the tropopause altitude of 11km. Above the tropopause, the standard day temperature is constant at T_T up to 20km.

The actual temperature profile is measured at different pressure altitude levels by rawindsonde balloon tracking. Associated with each pressure altitude ($h_{p1} - h_{p4}$ in Fig. B-3) there is a true altitude ($h_{t1} - h_{t4}$). The absolute temperature is read at these altitudes (e.g., $T_{*1} - T_{*4}$). From these, the lapse rate for each segment ($a_{*1} - a_{*3}$) can be computed. Temperature is usually given at each altitude as a deviation from standard day.

Note in Fig. B-3 that the actual tropopause may have an arbitrary definition as to altitude measurement. It is usually defined as where the temperature becomes constant or becomes larger with altitude. This point may be above or below the maximum cruise altitude of the 737 which is 35000 ft (pressure altitude).

After we read in the temperature - altitude data base but before we compute trajectories, we can compute the additional constants associated with each segment of the table, based on different true altitudes. This is done as follows.

Between two true altitudes (h_{t1} and h_{t2} ; $h_{t2} > h_{t1}$), we can write

$$\left(\frac{p}{P_{*1}}\right)^{a_{*1}R} = \frac{T}{T_{*1}} \quad . \quad (B.12)$$

Here,

- T_{*1} = reference temperature measured at h_{t1} ;
= $T(h_{p1}) + \Delta T(h_{p1})$;
- h_{p1} = pressure altitude corresponding to h_{t1} ;
- $T(h_{p1})$ = standard temperature at h_{p1} ;
- $\Delta T(h_{p1})$ = temperature variation from standard at h_{p1} ;
- P_{*1} = reference pressure at h_{t1} and h_{p1} ; standard pressure;
- a_{*1} = computed lapse rate for segment between h_{t1} and h_{t2} ;
- p = pressure at a higher altitude h_t ($h_{t1} \leq h_t \leq h_{t2}$);
- T = temperature at the higher altitude; and
- R = universal gas constant.

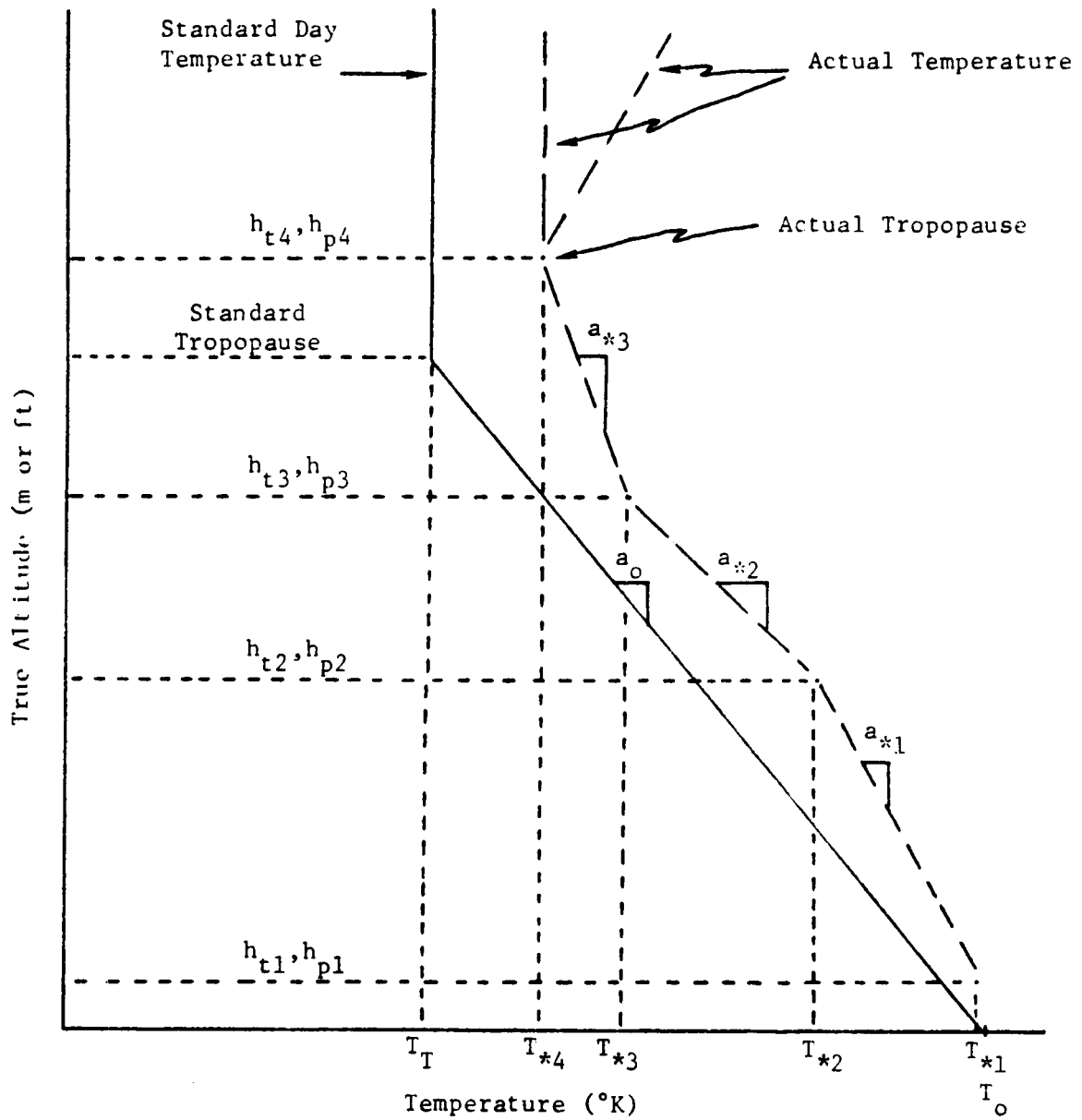


Figure B-3 Example Comparing Temperature - Altitude Profile on a Given Non-standard Day with Standard Day Profile.

To be able to solve Eq. (B.12), we need p_{*1} , and T_{*1} .

For each pressure altitude h_{p1} in the input table, we also read in the temperature variation $\Delta T(h_{p1})$. By using the standard model (call to subroutine ATLOW), we compute the standard temperature $T(h_{p1})$ at h_{p1} . Then, we compute

$$T_{*1} = T(h_{p1}) + \Delta T(h_{p1}) . \quad (B.13)$$

We also obtain (in ATLOW) the standard pressure at h_{p1} which is stored as p_{*1} . The lapse rate for this segment is computed as

$$a_{*1} = \frac{T_{*1} - T_{*2}}{h_{t2} - h_{t1}} . \quad (B.14)$$

This requires that T_{*2} be computed at h_{t2} . This process of obtaining and computing T_* , a_* , and p_* is done for each pressure/true altitude combination over a given location. The table is expanded to include these three parameters.

Note that a_* from Eq. (B.14) is usually positive except in the case of a "temperature inversion".

Climb/Descent Computations

With the above parameters initialized at the temperature profile read-in point, the following sequence is used to compute temperature, pressure, density ρ , and speed-of-sound V_{SS} during climb and descent. The true altitude h_t is used to interpolate in the table to find the next lower T_* , a_* , p_* , and h_{t*} . Then, the following computations are made

$$T = T_* - a_* (h_t - h_{t*}) , \quad (B.15a)$$

$$p = p_* (T/T_*)^{1/a_* R} , \quad (B.15b)$$

$$\rho = p/gRT , \quad (B.15c)$$

$$V_{SS} = (1.4 g RT)^{1/2} . \quad (B.15d)$$

If the lapse rate a_* is close to zero, as is sometimes true above the tropopause, we substitute

$$T = T_* e^{-[(h_t - h_{t*})/RT_*]} , \quad (B.16a)$$

$$p = p_* e^{-[(h_t - h_{t*})/RT_*]} , \quad (B.16b)$$

for Eqs. (B.15a) and (B.15b).

In solving for pressure altitude, we are computing what a baro-altimeter would read as a function of pressure. This is based on two formulae. Below the standard tropopause pressure p_T , (that is, $p \geq p_T$), we use

$$h_p = \frac{T_o}{a_o} \left(1 - \left(\frac{p}{p_o} \right)^{a_o R} \right) \quad (B.17a)$$

Above p_T (that is, $p < p_T$), we use

$$h_p = h_T - RT_T \ln(p/p_T) \quad (B.17b)$$

In these formulae, the parameters T_o , a_o , p_o , R , h_T , T_T , and p_T are fixed constants. They are the equivalent of constants used in the ATLOW subroutine.

Cruise Computations

For the cruise phases of flight, we begin with a given pressure altitude h_p . The computation sequence is then the reverse of the computations shown as Eqs. (B.15-B.17). We first call ATLOW to find the pressure associated with h_p .

We next interpolate h_p in the table to find the next lower T_* , a_* , P_* , and h_{t*} . Then, the true altitude h_t corresponding to h_p is either

$$h_t = \frac{T_*}{a_*} \left(1 - \left(\frac{P}{P_*} \right)^{a_* R} \right) + h_{t*} , \quad (\text{B.18a})$$

for $a_* \neq 0$, or

$$h_t = h_{t*} - R T_* \ln \left(\frac{P}{P_*} \right) , \quad (\text{B.18b})$$

for $a_* \approx 0$. Then temperature, density and speed-of-sound are computed using Eqs. (B.15a), (B.15c), and (B.15d).

During a step climb, we go from one pressure altitude to another. But this is done by computing smaller true altitude steps. This requires first solving for $h_{t,\text{bottom}}$ and $h_{t,\text{top}}$ corresponding to $h_{p,\text{bottom}}$ and $h_{p,\text{top}}$. These define the bottom and top altitudes of the step climb.

APPENDIX C

727-200 MODELING DETAIL

In developing the model for the 727-200 aircraft, further insight was gained in developing concise polynomials to represent drag coefficient and normalized thrust. For airborne applications, it is important to have as few as possible coefficients either for table lookup or for polynomial evaluation. A compact polynomial giving Z as a function of two variables x , and y , is

$$\begin{aligned}
 Z = & c_{11} + c_{12}x_2 + \dots c_{1n}x_2^{n-1} \\
 & + c_{21}x_1 + c_{22}x_1x_2 + \dots c_{2n-1}x_1x_2^{n-2} \\
 & + \dots \\
 & + c_{n-1,1}x_1^{n-2} + c_{n-1,2}x_1^{n-2}x_2 \\
 & + c_{n1}x_1^{n-1}.
 \end{aligned} \tag{C.1}$$

This was used during the modeling of the 727-200.

It was found that for representing drag coefficient C_D , a convenient way to model the natural increase in drag with Mach number was to define a new variable,

$$M_p = a + b * M^{13} \tag{C.2}$$

Then, the drag coefficient can be represented very accurately with a third or fourth order polynomial of the form of Eq. (C.1). Figure C-1 shows the input data points and the resulting curve fit lines for a third order polynomial consisting of ten coefficients. In OPTIM, a fourth order polynomial is used having only fifteen coefficients. Fit error is consistently better than 1%.

The drag coefficient requires previous computation of the lift coefficient C_L . This is shown for the 727-200 for zero flaps in Fig. C-2. The data points from Fig. C-2 are used to find the required lift term as a function of angle-of-attack when trimming the aircraft.

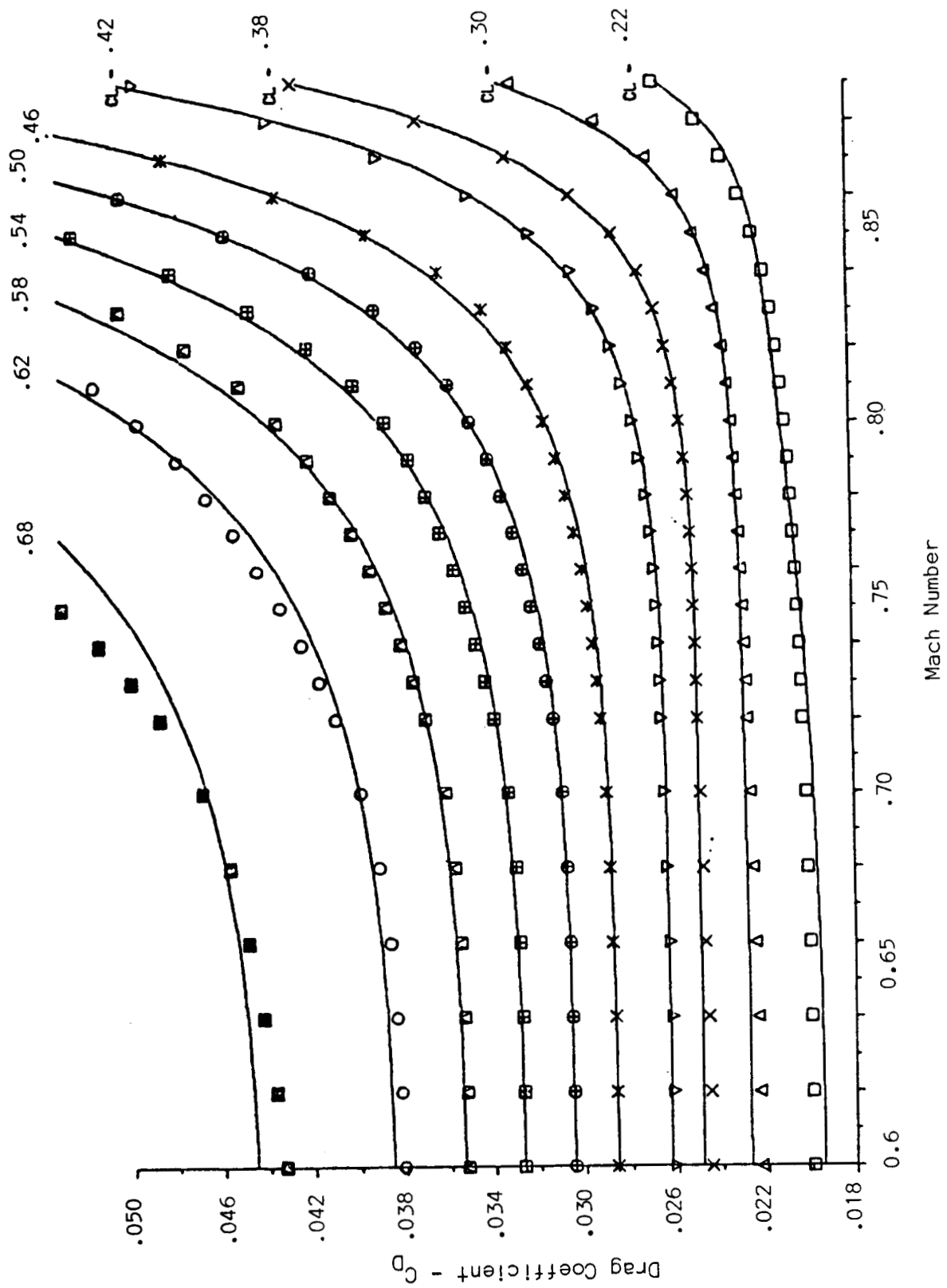


Figure C-1. Drag Coefficient Polynomial Fit vs C_L and Mach.

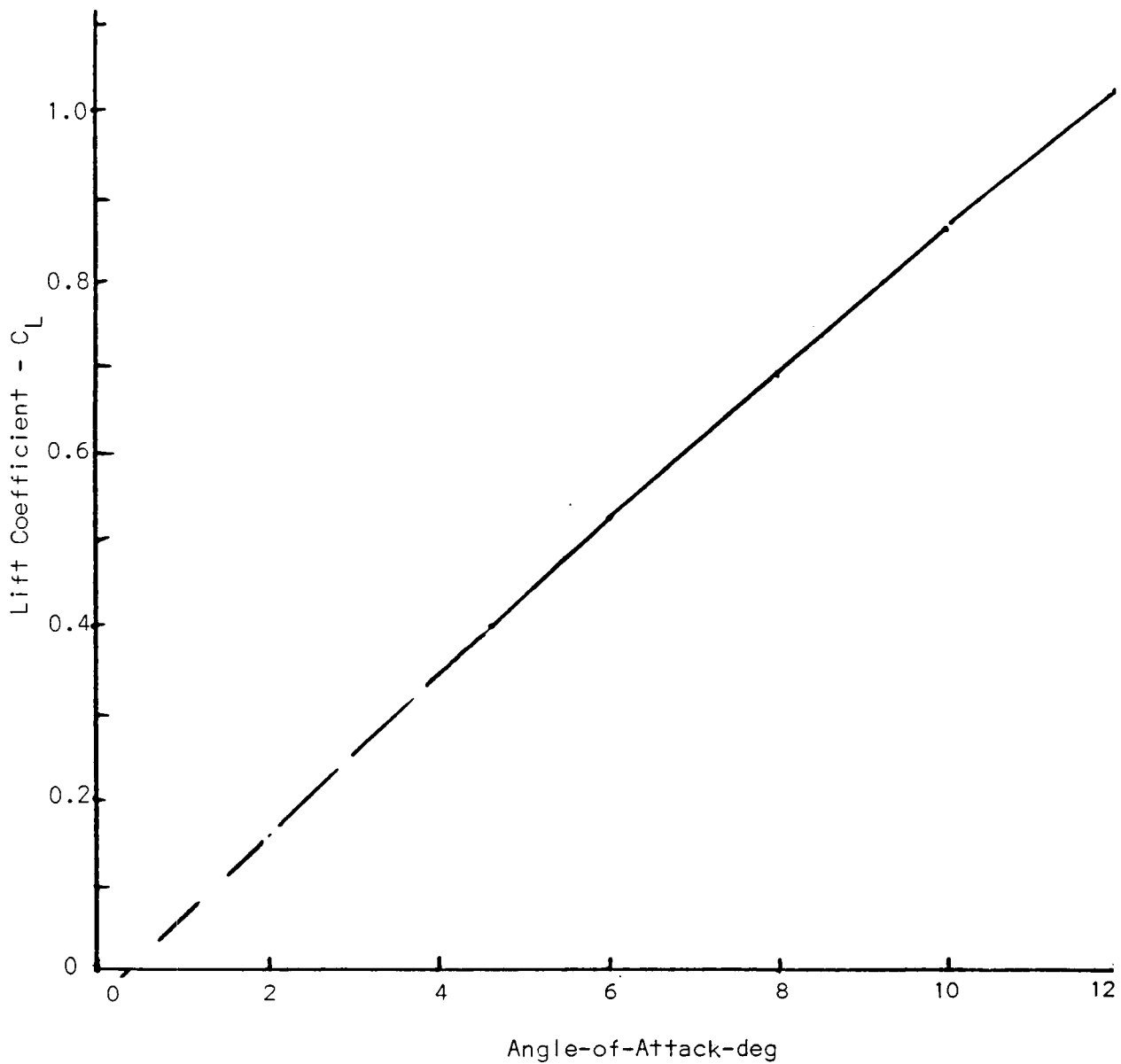


Figure C-2. Lift Coefficient vs Angle-of-Attack

To compute thrust for aircraft trim requires selection of the correct engine pressure ratio (EPR). This first requires that the maximum EPR setting for the aircraft altitude and atmospheric temperature be established. For cruise, the maximum value is found by interpolation from Table C-1. A similar table is used to compute maximum EPR for climb thrust.

The thrust divided by pressure ratio is a function of EPR and Mach number. This function can be modeled very nicely with a third order polynomial of the form of Eq. (C.1). Figure C-3 shows the input points and the resulting curve fit for the 727-200 center engine. This is a JT8D-15 engine. Only ten coefficients are required. A similar fit was made for the two pod engine thrust data.

Fuel flow data were obtained from plots such as shown in the example in Fig. C-4. Here, the corrected fuel flow curves are also smooth functions of EPR and Mach number. These data are currently in tabular form in OPTIM. They were not curve fit for budgetary reasons, but they are in the form that makes future curve fitting a relatively straight forward task.

Table C-1. Maximum Cruise EPR vs Altitude and Temperature

MAX CRUISE EPR										
TAT	PRESSURE ALT (1,000 FT)									
°C	5	10	20	29	31	33	35	37	39	41
45	1.53	1.52	1.51	1.50	1.50	1.50	1.49	1.49	1.48	1.48
40	1.57	1.56	1.55	1.54	1.54	1.54	1.53	1.53	1.52	1.52
35	1.61	1.60	1.59	1.58	1.58	1.58	1.57	1.57	1.56	1.56
30	1.65	1.65	1.64	1.63	1.63	1.63	1.62	1.61	1.60	1.60
25	1.70	1.69	1.68	1.67	1.67	1.67	1.66	1.66	1.65	1.65
20	1.75	1.74	1.73	1.72	1.72	1.72	1.71	1.71	1.70	1.70
15	1.80	1.80	1.79	1.78	1.78	1.78	1.77	1.77	1.76	1.76
10	1.87	1.85	1.85	1.84	1.84	1.84	1.83	1.83	1.82	1.82
5	1.93	1.93	1.92	1.91	1.91	1.91	1.90	1.90	1.89	1.89
0	1.99	1.99	1.98	1.97	1.97	1.96	1.95	1.95	1.94	1.94
-5	2.04	2.04	2.03	2.02	2.02	2.02	2.01	2.01	2.00	2.00
-10	2.09	2.09	2.08	2.07	2.07	2.07	2.06	2.05	2.04	2.04
-15	2.13	2.13	2.12	2.10	2.10	2.10	2.09	2.09	2.08	2.08
-20	2.16	2.15	2.14	2.13	2.13	2.13	2.12	2.12	2.11	2.11
-25	2.18	2.18	2.17	2.16	2.16	2.16	2.15	2.14	2.13	2.13
-30	2.20	2.19	2.18	2.17	2.17	2.17	2.16	2.16	2.15	2.15
-35	2.21	2.20	2.19	2.19	2.18	2.18	2.17	2.17	2.16	2.16
-40	2.22	2.22	2.21	2.20	2.20	2.20	2.19	2.18	2.17	2.17

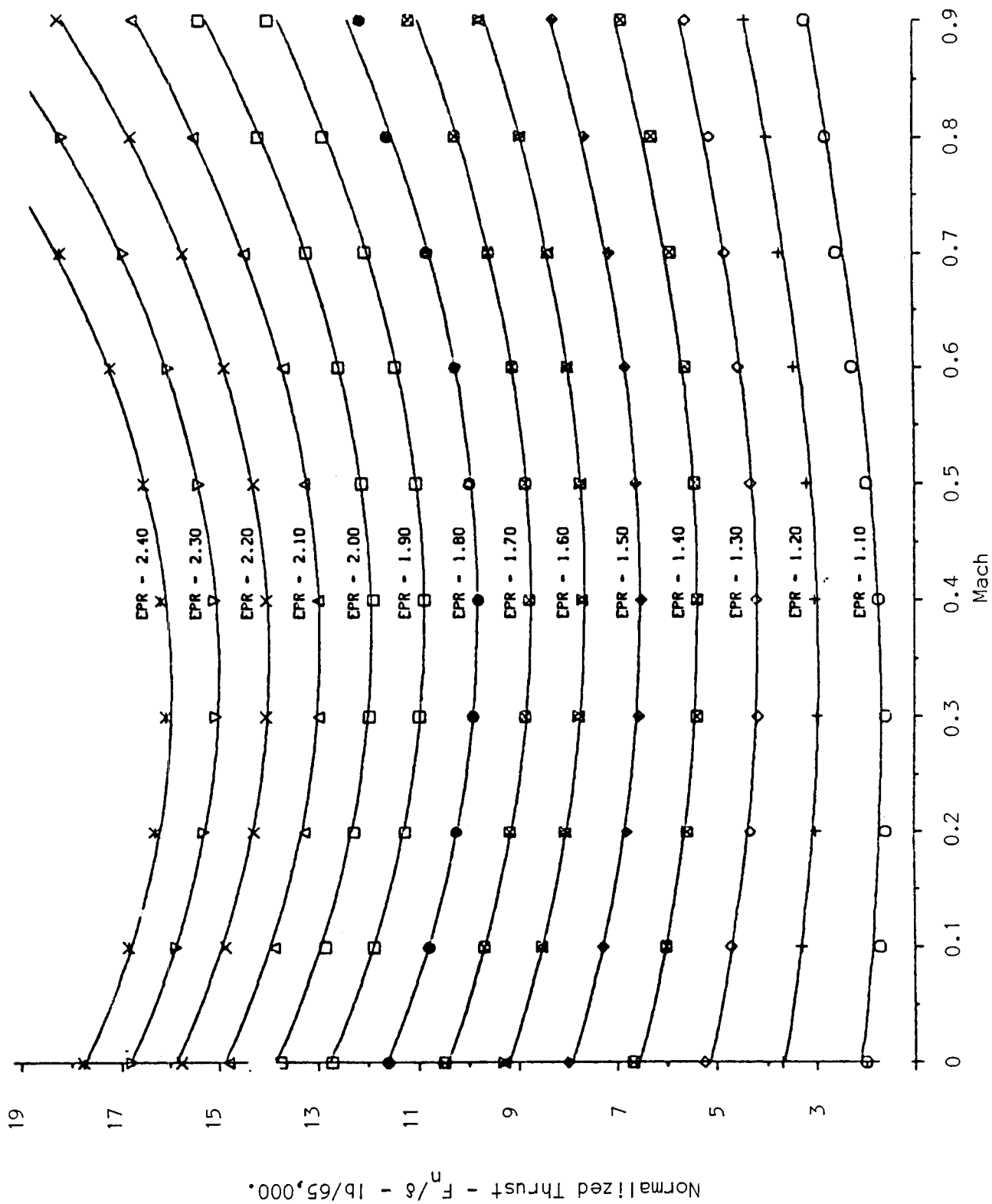


Figure C-3. Thrust Polynomial Fit vs Mach and EPR for Center Engine

ORIGINAL PAGE IS
OF POOR QUALITY

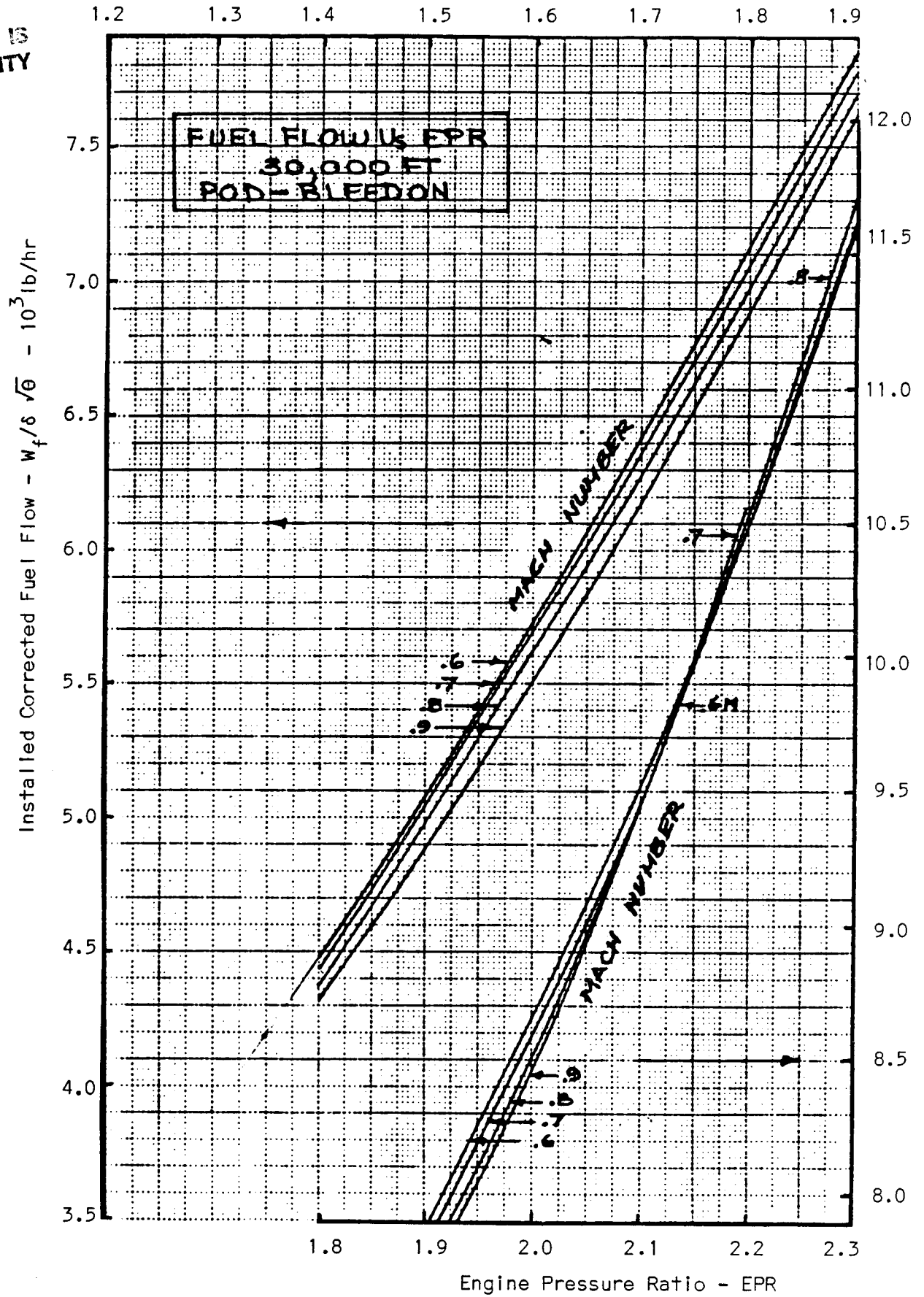


Figure C-4. Example of Fuel Flow Plots

APPENDIX D

WEATHER DATA BASE

EFPLAN uses the new NWS GRIB code weather format. EFPLAN is designed so that other formats could be used, but the GRIB code seems to be the likely candidate for near-future use.

Among the data available within the GRIB code structure, the following data are needed:

P	Pressure level	mb
H	Geopotential Height	m
T	Temperature	deg C
U	Eastward wind component	m/sec
V	Northward wind component	m/sec

The above data should be indexed according to latitude, longitude (over the contiguous US) and pressure level. The time dimension, corresponding to various forecast periods, can be taken care of via a corrected file. The above data should be organized in a file so that it conforms to the following COMMON definition:

```
COMMON/WXDATA/"header and file information",  
H(964,10), T(964,10), U(964,10), V(964,10), R(964,10).
```

Note that the pressure levels are imbedded in the second index. The first index signifies the 1.25 x 2.5 deg grid location over the contiguous U. S. according to the Product Definition Block (PDB) Byte No. 7 = 50. The following two tables define the indicies I and L of the above arrays (i.e., H(I,L)):

Table D-1. Definition of First Index I in Array A(I,L).

I	Latitude (deg North)	Longitude (deg West)
1 - 22 23 - 44 45 - 66 67 - 88	20 21.25 22.5 23.75	122.5 - 70 (every 2.5 deg)
89 - 112 113 - 136 137 - 160 161 - 184	25 26.25 27.5 28.75	125 - 67.5
185 - 210 211 - 236 237 - 262 263 - 288	30 31.25 32.5 33.75	127 - 65.0
289 - 316 317 - 344 345 - 372 373 - 400	35 36.25 37.5 38.75	130.0 - 62.5
401 - 430 431 - 460 461 - 490 491 - 520	40 41.25 42.5 43.75	132.5 - 60.0
521 - 552 553 - 584 585 - 616 617 - 648	45 46.25 47.5 48.75	135.0 - 57.5
649 - 682 683 - 716 717 - 750 751 - 784	50 51.25 52.5 53.75	137.5 - 55.0
785 - 820 821 - 856 857 - 892 893 - 928 929 - 964	55 56.25 57.5 58.75 60.0	140.0 - 52.5

Table D-2. Definition of Second Index L in Array A(I,L).

L	Pressure Level (mb)	Pressure Altitude (ft)
1	1000	371
2	850	4784
3	700	9884
4	500	18295
5	400	23586
6	300	30087
7	250	34010
8	200	38677
9	150	44667
10	100	53097

Figure D-1 is a sketch showing the horizontal layout of the latitude-longitude points representing the index L in the data arrays H, T, U, V and R. The EFPLAN User is responsible for creating and organizing the weather data file system following the outline given above.

A GRIB code practice tape is available to develop the pre-processing utility programs to strip and dump the data into the file system. This process should be automatic to the extent that the GRIB code will be received by the User's equipment every twelve hours. That is, the following sequence of events must be performed automatically and reliably by the User's computer (eventually, anyway):

- (a) open the NWS (or equivalent) digital data line;
- (b) store the incoming bit streams in a temporary buffer;
- (c) close the data line; and
- (d) strip, decode and store the weather data in the file system.

It is recommended that the User investigate providing high speed map or chart plotting capability as part of the output. This requires obtaining digital data bases for

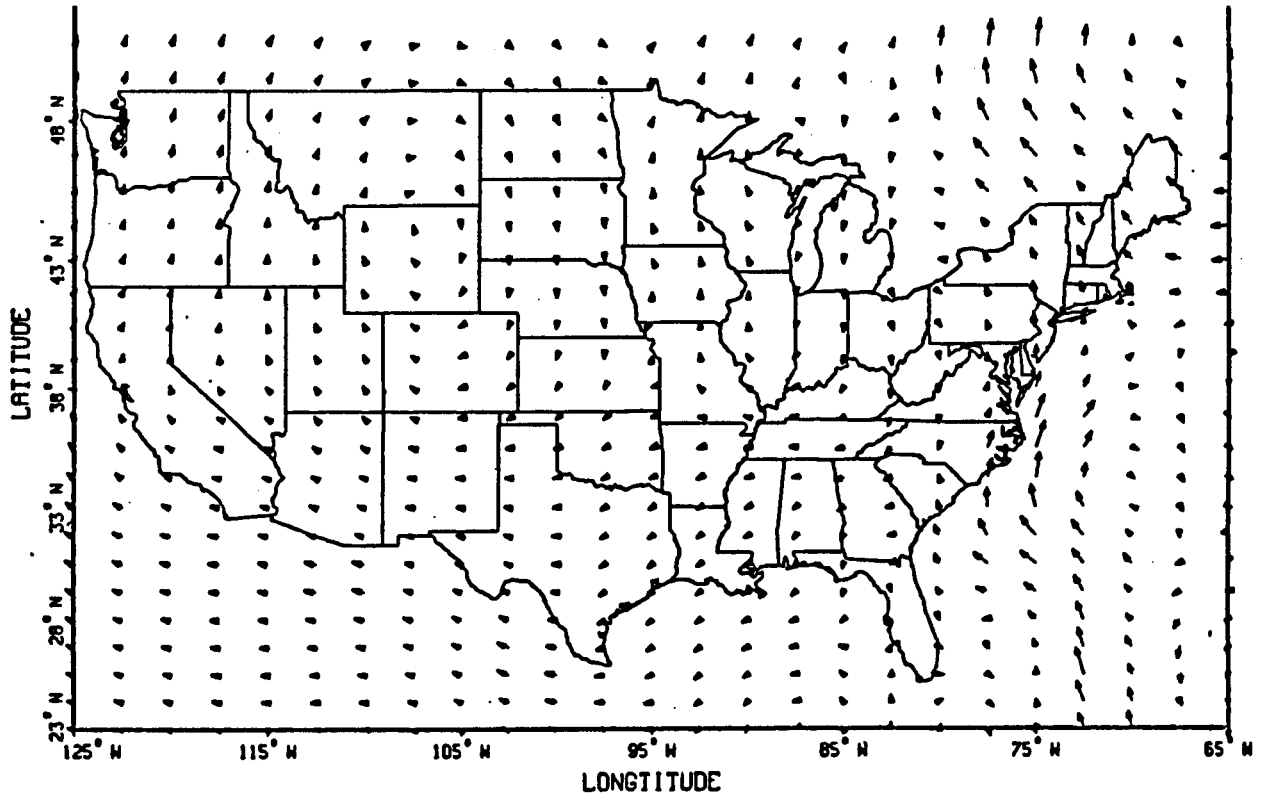
- (a) true north magnetic deviation; and
- (b) political border (state boundaries) and terrain contour information.

Combining the weather and flight routes on a terrain map could be a valuable way of presenting the flight plan output.

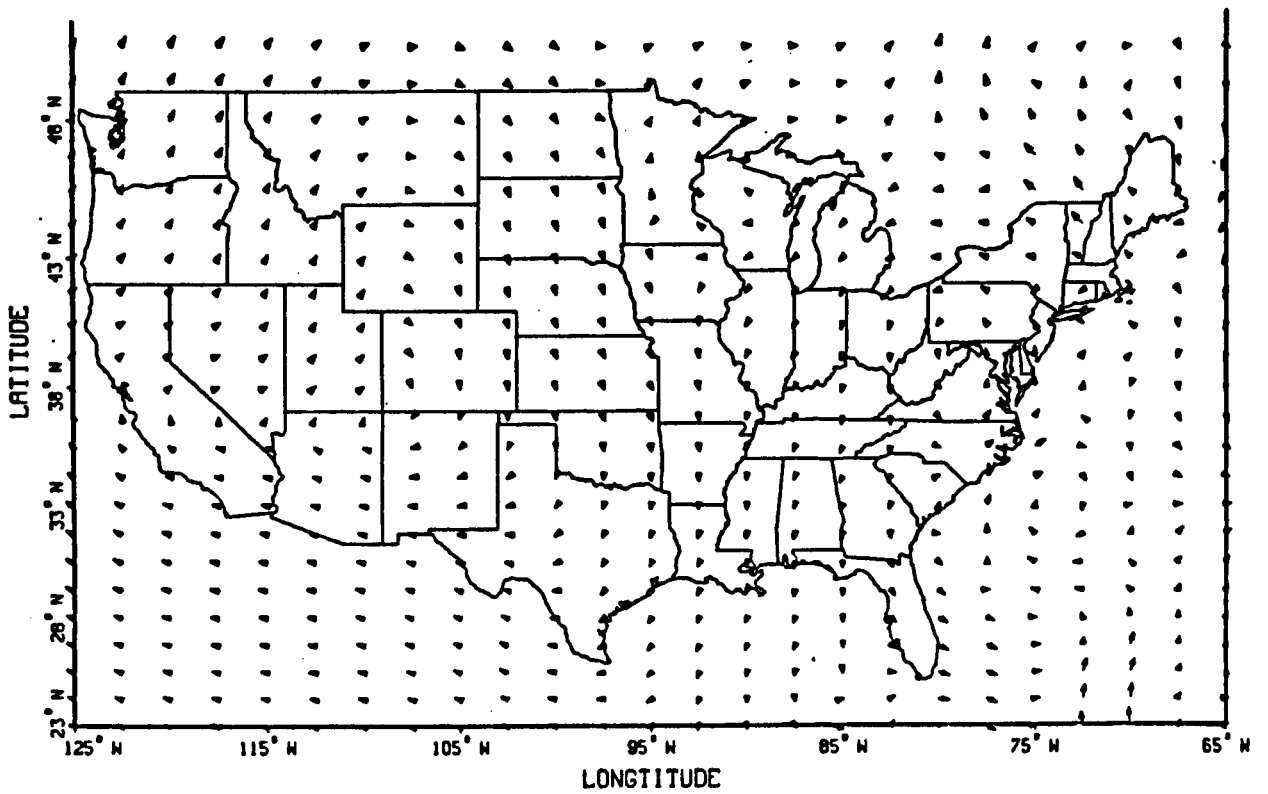
There is some pre-flight plan computation processing of the GRIB code data that is required for efficient usage. This includes:

- (a) transforming the data into the FAA mandatory flight levels rather than pressure levels as in the raw GRIB code format; and
- (b) converting the data units to conform to conventional aeronautical usage compatible with the AFPS.

The sample weather data used to develop EFPLAN was provided by Global Weather Dynamics, Inc. of Monterey, CA., courtesy of Pacific Southwest Airlines. These data represent a January 1985 day. The following plots are wind vectors at each of the ten pressure levels for a single forecast.

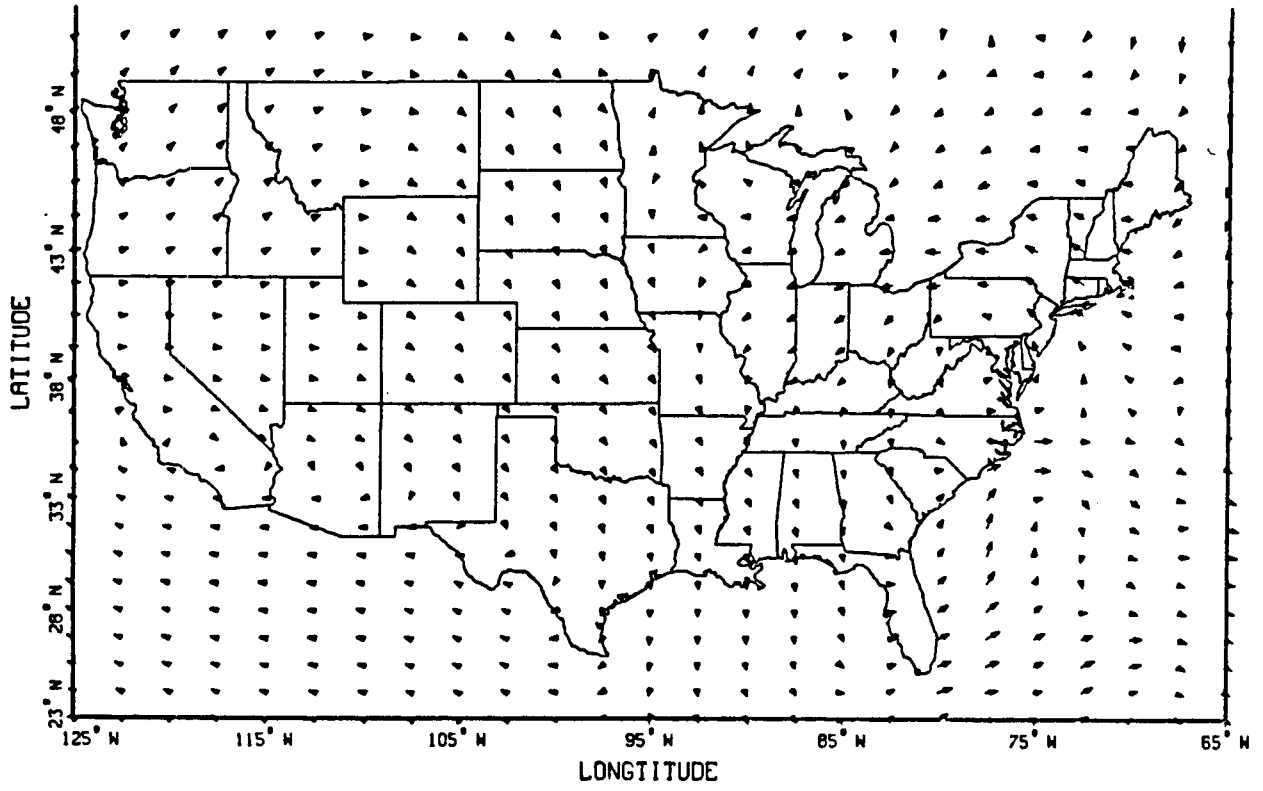


a. 1000 mb

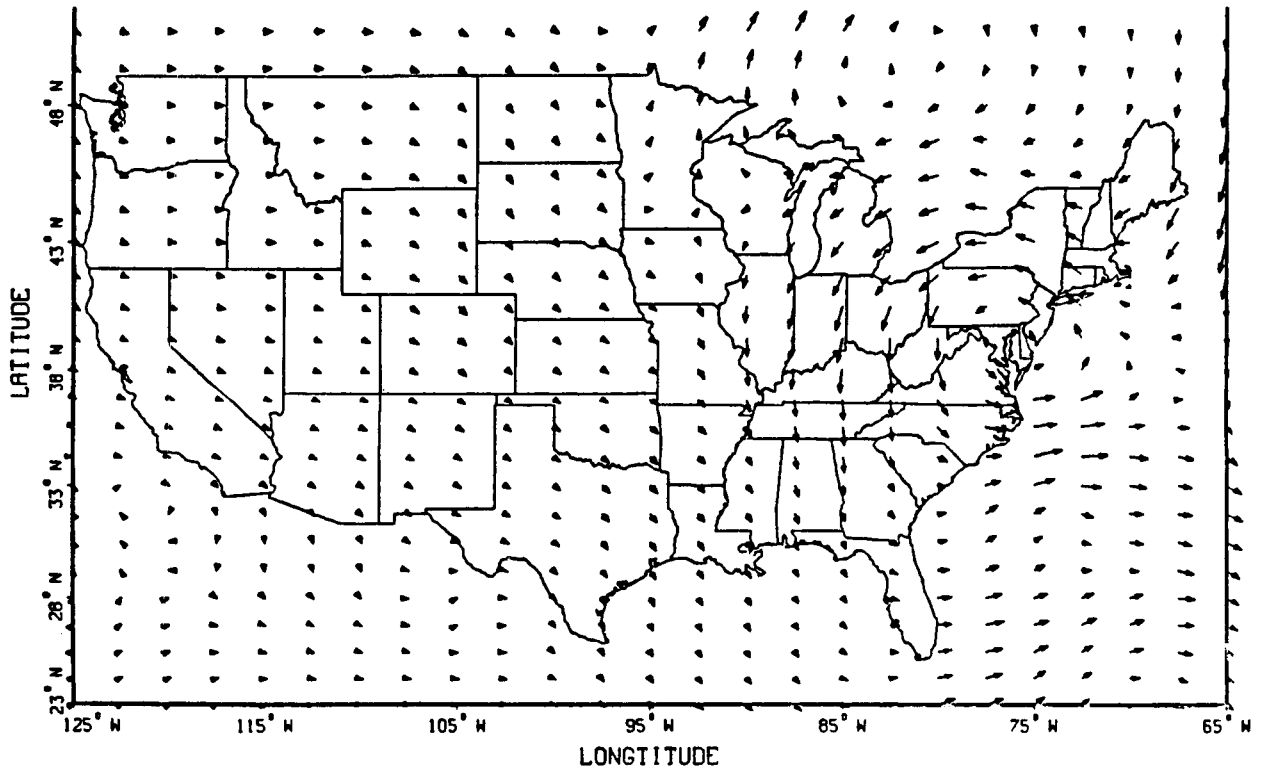


b. 850 mb

Figure D-2. January 14, 1985 Wind Forecast

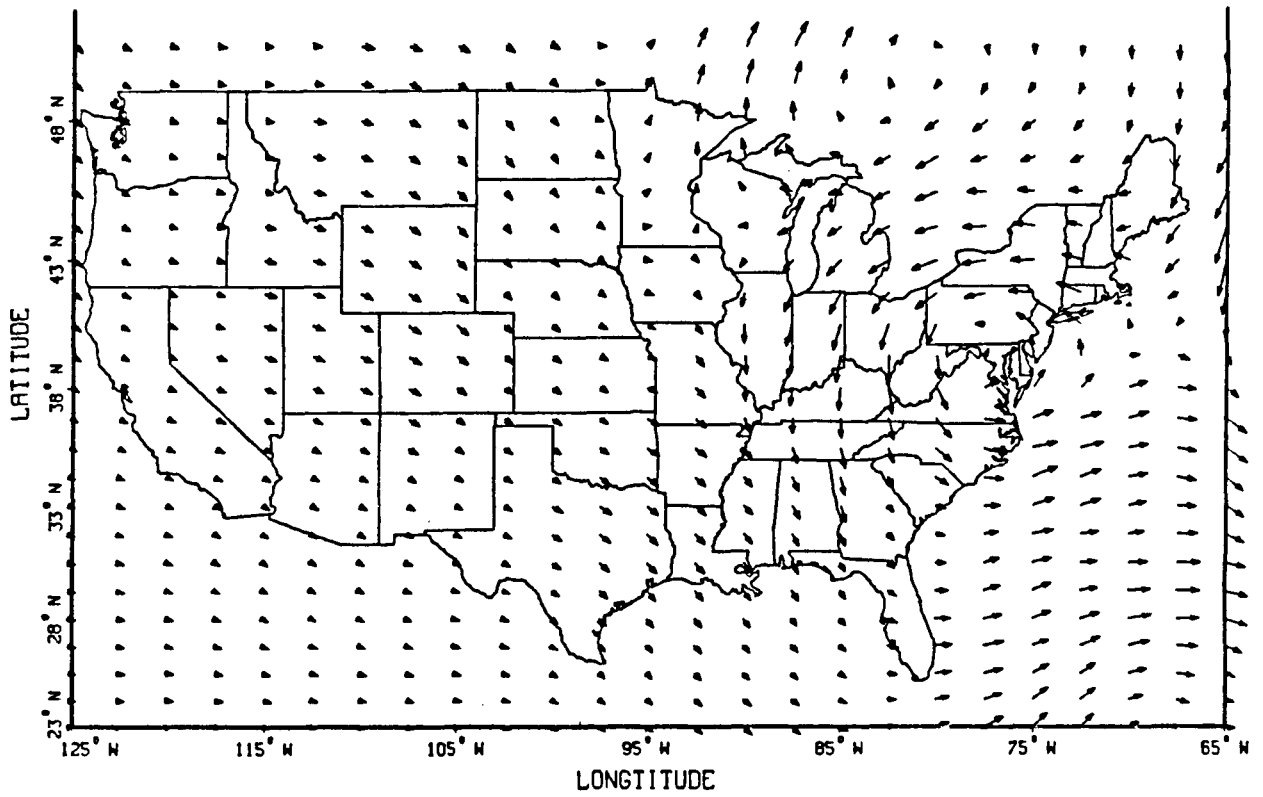


c. 700 mb

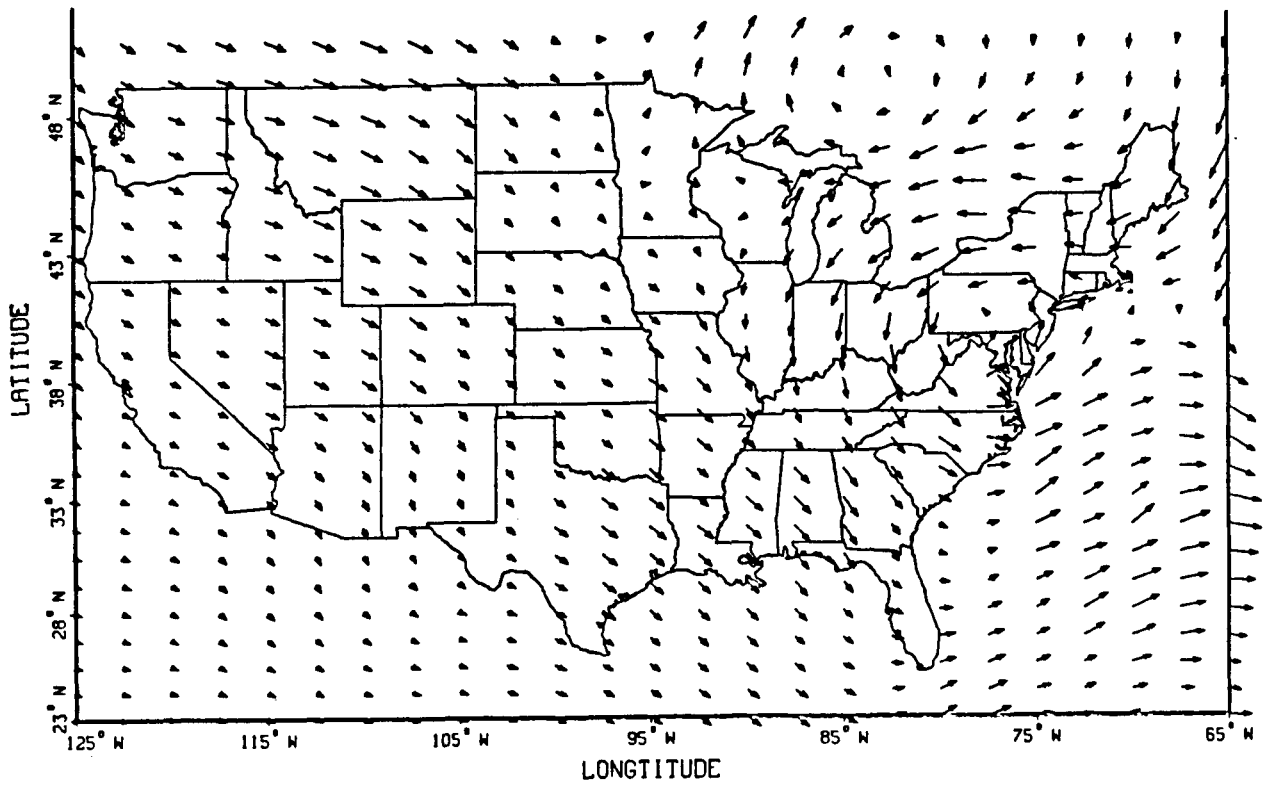


d. 500 mb

Figure D-2. Continued

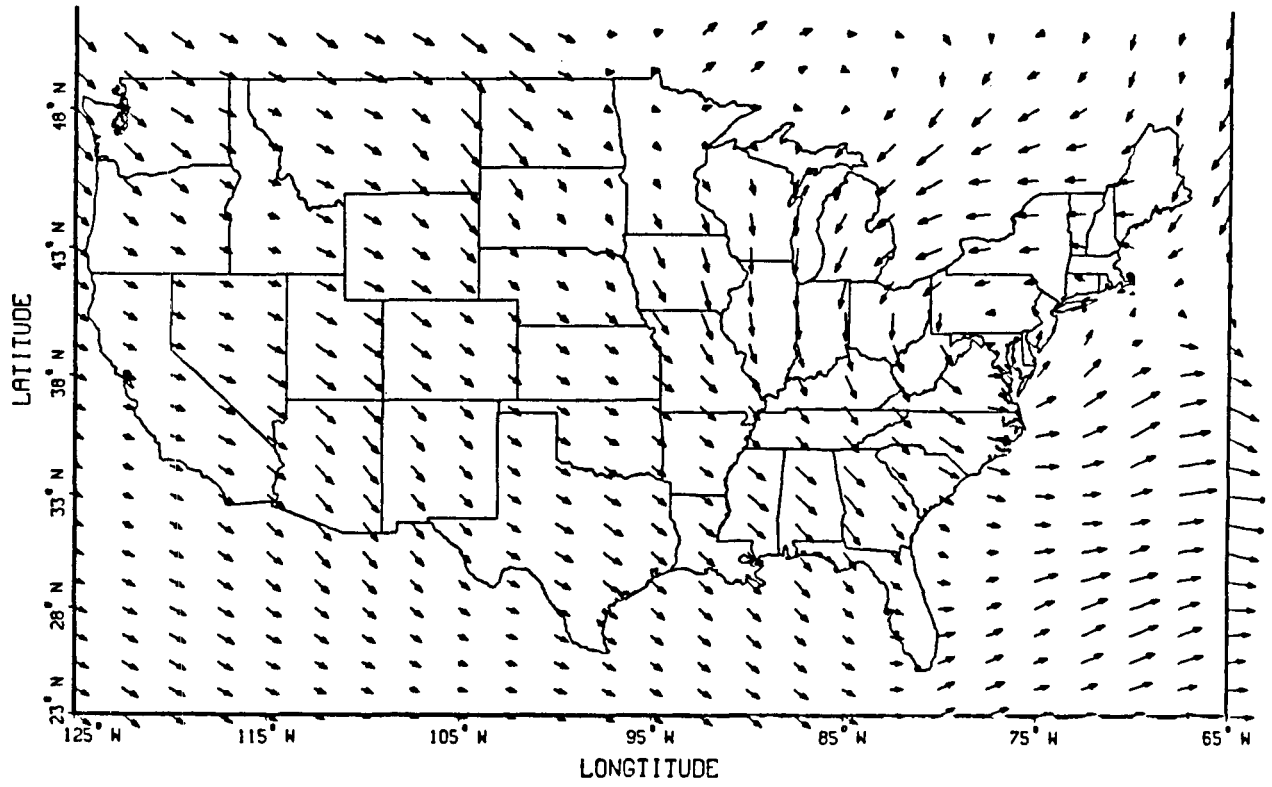


e. 400 mb

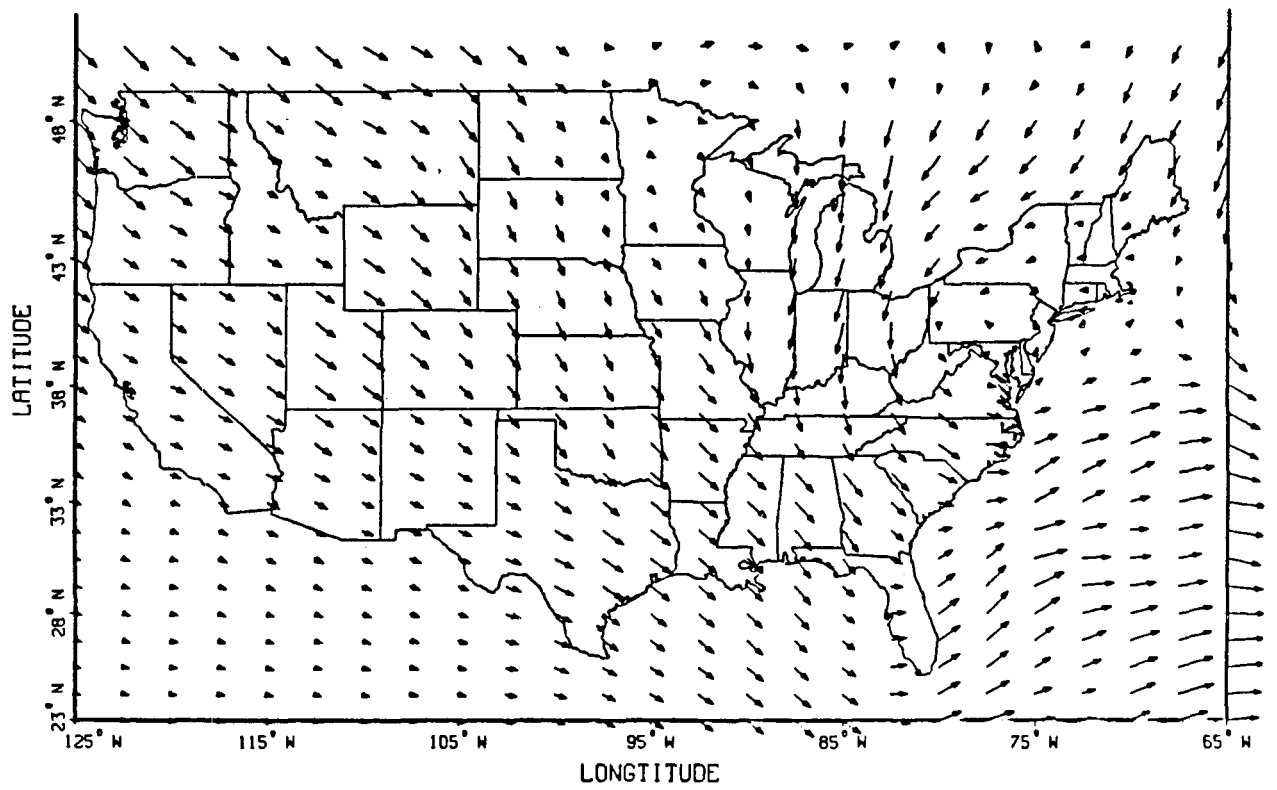


f. 300 mb

Figure D-2. Continued

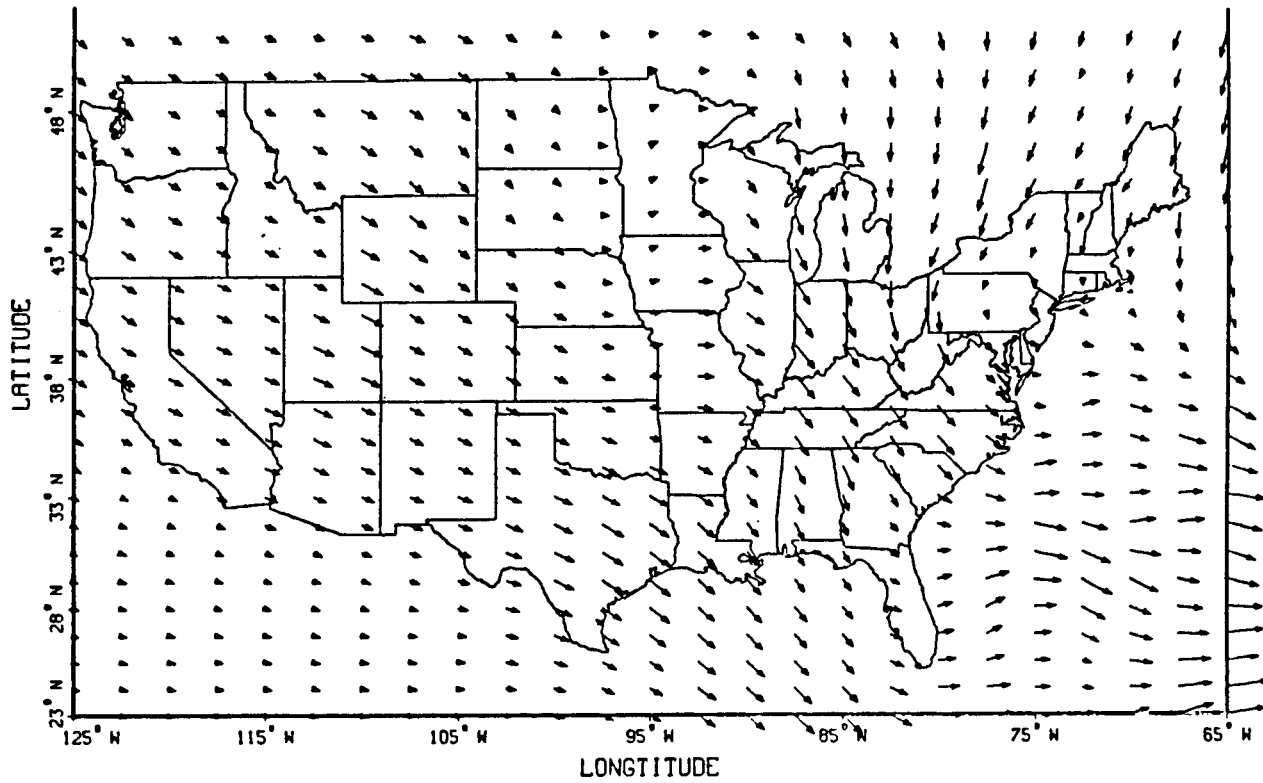


g. 250 mb

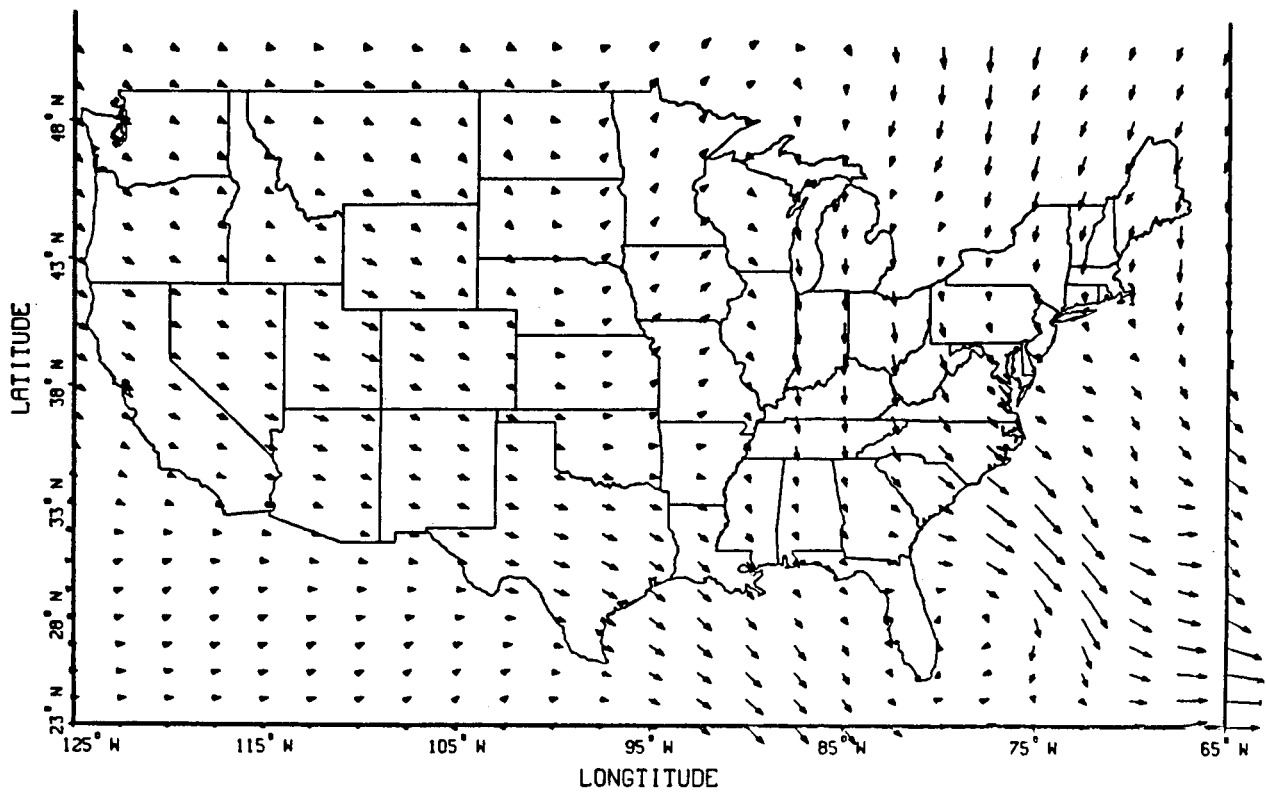


h. 200 mb

Figure D-2. Continued

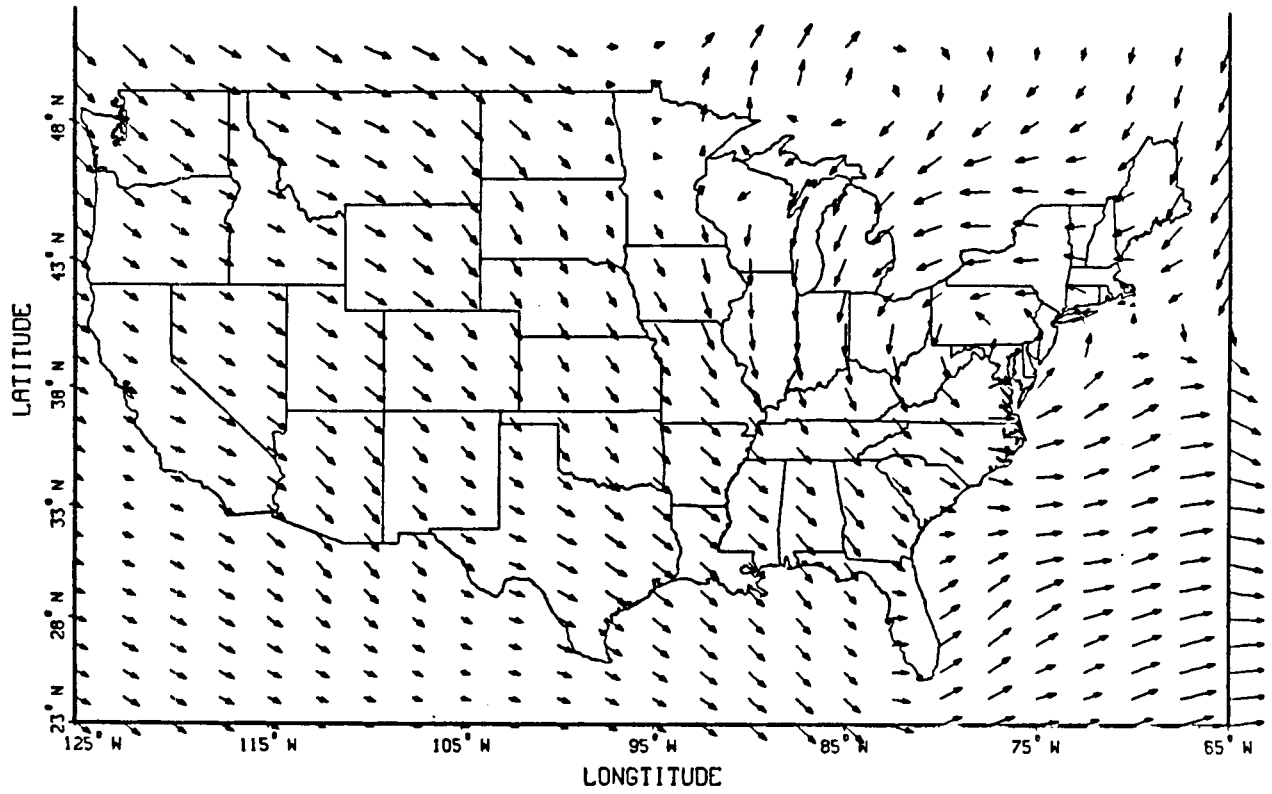


i. 150 mb



j. 100 mb

Figure D-2. Continued



k. Maximum Winds

Figure D-2. Continued

APPENDIX E

AIRCRAFT PERFORMANCE MODELING FOR FLIGHT PLANNING

This appendix presents tables and data used for representing the aircraft performance data base used in the EFPLAN flight planning program. These data are based on the performance characteristics used to model the NASA TSRV aircraft (B737-100). They are consistent with the aircraft model used in the OPTIM program.

Fuel Rate for Cruise

When computing the performance of the aircraft over a cruise segment, we wish to find the cruise speed V_a which minimizes the function

$$J = (C_f \dot{w} + C_t) / V_g \quad (E.1)$$

where C_f and C_t are the cost of fuel and cost of time, \dot{w} is the fuel rate, and V_g is the ground speed which is a function of cruise speed V_a . This requires that we find the cruise trim condition so that thrust and angle-of-attack are adjusted to balance the forces of drag and weight of the aircraft for straight, level, unaccelerated flight.

The process of trimming the aircraft mathematically requires an iterative solution to transcendental equations. Because there are so many segments over which to optimize cruise speed during the flight planning process of generating the reference profile, the trimming process would consume a very large amount of computer time. Thus, it was considered important to find a method to allow rapid solution to Eq. (E.1), without the trimming process being involved. This was done by devising a utility program that found a large spectrum of possible cruise trim conditions and generating an associated table of fuel rate \dot{w} as a function of the

PRECEDING PAGE BLANK NOT FILMED

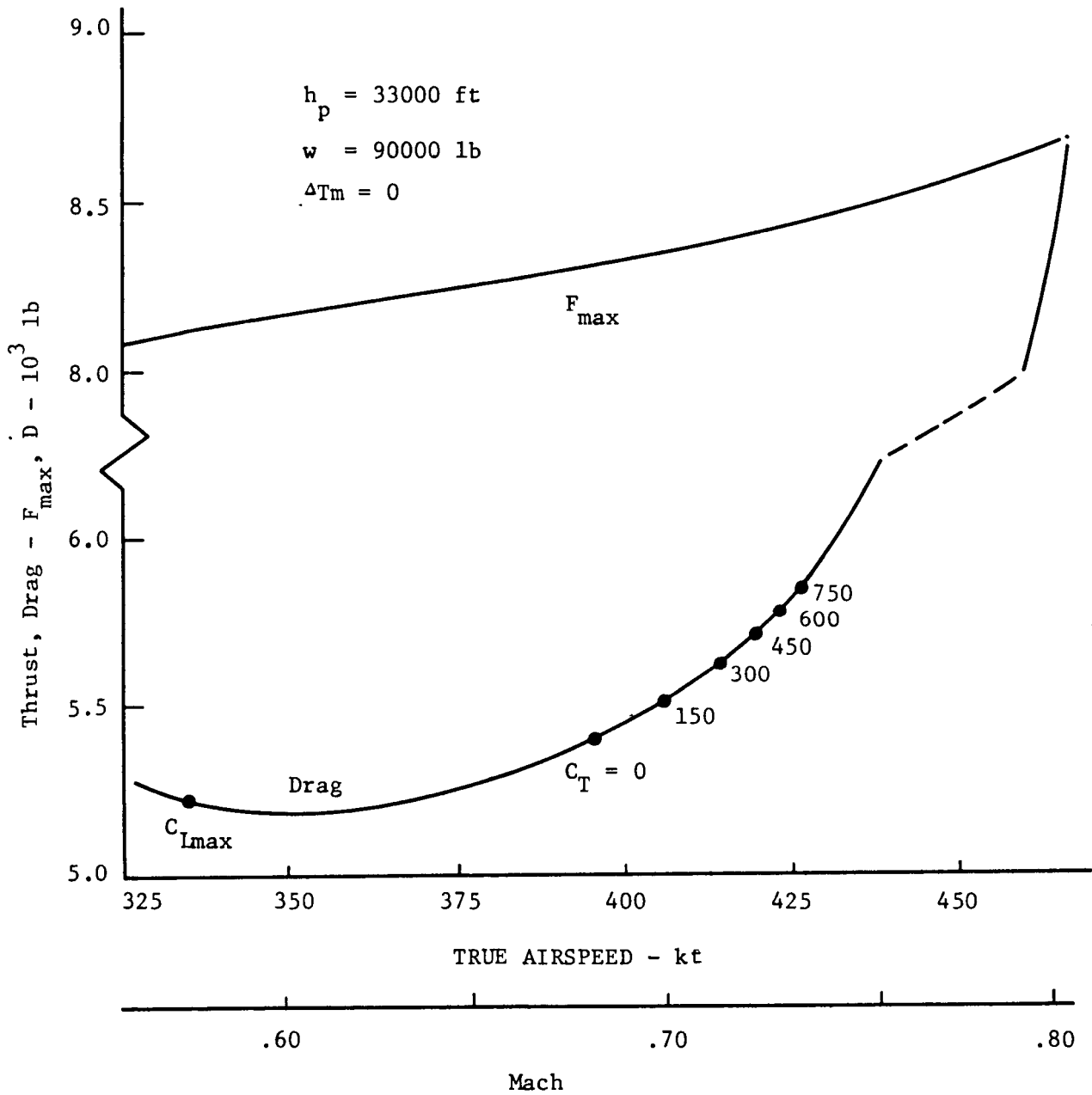


Figure E-1. Thrust Minus Drag Margin as a Function of Airspeed at 33000 ft.

important independent variables that govern the cruise trim condition. Then this table could be used for evaluating different cruise speeds in solving Eq. (E.1).

The independent variables that affect the cruise condition are aircraft weight, pressure altitude, temperature deviation from standard day, and cruise speed. By convention, speed is referenced as Mach number.

It was first required to specify the limits of the four independent variables that govern the fuel rate table. The TSRV has an upper altitude limit of 35000 ft, and we assume that all cruising will be done above 10000 ft. Thus, the altitude is assumed to vary from 10000 to 35000 ft in steps of 1000 ft.

The maximum inflight weight of the TSRV aircraft is 97000 lb, and the zero fuel weight, with no payload is 66500 lb. We assumed there would always be some payload, so the minimum cruise weight was assumed to be 70000 lb. The maximum weight was set to 105000 lb, to allow for expanded capability of the 737. The weight increments were set to 5000 lb.

The temperature was assumed to deviate by as much as 20 deg from Standard Day, although extrapolation is allowed for this variable. Thus, we use -20, 0 and +20 deg for the temperature variable.

For any cruise point, the altitude, weight, and temperature deviation will be known at that point. The main variable that is unknown is the Mach number. Figure E-1 shows how the maximum cruise thrust and aircraft drag vary with Mach number for a 90000 lb aircraft trimmed at 33000 ft on a standard day. This plot shows that there is excess thrust available up to a Mach number of about 0.80. Airspeed can go as low as the point of C_{Lmax} or about Mach 0.57. From a practical point, we would not want to cruise normally below the point where the cost of time is set to zero; this

is the minimum fuel or "long range cruise" point. In Fig. E-1, the trim points for minimizing Eq. (E.1) with cost of time varying from 0 to \$750/hr are indicated. Thus, for a windless day, Mach number at 33000 ft and 90000 lb would typically vary between 0.68 and 0.73.

However, in setting the Mach number bounds, we need to consider the effect of the wind on the optimum set point. Figure E-2 is a plot of optimum Mach settings for cost of time of 0 and \$750/hr. Parameters are aircraft weight of 70000 and 105000 lb, and maximum wind of 200 kt in both head and tail directions. The solid lines are the trimmed, optimized Mach settings for no wind. The dashed lines show the movement of these set points to the right and left because of wind. The dashed lines on the left are for the 200 kt tail wind with zero cost of time. The single dashed line on the right is for both weights, with \$750 cost of time and with the 200 kt head wind. Also shown are two dot-dash lines on the left which are the boundaries for the two weights with C_{Lmax} . Note that the 70000 lb aircraft would have an optimum Mach setting for zero cost of time that is slower than is feasible to fly with the 105000 lb aircraft. We realize, of course, that a 200 kt wind is an extreme case.

By examining Fig. E-2, it is apparent that the upper boundary on Mach number can be a constant 0.77. This is represented by the double solid vertical line on the right. However, the lower bound on Mach number must be a function of both altitude and aircraft weight. The boundary we chose is represented by the two double solid lines on the left that shows the table boundary increasing with increasing weight and altitude. Thus, our minimum Mach boundary was chosen to start at 0.35 at 10000 ft for the 70000 lb aircraft. The table boundary increases 0.01 Mach for every 1000 ft increase in altitude and 0.01 Mach for every 5000 lb increase in weight. We used our trim utility program to generate fuel rate data for each of the points between the two Mach boundaries, with Mach number varying in 0.01 steps.

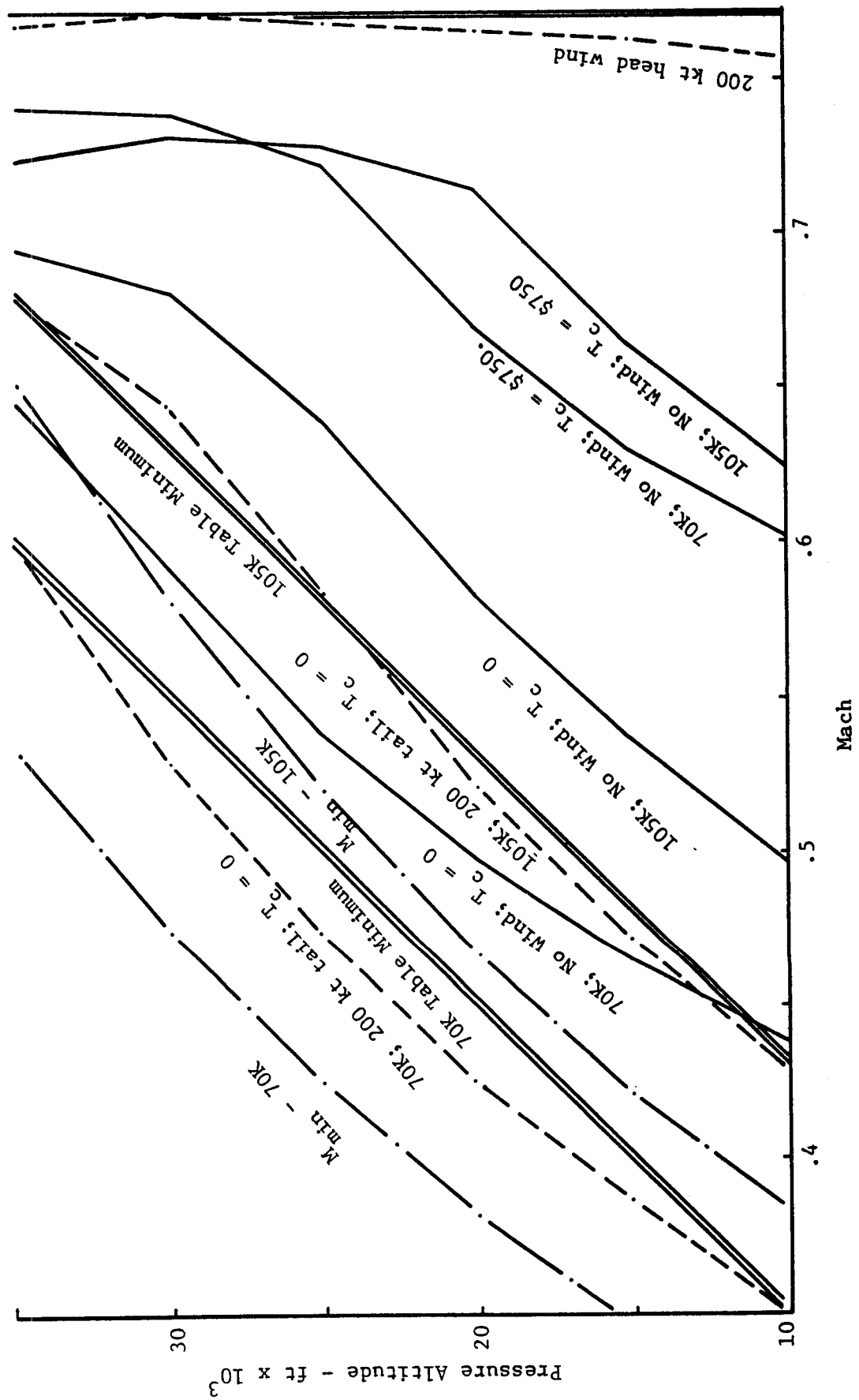


Figure E-2. Mach Limits to Define Fuel Rate Table Boundaries.

Figure E-3 shows a plot of typical fuel rates from the fuel rate table for different table parameter values. In Fig. E-3, the solid lines represent +20 deg, the dashed lines represent 0 deg, and the dot-dash lines represent -20 deg. Weights of 75000, 85000, and 95000 lb are indicated. Also, pressure altitudes of 31, 33, and 35000 ft are shown. As can be seen, the fuel rate increases with Mach number, weight, and temperature. It decreases with altitude. These curves are quite smooth, and the density of the table ultimately required will be determined by testing the EFPLAN program.

Step Climb and Descent Fuel and Time

In OPTIM, the requirements, in terms of time and fuel, to make a step climb in altitude are computed by simulating the 4000 ft climb in 500 ft steps. This is accurate but costly, in terms of computation time. Again, in the flight planning process, we need to evaluate both the step climb and the step descent for each segment in our grid network. Clearly, the simulation technique of OPTIM would be too time consuming for determining these incremental costs. Thus, it was decided to generate a table of step climb and step descent incremental fuel and time requirements for a spectrum of cruise conditions. With the table dense enough, the fuel and time increments could be interpolated from the table without use of simulation.

The values for the step climb and step descent incremental time and fuel were generated by another utility program that uses the same technique as OPTIM for generating a step climb profile. For the step, it is assumed that Mach number is held constant. For step climb, thrust is set at the maximum climb thrust. For step descent, thrust is set at idle.

For both climb and descent steps, the first table was generated for today's ATC altitude constraints. That is, the step climbs were assumed to be between flight levels 250 and 270, 270 and 290, 290 and 330, 260 and 280, 280 and 310, and 310 and 350. Step descents were with the same altitude pairs but in reverse order. The second table was generated for future altitude separations: 290 to 310, 310 to 330, 330 to 350, 280 to 300, 300 to 320, and 320 to 340.

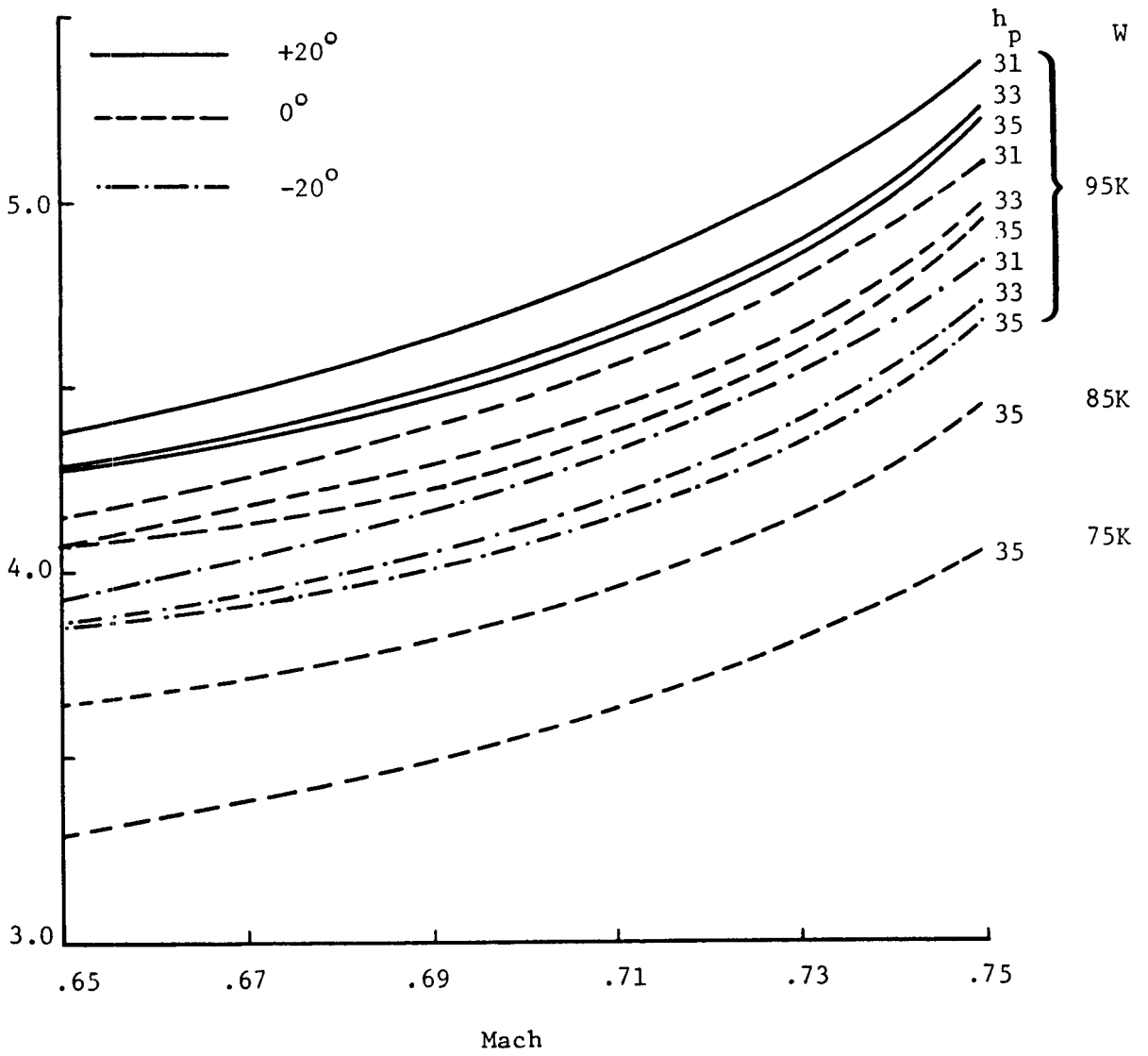


Figure E-3. Trimmed Cruise Fuel Rate as a Function of Mach, Altitude, Weight and Temperature.

Figure E-4 shows plots of time and fuel requirements for step climbs between 31000 and 35000 ft. The solid lines are for the aircraft weighing 85000 lb, the dashed line is for 73000 lb, and the dot-dash line is for 97000 lb. The effect of temperature variation is shown for 85000 lb. These curves are smooth with a rapid rise at higher Mach values. They lend themselves well for interpolation on weight and temperature.

Figure E-5 is a plot of fuel and time requirements for step descent from 35000 ft to 31000 ft. These are simpler than the step climb curves, and temperature deviation proved to not be important. Both fuel and time requirements decrease with increase in Mach number.

The values of fuel and time for step climb and step descent are used to adjust the segment cost as explained in Chapter III. This allows the program to assess whether it is beneficial to make a step change during any one segment of cruise.

Cruise-Climb Fuel Rate

Cruise-climb is the condition where the aircraft has the freedom to gradually change its cruise altitude as the winds change and fuel is burned. For a windless day, the optimum cruise altitude will gradually rise as fuel is burned and the aircraft weight decreases.

In flight planning, it is assumed that for cruise-climb, the altitude changes over a given range will vary with very small flight path angles. Thus, it is assumed that the time required to travel a given air distance does not vary with altitude change for a constant airspeed. However, fuel rate will vary because of gradual climb or descent, as compared to level flight. This change in fuel rate is primarily a function of altitude level, aircraft weight, and flight path angle.

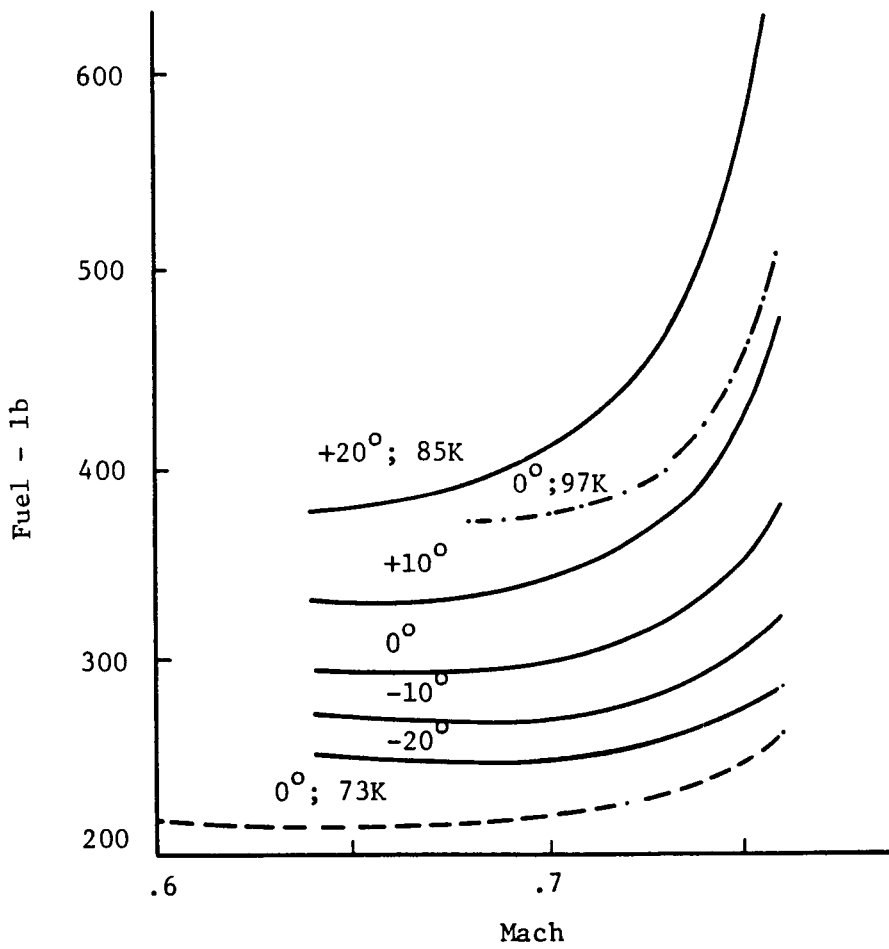
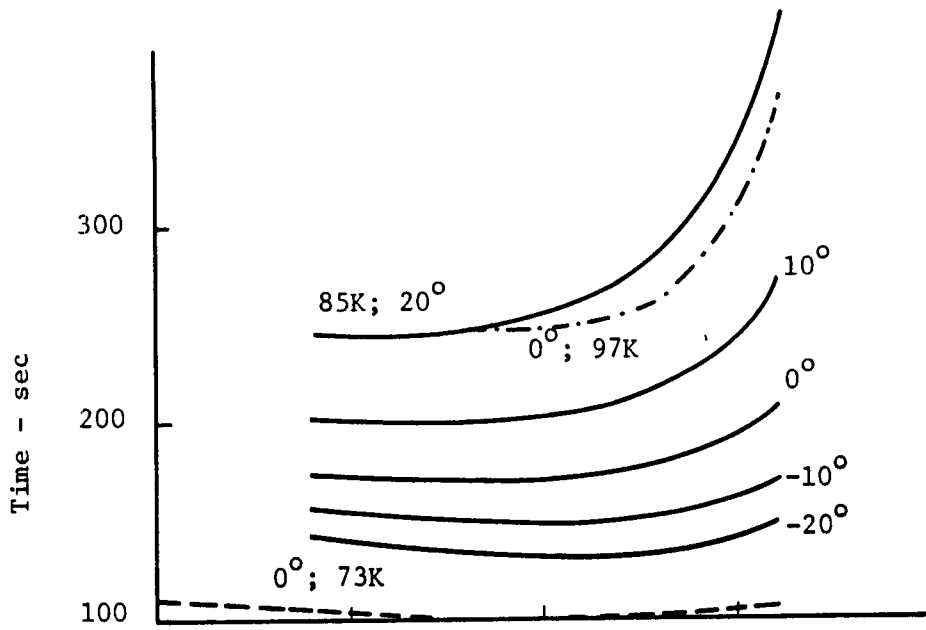


Figure E-4. Step Climb Fuel and Time from 31000 to 35000 ft.

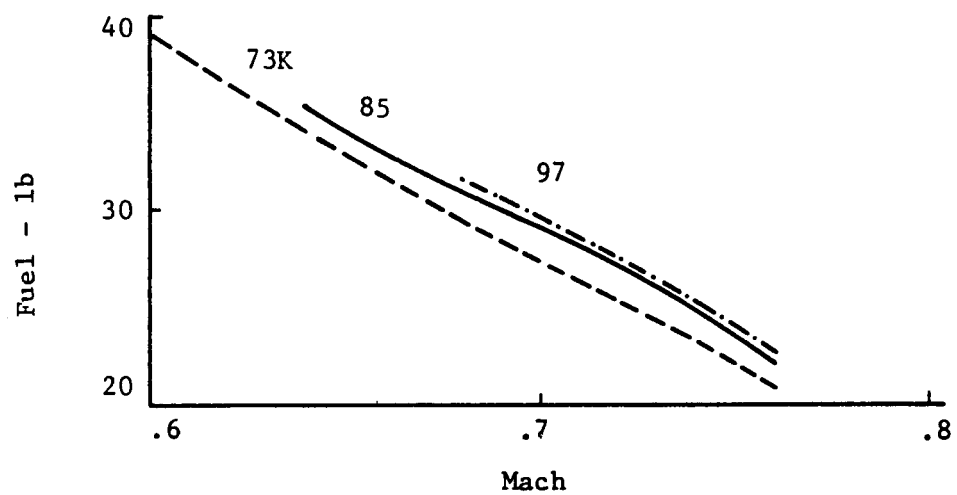
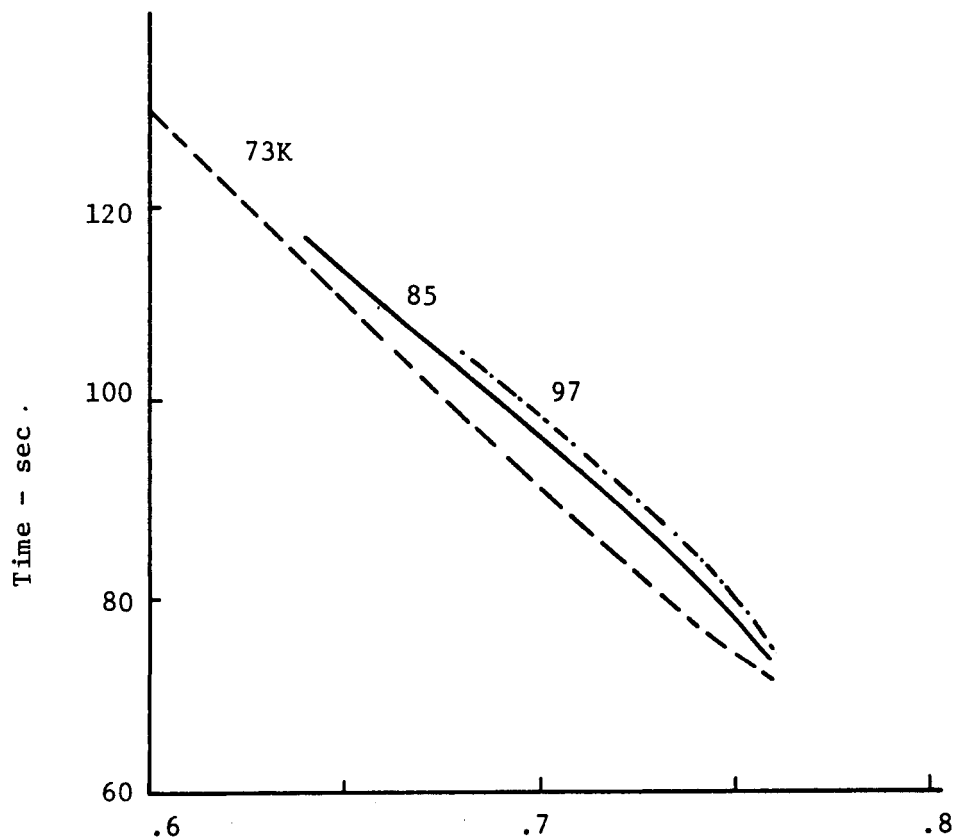


Figure E-5. Step Descent Fuel and Time from 35000 to 31000 ft.

A convenient way to model the cruise climb requirements is to multiply the fuel rate requirements for level cruise flight by a scale factor. A table of these scale factors was generated for altitudes between 25000 and 35000 ft. A plot of how the scale factor changes with Mach number and weight is presented as Fig. E-6.

In Fig. E-6, the aircraft weight is 73, 85, and 97000 lb. The altitudes are 25 and 35000 ft. Flight path angles of +0.4 to -0.4 deg are shown. A 0.4 deg flight path angle represents about a 1000 ft climb over 25 nmi of range.

In EFPLAN, it is currently assumed that for cruise climb, the aircraft would nominally vary its altitude about 1000 ft for a 100 nmi range segment. However, this 1000 ft increment can be adjusted in future studies. The scale factor table is large enough to encompass any feasible band of altitudes that may be tested.

Reserve Fuel

In computing the nominal aircraft landing weight to begin the flight planning process, it is required that an amount of fuel be added to the landing weight as a reserve for ATC holding. The Federal Aviation Regulations specify that for domestic travel, the reserve fuel be equal to what is required for 45 minutes of holding at aircraft empty weight. The holding altitude is not specified.

To include a reserve fuel allowance in EFPLAN, we assumed that the aircraft would hold at 25000 ft with the zero fuel landing weight. The speed would be held at the point of C_{Lmax} . A utility program was used to create a table of holding speed fuel rates with these assumptions. The weight was assumed to vary from 65000 to 90000 lb. The temperature deviation was assumed to vary from -20 to +20 deg. Figure E-7 is a plot of the fuel rate from this table.

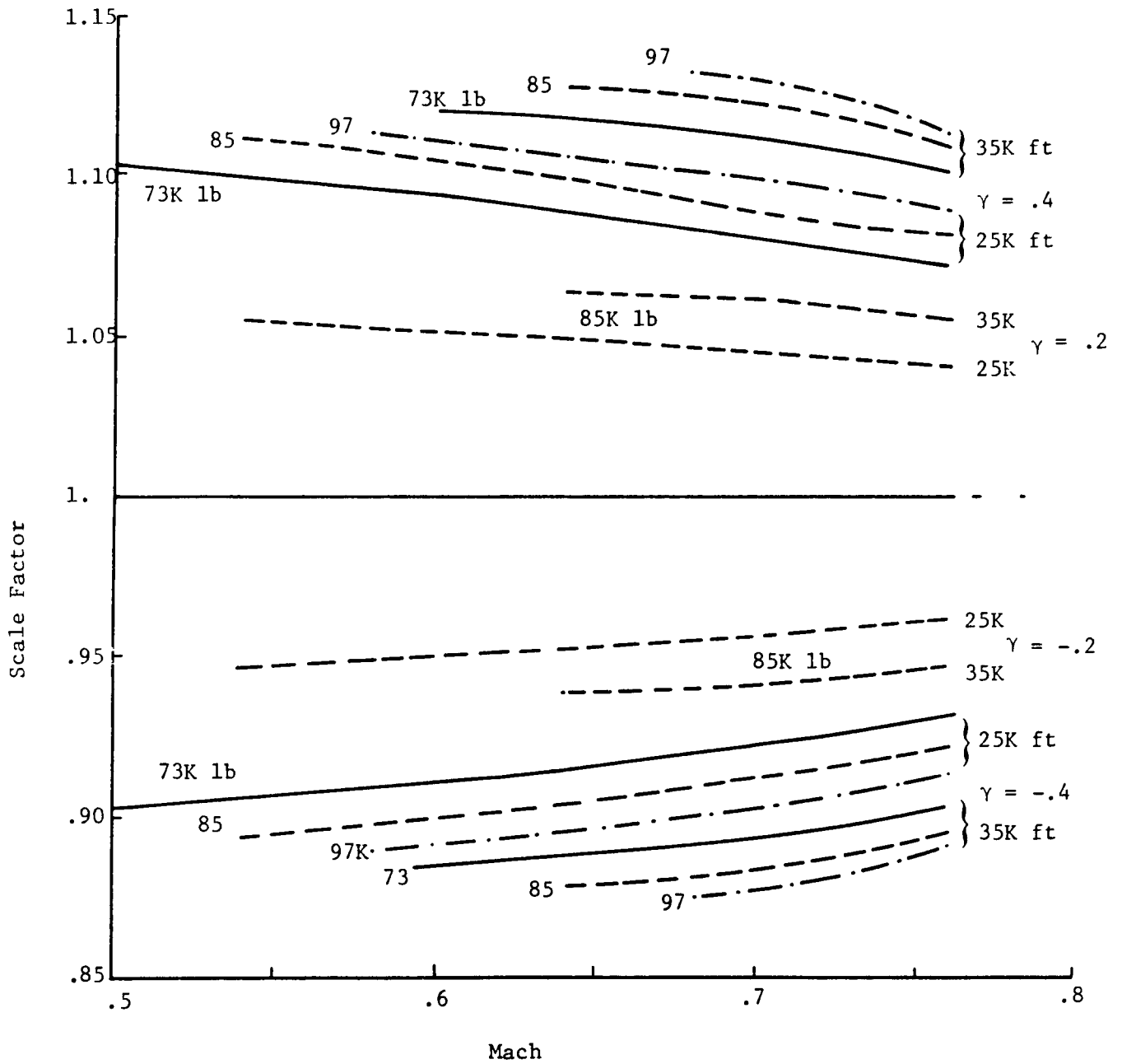


Figure E-6. Scale Factor on Fuel Flow for Cruise-Climb.

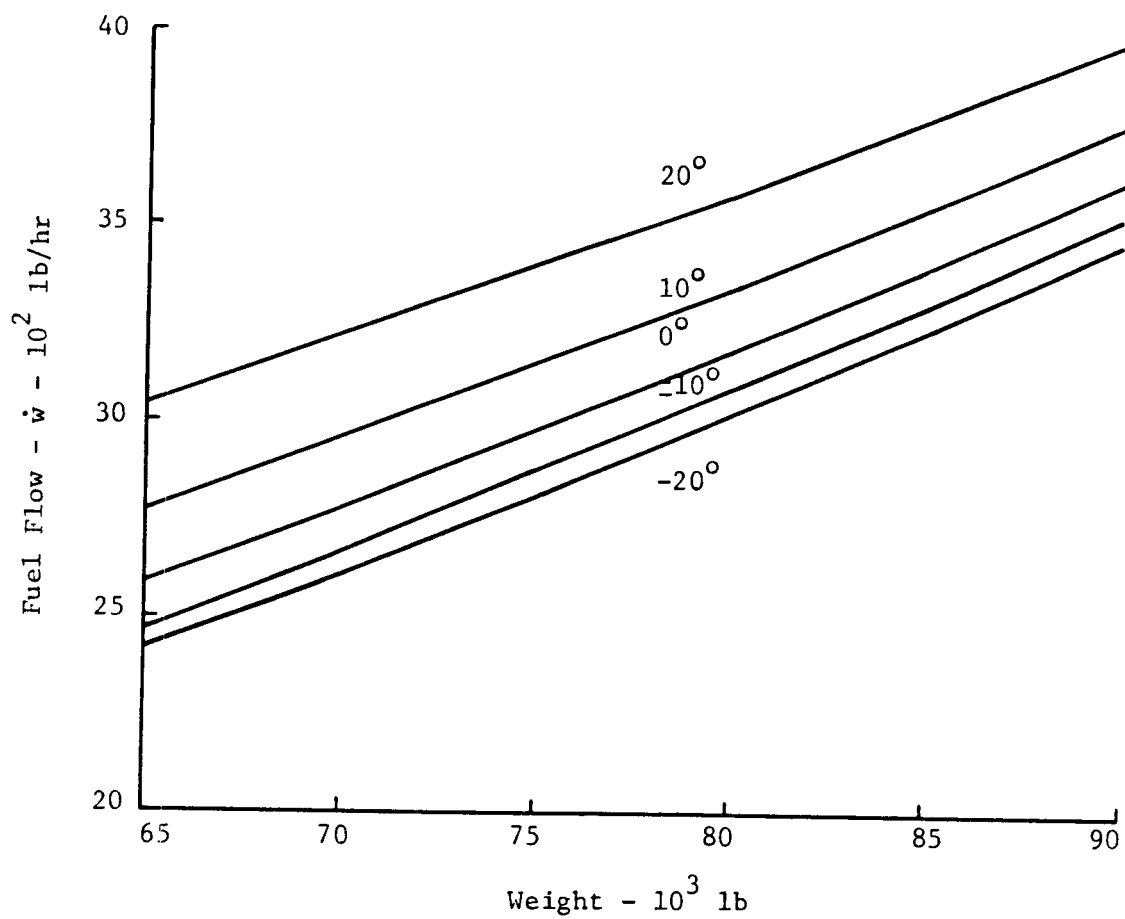


Figure E-7. Holding Speed Fuel Rate at 25000 ft.

An alternate holding pattern fuel rate table is presented as Table E-1. This gives fuel rate as a function of weight and altitude for an assumed holding speed (generally, 210 kt, indicated airspeed). It specifies that the rates shown be adjusted 1% for each 5 deg change in temperature. The results of this table are more conservative than those of Fig. E-7. However, every user tends to have their own way of computing these reserve requirements so our current table seems adequate for program testing purposes.

Computation of Landing Weight

In beginning to generate the reference profile for flight planning, we begin with the landing weight that we would like to achieve. The constraint numbers for computing this weight are shown in Table E-2. We begin with the zero fuel weight of 66500 lb. We next compute the payload consisting of passenger and cargo weight. This payload weight is checked against the payload limit constraint curve shown in Fig. E-8.

For the TSRV aircraft, there is a structural limit on payload of about 20,500 lb. This governs payload limits for short range. For range greater than 700 nmi, the governing limit is the aircraft maximum brake release weight of 97800 lb. Here, as range is increased, the payload must be decreased as fuel is increased. This limit continues until a range of about 2200 nmi is reached. At 2200 nmi, the point is reached where no more fuel can be traded against payload because the tanks are full. Beyond that point, range can be increased slightly to 2250 nmi by offloading more payload.

After the payload has been checked against Fig. E-8, we next add on fuel increments for alternate destination, reserve fuel, and any tankered or ferried fuel we may wish to carry along. The resulting weight must be tested against the maximum landing weight of 89700 lb.

Table E-1. 737 Holding Pattern Fuel Rate Requirements

EPR /
IAS - KTS
FF PER ENGINE - LB/HR
STD. DAY TAT - °C

FUEL FLOW BASED ON ISA
ADJUST FUEL FLOW ± 1%
PER ± 5°C ISA DEVIATION

PRESS ALT	GROSS WEIGHT - 1000 LB										
	115	110	105	100	95	90	85	80	75	70	65
35000		2.04	1.98	1.93	1.88	1.83	1.78	1.74	1.69	1.66	1.62
		228	223	217	211	210	210	210	210	210	210
		2740	2560	2400	2240	2130	2020	1920	1820	1740	1660
		-34	-35	-36	-37	-37	-37	-37	-37	-37	-37
30000	1.84	1.80	1.77	1.73	1.70	1.66	1.62	1.59	1.56	1.53	1.50
	231	226	220	215	210	210	210	210	210	210	210
	2670	2540	2410	2310	2200	2100	2000	1910	1830	1750	1680
	-27	-28	-28	-29	-30	-30	-30	-30	-30	-30	-30
25000	1.68	1.65	1.62	1.59	1.56	1.53	1.50	1.47	1.44	1.42	1.39
	229	224	218	213	210	210	210	210	210	210	210
	2640	2540	2430	2310	2210	2120	2030	1950	1870	1790	1730
	-20	-20	-21	-22	-22	-22	-22	-22	-22	-22	-22
20000	1.55	1.52	1.49	1.47	1.44	1.42	1.39	1.37	1.35	1.33	1.31
	227	222	217	211	210	210	210	210	210	210	210
	2650	2540	2430	2340	2250	2170	2090	2010	1940	1870	1810
	-12	-13	-13	-14	-14	-14	-14	-14	-14	-14	-14
15000	1.43	1.41	1.39	1.37	1.35	1.33	1.31	1.30	1.28	1.26	1.25
	226	221	216	210	210	210	210	210	210	210	210
	2720	2620	2510	2420	2320	2240	2160	2090	2020	1950	1890
	-4	-5	-5	-6	-6	-6	-6	-6	-6	-6	-6
10000	1.35	1.33	1.31	1.30	1.28	1.26	1.25	1.24	1.22	1.21	1.20
	225	220	215	210	210	210	210	210	210	210	210
	2800	2710	2610	2500	2410	2330	2260	2180	2110	2050	2000
	4	4	3	3	3	3	3	3	3	3	3
5000	1.28	1.27	1.25	1.24	1.22	1.21	1.20	1.19	1.18	1.17	1.16
	224	219	214	210	210	210	210	210	210	210	210
	2900	2800	2700	2600	2520	2440	2360	2300	2230	2170	2100
	13	12	12	12	12	12	12	12	12	12	12
1500	1.24	1.23	1.22	1.21	1.19	1.18	1.17	1.16	1.15	1.14	1.14
	224	219	214	210	210	210	210	210	210	210	210
	2970	2880	2780	2670	2590	2520	2450	2370	2310	2250	2180
	19	19	18	18	18	18	18	18	18	18	18
S.L.	1.23	1.21	1.20	1.19	1.18	1.17	1.16	1.15	1.14	1.14	1.13
	224	219	214	210	210	210	210	210	210	210	210
	3020	2920	2810	2720	2640	2560	2480	2410	2340	2280	2210
	22	21	21	21	21	21	21	21	21	21	21

HOLDING SPEED: 210 KIAS OR MINIMUM DRAG AIRSPEED - CLEAN.
FUEL FLOW IS BASED ON HOLDING IN A RACE TRACK
PATTERN. REDUCE FUEL FLOW BY 5% IF HOLDING
STRAIGHT AND LEVEL.

Table E-2. Weights and Limits for the TSRV (B737-100) Aircraft

Quantity	Value .(lb)
Empty weight (OEW)	64389
Zero Fuel Weight	66500.
Maximum Zero Fuel Weight	81700.
Fuel Capacity (4190 US gal)	27235
Maximum Landing Weight	89700.
Maximum Inflight Weight	
Flaps 0°	97000
30°	95000
40°	89700.
Maximum Taxi Weight (BRGW)	97800.

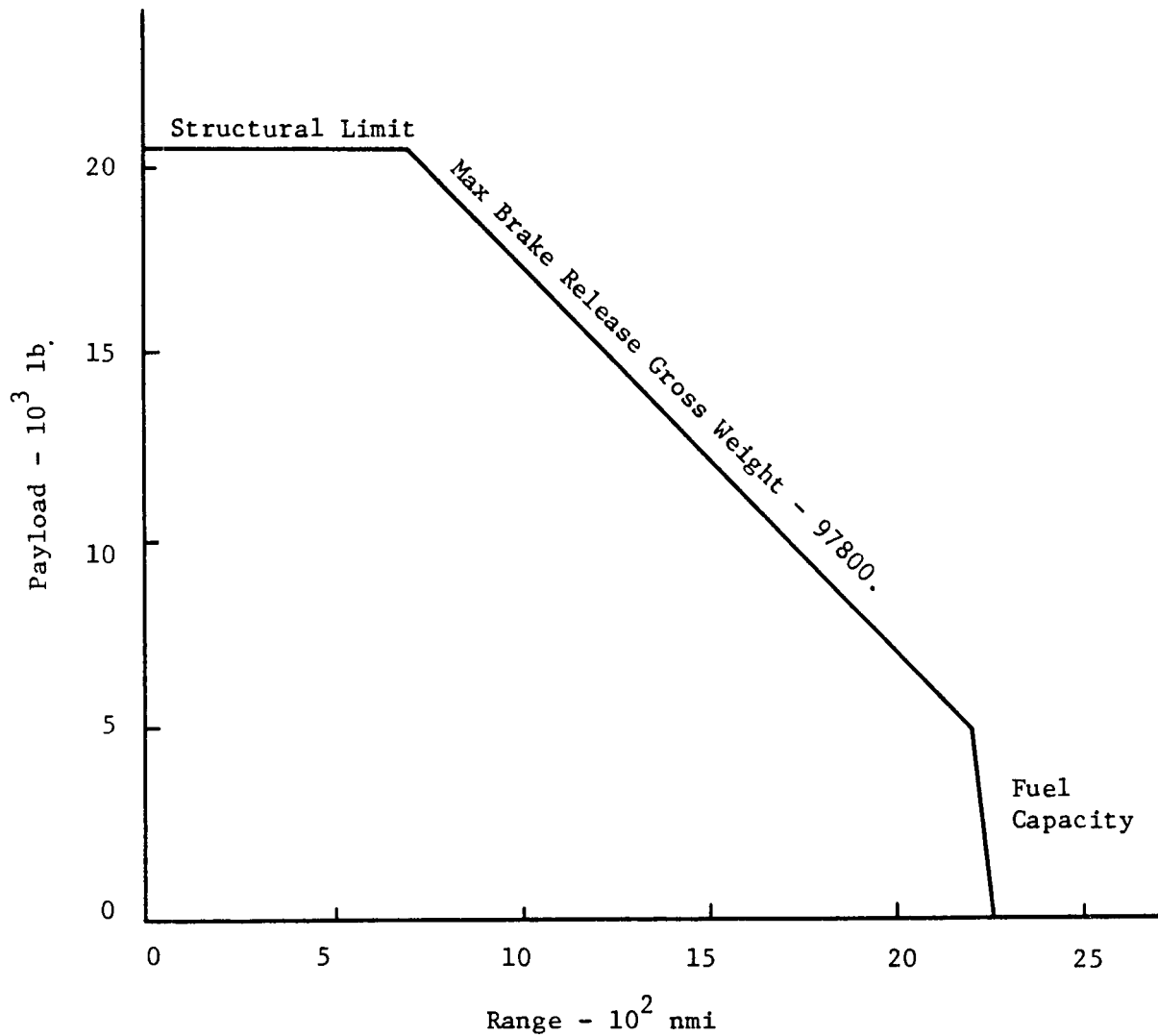


Figure E-8. Payload Limit as a Function of Range.

If the payload and fuel requirements meet these tests, a flight plan can be generated. The final test is to see if the takeoff weight that results from computing the reference profile backwards in time exceeds the takeoff limit of 97000 lb.

APPENDIX F

DYNAMIC PROGRAMMING FUNDAMENTALS

Dynamic programming is an optimization method concerned with problems which can be formulated as a sequence of decisions. This method was first developed by Richard Bellman in the early 1950's and is based on the principle of optimality. We shall make use of examples to introduce this method and Bellman's principle of optimality. [8,14]

Bellman's Principle of Optimality

Consider the multistage network shown in Fig. F-1. The problem is to find the path from A to H which minimizes the cost between A and H.

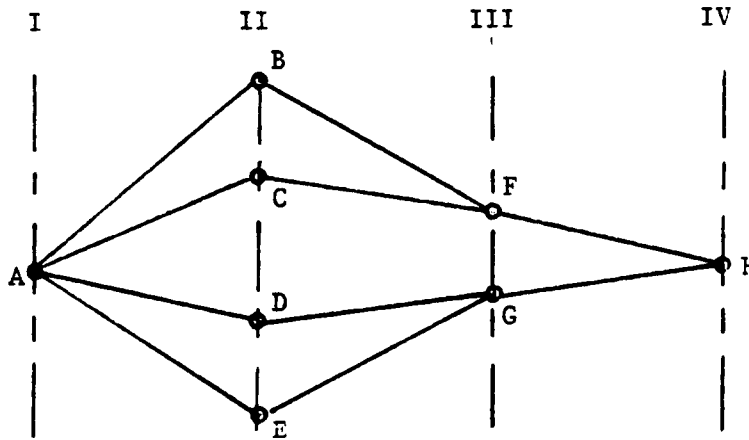


Figure F-1. Simple Multistage Network

PRECEDING PAGE BLANK NOT FILMED

In Stage I, one can decide to go to B, C, D or E, which are located in Stage II; in the same way, in Stage II one can choose to go to either F or G, provided that the corresponding link is available. Any path connecting A to H is called a policy. There are four possible policies in our example, namely; (ABFH), (ACFH), (ADGH) and (AEGH). Any continuous section of a policy is called a subpolicy. Sections AB, ABF, BFH and CFH are examples of subpolicies.

The principle of optimality states the following:

An optimal policy must contain only optimal subpolicies.

Proof: Let's assume that a subpolicy is extracted from an optimal policy. If such a subpolicy is not an optimal one, then there exist another subpolicy which, if substituted for the extracted one would improve the optimal policy, a deduction contrary to the hypothesis. Bellman [14] states the above principle in the following form:

"A policy is optimal if, at a stated period, whatever the preceding decision may have been, the decision still to be taken constitutes an optimal policy when the result of the previous decision is included."

Let us show how the above principle can be applied to the problem shown in Fig. F-1. Let $J_{\alpha\beta}$ be the cost to move from α to β and J_{γ} the minimum cost to move from point A (starting point) to point γ . Therefore, the minimum cost is obtained by:

$$J_{\min} = \min \left\{ J_{AB} + J_{BF} + J_{FH}, J_{AC} + J_{CF} + J_{FH}, J_{AD} + J_{DG} + J_{GH}, J_{AE} + J_{EG} + J_{GH} \right\}. \quad (F.1)$$

However, principle of optimality allows Eq. (F.1) to be written as:

$$J_{\min} = \min \{ J_F + J_{FH}, J_G + J_{GH} \}, \quad (F.2)$$

where

$$J_F = \min \{ J_{AB} + J_{BF}, J_{AC} + J_{CF} \}, \quad (F.3)$$

$$J_G = \min \{ J_{AD} + J_{DG}, J_{AE} + J_{EG} \}. \quad (F.4)$$

As seen, the principle of optimality replaces a choice between all alternatives given by Eq. (F.1) by a sequence of decisions between fewer alternatives given by Eq. (F.2).

Method of Computation

Consider the multistage network shown in Fig. F-2, and let the number on each link designate its associated cost. The problem is to obtain the least costly path from A to H.

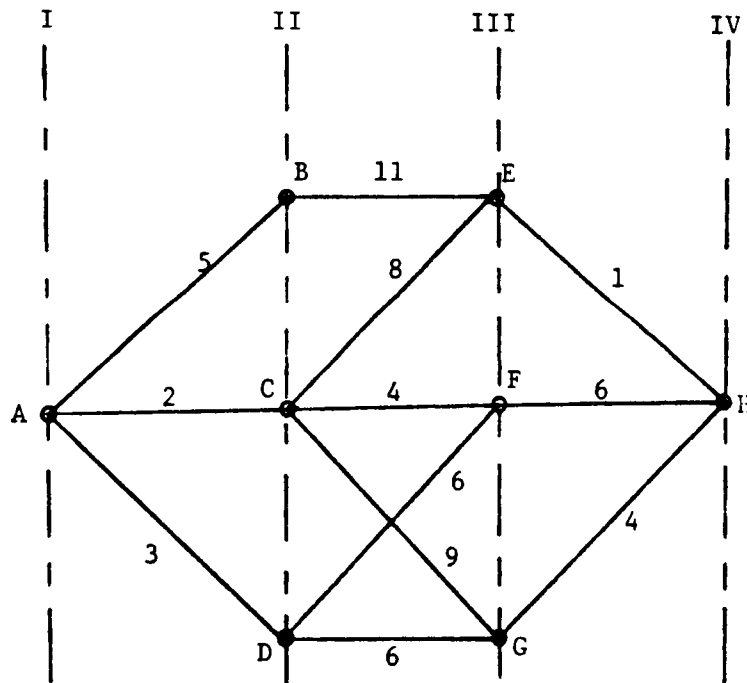


Figure F-2. Example Multistage Network

Let us define the following terms:

$J_i(x)$ = minimum cost to go to point x on the i^{th} stage from the origin,

$J_{\alpha\beta}$ = cost associated with segment $\alpha\beta$.

We first seek the minimum cost to arrive at Stage II for each of the points B, C and D. Therefore, we have

$$J_{II}(B) = 5, \tag{F.5}$$

$$J_{II}(C) = 2, \tag{F.6}$$

$$J_{II}(D) = 3. \tag{F.7}$$

Now we can move to the next stage

$$J_{III}(E) = \min_{x=B,C,D} [J_{II}(x) + J_{xe}], \tag{F.8}$$

$$J_{III}(F) = \min_{x=B,C,D} [J_{II}(x) + J_{xf}], \tag{F.9}$$

$$J_{III}(G) = \min_{x=B,C,D} [J_{II}(x) + J_{xg}], \tag{F.10}$$

using the numerical values in Eqs. (F.4)-(F.6), we have

$$J_{III}(E) = 10. \text{ with } x = C, \tag{F.11}$$

$$J_{III}(F) = 6. \text{ with } x = C, \tag{F.12}$$

$$J_{III}(G) = 9. \text{ with } x = D. \tag{F.13}$$

In the same way we move to the last stage

$$J_{IV}(G) = \min_{x=E,F,G} [J_{III}(x) + J_{xn}], \quad (F.14)$$

$$J_{IV}(G) = 11. \text{ with } x = E. \quad (F.15)$$

Therefore the least costly path is ACEH.

To derive a general equation for the above computation let us define the following terms.

$C(i,J,K)$ = Cost between the point J on the i^{th} stage and point K on the $(i + 1)^{\text{th}}$ stage.

$C_{\min}(i,J)$ = minimum cost between the origin and the J^{th} point on the i^{th} stage.

Therefore, we have

$$C_{\min}[i,K] = \min_{J=1, NP(i-1)} [C_{\min}(i-1,J) + C(i-1,J,K)] \quad (F.16)$$

where $NP(i-1)$ is the number of points on the $(i-1)^{\text{th}}$ stage.

Equation (F.16) summarizes the main computational algorithm.

Amount of Computation

In this section, we will show the amount of computation involved in determining the optimal path. This is done by computing the number of possible paths which have to be considered before the optimal path is obtained.

Consider the multistage network shown in Fig. F-3.

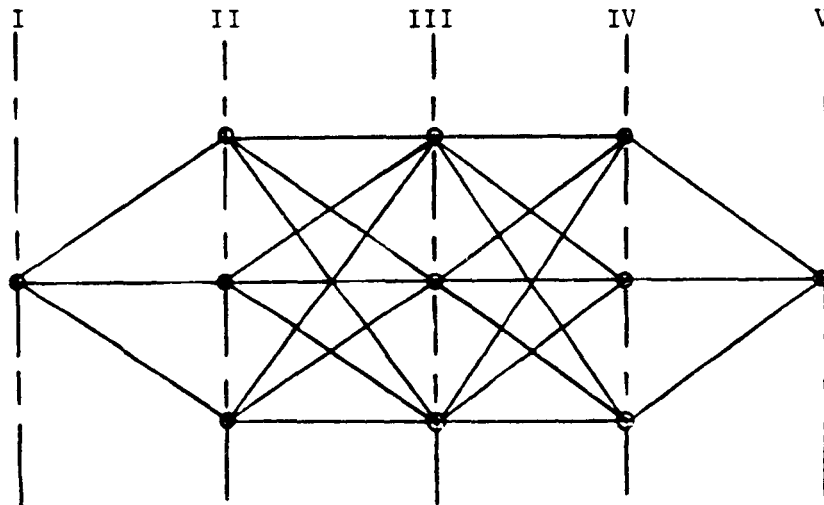


Figure F-3. Multistage Networks to Illustrate Computation Load

In going from Stage I to II, there are three alternative choices. Each of these choices offers three alternative choices to move from Stage II to Stage III. Therefore, we have nine possible routes to go to Stage III and in the same way twenty-seven possible routes to go to Stage IV. In general, if the network has N stages and each stage has r points, (with the exception of first and last stage which only have one point), then there are $r^{(N-2)}$ possibilities to reach the N^{th} stage. In the above example, there are twenty-seven paths leading to Stage V. However, using

the dynamic programming method, r possibilities to reach the second and last stages and r^2 possibilities for other stages must be considered. Therefore a total of $(2r + (N-3)r^2)$ possibilities must be considered. Using this approach, there are twenty-four possibilities to reach stage V. The important change that has taken place is the removal of N from the exponent. For a many-stage problem, the reduction in computation can be enormous. The reason for this reduction is in utilizing the principle of optimality at each stage. We discard a vast majority of path combinations and only save a comparatively few that are possible candidates for the overall optimal path.

REFERENCES

1. Sorensen, J. A., Waters, M. H., and Patmore, L. C., "Computer Programs for Generation and Evaluation of Near-Optimum Vertical Flight Profiles," NASA CR-3688, May 1983.
2. Sorensen, J. A., "The Flight Planning - Flight Management Connection," American Control Conference, San Diego, CA, June 1984.
3. Steinberg, R., "Airline flight Planning - The Weather Connection," Symposium on Commercial Aviation Energy Conservation Strategies, Washington, DC, April 1981.
4. Lee, H. W., and Erzberger, H., "Algorithm for Fixed-Range Optimal Trajectories," NASA TN-1565, July 1980.
5. Sorensen, J. A., and Waters, M. H., "Generation of Optimum Vertical Profiles for an Advanced Flight Management System," NASA CR-165674, March 1981.
6. True, H. C., and Winer, D. E., "Wind and Temperature Database for Flight Planning," SAE Aerospace Congress & Exposition, Anaheim, CA, October 1981.
7. Anon., "ACARS for the Eighties and Beyond," Aeronautical Radio, Inc., Annapolis, MD, January 1984.
8. Dreyfus, S. E., Dynamic Programming and the Calculus of Variations, Academic Press, New York, 1965.
9. Wang, K. Y., and D'Esopo, D. A., "Oceanic Area System Improvement Study," Vol. IX, FAA-EM-81-17, DOT, September 1981.
10. Bryson, A. E., and Ho, Y. C., Applied Optimal Control, Blaisdell, Waltham, MA, 1969.
11. Dorsky, S., and Sorensen, J. A., "EFPLAN - An Experimental Flight Planning Program. User's Guide," Analytical Mechanics Report No. 85-08, NASA Contract No. NAS1-17345, Mountain View, CA, October 1985.
12. Box, M. J., "A New Method of Constraint Optimization and a Comparison with Other Methods," Computer Journal, Vol. 8, 1968, p. 42.
13. Anon., "U. S. Standard Atmosphere, 1976," NOAA S/T76-1562, U. S. Government Printing Office, Washington, DC, October 1976.
14. Bellman, R., Dynamic Programming, Princeton Univ. Press, Princeton, NJ, 1957.

PRECEDING PAGE BLANK NOT FILMED

Standard Bibliographic Page

1. Report No. NASA CR-4063		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle Design and Analysis of Advanced Flight Planning Concepts				5. Report Date March 1987	
				6. Performing Organization Code	
7. Author(s) John A. Sorensen				8. Performing Organization Report No. AMA Report No. 85-09	
9. Performing Organization Name and Address Analytical Mechanics Associates, Inc. 2483 Old Middlefield Way Mountain View, CA 94043				10. Work Unit No.	
				11. Contract or Grant No. NAS1-17345	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546				13. Type of Report and Period Covered Contractor Report	
				14. Sponsoring Agency Code	
15. Supplementary Notes Langley Technical Monitor: Dan D. Vicroy Final Report					
16. Abstract The objectives of this continuing effort are to develop and evaluate new algorithms and advanced concepts for flight management and flight planning. This includes the minimization of fuel or direct operating costs, the integration of the airborne flight management and ground-based flight planning processes, and the enhancement of future traffic management systems design. Flight management (FMS) concepts are for on-board profile computation and steering of transport aircraft in the vertical plane between a city pair and along a given horizontal path. Flight planning (FPS) concepts are for the pre-flight ground based computation of the three-dimensional reference trajectory that connects the city pair and specifies the horizontal path, fuel load, and weather profiles for initializing the FMS. As part of these objectives, a new computer program called EFPLAN has been developed and utilized to study advanced flight planning concepts. EFPLAN represents an experimental version of an FPS. It has been developed to generate reference flight plans compatible as input to an FMS and to provide various options for flight planning research. This report describes EFPLAN and the associated research conducted in its development.					
17. Key Words (Suggested by Authors(s)) Flight Planning Flight Management Dynamic Programming Performance Modeling			18. Distribution Statement Unclassified - Unlimited Subject Category 03		
19. Security Classif.(of this report) Unclassified		20. Security Classif.(of this page) Unclassified		21. No. of Pages 162	22. Price A08

For sale by the National Technical Information Service, Springfield, Virginia 22161