

## **General Disclaimer**

### **One or more of the Following Statements may affect this Document**

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

(NASA-CR-169027) USE OF OPTIMIZATION TO  
PREDICT THE EFFECT OF SELECTED PARAMETERS ON  
COMMUTER AIRCRAFT PERFORMANCE Final Report  
(Stanford Univ.) 61 p HC A04/MF A01

N82-26279

CSCL 01C G3/05

Unclassified  
28083



**Department of AERONAUTICS and ASTRONAUTICS  
STANFORD UNIVERSITY**

SUDAAR 534

**USE OF OPTIMIZATION TO PREDICT THE  
EFFECT OF SELECTED PARAMETERS ON  
COMMUTER AIRCRAFT PERFORMANCE**

by

**Valana L. Wells and Richard S. Shevell**

**This Research Was Supported By  
NASA Langley Research Center  
Hampton, Virginia 23665  
Grant NAG 1-202**



**June 1982**

USE OF OPTIMIZATION TO PREDICT THE EFFECT OF  
SELECTED PARAMETERS ON COMMUTER AIRCRAFT PERFORMANCE

Final Report for  
NASA Research Grant NAG-1-202

Valana L. Wells and Richard S. Shevell

Department of Aeronautics and Astronautics  
Stanford University  
Stanford, California 94305

June 1982

Abstract

An optimizing computer program, developed as part of this study, determined the turboprop aircraft with lowest direct operating cost for various sets of cruise speed and field length constraints. External variables included wing area, wing aspect ratio and engine sea level static horsepower; tail sizes, climb speed and cruise altitude were varied within the function evaluation program. Direct operating cost was minimized for a 150 n. mi typical mission. Generally, DOC increased with increasing speed and decreasing field length but not by a large amount. Ride roughness, however, increased considerably as speed became higher and field length became shorter.

## TABLE OF CONTENTS

ABSTRACT . . . . .	iii
TABLE OF CONTENTS . . . . .	iv
I. INTRODUCTION . . . . .	1
II. PROCEDURE . . . . .	2
A The Optimizer . . . . .	2
B The Function Evaluation . . . . .	3
1 Airplane Geometry and Drag Parameters . . . . .	3
2 Range and Maximum Takeoff Weight . . . . .	5
3 Evaluation of Constraint Parameters . . . . .	9
(a) Maximum Cruise Thrust . . . . .	9
(b) Takeoff Distance . . . . .	10
(c) Landing Distance . . . . .	10
(d) Second Segment Climb Gradient . . . . .	10
(e) Enroute Climb Gradient . . . . .	10
4 Typical Mission . . . . .	11
C Ride Roughness . . . . .	11
III. RESULTS . . . . .	12
A Critical Field Lengths . . . . .	12
B Active Constraints and Optimal Variable Values . . . . .	13
C Sensitivity Studies . . . . .	14
1 Grid Search About an Optimal Point . . . . .	14
2 Non-Optimal Operation . . . . .	15
D Ride Roughness . . . . .	15
IV. CONCLUSIONS . . . . .	17
V. REFERENCES . . . . .	18
Table 1 Active Constraints . . . . .	19
Figure 1 Direct Operating Cost vs. Field Length . . . . .	20
Figure 2 Optimal Variable Values vs. Field Length . . . . .	21
Figure 3 Optimal Point Sensitivities Cruise Speed = 330 kt, Maximum Field Length = 4000 feet . . . . .	22
Figure 4 Cost for Non-Optimal Operation [330 kt designs flown at lower speeds] . . . . .	23
Figure 5 Ride Roughness vs. Field Length . . . . .	24
Figure 6 Theoretical Gust Response . . . . .	25
Figure 7	
Appendix I Program Listing . . . . .	27

## I. Introduction

The resurgence of interest in small, propeller-driven aircraft has sparked renewed analysis of the aerodynamics, structures and propulsion systems of such planes. Along with advanced technology research, which is the bent of much of the recent concern, there remains a need for the answer to a, perhaps, more basic question—that is, for what mission should this airplane be designed? The "mission" includes not just stage length (which is determined by the actual leg distances flown by commuter airlines) but also the speed at which to climb and cruise and the field length from which the aircraft must takeoff and land.

This study, rather than seeking to prescribe a particular design or mission, discovers the relationships between field length and cruise speed and aircraft direct operating cost. To do this, a gradient optimizing computer program was developed to minimize direct operating cost (DOC) as a function of airplane geometry. In this way, one can compare the best airplane operating under one set of constraints with the best operating under another. Best, in this case, means having the minimum DOC.

To compare different airplanes, one can make use of relatively simple techniques for some parameter estimation. For example, a complete stability and control analysis for tail size determination is superfluous for preliminary design when statistical correlations of tail sizes with wing and fuselage characteristics exist for similar airplanes. Thus several such statistical correlations methods appear in the program. However, one must also use more sophisticated procedures when a high degree of accuracy is required or when the particular calculation may have a major influence on the performance index. The program, therefore, has extensive and detailed routines for drag, climb, range and other critical values.

For this study a constant 30-passenger fuselage and "rubberized" engines based on the General Electric CT-7 were used as a baseline. All aircraft had to have a 600 nautical mile maximum range and were designed to FAR part 25 structural integrity and climb gradient regulations. Direct operating cost was minimized for a typical design mission of 150 nautical miles. For purposes of  $C_{L_{max}}$  calculation, all aircraft had double-slotted flaps but with no Fowler action.

## II. Procedure

### A. The Optimizer

The optimizer minimizes direct operating cost as a function of wing area, aspect ratio and engine sea-level static horsepower rating through use of a variable metric algorithm which is, in fact, a quasi-Newton's method. A true Newton's method utilizes the following strategy for size and direction of step:

$$\hat{x}_{j+1} = \hat{x}_j - H_j^{-1} \hat{g}_j$$

where  $\hat{x}$  represents the vector of variables,  $H_j$  is the Hessian (matrix of second derivatives) at step  $j$ , and  $\hat{g}_j$  is the gradient vector at step  $j$ . In the absence of second derivative information, a numerical approximation of the Hessian using known values of the first derivatives provides an adequate substitute. The variable metric method follows exactly this procedure.

Of course, for such a complicated function as the one in this study (the "function" is a thirty page FORTRAN program), even first derivatives do not exist in closed form. Thus, the program must calculate a gradient estimate using a forward difference approximation. The differencing step size is constrained to be rather large (one percent of the variable value) since noise in the function evaluation leads to incorrect gradients for small steps.

The variable metric method solves unconstrained problems only. Thus, in order to account for the inequality constraints which must hold in order for the aircraft to meet such mission requirements as maximum takeoff distance, minimum engine-out climb gradient, etc., the program uses what is termed the "penalty function" or "soft constraint" approach. In a mathematical sense, this method changes the problem to an unconstrained one by including the constraints in the goal function. The goal function becomes,

$$GOAL = DOC + K|constraint value - constraint value required|$$

where  $DOC =$  direct operating cost  
 $K =$  penalty coefficient  
 $= \begin{cases} 0 & \text{if constraint is met} \\ \text{large if constraint is not met.} \end{cases}$

A penalty is added to the goal function for each of the five inequality constraints:

- . takeoff distance
- . landing distance
- . available cruise power
- . second segment climb gradient
- . enroute climb gradient

## B. The Function Evaluation

The function evaluation program, which comprises the bulk of the calculations involved in the optimization, acts as a mathematical aircraft model. This routine determines, for a prescribed wing area, sea level static horsepower rating, and wing aspect ratio, the complete geometry, performance, and operating cost of the resulting airplane. For simplicity, it employs preliminary design methodology for estimating such parameters as zero-lift equivalent drag area, tail sizes,  $C_{L\max}$ , and airplane efficiency factor. The direct operating cost calculation is based on the 1967 ATA DOC method with corrections for inflation and commuter operation. The following outline briefly describes the function evaluation scheme.

### 1. Airplane Geometry and Drag Parameters

In order to compute the airplane geometry (wing span, wing mean aerodynamic chord, vertical and horizontal tail areas) the program assumes as constants:

wing average thickness ratio	.15
tail average thickness ratio	.1
wing taper ratio	.4
horizontal tail aspect ratio	4.0
vertical tail aspect ratio	1.8
wing sweep	0
tail surface sweep angle	20°

To avoid a complex iteration involving weight and balance, the horizontal and vertical tail lengths are estimated as 32 ft. and 30 ft., respectively, and a center of gravity range of 25% of the wing mac is allowed. Using these estimates, the program calculates tail areas as a function of fuselage and wing sizes according to ref. 1.

Once all surface areas are known, the program computes the zero-lift equivalent drag area,  $f$ . The formula for  $f$  of a component has the form

$$f_i = C_{f,i} K_i S_{wet,i}$$

where  $C_f$  = friction coefficient; function of Reynolds number

$K$  = form factor; function of fineness ratio or thickness ratio

$S_{wet}$  = component wetted area

$i$  refers to the  $i$ th component such as wing, fuselage, nacelle, etc.

A summation of all component drag areas, plus a 6% addition for miscellaneous components, gives the total airplane equivalent parasite drag area:

$$f = \sum_i f_i / .94$$

The zero-lift or parasite drag coefficient,  $C_{Dp}$ , is just:  $C_{Dp} = f/S_w$ ,  $S_w$  = wing reference area.

The program computes airplane efficiency factor,  $e$ , from:

$$e = \frac{1}{\pi R \left( \frac{1}{\pi R u_s} + .43 C_{Dp} \right)}$$

where  $u$  = induced drag factor due to planform; function of  $R$ , taper ratio, sweep.

$s$  = induced drag factor due to fuselage interferences function of wing span/fuselage diameter.

Inclusion of  $C_{Dp}$  in this formula accounts for the increase in profile drag with angle of attack.

## 2. Range and Maximum Takeoff Weight

For any combination of wing area, sea level horsepower, and wing aspect ratio (other possible variables assumed constant) there exists a takeoff weight necessary to travel a given distance at a given speed. This routine determines that takeoff weight required for the airplane described by those three variables to cover a maximum range of 600 N mi. at a prescribed cruising speed. The takeoff weight depends rather heavily on two other variables - cruise altitude and climb speed. Thus, in order to include these as variables, the program performs a two dimensional grid search on altitude and climb speed and saves the combination of the two which uses the least fuel to complete the 600 N mi. mission.

Determining the maximum takeoff weight is an iterative procedure completed through the use of a one dimensional minimization routine. The minimizer employs a "linear search with parabolic inverse interpolation" with the goal function defined as the square of the difference between the actual range and the desired range.

The range calculation itself has four major parts:

- (a) calculation of empty weight
- (b) climb
- (c) descent
- (d) cruise

The weight is calculated using a statistical method based on data from large and small commercial aircraft, (ref. 1).

The time, fuel, and distance to climb are calculated according to:

$$\text{time to climb} = \int_{h_{\min}}^{h_{\max}} \frac{dh}{R/C}$$

$$\text{fuel to climb} = \int_{h_{\min}}^{h_{\max}} \frac{\text{SHP} * \text{SFC}}{3600 * \text{R/C}} dh$$

$$\text{distance to climb} = \int_{h_{\min}}^{h_{\max}} \frac{V}{\text{R/C}} dh$$

where R/C = rate of climb in ft/sec

h = altitude in ft

SHP = shaft power in horsepower

SFC = specific fuel consumption in lb/SHP-hr

V = true airspeed in ft/s

The climb routine numerically evaluates these integrals making the following assumptions:

- . climb at constant equivalent airspeed
- . climb at maximum continuous power
- . SFC constant at maximum continuous power

The numerical integration uses a forward Euler technique and an altitude step size of 200 ft.

The descent method assumes:

- . descent at constant equivalent airspeed
- . descent at constant rate of descent
- . idle (minimum) power at 10% of maximum power

The aircraft rate of descent corresponds to a 300 feet per minute cabin pressure descent where the cabin has a 6000 foot pressure altitude in cruise. The program computes fuel and distance to descend using the following integrals:

$$\text{fuel to descend} = \int_{h_{\min}}^{h_{\max}} \frac{\text{SHP} * \text{SFC}}{3600 * \text{R/D}} dh$$

$$\text{distance to descend} = \int_{h\min}^{h\max} \frac{V}{R/D} dh$$

where  $h$  = altitude

SFC = specific fuel consumption

R/D = rate of descent, ft/sec

$V$  = descent true airspeed, ft/sec

SHP = shaft power

Since the airplane descends at constant equivalent airspeed and constant rate of descent, the distance integral becomes:

$$\text{distance to descend} = \frac{V_E}{R/D} \int_{h\min}^{h\max} (1 - 6.8634 \times 10^{-6} h)^{-2.1324} dh$$

where  $V_E$  = equivalent airspeed for descent

$$V_E = V (1 - 6.8634 \times 10^{-6} h)^{-2.1324}$$

Integrating gives:

$$\text{distance to descend} = \frac{V_E}{R/D} \left( \frac{1}{6.8634 * 1.1324 * 10^{-6}} \right)$$

$$\times \left[ (1 - 6.8634 \times 10^{-6} h)^{-1.1324} \right]_{h\min}^{h\max}$$

Letting  $h\min = 0$ ,

$$\text{distance to descend} = \frac{V_E}{R/D} \left( \frac{1}{6.8634 * 1.1324 * 10^{-6}} \right)$$

$$\times \left[ (1 - 6.8634 * 10^{-6} h\max)^{-1.1324} - 1 \right]$$

Fuel to descend is numerically calculated using an explicit Euler integration and an altitude step size of 500 ft. The fuel integral begins at the end of the descent and integrates backwards until the aircraft reaches the cruising altitude. The weight at the bottom of descent is the zero fuel weight plus additional fuel weight for an appropriate reserve mission (100 n. mi. at best specific range plus 45 minutes at best endurance).

The distance covered in the cruising portion of the mission depends on the weights at the end of climb and at the top of descent. For propeller-driven aircraft,

$$R = \int_{Wf}^{Wi} 325 \frac{n}{SFC} \frac{dW}{D}$$

where  $n$  = propeller efficiency in cruise

$D$  = drag

$Wi$  = weight at beginning of cruise

$Wf$  = weight at end of cruise

SFC = specific fuel consumption

$R$  = range, n. miles

Letting SFC be approximately constant and equal to the average SFC during cruise, and noting that

$$D = C_{Dp} q S + \frac{W^2}{q\pi b^2 e},$$

and letting

$$A1 = C_{Dp} q S$$

$$A2 = \frac{1}{q\pi b^2 e},$$

the integral becomes,

$$R = 325 \frac{n}{SFC} \int_{W_f}^{W_1} \frac{dW}{A_1 + A_2 W^2}$$

for constant cruise speed and cruise altitude. Integrating gives:

$$R = 325 \frac{n}{SFC} \frac{1}{\sqrt{A_1 A_2}} \left[ \tan^{-1} \frac{W}{\sqrt{A_1/A_2}} \right]_{W_f}^{W_1}$$

or

$$R = 325 \frac{n}{SFC} b \sqrt{\frac{\pi e}{C_D p S}} \left[ \tan^{-1} \frac{W}{q_b \sqrt{C_D p \pi e S}} \right]_{W_f}^{W_1}$$

This formula holds only for the case of constant dynamic pressure,  $q$ . Since the commuter cruises at constant speed and altitude, it satisfies the condition of invariant  $q$ .

### 3. Evaluation of Constraint Parameters

Five acceptability criteria constrain the aircraft design.

#### (a) Maximum Cruise Thrust

To fly at the prescribed cruising speed, the maximum thrust produced by the engines must equal or exceed the cruise drag. Maximum thrust depends on maximum cruise power according to the relation

$$T_H = 550 n \frac{SHP}{V}$$

where  $n$  = propeller efficiency

$V$  = true airspeed, ft/sec

$T_H$  = thrust, lb.

Maximum cruise shaft horsepower is determined as a function of airspeed, cruise altitude, and static sea level power rating. Power calculations are based on the General Electric CT-7 turboprop engine.

### (b) Takeoff Distance

Allowed takeoff distances range from 3500 feet to 4500 feet. FAR takeoff field lengths depend on the parameter:

$$\frac{TOW^2}{\sigma C_{L_{max}} S_W^{TH}}$$

where  $\sigma = \sqrt{\rho/\rho_0}$

$S_W$  = reference wing area,  $ft^2$

Takeoff distance is calculated for a hot day (ISA + 30.8°F) sea level.

### (c) Landing Distance

The allowed FAR landing distances range from 3500 feet to 4500 feet, and they depend on the square of the airplane's stalling speed. Since commuter airplanes do not usually have the ability to jettison fuel, the studied aircraft must land at their takeoff weights.

### (d) Second Segment Climb Gradient

To comply with the Federal Air Regulation, part 25, a twin-engined airplane must have a second segment climb gradient of 2.4%. The gradient is computed for hot day conditions (ISA + 30.8°F) at takeoff power and with one engine inoperative. The drag in this configuration includes that due to a feathered propeller, due to excess rudder deflection as a consequence of asymmetric thrust, and due to a 25 degree takeoff flap deflection.

### (e) Enroute Climb Gradient

According to FAR part 25, a twin-engined airplane must have a one engine-out enroute climb gradient of 1.1%. Since speed for best climb gradient for aircraft of this type is less than the minimum allowable speed, enroute climb gradient is computed at 1.3 times the stalling speed in the clean configuration. The obstacle clearance height used is 11,000 feet.

#### 4. Typical Mission

To optimize the commuter design with respect to operating cost, one must compute DOC (direct operating cost) for a typical shorthaul mission. Using a characteristic stage length of 150 N mi., the program finds the corresponding block fuel and block time for the airplane designed to meet the restrictions outlined above. Such values as takeoff weight necessary to meet the range are calculated according to the method described in the "Range and Maximum Takeoff Weight" section.

The direct operating cost routine assumes that commuter pilot pay rates are about one-third that of trunk carrier pilots. It also uses the following cost estimates:

Labor Rate	\$12/hr
Airframe First Cost	\$200/lb of airframe
Engine First Cost	Taken from Ref. 4; inflated 25%
Fuel Cost	\$1.50/gallon
Oil Cost	\$10/lb

The cost calculation proceeds as suggested in Ref. 2. Appendix I contains a complete listing of the program.

#### C. Ride Roughness

Though ride-roughness was not considered in the optimization portion of the study, a relative ride-roughness parameter was computed for each optimum airplane. This parameter, taken from the FAR gust-response/structures regulations is given by

$$\Delta n = K \frac{a}{498(W/S)} \frac{U_{de} V_e}{}$$

where  $U_{de}$  = equivalent gust velocity, ft/s

$V_e$  = equivalent speed, knots

$W/S$  = wing loading, lb/ft<sup>2</sup>

$a$  =  $dC_L/d\alpha$

$$K = \frac{.88\mu}{5.3 + \mu}$$

$$\mu = \frac{2(W/S)}{\rho c a g}$$

Reference 7 provides further discussion of this parameter.

### III. Results

The results of the optimization program show that the airplane with the lowest direct operating cost flies at 290 knot TAS with an allowed field length greater than or equal to 4,060 feet, Figure 1. For field lengths less than 3,650 feet, the 250 knot airplane fares best in terms of DOC as the large wings required for short landing distances cause excessive drag at the higher speeds. At greater than 3,650 foot field lengths, 290 knots is the best speed. The best 330 knot airplane, however, with a landing distance of 4,275 feet has only one percent worse direct operating cost than the best airplane overall. Direct operating cost as a function of field length and cruise speed is presented in Figure 1.

The optimization, aside from determining the effect of cruise speed and field length on DOC, produced the following crucial results:

#### A. Critical Field Lengths

Although, generally, direct operating cost decreases with increasing field length (for a given speed), for each speed there exists a critical field length beyond which there is no further improvement in DOC; the field length constraint becomes non-active. Two factors contribute to this phenomenon. First, though the wing area can decrease with increased takeoff or landing distance, the aircraft must still maintain a span adequate to meet climb gradient standards. The resulting increase in aspect ratio increases the weight enough to counteract the beneficial effects of the lower wing area. Secondly, a smaller wing area forces the aircraft to an inefficient  $C_L$  far from that for best L/D (which indicates best specific range for propeller-driven aircraft). A drop in cruise altitude improves the  $C_L$  but increases the non-lift dependent drag so the altitude modification is not worthwhile.

## B. Active Constraints and Optimal Variable Values

A rough rule of thumb governing the selection of aircraft geometry states that the landing field length requirement determines the wing area and the other operative constraint, whichever one it is, fixes the proper combination of aspect ratio (span) and engine power. In fact, though wing area is not quite independent of cruise speed for a given field length, wing loading (takeoff weight divided by wing area) does not vary with speed. Thus the landing distance has only secondary effect on aspect ratio and horsepower required.

Table 1 presents a list of the active constraints—that is, those limiting the design—for each cruise speed and field length tested. The table includes the critical field length for each speed. At the lower speeds, the required enroute climb gradient sizes the aspect ratio and engine power. Since, previously, commuter aircraft have not been designed to meet FAR part 25 regulations, they have not encountered as much difficulty with the one-engine-out enroute climb restriction. Though enroute climb rarely presents a problem for turbofan aircraft, the turboprop airplane, because its speed for best climb is lower than the minimum allowable speed (30% above the stall speed), is often restricted by this regulation if it is designed according to part 25 rules.

At the highest cruise speed, in most cases, minimum cruise power to fly at 330 knot determines both engine power and aspect ratio. Obviously, increasing the horsepower increases the maximum cruise speed, but, though not as important a factor in the power-restricted cases, increasing the aspect ratio also increases the maximum cruise speed due to the reduced induced drag. So, whether the second active constraint is minimum enroute climb gradient or power to cruise at a given cruise velocity, several combinations of aspect ratio and engine power exist to satisfy that constraint. The optimizer chooses the best, or lowest cost, combination of the two.

At a cruise speed of 330 knot and landing distance 4,275 feet or more, enroute climb gradient rather than available cruise power becomes the second operational constraint. This occurs because the wing area has decreased enough that the cruise drag (and, therefore, cruise power required) has also decreased to the extent that power to climb is greater than the power to maintain a 330 knot cruise speed.

Figure 2 shows the variations of optimal wing area, aspect ratio, and horsepower with field length and cruise velocity. As expected, wing area decreases as the field length gets longer. The aspect ratio, however, increases in an attempt to keep the same span in order to maintain the same climb gradient or induced drag. The 250 knot airplanes have higher aspect ratios than the 290 knot planes because they must meet identical climb gradients but with lower power levels. The slower airplanes have lower power ratings but higher spans than the 290 knot aircraft. The 330 knot airplanes have aspect ratios lying between those of the other two speed aircraft since the cruise speed constraint affects choice of aspect ratio differently from the enroute climb constraint.

Figure 2c provides an interesting insight into the effects of differing active constraints on optimum engine power. As wing areas decrease with increasing field length, the aspect ratios increase but, in general, not enough to maintain constant span. If enroute climb is critical, then, the engine power must increase for the airplane to meet the climb gradient for reduced span. At 250 knot and 290 knot this indeed happens. However, if meeting the required cruise velocity is critical, the smaller wing area reduces the parasite drag much more than the smaller span increases induced drag. Therefore, the aircraft requires less power to overcome the cruise drag, and the curve indicating a 330 knot aircraft follows this trend.

### C. Sensitivity Studies

1. Grid Search About an Optimal Point. Although the optimizing program chooses a lowest-cost airplane for a given set of constraint parameters, it gives little information about the effects of small changes in variable values about that optimum. Figures 3a-c show cost for values of wing area, aspect ratio, and engine power above and below those calculated as the optimum for cruise speed equal to 330 knot and a field length of 4,000 feet. Constraint barriers are included in these figures to indicate areas of impossible choices. At the smallest wing area ( $345 \text{ ft}^2$ ) no airplane can meet the 4,000 foot field length constraint whereas, at a wing area of  $385 \text{ ft}^2$ , all airplanes easily fall below the field length requirement.

As these figures illustrate, the optimizer chooses the lowest cost configuration which can meet all requirements. At the optimum point, the design is bounded by both cruise power and field length, and, as a consequence, it cannot move in a direction of lower cost. (See Figure 3b.)

The "kinks" in the highest power curves of Figures 3b and 3c occur because the program allows only discrete values of cruise altitude which leads to slight discontinuities in the goal function.

2. Non-Optimal Operation. The previous discussion deals with aircraft operation under the conditions for which that aircraft is designed. Possibly, however, a commuter operator would like to have the ability to fly his airplanes at a fast speed even if he normally flies much more slowly.

Figure 4 shows the cost penalty incurred for two cases of non-optimal operation. The costs for the optimum airplanes designed for cruise at 330 knots and field lengths of 3,500 and 4,000 feet, but actually flown at several lower cruise speeds over the 150 nautical mile typical stage length, are shown. Although the cost does decrease as the airplane slows down, it does not reach the economy level achieved for the optimized airplane at each speed. The difference in DOC between the optimized aircraft and the high-speed airplane flown at a lower speed reaches as high as 1.4% for airplanes meeting a 4,000 foot landing distance and as high as 5% for airplanes with 3,500 foot field lengths. The non-optimized airplanes cost more to operate at a given speed since their larger engines and higher wing areas contribute to higher weight and drag and, thus, to more fuel burned per mission.

#### D. Ride Roughness

Figure 5 shows ride roughness parameter,  $\Delta n$ , as a function of field length for the airplanes generated by the optimizing portion of this study. The plot does not form a family of smooth curves with speed as a parameter due to the discrete altitudes allowed in the optimizing routine. In particular, the lowest field length, 330 knot airplane flies at 25,000 feet because of the extremely sub-optimal  $C_L$  produced at lower altitude. As the field length increases and the wing becomes smaller, the airplanes come down to

20,000 feet since smaller engines provide the required power at the lower altitude.

Also shown in figure 5 are the roughness parameters of several comparable-mission aircraft. Notice that, according to the method employed here, three of the optimum commuters would, in fact, have more favorable gust response than the DC-9-30. If indeed true, this suggests that commuters need only increase wing loading by using a good flap system, or slow down to 250 kts, in order to solve the ride roughness problem. Because of qualitative assessments of commuter ride roughness, however, some skepticism remains as to the validity of using this particular parameter to compare these somewhat different aircraft.

The theory for response to a sharp-edged gust (see Jones, ref. 8) can be applied to the optimum aircraft and to the DC-9-30. Figure 6 shows the theoretical curves for maximum  $\Delta C_L$  vs. mass ratio and those points corresponding to the optimal commuters from this study. The ordinate for this plot, maximum  $\Delta C_L$ , is obtained from:

$$\Delta n = \frac{\Delta L}{L} = \frac{\Delta C_L}{C_L}$$

$$\Delta C_L = \Delta n C_L$$

Since, however,  $\Delta n$  is computed for a 30 ft/sec equivalent airspeed gust and the theory shows response to a unit gust, the result must be normalized by:

$$\text{maximum } \Delta C_L = \Delta n C_L \frac{V}{U}$$

where  $U$  = true gust velocity for which  $\Delta n$  is computed

$V$  = true airplane speed

The theoretical plot indicates, first of all, that the gust response method of reference 7 coincides quite well with the theory for a sharp-edged gust. Secondly, it shows that the DC-9, with comparable mass ratio

and aspect ratio to the optimal commuters, should indeed have comparable gust response. Even with this evidence, however, a need remains for verification of  $\Delta n$  as a useful gust response parameter, especially since the theoretical curve in figure 6 has been computed only up to mass ratios of 280 - far less than those of present-day airplanes.

Figure 7 shows a diagram of the optimal airplane for 330 knot cruise speed and a 4,000 feet field length.

#### IV. Conclusions

- . Increasing cruise speed (beyond 290 knot) and decreasing allowable field length tend to increase direct operating cost, but only a six percent difference in DOC exists between the best and worst airplanes studied. This occurs because each airplane is optimized with respect to direct operating cost for its particular mission.
- . One-engine-out enroute climb gradient requirements restrict the commuter aircraft with turboprops more than they do a turbofan aircraft because the commuter's speed for best climb gradient is less than its enroute minimum allowable speed.
- . The major drawback to increasing speed and decreasing runway length is the increased ride roughness due to both higher velocity and lower wing loading. The worst ride roughness calculated for an optimal airplane represents a 45 percent increase in the relative parameter  $\Delta n$  over the lowest value.
- . Some work remains to verify the FAR value  $\Delta n$ , as a useful parameter for comparing airplane ride roughness.

## V. References

- [1] Shevell, Richard S., "Aerospace Systems Synthesis and Analysis", Department of Aeronautics and Astronautics, Stanford University, September 1978.
- [2] . "Standard Method of Estimating Comparative Direct Operating Costs of Turbine Powered Transport Airplanes", Air Transport Association of America, December 1967.
- [3] Nash, J.C., Compact Numerical Methods for Computers, New York: John Wiley & Sons, 1979.
- [4] Lockheed-California Company, Application of Advanced Technologies to Small Short-Haul Transport Aircraft, NASA CR 152363, June 1980.
- [5] Smith, C.E., Hirschkron, R., Warren, R.E., Propulsion System Study for Small Transport Aircraft Technology (STAT), General Electric Company, NASA CR 165330, May 1981.
- [6] Ebeling, A., The Determination of Airplane Maximum Lift Coefficients, Douglas Aircraft Company, Inc., July 1964.
- [7] Shevell, Richard S., Bayan, Fawzi P., "A Brief Ride Roughness Analysis", Department of Aeronautics and Astronautics, Stanford University, SUDAAR No. 526, May 1981.
- [8] Jones, Robert T., "The Unsteady Lift of a Wing of Finite Aspect Ratio", NACA Report No. 681, 1940.

Table 1. Active Constraints

Cruise Speed (kts)	Field Length Constraint (ft)	Active Constraint <sup>1</sup>
250	3,500	Enroute climb gradient
	3,725 <sup>2</sup>	"
290	3,500	"
	3,750	"
	4,000	"
	4,060 <sup>2</sup>	"
330	3,500	Maximum cruise power
	3,750	"
	4,000	"
	4,275 <sup>2</sup>	enroute climb gradient

1. This column contains the second active constraint. The first active constraint is landing distance at the field length listed in column 2.
2. Critical field length above which field length does not determine wing area, power or aspect ratio.

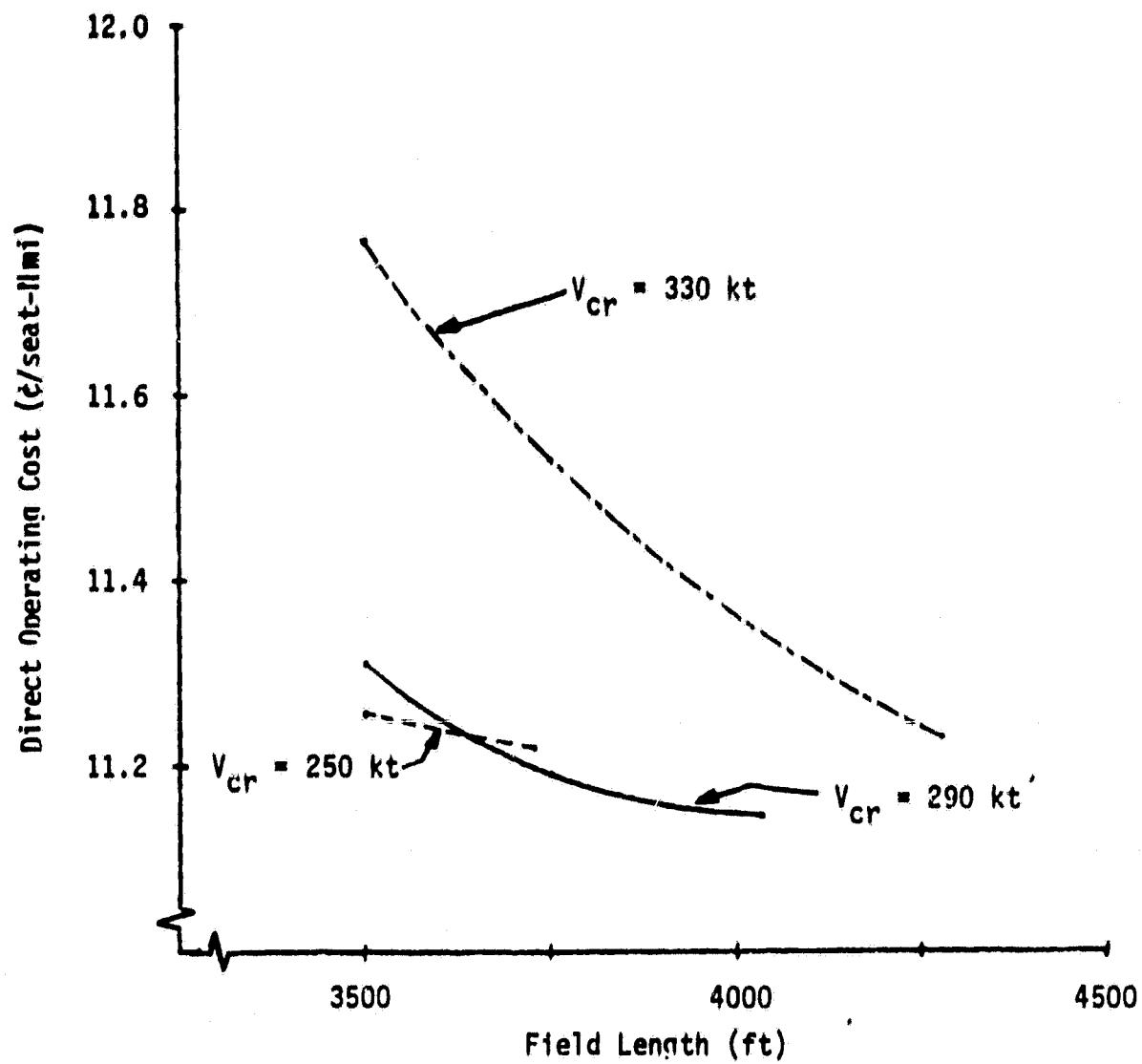
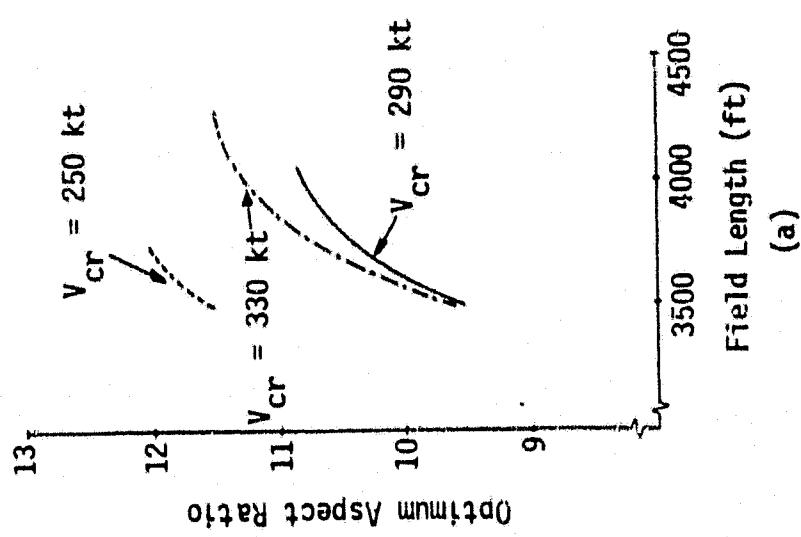
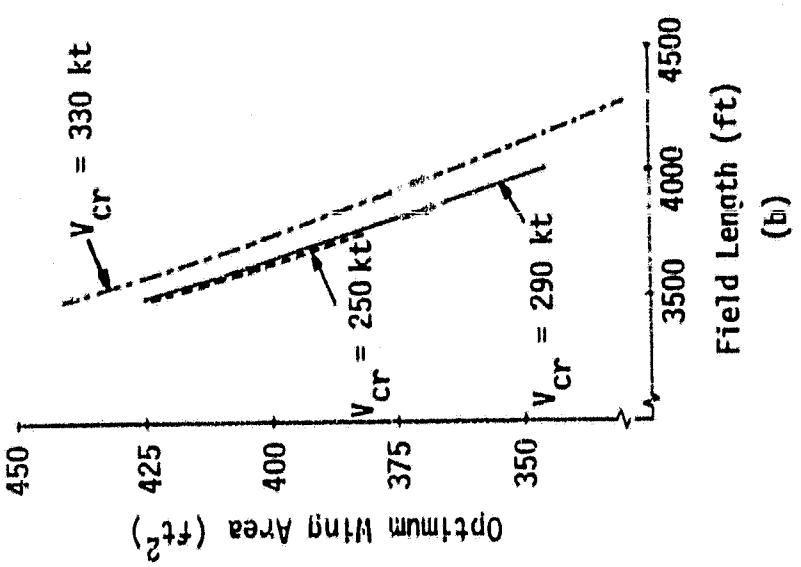


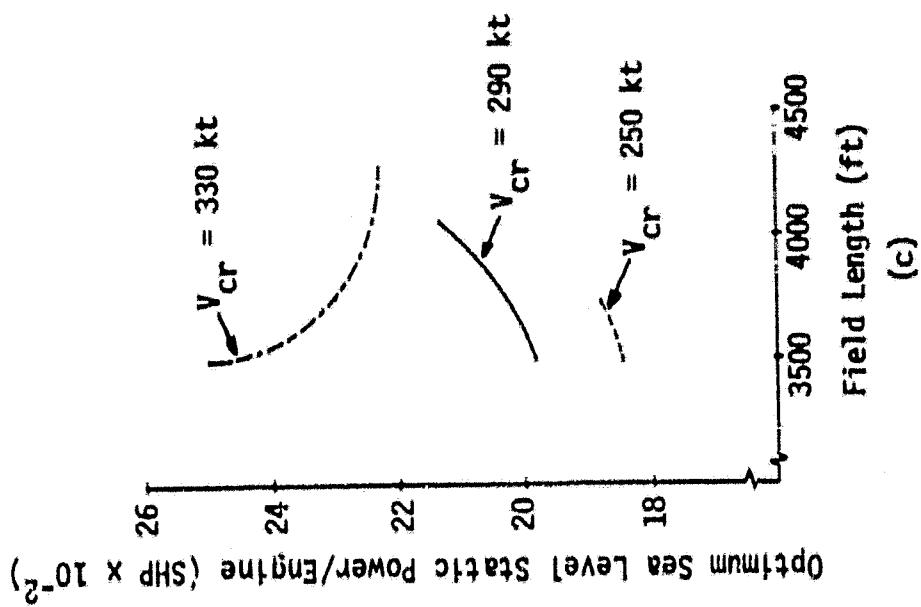
Figure 1. Direct Operating Cost vs. Field Length



(a)



(b)



(c)

Figure 2. Optimal Variable Values vs. Field Length

ORIGINAL PAGE IS  
OF POOR QUALITY

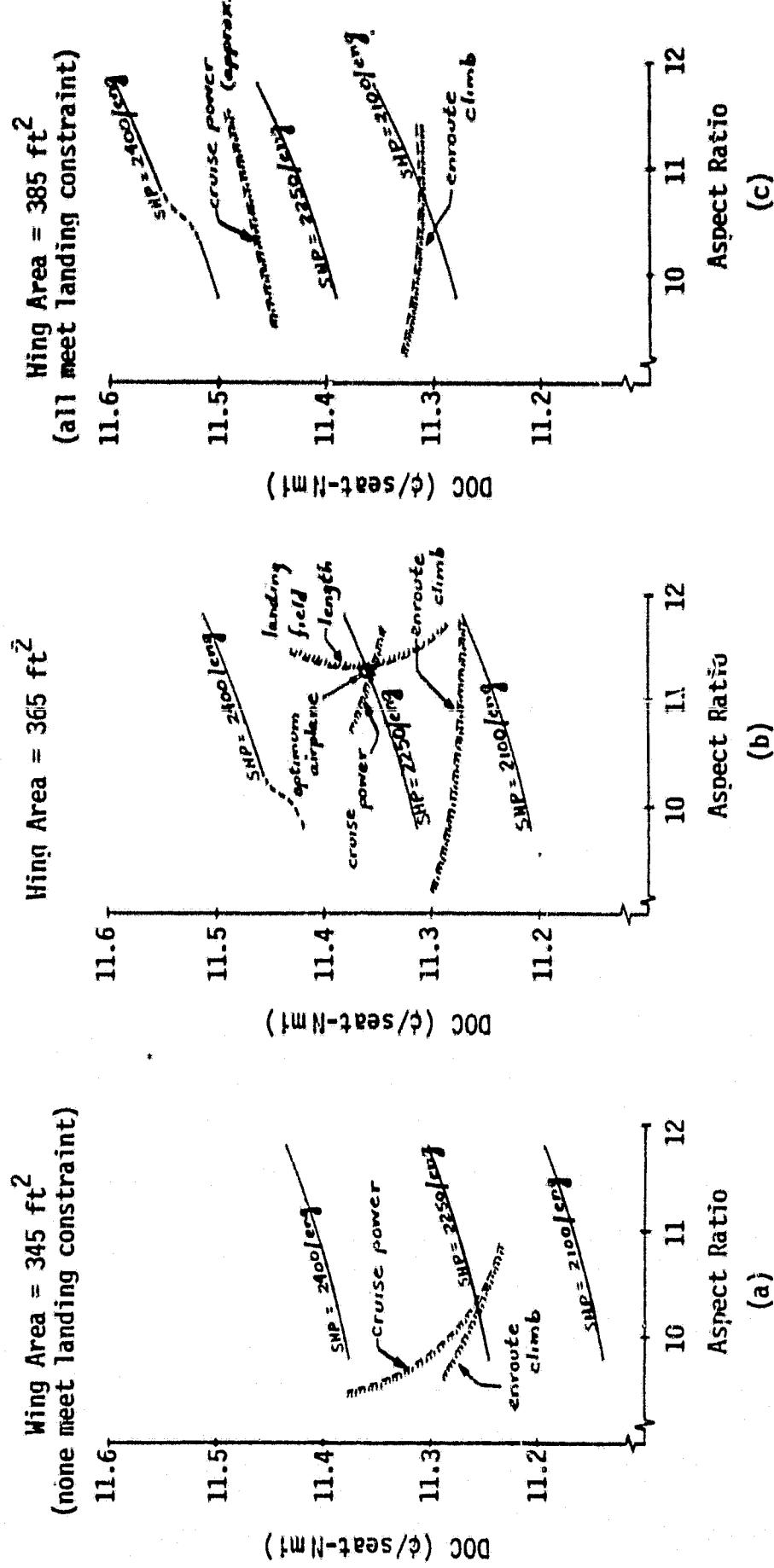


Figure 3. Optimal Point Sensitivities  
Cruise Speed = 330 kt, Maximum Field Length = 4000 feet

ORIGINAL PAGE IS  
OF POOR QUALITY

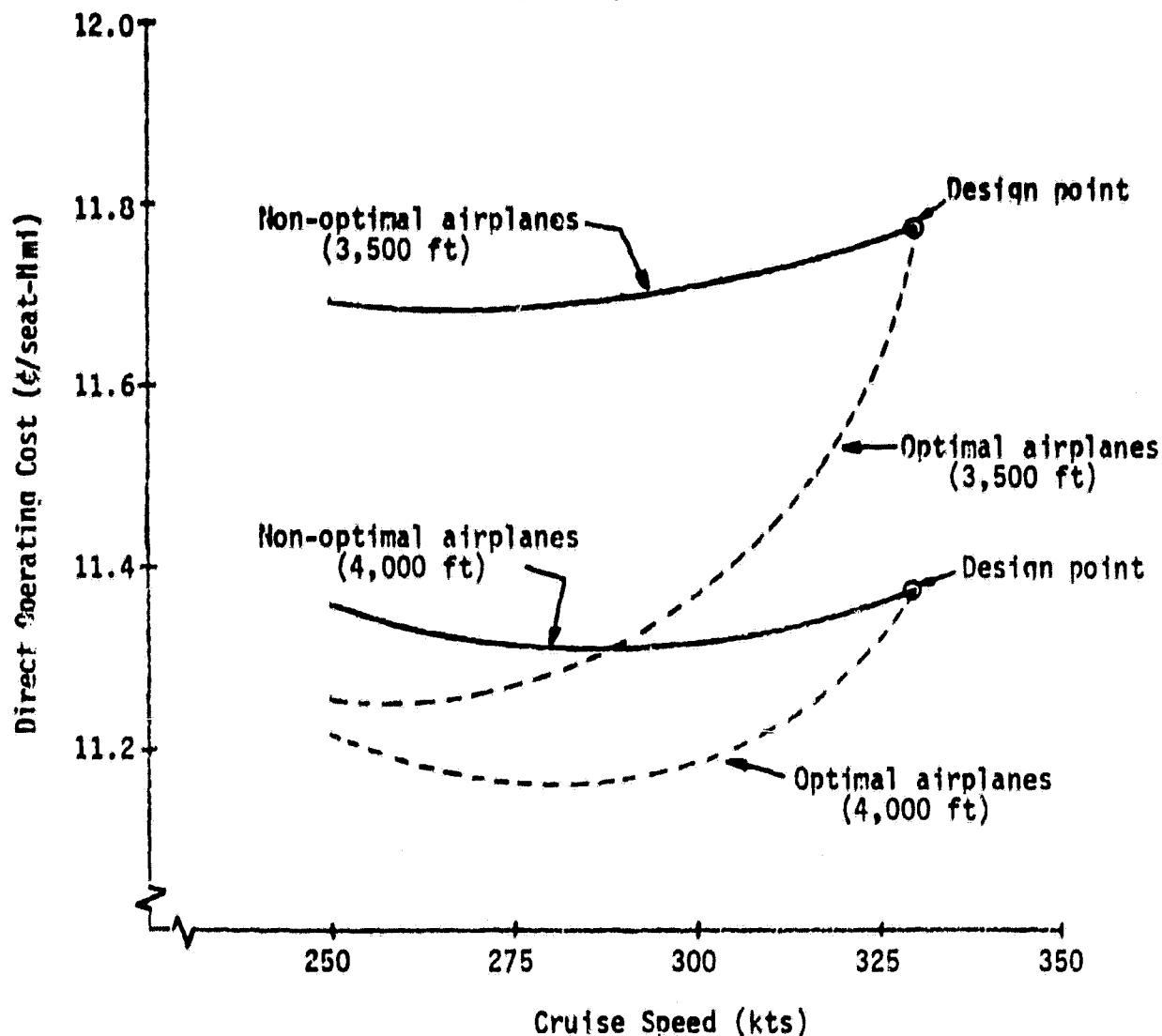


Figure 4. Cost for Non-Optimal Operation  
[330 kt designs flown at lower speeds]

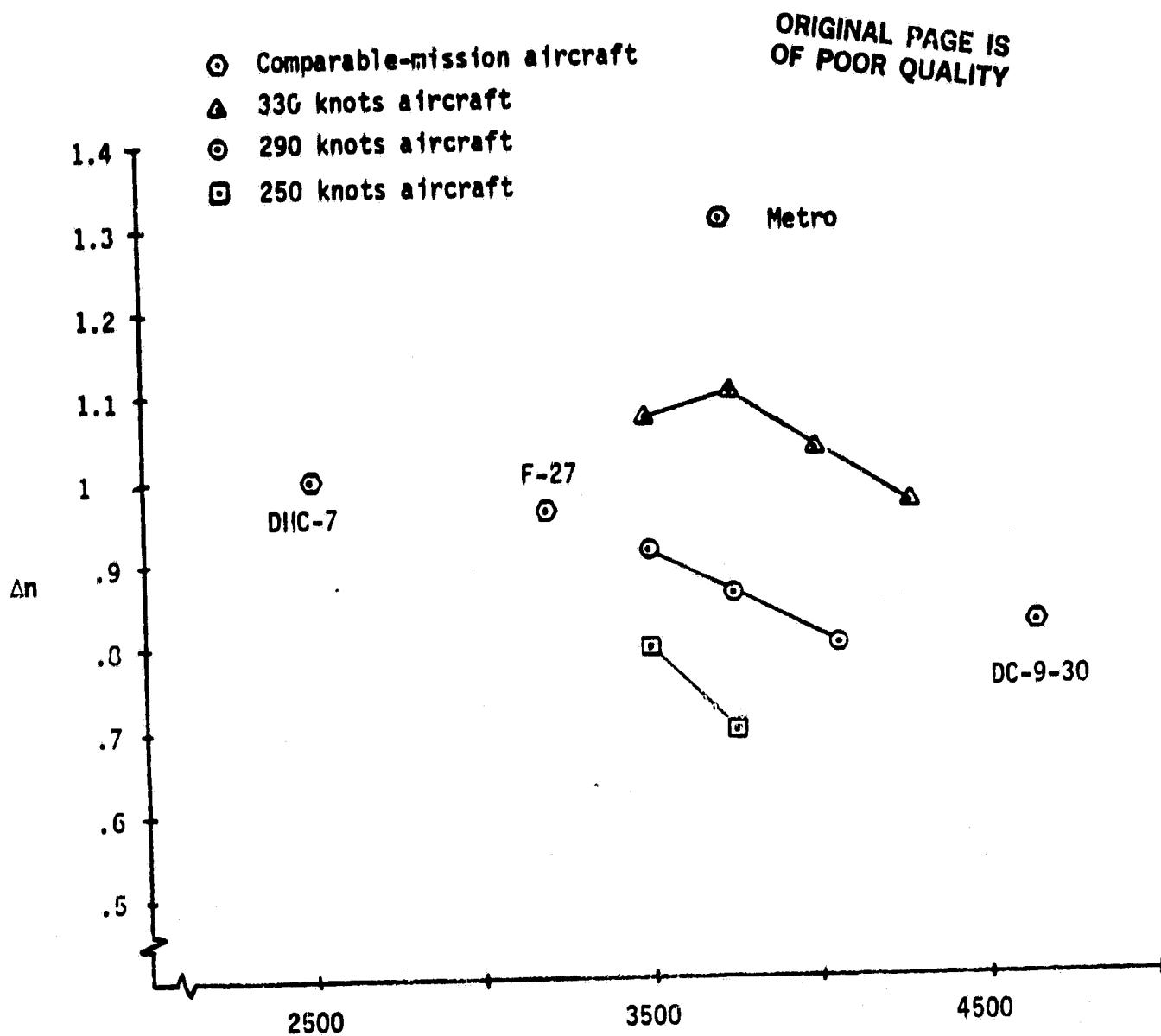
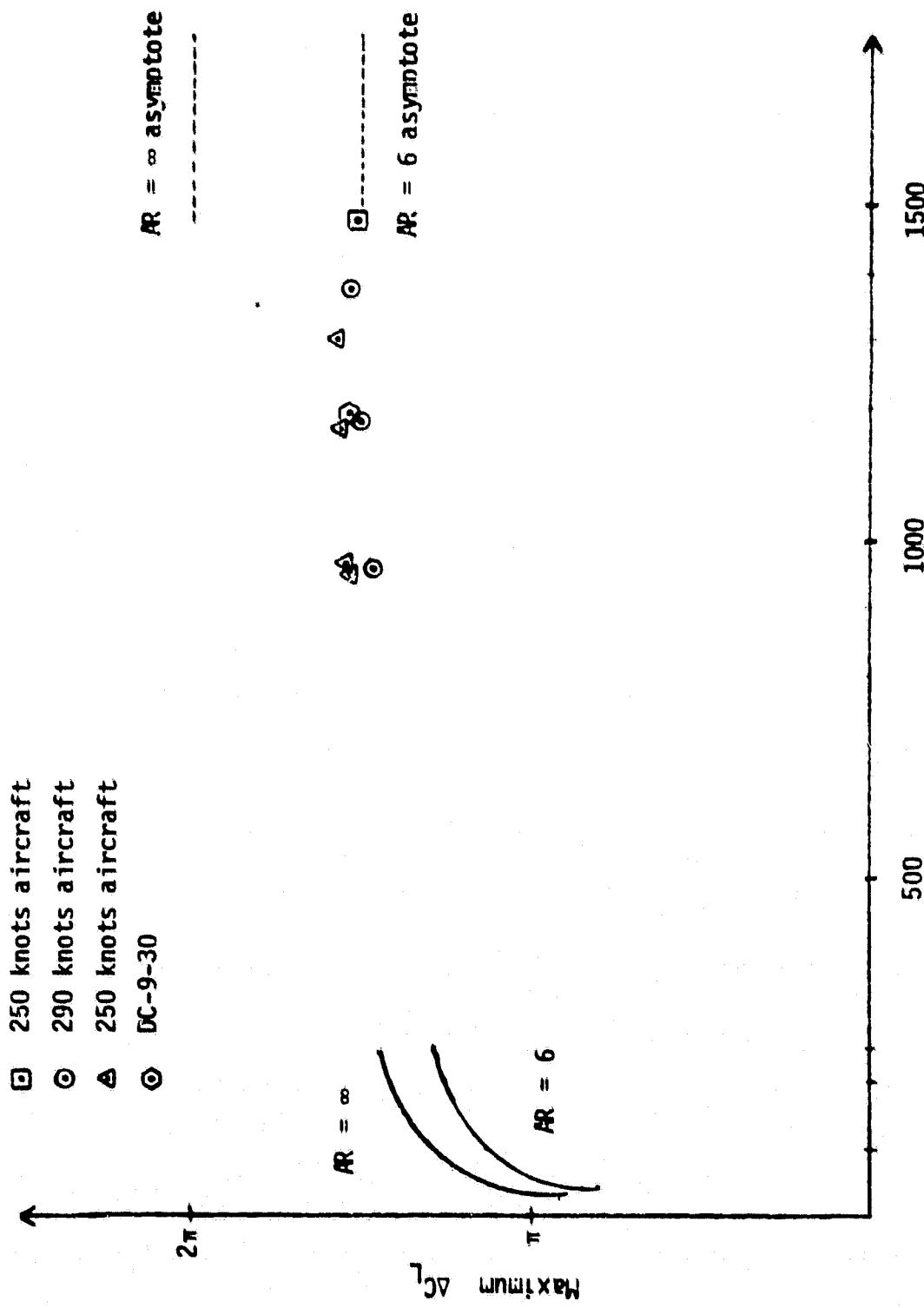


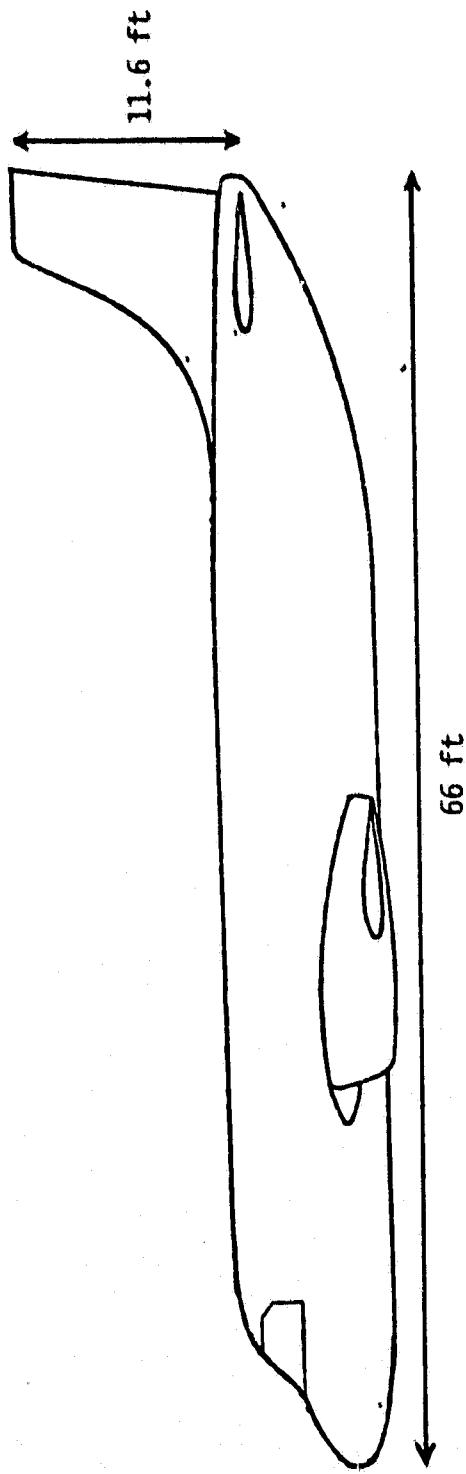
Figure 5. Ride Roughness vs. Field Length

ORIGINAL PAGE IS  
OF POOR QUALITY



$$\text{mass ratio} = \frac{4H}{\rho S C_L}$$

Figure 6. Theoretical Gust Response



Takeoff weight: 25,370 lbs  
Cruise speed : 330 kts  
Field length : 4,000 ft

ORIGINAL PAGE IS  
OF POOR QUALITY

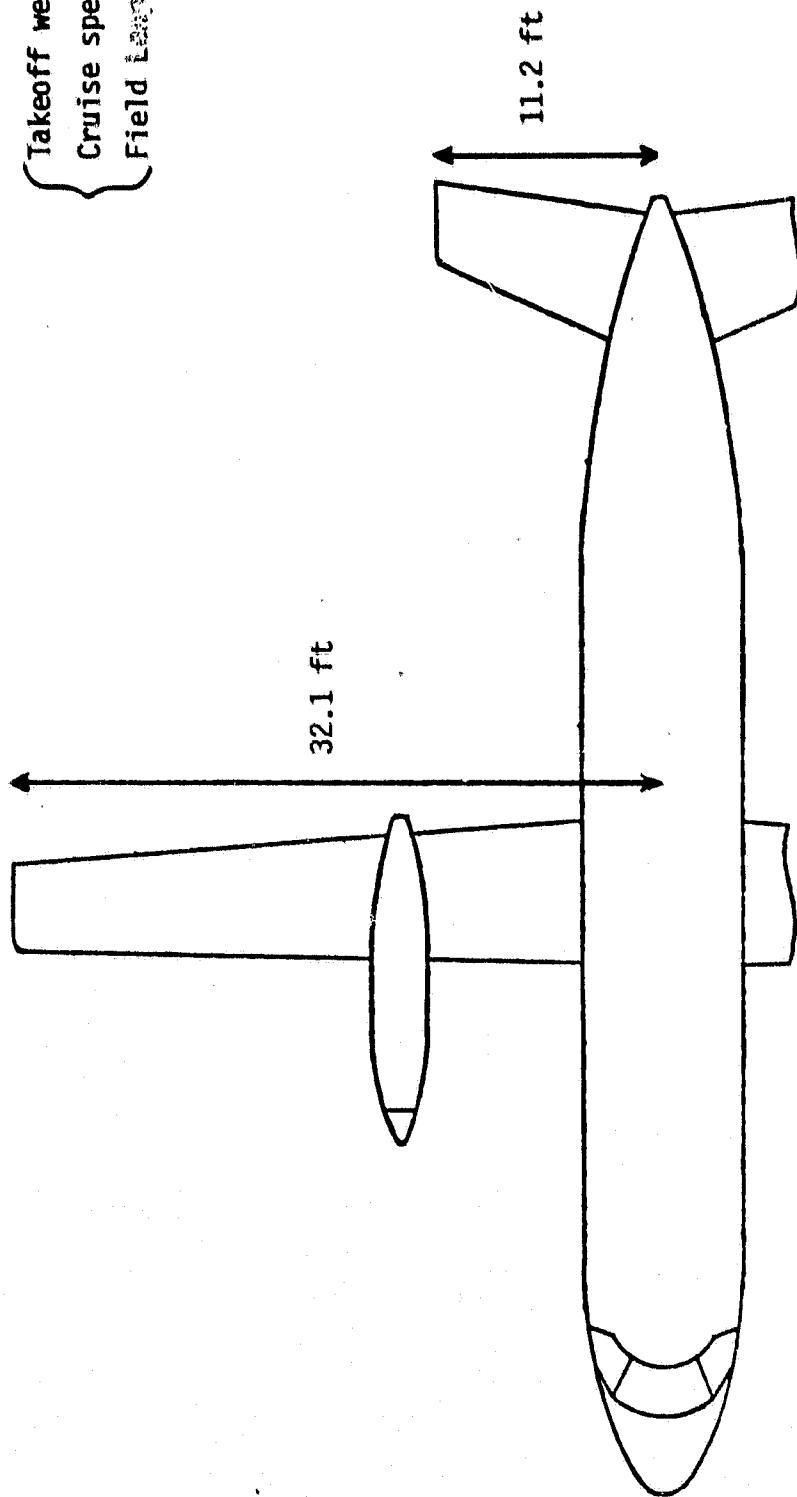


Figure 7. Optimized Commuter Aircraft

## **Appendix I**

### **Program Listing**

```

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINEMCHT=58,SIZE=600K,
      SOURCE,EBCDIC,NOLIST,NOECK,LOAD,MAP,NOEDIT,IO,NOXREF
      C PROGRAM TO MINIMIZE A FUNCTION OF N VARIABLES BY A 'VARIABLE
      C METRIC ALGORITHM', WHICH, IN FACT, IS A SECOND ORDER GRADIENT
      C METHOD USING AN ESTIMATED GRADIENT COMPUTATION (SUBROUTINE
      C GRODENT).

```

ISN 0030 21 CONTINUE  
 ISN 0031 20 B(I,I) = 1.  
 ISN 0032 C CONTINUE  
 C SET COUNTER  
 C ILAST = 16  
 C START MAIN ITERATION LOOP.  
 C  
 ISN 0034 4 KWRITE(6,100)IG,IFN,P0  
 ISN 0035 100 FORMAT(1X,'#GRADS,#FUNCSC,FUNC VALUE = ',2I5,F16.0)  
 ISN 0036 WRITE(6,79)(GR(I),I=1,N),(XB(I),I=1,N)  
 ISN 0037 79 FORMAT(1X,'GRADS = ',3F16.0,/, 'VARIABLES = ',3F16.0)  
 C  
 DO 22 I = 1,N  
 X(I) = XD(I)  
 C(I) = ER(I)  
 CONTINUE  
 C COMPUTE THE STEP VECTOR, T, AND THE DIRECTION INDICATOR, DI.  
 C  
 ISN 0039 ISN 0039  
 ISN 0040 ISN 0040  
 ISN 0041 22 CONTINUE  
 C NOW, DETERMINE WHICH DIRECTION WE ARE TRAVELLING.  
 C  
 ISN 0042 ISN 0042  
 ISN 0043 23 DI = 0.  
 ISN 0044 DO 23 I = 1,N  
 SS = 0.  
 DO 24 J = 1,N  
 SS = SS - B(I,J)\*GR(J)  
 CONTINUE  
 T(I) = SS  
 DI = DI - SS\*GR(I)  
 CONTINUE  
 C NOW, DETERMINE WHICH DIRECTION WE ARE TRAVELLING.  
 C  
 IF (DI .LE. 0.) GO TO 60  
 C IF WE ARE GOING DOWN, CONTINUE THE SEARCH.  
 C  
 ISN 0051 ISN 0051  
 ISN 0052 6 K = 1  
 COUNT = 0  
 C FIND THE NEXT STATE VECTOR.  
 C  
 ISN 0053 ISN 0053  
 ISN 0054 25 DO 25 I = 1,N  
 STEP = K\*T(I)  
 IF (ABS(STEP) .GT. .1) STEP = .1\*STEP/ABS(STEP)  
 XB(I) = X(I) + STEP  
 IF (XB(I) .EQ. X(I)) COUNT = COUNT + 1  
 CONTINUE  
 C DETERMINE CONVERGENCE  
 C  
 ISN 0055 ISN 0055  
 ISN 0056 95 IF (COUNT .GE. N) GO TO 60  
 ISN 0057 CALL EVAL(XB,N,P,FLEN)  
 ISN 0059 96 IFN = IFN + 1  
 ISN 0060 97  
 ISN 0062 98  
 C

C TRY ACCEPTANCE TEST. IF THE NEW POINT IS NOT WITHIN THE TOLERANCE.  
C REQUIRED, CONTINUE THE ITERATION. IF SO, TRY A NEW TOLERANCE.

ISN 0067 C PACC = P0 - D1\*K1TOL  
ISN 0068 IF (P .GE. PACC) GO TO 70  
ISN 0070 P0 = P

ISN 0071 C COMPUTE THE GRADIENT AT THE NEW POINT.

ISN 0072 C CALL GRDENT(P0,XB,N,GR,FLEN)  
ISN 0073 C GR(1) = 2.\*(XB(1) - 1.) - 400.\*XB(1)\*XB(2) - XB(1)\*\*2  
ISN 0074 C GR(2) = 200.\*XB(2) - XB(1)\*\*2  
ISN 0075 C TG = IG + 1

ISN 0076 C COMPUTE GRADIENT DIFFERENCE DIRECTION.

ISN 0077 C D1 = 0.  
ISN 0078 DO 26 I = 1,N  
ISN 0079 T(I) = K\*T(I)  
ISN 0080 C(I) = GR(I) - C(I)  
ISN 0081 D1 = D1 + T(I)\*C(I)

ISN 0082 26 CONTINUE

ISN 0083 C IF (D1 .LE. 0.) GO TO 3

ISN 0084 C HOW DO SOME STUFF THAT I DON'T COMPLETELY UNDERSTAND.

ISN 0085 C HAS TO DO WITH COMPUTING THE RATE OF CHANGE OF THE GRADIENT.

ISN 0086 C D2 = 0.  
ISN 0087 DO 27 I = 1,N  
ISN 0088 SS = 0.  
ISN 0089 DO 28 J = 1,N  
ISN 0090 SS = SS + B(I,J)\*C(J)

ISN 0091 28 CONTINUE

ISN 0092 X(I) = SS  
ISN 0093 D2 = D2 + SS\*C(I)

ISN 0094 27 CONTINUE

ISN 0095 C D2 = 1. + D2/D1  
ISN 0096 DO 29 I = 1,N  
ISN 0097 DO 29 J = 1,N  
ISN 0098 B(I,J) = B(I,J)-(T(I)\*X(J)+X(I)\*T(J))-D2\*T(I)\*T(J)/D1

ISN 0099 29 CONTINUE

ISN 0100 C 60 TO 4

ISN 0101 60 IF (ILAST .NE. 16) GO TO 3  
ISN 0102 60 TO 16

ISN 0103 C K = N\*K  
ISN 0104 70 GO TO 6

ISN 0105 C WRITE(6,102)(XB(I),I=1,N)  
ISN 0106 102 FORMAT(1X,'VARIABLES = ',3F16.8)  
ISN 0107 WRITE(6,100)IG,IFN,PO  
ISN 0108 CALL EVAL(XD,N,P0,FLEN)

ORIGINAL PAGE IS  
OF POOR QUALITY

ORIGINAL PAGE IS  
OF POOR QUALITY

PAGE 004

166.  
167.  
168.  
169.  
170.

CALL INFO(FLRN)

CONTINUE  
STOP  
END

c 500  
99  
ISN 0105  
ISN 0106  
ISN 0107  
ISN 0108  
ISN 0109

C  
C  
C SUBPROGRAM TO COMPUTE THE GRADIENT OF THE FUNCTION AT A POINT.  
C  
ISH 0002 C  
ISH 0003 C DIMENSION B(15),G(15),D(15)  
ISH 0004 C DO 10 I = 1,N  
ISH 0005 C DO 20 J = 1,N  
ISH 0006 C D(I,J) = B(I,J)  
ISH 0007 C CONTINUE  
ISH 0008 C H = ABS(B(I)) + 1.E-2\*I1.E-2  
ISH 0009 C D(I) = B(I) + H  
ISH 0010 C CALL EVAL(0,N,SP,FLEN)  
ISH 0011 C G(I) = (SP - S)/H  
ISH 0012 C CONTINUE  
ISH 0013 C RETURN  
ISH 0014 C  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91

ORIGINAL PAGE IS  
OF POOR QUALITY.



COMPILER OPTIONS - NAME=MAIN,OPT=02,LINECNT=58,SIZE=00000K,  
SOURCE,EBCDIC,NOLIST,NOECK,LOAD,MAP,NOEDIT,JO,NOXREF

ORIGINAL PAGE IS  
OF POOR QUALITY

( JUN 74 ) 05/360 FORTRAN H  
COMPILER OPTIONS - NAME= MAIN.OPT=02,LINECNT=58,SIZE=0000K,  
SUBCFC,FECDC,NOLIST,NODECK,LOAD,MAP,NDEDIT,IO,NOXREF

```

342.      SN 0002      C
343.      C   SUBROUTINE RANGE(DIST,MCT,VCRUS)
344.      C
345.      C   REAL HAVING
346.      C   COMMON/VARIB/TOW,S,SLSHIP,AR,VCLIMB,HCRUS
347.      C   COMMON/WEIGHT/QHEPL,AFHT
348.      C   COMMON/CONST/PI,G
349.      C   COMMON/CHARAC/COP,E,B
350.      C   COMMON/FLAG/KANT
351.      C   COMMON/AERO/VEQH,CLALPH
352.      C   COMMON/AERO/VEQH,CLALPH
353.      C
354.      C   COMPUTES THE TOW TO MEET THE MAXIMUM RANGE MISSION USING A FITTED
355.      C   PARABOLA MINIMIZATION PROCEDURE. USES A GRID SEARCH TO FIND THE
356.      C   BEST CLIMB SPEED AND CRUISE ALTITUDE FOR THE GIVEN AIRPLANE.
357.      C
358.      SN 0010      C
359.      SN 0011      C
360.      SN 0012      C
361.      C
362.      C   BEGIN SEARCH LOOP ON CLIMB SPEED.
363.      C
364.      DO 200 I = 250,300,25
365.      C
366.      CLMSRD = FLOAT(I)
367.      C
368.      C   BEGIN SEARCH LOOP ON CRUISE ALTITUDE.
369.      C
370.      DO 100 J = 20000,30000,5000
371.      HMAX = FLOAT(J)
372.      WRITE(6,300) HMAX,CLMSPO
373.      C
374.      300  FORMAT(//, HCRUS,VCLM = ',2F14.4)
375.      JINMAX = 1
376.      TEMP = 516.69 - .00356*HMAX
377.      XHACH = VCRUS/SQRT(1.4*1710.*TEMP)
378.      BETA2 = 1 - (1.07*XHACH)*X2
379.      CLALPH = (2.*PI*AR)/(2.+SQRT(AR*X2*BETA2*1.062844.))
380.      C
381.      C   CALCULATE DESIGN EQUIVALENT STRUCTURAL SPEED AT 10000 FT.
382.      C
383.      VECH = 1.07*VCRUS*SQRT(.0017553/.0023769)
384.      C
385.      RHO = 2.3769E-3*(1. - 6.0634E-6*HMAX)*(4.2648)
386.      Q = .5 * RHO * VCRUS*X2
387.      C
388.      C   GUESS TOW AND FIND EMPTY WEIGHT.
389.      C
390.      X2 = 25000.
391.      STEP = X2/50.
392.      HAVING = 600.
393.      C
394.      C   COMMON/FLAG/KANT
395.      C   COMMON/AERO/VEQH,CLALPH
396.      C
397.      C   COMMON/AERO/VEQH,CLALPH
398.      C
399.      C   COMMON/FLAG/KANT
400.      C
401.      C   COMMON/AERO/VEQH,CLALPH
402.      C
403.      C   COMMON/FLAG/KANT
404.      C
405.      C   COMMON/AERO/VEQH,CLALPH
406.      C
407.      C   COMMON/FLAG/KANT
408.      C
409.      C   COMMON/AERO/VEQH,CLALPH
410.      C
411.      C   COMMON/FLAG/KANT
412.      C
413.      C   COMMON/AERO/VEQH,CLALPH
414.      C
415.      C   COMMON/FLAG/KANT
416.      C
417.      C   COMMON/AERO/VEQH,CLALPH
418.      C
419.      C   COMMON/FLAG/KANT
420.      C
421.      C   COMMON/AERO/VEQH,CLALPH
422.      C
423.      C   COMMON/FLAG/KANT
424.      C
425.      C   COMMON/AERO/VEQH,CLALPH
426.      C
427.      C   COMMON/FLAG/KANT
428.      C
429.      C   COMMON/AERO/VEQH,CLALPH
430.      C
431.      C   COMMON/FLAG/KANT
432.      C
433.      C   COMMON/AERO/VEQH,CLALPH
434.      C
435.      C   COMMON/FLAG/KANT
436.      C
437.      C   COMMON/AERO/VEQH,CLALPH
438.      C
439.      C   COMMON/FLAG/KANT
440.      C
441.      C   COMMON/AERO/VEQH,CLALPH
442.      C
443.      C   COMMON/FLAG/KANT
444.      C
445.      C   COMMON/AERO/VEQH,CLALPH
446.      C
447.      C   COMMON/FLAG/KANT
448.      C
449.      C   COMMON/AERO/VEQH,CLALPH
450.      C
451.      C   COMMON/FLAG/KANT
452.      C
453.      C   COMMON/AERO/VEQH,CLALPH
454.      C
455.      C   COMMON/FLAG/KANT
456.      C
457.      C   COMMON/AERO/VEQH,CLALPH
458.      C
459.      C   COMMON/FLAG/KANT
460.      C
461.      C   COMMON/AERO/VEQH,CLALPH
462.      C
463.      C   COMMON/FLAG/KANT
464.      C
465.      C   COMMON/AERO/VEQH,CLALPH
466.      C
467.      C   COMMON/FLAG/KANT
468.      C
469.      C   COMMON/AERO/VEQH,CLALPH
470.      C
471.      C   COMMON/FLAG/KANT
472.      C
473.      C   COMMON/AERO/VEQH,CLALPH
474.      C
475.      C   COMMON/FLAG/KANT
476.      C
477.      C   COMMON/AERO/VEQH,CLALPH
478.      C
479.      C   COMMON/FLAG/KANT
480.      C
481.      C   COMMON/AERO/VEQH,CLALPH
482.      C
483.      C   COMMON/FLAG/KANT
484.      C
485.      C   COMMON/AERO/VEQH,CLALPH
486.      C
487.      C   COMMON/FLAG/KANT
488.      C
489.      C   COMMON/AERO/VEQH,CLALPH
490.      C
491.      C   COMMON/FLAG/KANT
492.      C
493.      C   COMMON/AERO/VEQH,CLALPH
494.      C
495.      C   COMMON/FLAG/KANT
496.      C
497.      C   COMMON/AERO/VEQH,CLALPH
498.      C
499.      C   COMMON/FLAG/KANT
500.      C
501.      C   COMMON/AERO/VEQH,CLALPH
502.      C
503.      C   COMMON/FLAG/KANT
504.      C
505.      C   COMMON/AERO/VEQH,CLALPH
506.      C
507.      C   COMMON/FLAG/KANT
508.      C
509.      C   COMMON/AERO/VEQH,CLALPH
510.      C
511.      C   COMMON/FLAG/KANT
512.      C
513.      C   COMMON/AERO/VEQH,CLALPH
514.      C
515.      C   COMMON/FLAG/KANT
516.      C
517.      C   COMMON/AERO/VEQH,CLALPH
518.      C
519.      C   COMMON/FLAG/KANT
520.      C
521.      C   COMMON/AERO/VEQH,CLALPH
522.      C
523.      C   COMMON/FLAG/KANT
524.      C
525.      C   COMMON/AERO/VEQH,CLALPH
526.      C
527.      C   COMMON/FLAG/KANT
528.      C
529.      C   COMMON/AERO/VEQH,CLALPH
530.      C
531.      C   COMMON/FLAG/KANT
532.      C
533.      C   COMMON/AERO/VEQH,CLALPH
534.      C
535.      C   COMMON/FLAG/KANT
536.      C
537.      C   COMMON/AERO/VEQH,CLALPH
538.      C
539.      C   COMMON/FLAG/KANT
540.      C
541.      C   COMMON/AERO/VEQH,CLALPH
542.      C
543.      C   COMMON/FLAG/KANT
544.      C
545.      C   COMMON/AERO/VEQH,CLALPH
546.      C
547.      C   COMMON/FLAG/KANT
548.      C
549.      C   COMMON/AERO/VEQH,CLALPH
550.      C
551.      C   COMMON/FLAG/KANT
552.      C
553.      C   COMMON/AERO/VEQH,CLALPH
554.      C
555.      C   COMMON/FLAG/KANT
556.      C
557.      C   COMMON/AERO/VEQH,CLALPH
558.      C
559.      C   COMMON/FLAG/KANT
560.      C
561.      C   COMMON/AERO/VEQH,CLALPH
562.      C
563.      C   COMMON/FLAG/KANT
564.      C
565.      C   COMMON/AERO/VEQH,CLALPH
566.      C
567.      C   COMMON/FLAG/KANT
568.      C
569.      C   COMMON/AERO/VEQH,CLALPH
570.      C
571.      C   COMMON/FLAG/KANT
572.      C
573.      C   COMMON/AERO/VEQH,CLALPH
574.      C
575.      C   COMMON/FLAG/KANT
576.      C
577.      C   COMMON/AERO/VEQH,CLALPH
578.      C
579.      C   COMMON/FLAG/KANT
580.      C
581.      C   COMMON/AERO/VEQH,CLALPH
582.      C
583.      C   COMMON/FLAG/KANT
584.      C
585.      C   COMMON/AERO/VEQH,CLALPH
586.      C
587.      C   COMMON/FLAG/KANT
588.      C
589.      C   COMMON/AERO/VEQH,CLALPH
590.      C
591.      C   COMMON/FLAG/KANT
592.      C
593.      C   COMMON/AERO/VEQH,CLALPH
594.      C
595.      C   COMMON/FLAG/KANT
596.      C
597.      C   COMMON/AERO/VEQH,CLALPH
598.      C
599.      C   COMMON/FLAG/KANT
600.      C
601.      C   COMMON/AERO/VEQH,CLALPH
602.      C
603.      C   COMMON/FLAG/KANT
604.      C
605.      C   COMMON/AERO/VEQH,CLALPH
606.      C
607.      C   COMMON/FLAG/KANT
608.      C
609.      C   COMMON/AERO/VEQH,CLALPH
610.      C
611.      C   COMMON/FLAG/KANT
612.      C
613.      C   COMMON/AERO/VEQH,CLALPH
614.      C
615.      C   COMMON/FLAG/KANT
616.      C
617.      C   COMMON/AERO/VEQH,CLALPH
618.      C
619.      C   COMMON/FLAG/KANT
620.      C
621.      C   COMMON/AERO/VEQH,CLALPH
622.      C
623.      C   COMMON/FLAG/KANT
624.      C
625.      C   COMMON/AERO/VEQH,CLALPH
626.      C
627.      C   COMMON/FLAG/KANT
628.      C
629.      C   COMMON/AERO/VEQH,CLALPH
630.      C
631.      C   COMMON/FLAG/KANT
632.      C
633.      C   COMMON/AERO/VEQH,CLALPH
634.      C
635.      C   COMMON/FLAG/KANT
636.      C
637.      C   COMMON/AERO/VEQH,CLALPH
638.      C
639.      C   COMMON/FLAG/KANT
640.      C
641.      C   COMMON/AERO/VEQH,CLALPH
642.      C
643.      C   COMMON/FLAG/KANT
644.      C
645.      C   COMMON/AERO/VEQH,CLALPH
646.      C
647.      C   COMMON/FLAG/KANT
648.      C
649.      C   COMMON/AERO/VEQH,CLALPH
650.      C
651.      C   COMMON/FLAG/KANT
652.      C
653.      C   COMMON/AERO/VEQH,CLALPH
654.      C
655.      C   COMMON/FLAG/KANT
656.      C
657.      C   COMMON/AERO/VEQH,CLALPH
658.      C
659.      C   COMMON/FLAG/KANT
660.      C
661.      C   COMMON/AERO/VEQH,CLALPH
662.      C
663.      C   COMMON/FLAG/KANT
664.      C
665.      C   COMMON/AERO/VEQH,CLALPH
666.      C
667.      C   COMMON/FLAG/KANT
668.      C
669.      C   COMMON/AERO/VEQH,CLALPH
670.      C
671.      C   COMMON/FLAG/KANT
672.      C
673.      C   COMMON/AERO/VEQH,CLALPH
674.      C
675.      C   COMMON/FLAG/KANT
676.      C
677.      C   COMMON/AERO/VEQH,CLALPH
678.      C
679.      C   COMMON/FLAG/KANT
680.      C
681.      C   COMMON/AERO/VEQH,CLALPH
682.      C
683.      C   COMMON/FLAG/KANT
684.      C
685.      C   COMMON/AERO/VEQH,CLALPH
686.      C
687.      C   COMMON/FLAG/KANT
688.      C
689.      C   COMMON/AERO/VEQH,CLALPH
690.      C
691.      C   COMMON/FLAG/KANT
692.      C
693.      C   COMMON/AERO/VEQH,CLALPH
694.      C
695.      C   COMMON/FLAG/KANT
696.      C
697.      C   COMMON/AERO/VEQH,CLALPH
698.      C
699.      C   COMMON/FLAG/KANT
700.      C
701.      C   COMMON/AERO/VEQH,CLALPH
702.      C
703.      C   COMMON/FLAG/KANT
704.      C
705.      C   COMMON/AERO/VEQH,CLALPH
706.      C
707.      C   COMMON/FLAG/KANT
708.      C
709.      C   COMMON/AERO/VEQH,CLALPH
710.      C
711.      C   COMMON/FLAG/KANT
712.      C
713.      C   COMMON/AERO/VEQH,CLALPH
714.      C
715.      C   COMMON/FLAG/KANT
716.      C
717.      C   COMMON/AERO/VEQH,CLALPH
718.      C
719.      C   COMMON/FLAG/KANT
720.      C
721.      C   COMMON/AERO/VEQH,CLALPH
722.      C
723.      C   COMMON/FLAG/KANT
724.      C
725.      C   COMMON/AERO/VEQH,CLALPH
726.      C
727.      C   COMMON/FLAG/KANT
728.      C
729.      C   COMMON/AERO/VEQH,CLALPH
730.      C
731.      C   COMMON/FLAG/KANT
732.      C
733.      C   COMMON/AERO/VEQH,CLALPH
734.      C
735.      C   COMMON/FLAG/KANT
736.      C
737.      C   COMMON/AERO/VEQH,CLALPH
738.      C
739.      C   COMMON/FLAG/KANT
740.      C
741.      C   COMMON/AERO/VEQH,CLALPH
742.      C
743.      C   COMMON/FLAG/KANT
744.      C
745.      C   COMMON/AERO/VEQH,CLALPH
746.      C
747.      C   COMMON/FLAG/KANT
748.      C
749.      C   COMMON/AERO/VEQH,CLALPH
750.      C
751.      C   COMMON/FLAG/KANT
752.      C
753.      C   COMMON/AERO/VEQH,CLALPH
754.      C
755.      C   COMMON/FLAG/KANT
756.      C
757.      C   COMMON/AERO/VEQH,CLALPH
758.      C
759.      C   COMMON/FLAG/KANT
760.      C
761.      C   COMMON/AERO/VEQH,CLALPH
762.      C
763.      C   COMMON/FLAG/KANT
764.      C
765.      C   COMMON/AERO/VEQH,CLALPH
766.      C
767.      C   COMMON/FLAG/KANT
768.      C
769.      C   COMMON/AERO/VEQH,CLALPH
770.      C
771.      C   COMMON/FLAG/KANT
772.      C
773.      C   COMMON/AERO/VEQH,CLALPH
774.      C
775.      C   COMMON/FLAG/KANT
776.      C
777.      C   COMMON/AERO/VEQH,CLALPH
778.      C
779.      C   COMMON/FLAG/KANT
780.      C
781.      C   COMMON/AERO/VEQH,CLALPH
782.      C
783.      C   COMMON/FLAG/KANT
784.      C
785.      C   COMMON/AERO/VEQH,CLALPH
786.      C
787.      C   COMMON/FLAG/KANT
788.      C
789.      C   COMMON/AERO/VEQH,CLALPH
790.      C
791.      C   COMMON/FLAG/KANT
792.      C
793.      C   COMMON/AERO/VEQH,CLALPH
794.      C
795.      C   COMMON/FLAG/KANT
796.      C
797.      C   COMMON/AERO/VEQH,CLALPH
798.      C
799.      C   COMMON/FLAG/KANT
800.      C
801.      C   COMMON/AERO/VEQH,CLALPH
802.      C
803.      C   COMMON/FLAG/KANT
804.      C
805.      C   COMMON/AERO/VEQH,CLALPH
806.      C
807.      C   COMMON/FLAG/KANT
808.      C
809.      C   COMMON/AERO/VEQH,CLALPH
810.      C
811.      C   COMMON/FLAG/KANT
812.      C
813.      C   COMMON/AERO/VEQH,CLALPH
814.      C
815.      C   COMMON/FLAG/KANT
816.      C
817.      C   COMMON/AERO/VEQH,CLALPH
818.      C
819.      C   COMMON/FLAG/KANT
820.      C
821.      C   COMMON/AERO/VEQH,CLALPH
822.      C
823.      C   COMMON/FLAG/KANT
824.      C
825.      C   COMMON/AERO/VEQH,CLALPH
826.      C
827.      C   COMMON/FLAG/KANT
828.      C
829.      C   COMMON/AERO/VEQH,CLALPH
830.      C
831.      C   COMMON/FLAG/KANT
832.      C
833.      C   COMMON/AERO/VEQH,CLALPH
834.      C
835.      C   COMMON/FLAG/KANT
836.      C
837.      C   COMMON/AERO/VEQH,CLALPH
838.      C
839.      C   COMMON/FLAG/KANT
840.      C
841.      C   COMMON/AERO/VEQH,CLALPH
842.      C
843.      C   COMMON/FLAG/KANT
844.      C
845.      C   COMMON/AERO/VEQH,CLALPH
846.      C
847.      C   COMMON/FLAG/KANT
848.      C
849.      C   COMMON/AERO/VEQH,CLALPH
850.      C
851.      C   COMMON/FLAG/KANT
852.      C
853.      C   COMMON/AERO/VEQH,CLALPH
854.      C
855.      C   COMMON/FLAG/KANT
856.      C
857.      C   COMMON/AERO/VEQH,CLALPH
858.      C
859.      C   COMMON/FLAG/KANT
860.      C
861.      C   COMMON/AERO/VEQH,CLALPH
862.      C
863.      C   COMMON/FLAG/KANT
864.      C
865.      C   COMMON/AERO/VEQH,CLALPH
866.      C
867.      C   COMMON/FLAG/KANT
868.      C
869.      C   COMMON/AERO/VEQH,CLALPH
870.      C
871.      C   COMMON/FLAG/KANT
872.      C
873.      C   COMMON/AERO/VEQH,CLALPH
874.      C
875.      C   COMMON/FLAG/KANT
876.      C
877.      C   COMMON/AERO/VEQH,CLALPH
878.      C
879.      C   COMMON/FLAG/KANT
880.      C
881.      C   COMMON/AERO/VEQH,CLALPH
882.      C
883.      C   COMMON/FLAG/KANT
884.      C
885.      C   COMMON/AERO/VEQH,CLALPH
886.      C
887.      C   COMMON/FLAG/KANT
888.      C
889.      C   COMMON/AERO/VEQH,CLALPH
890.      C
891.      C   COMMON/FLAG/KANT
892.      C
893.      C   COMMON/AERO/VEQH,CLALPH
894.      C
895.      C   COMMON/FLAG/KANT
896.      C
897.      C   COMMON/AERO/VEQH,CLALPH
898.      C
899.      C   COMMON/FLAG/KANT
900.      C
901.      C   COMMON/AERO/VEQH,CLALPH
902.      C
903.      C   COMMON/FLAG/KANT
904.      C
905.      C   COMMON/AERO/VEQH,CLALPH
906.      C
907.      C   COMMON/FLAG/KANT
908.      C
909.      C   COMMON/AERO/VEQH,CLALPH
910.      C
911.      C   COMMON/FLAG/KANT
912.      C
913.      C   COMMON/AERO/VEQH,CLALPH
914.      C
915.      C   COMMON/FLAG/KANT
916.      C
917.      C   COMMON/AERO/VEQH,CLALPH
918.      C
919.      C   COMMON/FLAG/KANT
920.      C
921.      C   COMMON/AERO/VEQH,CLALPH
922.      C
923.      C   COMMON/FLAG/KANT
924.      C
925.      C   COMMON/AERO/VEQH,CLALPH
926.      C
927.      C   COMMON/FLAG/KANT
928.      C
929.      C   COMMON/AERO/VEQH,CLALPH
930.      C
931.      C   COMMON/FLAG/KANT
932.      C
933.      C   COMMON/AERO/VEQH,CLALPH
934.      C
935.      C   COMMON/FLAG/KANT
936.      C
937.      C   COMMON/AERO/VEQH,CLALPH
938.      C
939.      C   COMMON/FLAG/KANT
940.      C
941.      C   COMMON/AERO/VEQH,CLALPH
942.      C
943.      C   COMMON/FLAG/KANT
944.      C
945.      C   COMMON/AERO/VEQH,CLALPH
946.      C
947.      C   COMMON/FLAG/KANT
948.      C
949.      C   COMMON/AERO/VEQH,CLALPH
950.      C
951.      C   COMMON/FLAG/KANT
952.      C
953.      C   COMMON/AERO/VEQH,CLALPH
954.      C
955.      C   COMMON/FLAG/KANT
956.      C
957.      C   COMMON/AERO/VEQH,CLALPH
958.      C
959.      C   COMMON/FLAG/KANT
960.      C
961.      C   COMMON/AERO/VEQH,CLALPH
962.      C
963.      C   COMMON/FLAG/KANT
964.      C
965.      C   COMMON/AERO/VEQH,CLALPH
966.      C
967.      C   COMMON/FLAG/KANT
968.      C
969.      C   COMMON/AERO/VEQH,CLALPH
970.      C
971.      C   COMMON/FLAG/KANT
972.      C
973.      C   COMMON/AERO/VEQH,CLALPH
974.      C
975.      C   COMMON/FLAG/KANT
976.      C
977.      C   COMMON/AERO/VEQH,CLALPH
978.      C
979.      C   COMMON/FLAG/KANT
980.      C
981.      C   COMMON/AERO/VEQH,CLALPH
982.      C
983.      C   COMMON/FLAG/KANT
984.      C
985.      C   COMMON/AERO/VEQH,CLALPH
986.      C
987.      C   COMMON/FLAG/KANT
988.      C
989.      C   COMMON/AERO/VEQH,CLALPH
990.      C
991.      C   COMMON/FLAG/KANT
992.      C
993.      C   COMMON/AERO/VEQH,CLALPH
994.      C
995.      C   COMMON/FLAG/KANT
996.      C
997.      C   COMMON/AERO/VEQH,CLALPH
998.      C
999.      C   COMMON/FLAG/KANT

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

394.
395.
396.
397.
398.
399.
400.
401.
402.
403.
404.
405.
406.
407.
408.
409.
410.
411.
412.
413.
414.
415.
416.
417.
418.
419.
420.
421.
422.
423.
424.
425.
426.
427.
428.
429.
430.
431.
432.
433.
434.
435.
436.
437.
438.
439.
440.
441.
442.
443.
444.
445.
446.
447.
448.
449.

C NOW CALL THE FUNCTION EVALUATION ROUTINE TO FIND THE TOW FOR THE
C GIVEN RANGE.
C MINIMIZATION ROUTINE
C
C CALL DISTR(X2,F2,H,VCL,MD,Q,MAXRG,MC,FI,VCRUS,ZFH,WTAF)
C
C TEST FOR CLIMB VIOLATION. KANT IS 1 IF CLIMB GRADIENT < 0.
C
      ISN 0040      IF (KANT .EQ. 0) GO TO 10
      ISN 0041      IF (ICOUNT .EQ. 1) GO TO 30
      ISN 0042      IF (ICOUNT .NE. 1) GO TO 100
      ISN 0043      X3 = X2 + STEP
      ISN 0044      X1 = X2 - STEP
      ISN 0045      CALL DISTR(X1,F1,H,VCL,MD,Q,MAXRG,MC,FI,VCRUS,ZFH,WTAF)
      ISN 0046      CALL DISTR(X3,F3,H,VCL,MD,Q,MAXRG,MC,FI,VCRUS,ZFH,WTAF)
      ISN 0047      INTRRN = INTRRN + 1
      ISN 0048      IF (INTRRN .GT. 50) GO TO 89
      ISN 0049      IF (F1 .LE. F2) GO TO 1
      ISN 0050      IF (F3 .LE. F2) GO TO 2
      ISN 0051      WHERE F2 IS LESS THAN F1 AND F3, FIT A PARABOLA, FIND MINIMUM AND
      ISN 0052      USE THAT AS THE NEXT X2.
      ISN 0053      X2 = X2 + .5*STEP*(F1-F3)/(F3-2.*F2+F1)
      ISN 0054      CALL DISTR(X2,F2,H,VCL,MD,Q,MAXRG,MC,FI,VCRUS,ZFH,WTAF)
      ISN 0055      IF (F2 .LE. CONV) GO TO 99
      ISN 0056      STEP = STEP/3.
      ISN 0057      GO TO 10
      ISN 0058      WHERE F2 IS BETWEEN OR GREATER THAN F1 AND F3, MAKE X2 THE VALUE
      ISN 0059      FOR A MINIMUM
      ISN 0060      IF (F3 .LT. F1) GO TO 2
      ISN 0061      X3 = X2
      ISN 0062      F2 = F1
      ISN 0063      X1 = X1 - STEP
      ISN 0064      CALL DISTR(X1,F1,H,VCL,MD,Q,MAXRG,MC,FI,VCRUS,ZFH,WTAF)
      ISN 0065      GO TO 3
      ISN 0066      X1 = X2
      ISN 0067      F1 = F2
      ISN 0068      X2 = X3
      ISN 0069      F2 = F3
      ISN 0070      X3 = X3 + STEP
      ISN 0071      CALL DISTR(X3,F3,H,VCL,MD,Q,MAXRG,MC,FI,VCRUS,ZFH,WTAF)
      ISN 0072      GO TO 3
      ISN 0073      NOT CONVERGING IF INTRRN > 50. PRINT ERROR MESSAGE.
      ISN 0074      WRITE(6,302)
      ISN 0075      302 FORMAT(' INTRRN = 50, CONTINUING ...')
      ISN 0076
      ISN 0077
      ISN 0078
      ISN 0079
      ISN 0080
      ISN 0081
      ISN 0082
      ISN 0083
      ISN 0084
      ISN 0085
      ISN 0086
      ISN 0087
      ISN 0088
      ISN 0089
      ISN 0090
      ISN 0091
      ISN 0092
      ISN 0093
      ISN 0094
      ISN 0095
      ISN 0096
      ISN 0097
      ISN 0098
      ISN 0099
      ISN 0100
      ISN 0101
      ISN 0102
      ISN 0103
      ISN 0104
      ISN 0105
      ISN 0106
      ISN 0107
      ISN 0108
      ISN 0109
      ISN 0110
      ISN 0111
      ISN 0112
      ISN 0113
      ISN 0114
      ISN 0115
      ISN 0116
      ISN 0117
      ISN 0118
      ISN 0119
      ISN 0120
      ISN 0121
      ISN 0122
      ISN 0123
      ISN 0124
      ISN 0125
      ISN 0126
      ISN 0127
      ISN 0128
      ISN 0129
      ISN 0130
      ISN 0131
      ISN 0132
      ISN 0133
      ISN 0134
      ISN 0135
      ISN 0136
      ISN 0137
      ISN 0138
      ISN 0139
      ISN 0140
      ISN 0141
      ISN 0142
      ISN 0143
      ISN 0144
      ISN 0145
      ISN 0146
      ISN 0147
      ISN 0148
      ISN 0149
      ISN 0150
      ISN 0151
      ISN 0152
      ISN 0153
      ISN 0154
      ISN 0155
      ISN 0156
      ISN 0157
      ISN 0158
      ISN 0159
      ISN 0160
      ISN 0161
      ISN 0162
      ISN 0163
      ISN 0164
      ISN 0165
      ISN 0166
      ISN 0167
      ISN 0168
      ISN 0169
      ISN 0170
      ISN 0171
      ISN 0172
      ISN 0173
      ISN 0174
      ISN 0175
      ISN 0176
      ISN 0177
      ISN 0178
      ISN 0179
      ISN 0180
      ISN 0181
      ISN 0182
      ISN 0183
      ISN 0184
      ISN 0185
      ISN 0186
      ISN 0187
      ISN 0188
      ISN 0189
      ISN 0190
      ISN 0191
      ISN 0192
      ISN 0193
      ISN 0194
      ISN 0195
      ISN 0196
      ISN 0197
      ISN 0198
      ISN 0199
      ISN 0200
      ISN 0201
      ISN 0202
      ISN 0203
      ISN 0204
      ISN 0205
      ISN 0206
      ISN 0207
      ISN 0208
      ISN 0209
      ISN 0210
      ISN 0211
      ISN 0212
      ISN 0213
      ISN 0214
      ISN 0215
      ISN 0216
      ISN 0217
      ISN 0218
      ISN 0219
      ISN 0220
      ISN 0221
      ISN 0222
      ISN 0223
      ISN 0224
      ISN 0225
      ISN 0226
      ISN 0227
      ISN 0228
      ISN 0229
      ISN 0230
      ISN 0231
      ISN 0232
      ISN 0233
      ISN 0234
      ISN 0235
      ISN 0236
      ISN 0237
      ISN 0238
      ISN 0239
      ISN 0240
      ISN 0241
      ISN 0242
      ISN 0243
      ISN 0244
      ISN 0245
      ISN 0246
      ISN 0247
      ISN 0248
      ISN 0249
      ISN 0250
      ISN 0251
      ISN 0252
      ISN 0253
      ISN 0254
      ISN 0255
      ISN 0256
      ISN 0257
      ISN 0258
      ISN 0259
      ISN 0260
      ISN 0261
      ISN 0262
      ISN 0263
      ISN 0264
      ISN 0265
      ISN 0266
      ISN 0267
      ISN 0268
      ISN 0269
      ISN 0270
      ISN 0271
      ISN 0272
      ISN 0273
      ISN 0274
      ISN 0275
      ISN 0276
      ISN 0277
      ISN 0278
      ISN 0279
      ISN 0280
      ISN 0281
      ISN 0282
      ISN 0283
      ISN 0284
      ISN 0285
      ISN 0286
      ISN 0287
      ISN 0288
      ISN 0289
      ISN 0290
      ISN 0291
      ISN 0292
      ISN 0293
      ISN 0294
      ISN 0295
      ISN 0296
      ISN 0297
      ISN 0298
      ISN 0299
      ISN 0300
      ISN 0301
      ISN 0302
      ISN 0303
      ISN 0304
      ISN 0305
      ISN 0306
      ISN 0307
      ISN 0308
      ISN 0309
      ISN 0310
      ISN 0311
      ISN 0312
      ISN 0313
      ISN 0314
      ISN 0315
      ISN 0316
      ISN 0317
      ISN 0318
      ISN 0319
      ISN 0320
      ISN 0321
      ISN 0322
      ISN 0323
      ISN 0324
      ISN 0325
      ISN 0326
      ISN 0327
      ISN 0328
      ISN 0329
      ISN 0330
      ISN 0331
      ISN 0332
      ISN 0333
      ISN 0334
      ISN 0335
      ISN 0336
      ISN 0337
      ISN 0338
      ISN 0339
      ISN 0340
      ISN 0341
      ISN 0342
      ISN 0343
      ISN 0344
      ISN 0345
      ISN 0346
      ISN 0347
      ISN 0348
      ISN 0349
      ISN 0350
      ISN 0351
      ISN 0352
      ISN 0353
      ISN 0354
      ISN 0355
      ISN 0356
      ISN 0357
      ISN 0358
      ISN 0359
      ISN 0360
      ISN 0361
      ISN 0362
      ISN 0363
      ISN 0364
      ISN 0365
      ISN 0366
      ISN 0367
      ISN 0368
      ISN 0369
      ISN 0370
      ISN 0371
      ISN 0372
      ISN 0373
      ISN 0374
      ISN 0375
      ISN 0376
      ISN 0377
      ISN 0378
      ISN 0379
      ISN 0380
      ISN 0381
      ISN 0382
      ISN 0383
      ISN 0384
      ISN 0385
      ISN 0386
      ISN 0387
      ISN 0388
      ISN 0389
      ISN 0390
      ISN 0391
      ISN 0392
      ISN 0393
      ISN 0394
      ISN 0395
      ISN 0396
      ISN 0397
      ISN 0398
      ISN 0399
      ISN 0400
      ISN 0401
      ISN 0402
      ISN 0403
      ISN 0404
      ISN 0405
      ISN 0406
      ISN 0407
      ISN 0408
      ISN 0409
      ISN 0410
      ISN 0411
      ISN 0412
      ISN 0413
      ISN 0414
      ISN 0415
      ISN 0416
      ISN 0417
      ISN 0418
      ISN 0419
      ISN 0420
      ISN 0421
      ISN 0422
      ISN 0423
      ISN 0424
      ISN 0425
      ISN 0426
      ISN 0427
      ISN 0428
      ISN 0429
      ISN 0430
      ISN 0431
      ISN 0432
      ISN 0433
      ISN 0434
      ISN 0435
      ISN 0436
      ISN 0437
      ISN 0438
      ISN 0439
      ISN 0440
      ISN 0441
      ISN 0442
      ISN 0443
      ISN 0444
      ISN 0445
      ISN 0446
      ISN 0447
      ISN 0448
      ISN 0449
      ISN 0450
      ISN 0451
      ISN 0452
      ISN 0453
      ISN 0454
      ISN 0455
      ISN 0456
      ISN 0457
      ISN 0458
      ISN 0459
      ISN 0460
      ISN 0461
      ISN 0462
      ISN 0463
      ISN 0464
      ISN 0465
      ISN 0466
      ISN 0467
      ISN 0468
      ISN 0469
      ISN 0470
      ISN 0471
      ISN 0472
      ISN 0473
      ISN 0474
      ISN 0475
      ISN 0476
      ISN 0477
      ISN 0478
      ISN 0479
      ISN 0480
      ISN 0481
      ISN 0482
      ISN 0483
      ISN 0484
      ISN 0485
      ISN 0486
      ISN 0487
      ISN 0488
      ISN 0489
      ISN 0490
      ISN 0491
      ISN 0492
      ISN 0493
      ISN 0494
      ISN 0495
      ISN 0496
      ISN 0497
      ISN 0498
      ISN 0499
      ISN 0500
      ISN 0501
      ISN 0502
      ISN 0503
      ISN 0504
      ISN 0505
      ISN 0506
      ISN 0507
      ISN 0508
      ISN 0509
      ISN 0510
      ISN 0511
      ISN 0512
      ISN 0513
      ISN 0514
      ISN 0515
      ISN 0516
      ISN 0517
      ISN 0518
      ISN 0519
      ISN 0520
      ISN 0521
      ISN 0522
      ISN 0523
      ISN 0524
      ISN 0525
      ISN 0526
      ISN 0527
      ISN 0528
      ISN 0529
      ISN 0530
      ISN 0531
      ISN 0532
      ISN 0533
      ISN 0534
      ISN 0535
      ISN 0536
      ISN 0537
      ISN 0538
      ISN 0539
      ISN 0540
      ISN 0541
      ISN 0542
      ISN 0543
      ISN 0544
      ISN 0545
      ISN 0546
      ISN 0547
      ISN 0548
      ISN 0549
      ISN 0550
      ISN 0551
      ISN 0552
      ISN 0553
      ISN 0554
      ISN 0555
      ISN 0556
      ISN 0557
      ISN 0558
      ISN 0559
      ISN 0560
      ISN 0561
      ISN 0562
      ISN 0563
      ISN 0564
      ISN 0565
      ISN 0566
      ISN 0567
      ISN 0568
      ISN 0569
      ISN 0570
      ISN 0571
      ISN 0572
      ISN 0573
      ISN 0574
      ISN 0575
      ISN 0576
      ISN 0577
      ISN 0578
      ISN 0579
      ISN 0580
      ISN 0581
      ISN 0582
      ISN 0583
      ISN 0584
      ISN 0585
      ISN 0586
      ISN 0587
      ISN 0588
      ISN 0589
      ISN 0590
      ISN 0591
      ISN 0592
      ISN 0593
      ISN 0594
      ISN 0595
      ISN 0596
      ISN 0597
      ISN 0598
      ISN 0599
      ISN 0600
      ISN 0601
      ISN 0602
      ISN 0603
      ISN 0604
      ISN 0605
      ISN 0606
      ISN 0607
      ISN 0608
      ISN 0609
      ISN 0610
      ISN 0611
      ISN 0612
      ISN 0613
      ISN 0614
      ISN 0615
      ISN 0616
      ISN 0617
      ISN 0618
      ISN 0619
      ISN 0620
      ISN 0621
      ISN 0622
      ISN 0623
      ISN 0624
      ISN 0625
      ISN 0626
      ISN 0627
      ISN 0628
      ISN 0629
      ISN 0630
      ISN 0631
      ISN 0632
      ISN 0633
      ISN 0634
      ISN 0635
      ISN 0636
      ISN 0637
      ISN 0638
      ISN 0639
      ISN 0640
      ISN 0641
      ISN 0642
      ISN 0643
      ISN 0644
      ISN 0645
      ISN 0646
      ISN 0647
      ISN 0648
      ISN 0649
      ISN 0650
      ISN 0651
      ISN 0652
      ISN 0653
      ISN 0654
      ISN 0655
      ISN 0656
      ISN 0657
      ISN 0658
      ISN 0659
      ISN 0660
      ISN 0661
      ISN 0662
      ISN 0663
      ISN 0664
      ISN 0665
      ISN 0666
      ISN 0667
      ISN 0668
      ISN 0669
      ISN 0670
      ISN 0671
      ISN 0672
      ISN 0673
      ISN 0674
      ISN 0675
      ISN 0676
      ISN 0677
      ISN 0678
      ISN 0679
      ISN 0680
      ISN 0681
      ISN 0682
      ISN 0683
      ISN 0684
      ISN 0685
      ISN 0686
      ISN 0687
      ISN 0688
      ISN 0689
      ISN 0690
      ISN 0691
      ISN 0692
      ISN 0693
      ISN 0694
      ISN 0695
      ISN 0696
      ISN 0697
      ISN 0698
      ISN 0699
      ISN 0700
      ISN 0701
      ISN 0702
      ISN 0703
      ISN 0704
      ISN 0705
      ISN 0706
      ISN 0707
      ISN 0708
      ISN 0709
      ISN 0710
      ISN 0711
      ISN 0712
      ISN 0713
      ISN 0714
      ISN 0715
      ISN 0716
      ISN 0717
      ISN 0718
      ISN 0719
      ISN 0720
      ISN 0721
      ISN 0722
      ISN 0723
      ISN 0724
      ISN 0725
      ISN 0726
      ISN 0727
      ISN 0728
      ISN 0729
      ISN 0730
      ISN 0731
      ISN 0732
      ISN 0733
      ISN 0734
      ISN 0735
      ISN 0736
      ISN 0737
      ISN 0738
      ISN 0739
      ISN 0740
      ISN 0741
      ISN 0742
      ISN 0743
      ISN 0744
      ISN 0745
      ISN 0746
      ISN 0747
      ISN 0748
      ISN 0749
      ISN 0750
      ISN 0751
      ISN 0752
      ISN 0753
      ISN 0754
      ISN 0755
      ISN 0756
      ISN 0757
      ISN 0758
      ISN 0759
      ISN 0760
      ISN 0761
      ISN 0762
      ISN 0763
      ISN 0764
      ISN 0765
      ISN 0766
      ISN 0767
      ISN 0768
      ISN 0769
      ISN 0770
      ISN 0771
      ISN 0772
      ISN 0773
      ISN 0774
      ISN 0775
      ISN 0776
      ISN 0777
      ISN 0778
      ISN 0779
      ISN 0780
      ISN 0781
      ISN 0782
      ISN 0783
      ISN 0784
      ISN 0785
      ISN 0786
      ISN 0787
      ISN 0788
      ISN 0789
      ISN 0790
      ISN 0791
      ISN 0792
      ISN 0793
      ISN 0794
      ISN 0795
      ISN 0796
      ISN 0797
      ISN 0798
      ISN 0799
      ISN 0800
      ISN 0801
      ISN 0802
      ISN 0803
      ISN 0804
      ISN 0805
      ISN 0806
      ISN 0807
      ISN 0808
      ISN 0809
      ISN 0810
      ISN 0811
      ISN 0812
      ISN 0813
      ISN 0814
      ISN 0815
      ISN 0816
      ISN 0817
      ISN 0818
      ISN 0819
      ISN 0820
      ISN 0821
      ISN 0822
      ISN 0823
      ISN 0824
      ISN 0825
      ISN 0826
      ISN 0827
      ISN 0828
      ISN 0829
      ISN 0830
      ISN 0831
      ISN 0832
      ISN 0833
      ISN 0834
      ISN 0835
      ISN 0836
      ISN 0837
      ISN 0838
      ISN 0839
      ISN 0840
      ISN 0841
      ISN 0842
      ISN 0843
      ISN 0844
      ISN 0845
      ISN 0846
      ISN 0847
      ISN 0848
      ISN 0849
      ISN 0850
      ISN 0851
      ISN 0852
      ISN 0853
      ISN 0854
      ISN 0855
      ISN 0856
      ISN 0857
      ISN 0858
      ISN 0859
      ISN 0860
      ISN 0861
      ISN 0862
      ISN 0863
      ISN 0864
      ISN 0865
      ISN 0866
      ISN 0867
      ISN 0868
      ISN 0869
      ISN 0870
      ISN 0871
      ISN 0872
      ISN 0873
      ISN 0874
      ISN 0875
      ISN 0876
      ISN 0877
      ISN 0878
      ISN 0879
      ISN 0880
      ISN 0881
      ISN 0882
      ISN 0883
      ISN 0884
      ISN 0885
      ISN 0886
      ISN 0887
      ISN 0888
      ISN 0889
      ISN 0890
      ISN 0891
      ISN 0892
      ISN 0893
      ISN 0894
      ISN 0895
      ISN 0896
      ISN 0897
      ISN 0898
      ISN 0899
      ISN 0900
      ISN 0901
      ISN 0902
      ISN 0903
      ISN 0904
      ISN 0905
      ISN 0906
      ISN 0907
      ISN 0908
      ISN 0909
      ISN 0910
      ISN 0911
      ISN 0912
      ISN 0913
      ISN 0914
      ISN 0915
      ISN 0916
      ISN 0917
      ISN 0918
      ISN 0919
      ISN 0920
      ISN 0921
      ISN 0922
      ISN 0923
      ISN 0924
      ISN 0925
      ISN 0926
      ISN 0927
      ISN 0928
      ISN 0929
      ISN 0930
      ISN 0931
      ISN 0932
      ISN 0933
      ISN 0934
      ISN 0935
      ISN 0936
      ISN 0937
      ISN 0938
      ISN 0939
      ISN 0940
      ISN 0941
      ISN 0942
      ISN 0943
      ISN 0944
      ISN 0945
      ISN 0946
      ISN 0947
      ISN 0948
      ISN 0949
      ISN 0950
      ISN 0951
      ISN 0952
      ISN 0953
      ISN 0954
      ISN 0955
      ISN 0956
      ISN 0957
      ISN 0958
      ISN 0959
      ISN 0960
      ISN 0961
      ISN 0962
      ISN 0963
      ISN 0964
      ISN 0965
      ISN 0966
      ISN 0967
      ISN 0968
      ISN 0969
      ISN 0970
      ISN 0971
      ISN 0972
      ISN 0973
      ISN 0974
      ISN 0975
      ISN 0976
      ISN 0977
      ISN 0978
      ISN 0979
      ISN 0980
      ISN 0981
      ISN 0982
      ISN 0983
      ISN 0984
      ISN 0985
      ISN 0986
      ISN 0987
      ISN 0988
      ISN 0989
      ISN 0990
      ISN 0991
      ISN 0992
      ISN 0993
      ISN 0994
      ISN 0995
      ISN 0996
      ISN 0997
      ISN 0998
      ISN 0999
      ISN 1000
      ISN 1001
      ISN 1002
      ISN 1003
      ISN 1004
      ISN 1005
      ISN 1006
      ISN 1007
      ISN 1008
      ISN 1009
      ISN 1010
      ISN 1011
      ISN 1012
      ISN 1013
      ISN 1014
      ISN 1015
      ISN 1016
      ISN 1017
      ISN 1018
      ISN 1019
      ISN 1020
      ISN 1021
      ISN 1022
      ISN 1023
      ISN 1024
      ISN 1025
      ISN 1026
      ISN 1027
      ISN 1028
      ISN 1029
      ISN 1030
      ISN 1031
      ISN 1032
      ISN 1033
      ISN 1034
      ISN 1035
      ISN 1036
      ISN 1037
      ISN 1038
      ISN 1039
      ISN 1040
      ISN 1041
      ISN 1042
      ISN 1043
      ISN 1044
      ISN 1045
      ISN 1046
      ISN 1047
      ISN 1048
      ISN 1049
      ISN 1050
      ISN 1051
      ISN 1052
      ISN 1053
      ISN 1054
      ISN 1055
      ISN 1056
      ISN 1057
      ISN 1058
      ISN 1059
      ISN 1060
      ISN 1061
      ISN 1062
      ISN 1063
      ISN 1064
      ISN 1065
      ISN 1066
      ISN 1067
      ISN 1068
      ISN 1069
      ISN 1070
      ISN 1071
      ISN 1072
      ISN 1073
      ISN 1074
      ISN 1075
      ISN 1076
      ISN 1077
      ISN 1078
      ISN 1079
      ISN 1080
      ISN 1081
      ISN 1082
      ISN 1083
      ISN 1084
      ISN 1085
      ISN 1086
      ISN 1087
      ISN 1088
      ISN 1089
      ISN 1090
      ISN 1091
      ISN 1092
      ISN 1093
      ISN 1094
      ISN 1095
      ISN 1096
      ISN 1097
      ISN 1098
      ISN 1099
      ISN 1100
      ISN 1101
      ISN 1102
      ISN 1103
      ISN 1104
      ISN 1105
      ISN 1106
      ISN 1107
      ISN 1108
      ISN 1109
      ISN 1110
      ISN 1111
      ISN 1112
      ISN 1113
      ISN 1114
      ISN 1115
      ISN 1116
      ISN 1117
      ISN 1118
      ISN 1119
      ISN 1120
      ISN 1121
      ISN 1122
      ISN 1123
      ISN 1124
      ISN 1125
      ISN 1126
      ISN 1127
      ISN 1128
      ISN 1129
      ISN 1130
      ISN 1131
      ISN 1132
      ISN 1133
      ISN 1134
      ISN 1135
      ISN 1136
      ISN 1137
      ISN 1138
      ISN 1139
      ISN 1140
      ISN 1141
      ISN 1142
      ISN 1143
      ISN 1144
      ISN 1145
      ISN 1146
      ISN 1147
      ISN 1148
      ISN 1149
      ISN 1150
      ISN 1151
      ISN 1152
      ISN 1153
      ISN 1154
      ISN 1155
      ISN 1156
      ISN 1157
      ISN 1158
      ISN 1159
      ISN 1160
      ISN 1161
      ISN 1162
      ISN 1163
      ISN 1164
      ISN 1165
      ISN 1166
      ISN 1167
      ISN 1168
      ISN 1169
      ISN 1170
      ISN 1171
      ISN 1172
      ISN 1173
      ISN 1174
      ISN 1175
      ISN 1176
      ISN 1177
      ISN 1178
      ISN 1179
      ISN 1180
      ISN 1181
      ISN 1182
      ISN 1183
      ISN 1184
      ISN 1185
      ISN 1186
      ISN 1187
      ISN 1188
      ISN 1189
      ISN 1190
      ISN 1191
      ISN 1192
      ISN 1193
      ISN 1194
      ISN 1195
      ISN 1196
      ISN 1197
      ISN 1198
      ISN 1199
      ISN 1200
      ISN 1201
      ISN 1202
      ISN 1203
      ISN 1204
      ISN 1205
      ISN 1206
      ISN 1207
      ISN 1208
      ISN 1209
      ISN 1210
      ISN 1211
      ISN 1212
      ISN 1213
      ISN 1214
      ISN 1215
      ISN 1216
      ISN 1217
      ISN 1218
      ISN 1219
      ISN 1220
      ISN 1221
      ISN 1222
      ISN 1223
      ISN 1224
      ISN 1225
      ISN 1226
      ISN 1227
      ISN 1228
      ISN 1229
```

ORIGINAL PAGE IS  
OF POOR QUALITY

ISN 0075 C 60 TO 100  
ISN 0076 C IF (X2 .GE. T0H) GO TO 100  
ISN 0076 CALL HAXTR(HC,VCRUS,H,DIFF)  
ISN 0076 IF (DIFF .LT. 0 .AND. HMAX .NE. 20000.1) GO TO 100  
ISN 0076 HCH = HC  
ISN 0076 AFHT1 = HTAF  
ISN 0076 GHEPL1 = ZFH  
ISN 0076 DIST1 = F1  
ISN 0076 T0H = X2  
ISN 0076 HCRUS = HMAX  
ISN 0076 CONTINUE  
ISN 0077 C 100  
ISN 0077 IF (T0H .GE. T0H) GO TO 200  
ISN 0077 T0W = T0H  
ISN 0077 VCLIMB = CUMSPD  
ISN 0077 NCT = HC  
ISN 0077 AFHT = AFHT1  
ISN 0077 GHEPL = GHEPL1  
ISN 0077 DIST = DIST1  
ISN 0077 CONTINUE  
ISN 0078 C 200  
ISN 0078 RETURN  
ISN 0079 C 30 CALL INFO  
ISN 0079 STOP  
ISN 0080 C END  
ISN 0081  
ISN 0082  
ISN 0083  
ISN 0084  
ISN 0085  
ISN 0086  
ISN 0087  
ISN 0088  
ISN 0089  
ISN 0090  
ISN 0091  
ISN 0092  
ISN 0093  
ISN 0094  
ISN 0095  
ISN 0096  
ISN 0097  
ISN 0098  
ISN 0099  
ISN 0100

LEVEL 21.5 ( JUN 74 )

05/360 FORTRAN H

DATE 02.016/13.51.58

COMPILER OPTIONS - NAME= HAIN,OPT=02,LINECNT=58,SIZE=0000K,  
SOURCE,EBCDIC,NOLIST,NOECK,LOAD,MAP,NOEDIT,IO,NOXREF

C  
C  
C SUBROUTINE TAKEOFF(TOFL)  
C  
ISN 0003 C COMPUT/VARIAB/TOM,S,SLSHP,AR,VCLIM,HCRUS  
C  
C COMPUTES FAR TAKEOFF DISTANCE BASED ON PROF SIEVELL'S CURVE  
C FOR TWIN ENGINE AIRCRAFT IN AA 241 NOTES. CURVE HAS FITTED  
C TO A QUADRATIC -- 952.8 + 26.672A + .0255AC2 -- WHERE A  
C IS THE PARAMETER  $W_2/\Sigma(\Delta X \times S \times H)$ . TAKEOFF POWER VS.  
C SPEED FOR SEA LEVEL AND ISA + 30.8 DEGREES F WAS FITTED TO  
C A CUBIC. POWER RATIO WAS BASED ON THE GENERAL ELECTRIC CT-7  
C ENGINE. ETA OF .65 IS ASSUMED FOR TAKEOFF.  
C  
ISN 0004 C ETA = .65  
ISN 0005 C CLMAX = 2.25  
ISN 0006 C V = .84 \* SQRT((2.\*TOM)/( .002244\*CLMAX\*S ))  
C  
ISN 0007 C SHP = SLSHP\*( .89161-4.057E-4\*V+3.276E-6\*V\*\*2-5.2103E-9\*V\*\*3 )  
ISN 0008 C TH = 550.\*ETA\*SHP/V  
C  
ISN 0009 C A = TOM\*\*2/( .9441\*CLMAX\*S\*TH )  
C  
ISN 0010 C TOFL = 952.8 + 26.672A + .0255\*AX#2  
C  
ISN 0011 C RETURN  
ISN 0012 C END

ORIGINAL PAGE IS  
OF POOR QUALITY.

LEVEL 21.81 JUN 74 ]

05/360 FORTRAN II

DATE 02.016/13.52.00

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=50,SIZE=000K,  
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,10,NOXREF

```
C  
C  
C      SUBROUTINE LANDING(XLFL)  
C  
ISN 0002    C  
COTON/VARIAB/TOW,S,SLSHIP,AR,VCLIMB,MCRS  
C  
ISI 0003    C COMPUTES LANDING DISTANCE USING LINEAR FITTED CURVE OF LANDING  
C DISTANCE VS STALL SPEED SQUARED. ASSUMES DOUBLE SLOTTED FLAPS BUT  
C NO SLATS. MAX LANDING WEIGHT EQUALS TOW.  
C  
ISN 0004    C  
ISI 0005    C   VS2 = 2.*TOW/(1.002244*2.67*5*1.669**2)  
C   XLFL = .4*VS2 + 750.  
C  
ISN 0006    C   RETURN  
ISI 0007    C   END
```

ORIGINAL PAGE IS  
OF POOR QUALITY

LEVEL 21.8 ( JUN 74 )

05/360 FORTRAN H

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINENCNT=56,SIZE=000K,  
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,IO,NOXREF

```

C
C          SUBROUTINE SECSEG(GANSS)
C
ISN 0002      C          COMTCH/VARIAB/TOM,S,SLSHIP,AR,VCLIB,ICRUS
ISN 0003      C          COMPN/CONST/PI,G
ISN 0004      C          COMPN/CHARAC/CDP,E,B
ISN 0005      C          COMPUTES SECOND SEGMENT CLIMB GRADIENT (MUST BE GREATER THAN
C          OR EQUAL TO 2.4% FOR TWIN ENGINE AIRPLANES TO MEET PART 25).
C          ETA IS CONSIDERED TO BE .75 FOR 2ND SEGMENT CLIMB. CALCULATED
C          FOR HOT DAY (ISA + 30.6 DEGREES F) BUT MAY BE CHANGED BY
C          CHANGING DENSITY.
C
ISN 0006      C          H = 400.
ISN 0007      C          CLMAX = 2.25
ISN 0008      C          V = 1.2 * SQRT((2*TOM)/(1.00224445*CLMAX) )
ISN 0009      C          HP = SLSHIP/2.
ISN 0010      C          CDP = CDP + .0000049.*PI/S + .0027 + .0155
ISN 0011      C          ETA = .7
ISN 0012      C          SLIP = HP*(.69161-6.057E-4*V+3.27605E-6*V**2-5.2103E-9*V**3)
ISN 0013      C          THI = 550.*ETA*SLIP/V
ISN 0014      C          D = DRAGH(TOM,V,CORO,S)
ISN 0015      C          DVDH = 1.4636E-5*V*(1.1.-6.8634E-6*H)**(-3.1324)
ISN 0016      C          GANSS = (TH - D)/(TOH*(1.-(V/G)*DVDH))
ISN 0017      C          RETURN
ISN 0018      C          END

```

ORIGINAL PAGE IS  
OF POOR QUALITY.

LEVEL 21:3 ( JUN 74 )

05/360 FORTRAN H

DATE 02.016/13.52.03

**COMPILER OPTIONS - NAME= MAIN,OPT=02,LINJECT=58,SIZE=0000K,  
 SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,TD,NOREF**

```

577.      C
578.      C
579.      C
580.      C
581.      C
582.      C
583.      C
584.      C
585.      C
586.      C
587.      C
588.      C
589.      C
590.      C
591.      C
592.      C
593.      C
594.      C
595.      C
596.      C
597.      C
598.      C
599.      C
600.      C
601.      C
602.      C
603.      C
604.      C
605.      C
606.      C
607.      C
608.      C
609.      C
610.      C
611.      C

ISN 0002      C
ISN 0003      C
ISN 0004      C
ISN 0005      C
ISN 0006      C
ISN 0007      C
ISN 0008      C
ISN 0009      C
ISN 0010      C
ISN 0011      C
ISN 0012      C
ISN 0013      C
ISN 0014      C
ISN 0015      C
ISN 0016      C
ISN 0017      C
ISN 0018      C
ISN 0019      C
ISN 0020      C
ISN 0021      C
ISN 0022      C

C SUBROUTINE ENRCLM(GAMER)
C
C COTTON/VARIAB/TOW,S,SLSHIP,AR,VCLIMB,ICRUS
C COTTON/CHARAC/COP,EIB
C COTTON/CONST/PI,G
C
C COMPUTE EIRROUTE CLIMB GRADIENT AT V = 1.2 VS IN THE CRUISE
C CONFIGURATION. ETA IS .75. GRADIENT MUST BE 1.1/2 TO MEET
C PART 25 REGULATIONS.
C
C H = 11000.
C YEQ = VCLIMB
C ETA = .6
C COPD = COP
C CALL CLIMB(S,SLSHIP,TOW,H,YEQ,COP,ETA,FC,DC,TC,ICRUS,HC)
C
C NOW WE HAVE THE WEIGHT AT 11000 FT (OBSTACLE CLEARANCE HEIGHT).
C COMPUTE THE CLIMB GRADIENT AT THAT HEIGHT.
C
C RHO = 2.3769E-3*(1. - 6.6631E-6*H)^4(4.2648)
C CLMAX = 1.58
C V = 1.3 * SQRT((2.*WC)/(RHO*S*CLMAX))
C HP = SLSHIP/2.
C ETA = .75
C SUP = POWER(HP,V,H)
C TH = 550.*ETA*SUP/V
C COPD = COP + .0008*49.*PI/S + .0027
C D = DRAG(H,WC,V,COPD,S)
C GAMER = (TH - D)/WC
C RETURN
C

```

-41-



ORIGINAL PAGE IS  
OF POOR QUALITY

```

666.
667.
668.
669.
670.
671.
672.
673.
674.
675.
676.
677.
678.
679.
680.
681.
682.
683.
684.
685.
686.
687.
688.
689.
690.
691.
692.
693.
694.
695.
696.
697.
698.
699.
700.
701.
702.
703.
704.
705.
706.
707.
708.
709.
710.
711.
712.
713.
714.
715.
716.
717.
718.
719.
720.
721.

C EXPOSED AREA IS THE TOTAL AREA MINUS THAT AREA COVERED BY THE
C FUSELAGE.
C
C SE = ((B - 0.116)/2.)*(CRF + .4*CR)
C SHET = 2.04*SE
ISN 0026
ISN 0027

C FIND THE EXPOSED MAC AND THE RETROIDS NUMBER ASSOCIATED WITH THAT
C LENGTH AT A CRUISE SPEED OF 290 KT AT 25,000 FT.
C
C XMAC = .6667*(CRF+.4*CR)*CP/(CRF+.4*CR))
C RE = 1.867E6*XMAC

ISN 0028
ISN 0029

C FRICTION COEFFICIENT IS A FUNCTION OF THE LOG(BASE 10) OF THE
C REYNOLD NUMBER.
C
C RELOG = ALOG10(RE)
C CF = (178.868-26.463*(RELOG)+3.1025*(RELOG)**2-.12417)*(RELOG)
C
1   **3)*1.E-3

ISN 0030
ISN 0031

C FIND THE FORM FACTOR USING THE FORMULA GIVEN IN THE AA241 NOTES.
C
C Z = 1.75*COS(SHEEP)/SQRT(1.-.25*(COS(SHEEP))**2)
C K = 1. + Z*TCH + 100.*TCW**4
C WRITE(6,*),SHET,CF,K,TCW,Z
C
C NOW FIND THE F OF THE WING.
C
C FWING = CF*K*SHET
ISI 0034

C DO THE SAME THING FOR THE HORIZONTAL TAIL.
C
C SHET = 2.04*SHE
C RE = 1.867E6*SHE/BH
C RELOG = ALOG10(RE)
C CF = (178.868-26.463*(RELOG)+3.1025*(RELOG)**2-.12417)*(RELOG)
C
1   **3)*1.E-3
C Z = 1.75*COS(SHEEP)/SQRT(1.-.25*(COS(SHEEP))**2)
C K = 1. + Z*TCH + 100.*TCW**4
C FHORIZ = CF*K*SHET
C
C NOW DO THE SAME FOR THE VERTICAL TAIL.
C
C SHET = 2.04*SVE
C RE = 1.867E6*SVE/BV
C RELOG = ALOG10(RE)
C CF = (178.868-26.463*(RELOG)+3.1025*(RELOG)**2-.12417)*(RELOG)
C
1   **3)*1.E-3
C Z = 1.75*COS(SHEEP)/SQRT(1.-.25*(COS(SHEEP))**2)
C K = 1. + Z*TCV + 100.*TCV**4
C FVERT = CF*K*SHET
C
C FIND THE GAP DRAG USING THE METHOD OF AA241.
C
C FGAP = .0042*(RE*(COS(SHEEP))**2*B/4.)
C
C THE CONSTANT F'S HAVE BEEN DETERMINED AS,
ISN 0049

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
C FFUSE = 3.799          723.  
ISH 0050 C FNACPY = 1.15    724.  
ISH 0051 C NOW FIND THE TOTAL F USING MISCELLANEOUS DRAG TO EQUAL 62. 725.  
C 726.  
C FTOT = (FWING+FHORIZ+FVERT+FFUSE+FNACPY+FGAP)/.96 727.  
ISH 0052 C 728.  
C TO FIND THE PARASITE DRAG COEFFICIENT, DIVIDE BY THE WING AREA. 729.  
C 730.  
C 731.  
C COP = FTOT/S          732.  
ISH 0053 C EFFICIENCY FACTOR IS CALCULATED USING INDUCED DRAG FACTORS 733.  
C FOUND IN SIEVELL'S NOTES. 734.  
C 735.  
C SS = 1. - .0745*(0.116/B) - 1.6136*(0.116/B)**2 736.  
ISH 0054 C U = .990674*3.3366*E-6*AR**2+2.02546*E-6*AR**3 737.  
ISH 0055 C E = 1. / (PI*AR*(1. / (PI*AR*USS)+.4368*COP)) 738.  
ISH 0056 C 739.  
C RETURN 740.  
ISH 0057 C END 741.  
ISH 0058 C 742.
```

COMPILER OPTIONS - NAME= NAME,OPT=02,LINECNT=56,SIZE=0000K,  
SOURCE,ESCDIC,NDLIST,NODECK,LOAD,MAP,INDEXIT,TD,NOXREF

```

C      C SUBROUTINE POARDS(TOW,S,SLSHP,ZFW,HTAFL)
C      C
C      C TRANSLATION OF BASIC WEIGHT PROGRAM USED IN AA241 -- SHEVELL'S
C      C WEIGHT METHOD. CONSTANT THIRTY PASSENGER SIZE FUSelage OF
C      C DIAMETER 6.116 FT AND LENGTH 66 FT. PAYLOAD OF 6270 LBS.
C      C PROGRAM MUST KNOW TOW, S AND SLSHIP AS WELL AS VALUES IN
C      C COTDCH/GEOM. MUST ALSO BE GIVEN MAX GUST VELOCITY (50 FPS).
C      C
C      ISN 0003          C COTDCH/CONST/PI.G
C      ISN 0004          C COTDCH/CHARAC/CDP,E,B
C      ISN 0005          C COTDCH/GEOM/XMAC,SIE,SHG,BH,SVE,SVG,BV
C      ISN 0006          C COTDCH/AERO/VEQ,CLALPH
C
C      C DESIGN STRUCTURAL SPEED DETERMINED AT 10000 FT.
C      C
C      ISN 0007          C RHO = .0017553
C      ISN 0008          C T7 = 9601.15
C      ISN 0009          C U9 = 50.
C      ISN 0010          C F9 = 3500./TOW
C
C      C LANDING GEAR WEIGHT
C      C
C      ISH 0011          C W7 = .04 * TOW
C      ISH 0012          C Z2 = TOW * (1. - F9)
C
C      ISN 0013          C 10   Z1 = 22
C      ISN 0014          C V1 = 2.*Z1*B/(RHO*G*S*SH2*CLALPH)
C      ISN 0015          C XX1 = .05*V1/(5.3 + V1)
C      ISN 0016          C V2 = 2.*TOW*B/(RHO*G*S*SH2*CLALPH)
C      ISN 0017          C XZ2 = .05*V2/(5.3 + V2)
C      ISH 0018          C F1 = 1. + (XX1*CLALPH*W7*VEQ)/(S*VEQ1)/(S*VEQ1)
C      ISH 0019          C F3 = 1. + (XX2*CLALPH*W7*VEQ)/(S*VEQ1)/(S*VEQ1)
C      IGN 0020          C F5 = 2.1 + (24000./TOW + 10000.)
C      ISH 0021          C IF (F5 .GT. 3.0) F5 = 3.0
C      ISH 0023          C IF (F5 .LT. 2.5) F5 = 2.5
C      ISH 0025          C F2 = 1.5 * F3
C      ISN 0027          C
C      C WING WEIGHT; AVERAGE THICKNESS = .15
C      C
C      ISN 0028          C XI1=(F2*B*3*SQRT(TOW*Z1)*1.0E-6)/(1.1545*2*1.4)
C      ISN 0029          C HI = (1.642*XI1 + 4.22)*5
C      C WRITE(6,*),XI1,HI
C
C      C HORIZONTAL TAIL HEIGHT; AVERAGE THICKNESS = .12
C      C
C      ISN 0030          C XI2= (F2*B*3*TOW*XMAC*SQRT(SHE)*1E-6)/(6.31125*SHG*1.5*SHE)
C      ISN 0031          C W2 = (1.5*XI2 + 5.25)*SIE

```

```

C VERTICAL TAIL AND RUDDER WEIGHT; VERTICAL AVG T/C = .12
C
ISN 0032      X13=(F2*BV4*3*(0.+.44*TDX/S1)*1E-3)/(1.12*.96*.75*SVE)    798.
ISN 0033      W3 = (0.0145*X13 + 3.51)*.75*SVE   799.
ISN 0034      W4 = 1.6*X13/3.   800.
C
C SURFACE CONTROLS WEIGHT
C
ISN 0035      C      W5 = 1.7*(SHG+SVG)   801.
ISN 0036      C      IF (F1 .LT. 2.5) F1 = F3   802.
C
C FUSELAGE HEIGHT
C
ISN 0038      SLSHPH = SLSHP/2.   803.
ISN 0039      WENG = 2*(1-16.56 + 12.58*SART(SLSHPH)) + .0618*SLSHPH)   804.
ISN 0040      W9 = 6528. + WENG   805.
ISN 0041      H0 = 1948. + WENG   806.
C
ISN 0042      T1 = .6*F1*(Z1-W1-W0)/(FT*B.116**2)   807.
ISN 0043      T2 = T1 - T7   808.
ISN 0044      IF (T2 .LT. 0.) T2 = 0.   809.
ISN 0046      X16 = (T7 + (T2**2/(2.*T1)))*1E-3   810.
ISN 0047      R6 = (.102*X16 + 1.051)*1472.47   811.
C
C DETERMINE ZFH.  COMPARE WITH ESTIMATED ZFH.  IF NOT SAME,
C ITERATE.
C
ISN 0048      Z2 = W1**W2**W3**W4**W5**W6**W7*.65**5**W9**6270.   812.
ISN 0049      IF (ABS((Z2 - Z1)/Z1) .GT. .0001) GO TO 10   813.
ISN 0051      ZFH = Z2   814.
ISN 0052      WTAF = Z2 - 7270. - WENG   815.
C
ISN 0053      C      RETURN   816.
ISN 0054      END   817.

```

LEVEL 21.6 ( JUN 74 )

05/360 FORTRAN H

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=50,SIZE=0000K,  
SOURCE,EBCDIC,NOLIST,NOECK,LOAD,MAP,NOEDIT,IO,NOXREF

C  
C  
ISN 0002 C REAL FUNCTION DRAGINH,X,Y,COPD,S!  
ISN 0003 C COMMON/CHARAC/COP,E,B  
ISN 0004 C COMMON/CONST/PI,G  
C COMPUTES DRAG FOR A GIVEN FLIGHT CONDITION  
C  
ISN 0005 C RHO = 2.3769E-3\*(  
ISN 0006 C Q = .5\*RHO\*V\*\*2  
C DRAG = COP\*Q\*S + (W\*\*2)/(Q\*PI\*(B\*\*2)\*E)  
ISN 0007 C RETURN  
ISN 0008 END  
ISN 0009

DATE 02.016/13.52.09

ORIGINAL PAGE IS  
OF POOR QUALITY

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=59,SIZE=000CK,  
SOURCE,EBCDIC,NODECK,LOAD,MAP,NOEDIT,IO,NOREF

```

C
C
C      SUBROUTINE CLIMB(S,SLSHIP,T0A,INMAX,VEQ,CPOD,ETA,FC,TC,DC,
C      XGAMMAC,H)
C
C      COMMON/CHARAC/CPOD,E,B
C      COMMON/CONST/P1,G
C      COMMON/FLAG/KANT
C
C      COMPUTES TIME, FUEL AND DISTANCE TO CLIMB TO HMAX. VEQ IS
C      CONSTANT EQUIVALENT AIRSPEED FOR CLIMB. RETURNS CLIMB
C      GRADIENT, GAMMAC, AND HEIGHT AT END OF CLIMB. CPOD IS THE
C      BASIC CPO PLUS WHATEVER ADDITIONAL DRAG IS INCURRED DUE TO
C      CONFIGURATION (IE, FLAP DRAG, SPOTLER DRAG, J
C      TC,FC AND DC ARE TIME, FUEL AND DISTANCE TO CLIMB, RESPEC-
C      TIVELY. THE PROGRAM INTEGRATES IN ALTITUDE STEPS -- SIMPLE
C      BACKWARD EULER INTEGRATION.
C
C      KANT = 0
C      DH = 200.
C      H = .993*TH
C      H = 0.
C      TC = 0.
C      FC = 0.
C      DC = 0.
C      SFC = .47
C
C      BEGIN INTEGRATION.
C
C      10   V = VEQ*(1-6.8634E-6*H)**(-2.1324)
C            DVDH = 1.4636E-5*VEQ*(1-6.8634E-6*H)**(-3.1324)
C            SHIP = POWER(SLSHIP,V,H)
C            TH = 550. * METASHP/V
C
C      FIRST, ESTIMATE DRAG USING LIFT = WEIGHT. THEN USE LIFT
C      WEIGHT * THE COSINE OF THE ESTIMATED GAMMA FOR A BETTER
C      VALUE OF CLIMB GRADIENT.
C
C      D = DRAG(H,W,V,CPOD,S)
C      GAMMA = (TH - D)/(W*(1+(V/E)**DVDH))
C      XLIFT = W * COS(GAMMA)
C      D = DRAG(H,XLIFT,V,CPOD,S)
C
C      NOW FIND THE CORRECT CLIMB GRADIENT
C
C      GAMMAC = (TH - D)/(W*(1+(V/E)**DVDH))
C
C      IF GAMMA IS NEGATIVE THE AIRPLANE CAN'T CLIMB -- GIVE AN
C      ERROR MESSAGE.
C
C      IF (GAMMAC .LT. 0.) GO TO 50
C
C      CALCULATE RATE OF CLIMB

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

ISN 0025      C   RC = VNGARHAC
ISN 0026      C   CALCULATE FUEL TO CLIMB
ISN 0028      C   IF (H .EQ. 0.) GO TO 20
ISN 0029      C   DELTAF = (SHPSFC)/(3600.*RC)*DH
ISN 0030      C   FC = FC + DELTAF
ISN 0031      C   H = H - DELTAF
ISN 0032      C   CALCULATE DISTANCE TO CLIMB
ISN 0033      C   DELTAD = VNDH/RC
ISN 0034      C   DC = DC + DELTAD
ISN 0035      C   CALCULATE TIME TO CLIMB
ISN 0036      C   DELTAT = DV/RC
ISN 0037      C   TC = TC + DELTAT
ISN 0038      C   TAKE A STEP
ISN 0039      C   20   H = H + DH
ISN 0040      C   IF (H .LE. HMAX) GO TO 10
ISN 0041      C   CONTINUE
ISN 0042      C   60   GO TO 70
ISN 0043      C   ERROR MESSAGE
ISN 0044      C   50   WRITE(6,100)
ISN 0045      C   100  FORMAT(1X,'AIRPLANE CANNOT CLIMB -- INADEQUATE FUEL OR HP.')
ISN 0046      C   KANT = 1
ISN 0047      C   70   RETURN
ISN 0048      C   END

```

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=50,SIZE=0000K,  
SOURCE,EBCDIC,NDLIST,LOAD,MAP,NOEDIT,IO,INDREF

```

C
C
C      SUBROUTINE DESENT(SLSHP,S,HMAX,ETA,FD,TD,DD,H,OHEPL)
C
C      C DESCENT COMPUTES TIME, FUEL AND DISTANCE TO DESCEND FROM HMAX.
C      C IT REQUIRES A CONSTANT EQUIVALENT AIRSPEED FOR DESCENT, VEQ,
C      C AND A CONSTANT RATE OF DESCENT, RD, IN FEET PER SEC. IT
C      C USES A SIMPLE BACKWARD EULER INTEGRATION TO WORK BACKWARDS
C      C FROM THE GROUND TO HMAX.
C
C      RD = HMAX/(20.*60.)
C      VEQ = 300.
C      H = 0.
C      H = OHEPL/.977
C      DD = 500.
C      FD = 0.
C      DD = 0.
C      TD = 0.
C
C      BEGIN INTEGRATION. DEPENDING ON THE RATE OF DESCENT, MAY
C      C NEED SOME SPOILERS. SINCE IDLE POWER SEEMS TO BE AROUND .1 OF
C      C MAX POWER, USE SPOILERS IF THE REQUIRED POWER TURNS OUT TO BE LESS
C      C THAN .1 OF MAX.
C
C      CDP0 = CDP
C
C      10   V = VEQ * ((1 - 6.8634E-6*H)**(-2.1324))
C      DVDD = 11.4636E-5*VEQ*((1-6.8634E-6*H)**(-3.1324))
C
C      FIND DESCENT GRADIENT, GAMMAD, AND THE LIFT PRODUCED FOR THIS CONDI-
C      C TION. THEN DETERMINE THE DRAG AND THRUST REQUIRED.
C
C      GAMMAD = RD/V
C      XLIIFT = H * COS(GAMMAD)
C      D = DRAG(H,XLIIFT,V,CDPO,S)
C      TH = D - H*GAMMAD*(1-(V/G)*DVDD)
C      SHP = (TH * V)/(1550 * ETA)
C
C      FIND THE SFC FOR THIS POWER SETTING. Z IS RATIO OF SHP TO SHPMAX.
C
C      SHPMAX = POWER(SLSHP,V,H)
C      Z = SHP/SHPMAX
C      IF ((Z .LT. .1) Z = .1
C
C      SFC = .43 + 2.07E-2/Z + 2.02E-2/Z**2 - 1.04E-3/Z**3
C
C      CALCULATE FUEL TO DESCEND
C
C      DELTAF = (Z*SHPMAX * SFC)/13600. * RD) * DH
C      FD = FD + DELTAF
C      H = H + DELTAF

```

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=50,SIZE=0000K,  
SOURCE FBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT, ID, NOREF

```

      C
      C REAL FUNCTION POWER(SLSHP,V,H)
      C
      C CALCULATES AVAILABLE MAX CRUISE AND CLIMB HORSEPOWER AS A
      C FUNCTION OF ALTITUDE AND SPEED (IN FT/SEC). CURVES ARE FIT
      C FOR GENERAL ELECTRIC CT-7 ENGINE. CALCULATES RATIO OF AVAIL-
      C ABLE POWER TO SEA LEVEL STATIC POWER RATING. MAX POWER IS
      C FOUND BY MULTIPLYING RATIO BY SEA LEVEL SHP.
      C
      C1 = .6461 - 1.802E-5MH
      C2 = .5921H(-1.156E-4-1.443E-6)VH+3.260E-13VH*2+5.133E-10VH*3
      C3 = .3505H(1.904E-6+1.758E-10VH-5.875E-15VH*2)
      C4 = .2075H(-3.315E-9-2.26E-13VH+9.301E-16VH*2)
      C
      SLPF = C1 + C2VH + C3VH*2 + C4VH*3
      POWER = SLSHP * SLPF
      RETURN
      END

      ISN 0003
      ISN 0004
      ISN 0005
      ISN 0006
      ISN 0007
      ISN 0008
      ISN 0009
      ISN 0010

```

ORIGINAL PAGE IS  
OF POOR QUALITY

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECHT=50,SIZE=0000K,  
SOURCE,EBCDIC,NOLIST,NODECK,LOAD,MAP,NOEDIT,IO,NOXREF

```

C
C      SUBROUTINE BLOCK(VCRUS,BLOCKT,BLOCKF,STAGE)
C
C      COMMON/VARLAB/T0H,S,SLSHP,AR,VCLIMB,NCRUS
C      COMMON/WEIGHT/ONEPL,AFWT
C      COMMON/CONST/PI,G
C      COMMON/CHARAC/CDP,E,B
C
C      CALCULATES BLOCK TIME AND BLOCK FUEL FOR THE STAGE LENGTH TO BE
C      OPTIMIZED. USES AN ITERATIVE PROCEDURE TO DETERMINE THE TOW
C      FOR THE GIVEN RANGE. BLOCK TIME AND FUEL ARE NECESSARY FOR
C      FINDING THE DIRECT OPERATING COSTS.
C
C      ISH 0007          STAGE = 150.
C      ISH 0008          HMAX = 15000.
C      ISH 0009          VEQ = VCLIMB
C      ISH 0010          ETA = .8
C      ISH 0011          R010 = 2.3769E-3*(1. - 6.8634E-6*HMAX)**(4.2640)
C      ISH 0012          Q = .5 * RHO * VCRUS**2
C
C      CALCULATE TIME, FUEL, AND DISTANCE TO DESCEND ASSUMING WEIGHT
C      AT END OF DESCENT IS ONE + PAYLOAD + RESERVES.
C
C      CALL DESENT(SLSHP,S,HMAX,ETA,FD,TD,DD,MD,ONEPL)
C
C      C ASSUME A TAKEOFF WEIGHT. CALCULATE CLIMB AND CRUISE UNTIL THE
C      C HEIGHT IS RIGHT.
C
C      ISH 0014          X2 = WD + 1000.
C      ISH 0015          STEP = X2/100.
C      ISH 0016          CONV = .1
C      ISH 0017          STAGEN = STAGE - DD/6072.
C      ISH 0018          INTNUM = 0
C
C      C MINIMIZATION ROUTINE
C
C      ISH 0019          CALL DISTN(X2,F2,HMAX,VCLIMB,MD,Q,STAGEN,INC,F1,TC,FC,VCRUS)
C      ISH 0020          10   X3 = X2 + STEP
C      ISH 0021          X1 = X2 - STEP
C      ISH 0022          CALL DISTN(X1,F1,HMAX,VCLIMB,MD,Q,STAGEN,INC,F1,TC,FC,VCRUS)
C      ISH 0023          CALL DISTN(X3,F3,HMAX,VCLIMB,MD,Q,STAGEN,INC,F1,TC,FC,VCRUS)
C
C      ISH 0024          3   INTNUM = INTNUM + 1
C      ISH 0025          C   IF (INTNUM .GT. 50) GO TO 89
C
C      ISH 0027          C   IF (F1 .LT. F2) GO TO 1
C      ISH 0029          C   IF (F3 .LT. F2) GO TO 2
C
C      WHERE F2 IS LESS THAN F1 AND F3, FIT A PARABOLA, FIND MINIMUM AND
C      USE THAT AS THE NEXT X2
C
C      1027.          1028.
C      1029.          1030.
C      1031.          1032.
C      1033.          1034.
C      1035.          1036.
C      1037.          1038.
C      1039.          1040.
C      1041.          1042.
C      1043.          1044.
C      1045.          1046.
C      1047.          1048.
C      1049.          1050.
C      1051.          1052.
C      1053.          1054.
C      1055.          1056.
C      1057.          1058.
C      1059.          1060.
C      1061.          1062.
C      1063.          1064.
C      1065.          1066.
C      1067.          1068.
C      1069.          1070.
C      1071.          1072.
C      1073.          1074.
C      1075.          1076.
C      1077.          1078.
C      1079.          1080.

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

ISN 0031      X2 = X2 + .5*STEP*(F1-F3)/(F3-2.*F2+F1)          1081.
ISN 0032      CALL DISTN(X2,F2,HMAX,YCLIMB,HD,Q,STAGEM,MC,FI,TC,FC,VCRUS) 1082.
ISN 0033      IF (F2 .LE. CONV) GO TO 99                      1083.
ISN 0034      STEP = STEP/4.                                1084.
ISN 0035      GO TO 10                                1085.
ISN 0036

C WHERE F2 IS BETWEEN OR GREATER THAN F1 AND F3, MAKE X2 THE VALUE
C FOR A MINIMUM
C
C   1   IF (F3 .LT. F1) GO TO 2
      X2 = X1
      F2 = F1
      X1 = X1 - STEP
      CALL DISTN(X1,F1,HMAX,YCLIMB,HD,Q,STAGEM,MC,FI,TC,FC,VCRUS)
      GO TO 3
C
      ISN 0043      X1 = X2
      F1 = F2
      X2 = X3
      F2 = F3
      X3 = X3 + STEP
      CALL DISTN(X3,F3,HMAX,YCLIMB,HD,Q,STAGEM,MC,FI,TC,FC,VCRUS)
      GO TO 3
C
      ISN 0044      69  WRITE(6,302)
      ISN 0045      302  FORMAT(' INTMAX = 50, CONTINUING ... ')
C
      ISN 0046      C
      ISN 0047      C
      ISN 0048      C
      ISN 0049      C
      ISN 0050      C
      ISN 0051      69  CONTINUE
      ISN 0052      59  BLOCKF = FD + FC + (MC - HD) + .002*TOW
C
      ISN 0053      99  BLOCKT = TD/60. + TC/3600. + FI*6072./(VCRUS*3600.) + .25
      ISN 0054
      ISN 0055      C
      ISN 0056      C
      ISN 0057      C
      RETURN
      END

```

EVEL 21:1 JUN 74 1

OS/360 FORTRAN H

COMPILER OPTIONS - NAME = SOURCE,EBCDIC,NOECK,LOAD,SIZE=1024,HECIN=38,NOREF

ORIGINAL PAGE IS  
OF POOR QUALITY

DATE 02-016/13.52.25

DATE 02-016/13.52.25

C AIRFRAME MATERIAL  
C AFHAT = ((3.00\*COSTAFAF\*T1)\*(6.24\*COSTAFAF))/(1E6\*STAGES)

C ENGINE LABOR  
C XLABEF = ( .65+((.03\*SLSHH)/1000.))\*.2.

ISN 0019 C XLABEH = (.3+((.03\*SLSHH)/1000.))\*.2.

ISN 0020 C ENGLAB = (XLABEF\*T1\*XLADEH)/STAGES \* 12.  
ISN 0021 C

C ENGINE MATERIALS  
C ERMAT = ((2.5\*COSTEN)\*T1 + 2.\*COSTEN)/(1E5\*STAGES)

C MAINTENANCE BURDEN  
C BURDEN = 1.8 \* (AFLAB + ENGLAB)

ISN 0023 C TOTAL MAINTENANCE COST  
C TOTHAI = AFLAB + AFHAT + ENGLAB + ERMAT + BURDEN

ISN 0024 C DEPRECIATION COST  
C DEPR= ((COSTAFAF+COSTEN)+.1\*COSTAFAF+.4\*COSTEN)/(BLOCKS\*15.\*2800.)

ISN 0025 C CENTS PER SEAT STATUTE MILE

C CENTS = ((CRACST+FOLCST+XINCST+TOTHAI+DEPR)\*100./30.

C CENTS PER SEAT NAUTICAL MILE  
C CENT = CENTS \* 1.15

ISN 0026 C RETURN  
ISN 0027 C END  
ISN 0028 C  
ISN 0029 C

ORIGINAL PAGE IS  
OF POOR QUALITY

LEVEL 21.0 ( JUN 74 )

05/360 FORTRAN H

DATE 02-016/13-52-27

COMPILER OPTIONS - NAME= MATH,OPT=02,LINECHT=50,SIZE=0000K,  
SOURCE,EBCDIC,NOLIST,NOECK,LOAD,MAP,INCEDIT,JO,NOXREF

C  
C  
C SUBROUTINE DISTN(X,F,HMAX,VEQ,WD,Q,STAGEM,WC,F1,TC,FC,VCRUS)  
ISN 0002 C 1205.  
C 1206.  
COTON/CHARAC/COP,E,B  
COTON/CONST/PI,G  
COTON/VARIAB/TCH,S,SLSHP,AR,VCLIMB,MCUS  
ISN 0003 C 1207.  
ISN 0004 C 1208.  
ISN 0005 C 1209.  
C SUBROUTINE TO COMPUTE THE GOAL FUNCTION (OR THAT TO BE MINIMIZED)  
C FOR THE BLOCK TIME AND FUEL DETERMINIZATION.  
ISN 0006 C 1210.  
ISN 0007 C 1211.  
ISN 0008 C 1212.  
C 1213.  
C 1214.  
C 1215.  
ISN 0009 C 1216.  
ISI 0010 C 1217.  
ISI 0011 C 1218.  
ISI 0012 C 1219.  
ISN 0013 C 1220.  
ISI 0014 C 1221.  
ISI 0015 C 1222.  
ISN 0016 C 1223.  
ISI 0017 C 1224.  
ISN 0018 C 1225.  
ISI 0019 C 1226.  
C 1227.  
C 1228.  
C 1229.  
C 1230.  
C 1231.  
C 1232.  
C 1233.  
C 1234.

ORIGINAL PAGE IS  
OF POOR QUALITY

COMPILER OPTIONS - NAME = MATH, OPT=02, LINECNT=50, SIZE=60000,  
SOURCE, EBORDC, IBLIST, NOECK, LOAD, MAP, NOEDIT, ID, NOSEE

```

1235. ISN 0002 C
1236. ISN 0003 C
1237. ISN 0004 C
1238. ISN 0005 C
1239. ISN 0006 C
1240. ISN 0007 C
1241. ISN 0008 C
1242. ISN 0009 C
1243. ISN 0010 C
1244. ISN 0011 C
1245. ISN 0012 C
1246. ISN 0013 C
1247. ISN 0014 C
1248. ISN 0015 C
1249. ISN 0016 C
1250. ISN 0017 C
1251. ISN 0018 C
1252. ISN 0019 C
1253. ISN 0020 C
1254. ISN 0021 C
1255. ISN 0022 C
1256. ISN 0023 C
1257. ISN 0024 C
1258. ISN 0025 C
1259. ISN 0026 C
1260. ISN 0027 C
1261. ISN 0028 C
1262. ISN 0029 C
1263. ISN 0030 C
1264. ISN 0031 C

C SUBROUTINE DISTRI(X,F,IMAX,EAS,MD,Q,MAXING,MC,F1,VCRUS,ZFH,HTAF)
C
C   REAL MAXING
C   COMMON/CONST/PI,G
C   COMMON/CHARAC/COP,E,B
C   COMMON/VARIA/TCH,S,SLSHIP,AR,VLCLND,HRUS
C   COMMON/GEOM/XPHC,SUE,SIG,BH,SVE,SVG,BV
C   COMMON/WEIGHT/DCEPL,AFHT
C   COMMON/FLAG/KANT
C
C   FUNCTION EVALUATION FOR FINDING THE TCH FOR THE REQUIRED RANGE.
C
C   ETA = .6
C   COP0 = COP
C   CALL PORDIS(X,S,SLSHIP,ZFH,HTAF)
C   CALL DESCENT(SLSHIP,S,IMAX,ETA,FD,TD,DD,WD,ZFH)
C   CALL CLINBIS(SLSHIP,X,IMAX,EAS,COP0,ETA,FC,TC,DC,GC,MC)
C   IF (WD .GT. MC) KANT = 1
C   IF (KANT .NE. 0) GO TO 10
C
C   DAV = COP0*Q45 + (MC**2*WD**2)/(2.*SQRT(PI*(DD*ZFH)))
C   SIPAV = DAV*VCRUS/(550.*.05)
C   SHIMAX = POWER(SLSHIP,VCRUS,IMAX)
C   Z = SIPAV/SHIMAX
C   SFC = -.43 + 2.07E-2/Z + 2.02E-2/Z**2 - 1.04E-3/Z**3
C
C   A1 = 325.*L(.05/SFC)*MD*SQRT(PI*(DD*MC))
C   A2 = Q0*SQRT(COP0*PI*E*S)
C
C   F1 = A1*ATAN(MC/A2) - ATAN(WD/A2) + (DD + DC)/6072.
C   10 TO 20
C
C   IF AIRPLANE CANNOT CLIMB, LET THE RANGE BE EQUAL TO GMAXC..
C
C   10   F1 = GC
C   20.  F = (F1 - MAXING)**2
C
C   RETURN
C   END

```

**ORIGINAL PAGE IS  
OF POOR QUALITY**