

AD-A151 488

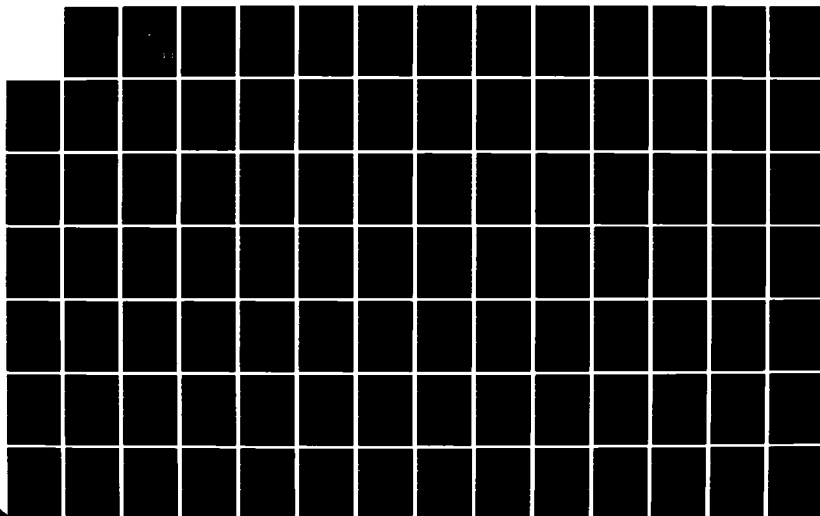
PRELIMINARY HELICOPTER DESIGN DECISION MAKING BASED ON
FLIGHT PERFORMANCE FACTORS(U) NAVAL POSTGRADUATE SCHOOL
MONTEREY CA P V ADAMCIK SEP 84

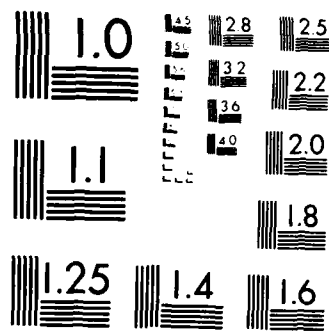
1/2

UNCLASSIFIED

F/G 6/3

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

AD-A151 488

NAVAL POSTGRADUATE SCHOOL
Monterey, California



THESIS

PRELIMINARY HELICOPTER DESIGN DECISION
MAKING BASED ON FLIGHT PERFORMANCE FACTORS

by

Patrick V. Adamcik

September 1984

Thesis Advisor:

Donald M. Layton

Approved for public release: Distribution unlimited

DTIC
ELECTE
MAR 22 1985

DTIC FILE COPY

85 03 11 117

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER		2. GOVT ACCESSION NO. A151 488	
4. TITLE (and Subtitle) Preliminary Helicopter Design Decision Making Based on Flight Performance Factors		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1984	
7. AUTHOR(s) Patrick V. Adamcik		6. PERFORMING ORG. REPORT NUMBER	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		8. CONTRACT OR GRANT NUMBER(s)	
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE September 1984	
		13. NUMBER OF PAGES 97	
		15. SECURITY CLASS. (of this report) Unclassified	
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release, distribution unlimited			
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		Accession For NTIS GRA&I ☒ DTIC TAB ☐ Unannounced ☐ Justification _____ By _____ Distribution/ _____ Availability Codes Dist Avail and/or A1 Special	
18. SUPPLEMENTARY NOTES			
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Helicopter Design, Helicopter Performance, Helicopter Flight Performance, Helicopter Performance Factors.			
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This thesis will assist those evaluating helicopter design to make preliminary judgments about the feasibility of new designs. By using the computer program developed in this thesis, a designer can produce estimates for power requirements, endurance velocity, rate of climb, range velocity, hover ceiling, and service ceiling versus main rotor radius. These			

20. continuation

estimates can also be examined for the effects of changes in main rotor radius, chord, and rotational velocity.

Approved for public release: distribution unlimited

Preliminary Helicopter Design Decision Making Based
on Flight Performance Factors

by

Patrick V. Adamcik
Captain, United States Army
B.S., University of Texas at Austin, 1977

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1984

Author:

Patrick V. Adamcik

Patrick V. Adamcik

Approved by:

D. M. Layton

D. M. Layton
Thesis Advisor

M. F. Platzer

M. F. Platzer
Chairman, Department of Aeronautics

J. N. Dyer

J. N. Dyer
Dean of Science and Engineering

ABSTRACT

This thesis will assist those evaluating helicopter design to make preliminary judgments about the feasibility of new designs. By using the computer program developed in this thesis, a designer can produce estimates for power requirements, endurance velocity, rate of climb, range velocity, hover ceiling, and service ceiling versus main rotor radius. These estimates can also be examined for the effects of changes in main rotor radius, chord, and rotational velocity.

TABLE OF CONTENTS

I.	INTRODUCTION	9
II.	APPROACH TO THE PROBLEM	11
III.	SOLUTION TO THE PROBLEM	13
IV.	RESULTS	18
V.	CONCLUSIONS	24
	APPENDIX A. COMPUTER PROGRAM OUTPUT - PLOTS	35
	APPENDIX B. COMPUTER PROGRAM LISTING	57
	LIST OF REFERENCES	96
	INITIAL DISTRIBUTION LIST	97

LIST OF TABLES

1.	Sample Data for Use in Program	13
2.	Assumptions Made in the Program	14
3.	Case Use of Variables	15
4.	Case Evaluation of Maximum Rate of Climb	19

LIST OF FIGURES

3.1	Example of Left Hand Plot	16
3.2	Example of Right Hand Plot	17
4.1	Example Maximum Rate of Climb - Case 1	20
4.2	Example Maximum Rate of Climb - Case 2	21
4.3	Example Maximum Rate of Climb - Case 3	22
4.4	Example Maximum Rate of Climb - Case 4	23

ACKNOWLEDGMENT

Acknowledgment is made of the assistance provided by Professor Donald M. Layton, thesis advisor, who helped in guiding the direction of this thesis. Much of the necessary information for this thesis was found during Professor Layton's excellent classes on performance and design of helicopters.

I. INTRODUCTION

A military engineering officer in a program or project office would probably never be called on to prepare a conceptual design of a helicopter, but he may well be required to evaluate a proposal submitted by a commercial contractor.

The requirement for such an evaluation might be stated as:

Determine if this design meets or exceeds the performance factors listed on the next page and if the design can be changed to optimize the performance.

There are several approaches one could use to complete this task. One could use a hand held calculator and enter the data for the many equations, or one could write a program for a micro or main frame computer to produce the required information. Both these options are laborious and time-consuming. By choosing the first option, one may be able to complete the task in two or three weeks or if it is decided to write a program, one may finish in one week provided that there are no "bugs".

The best solution of course, is to use an existing program which can determine the necessary information for the specification parameters. In addition, this program should be able to show what happens if certain parameters

are allowed to vary, thus providing the formation for optimization.

The objective of this thesis was to develop such a program that will graphically represent power requirements, endurance velocity, rate of climb, range velocity, hover ceiling and service ceiling all as a function of main rotor radius for four different cases. The graphs that result from this program should enable one not only to evaluate the basic helicopter design, but also to make recommendations for design improvement.

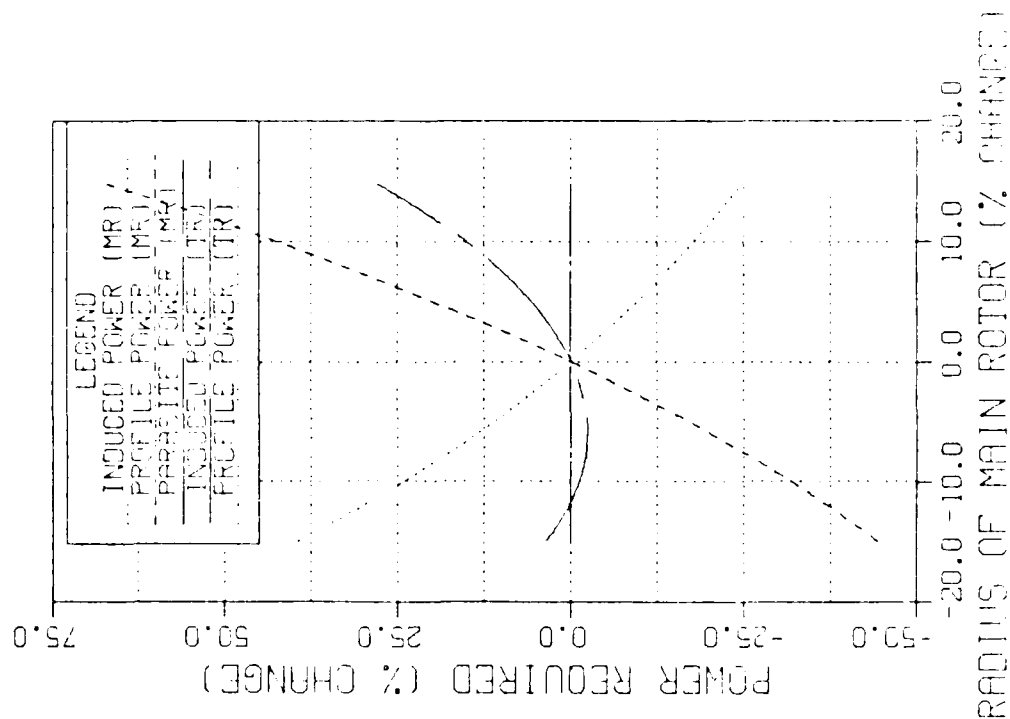
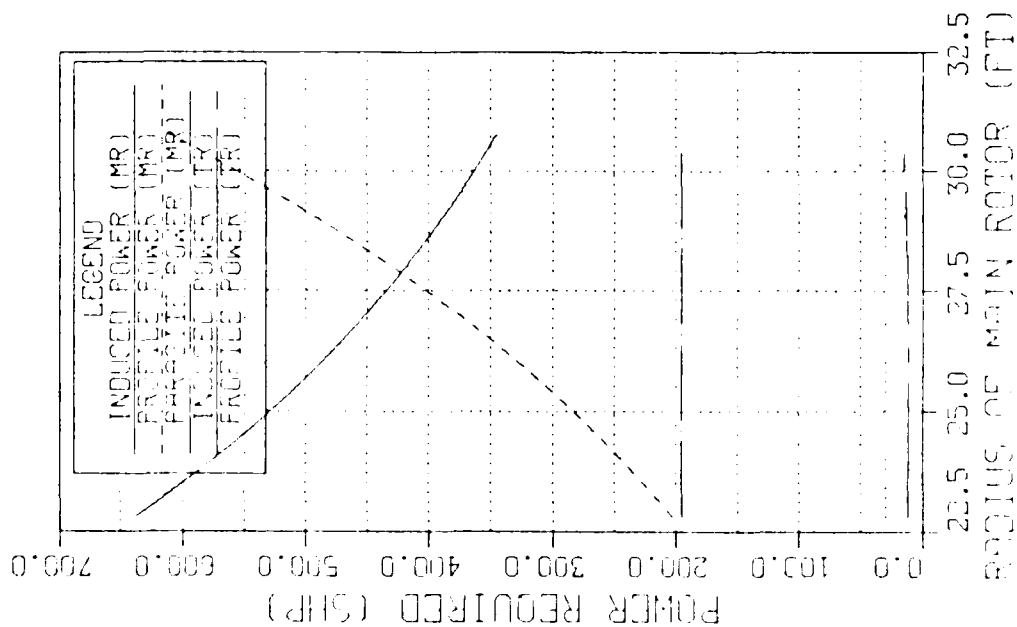
II. APPROACH TO THE PROBLEM

Six performance factors (power requirements, endurance, velocity, rate of climb, range velocity, hover ceiling and service ceiling) were determined to be major in designing of a helicopter. In order to obtain estimates for these, a Fortran program was to be written to be used with "DISSELA" (Display Integrated Software System and Plotting Language). [Ref. 1], to plot the results of the performance factors versus main rotor radius on either a screen or as a hard copy from the main frame computer.

The equations required for calculating these factors were obtained from "Helicopter Performance", [Ref. 2], and "Helicopter Design Manual", [Ref. 3], both written by Donald M. Layton. These equations were examined and then grouped for the purpose of writing subroutines to ensure an effective use of computer time. The result was six subroutines, Sub-group A, [Appendix B], are used in all the performance calculations. Sub-group A consist of:

1. RHO - Calculates density from pressure altitude and temperature or given density altitude. Also determines pressure and temperature ratio for altitude versus sea level.
2. VELMR - Calculates induced velocity for main rotor.
3. VELTR - Calculates induced velocity for tail rotor.

POWER VERSUS RADIUS CHORD & ROTATIONAL VELOCITY HELD CONSTANT SOLIDITY, TIP VELOCITY & ADVANCE RATIO ALLOWED TO VARY WITH RADIUS



APPENDIX A

COMPUTER PROGRAM OUTPUT - PLOTS

V. CONCLUSIONS

The objective of this thesis to design a program that evaluates a given set of helicopter design values was accomplished. The program will generate thirty two (32) graphs of six (6) performance factors for any single-rotor helicopter. The program does allow for one to optimize a design through the evaluation of the variation of parameters for each performance factor.

At this point, the program takes into account neither compressibility nor blade stall. These factors would improve the estimations and may prove to be a worthwhile project for further work. Other points which could be examined are twin-blade and no-tail-rotor helicopters. Of the points mentioned above, blade design would probably be of most importance since little has been done to examine the optimization of blade twist and composite structure of blades.

MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE
 SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
 CHORD & ROTATIONAL VELOCITY
 ALLOWED TO VARY WITH RADIUS

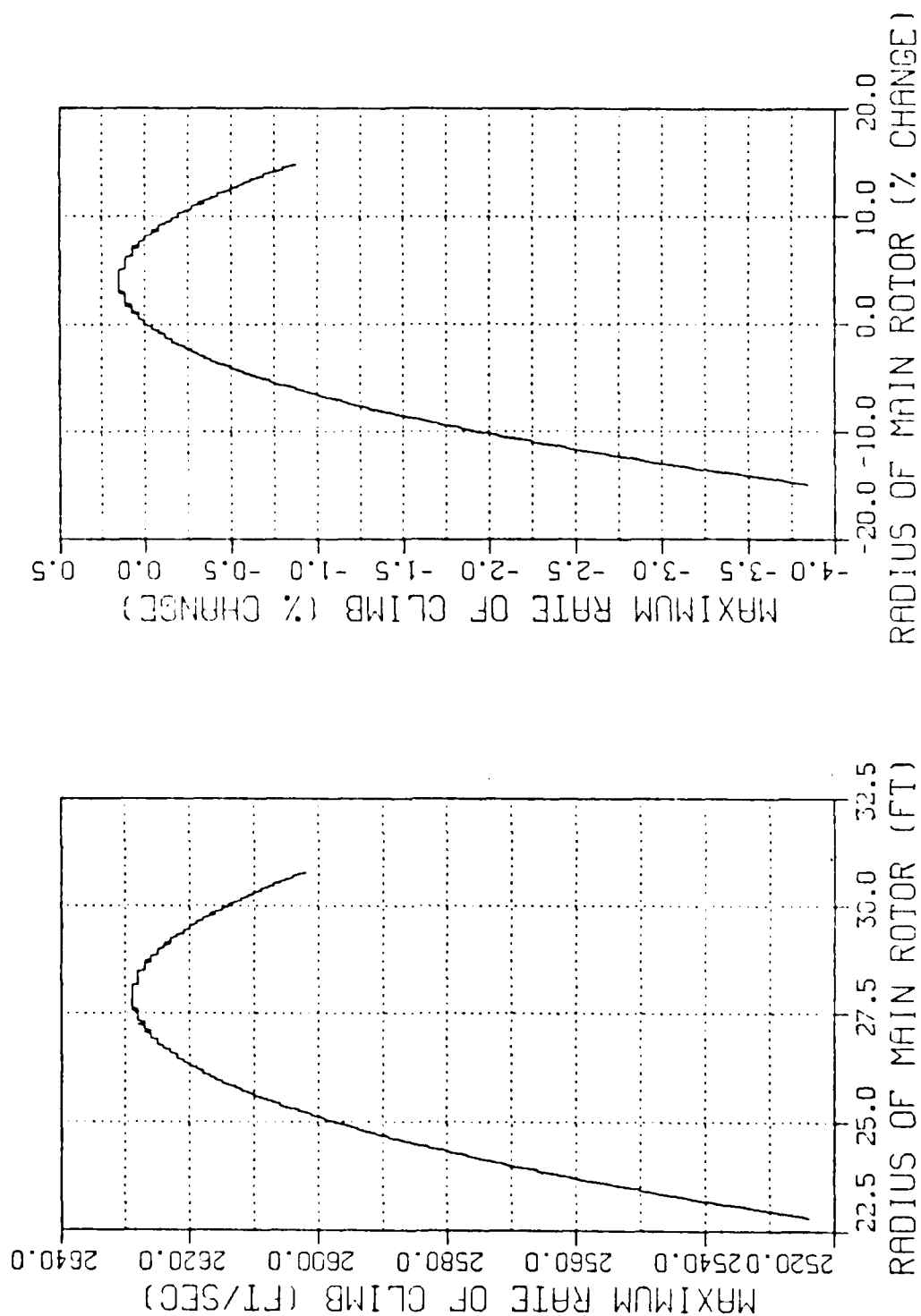


Figure 4.4 Example Maximum Rate of Climb - Case 4

MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE
 SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
 CHORD, ADVANCE RATIO & TIP VELOCITY
 ALLOWED TO VARY WITH RADIUS

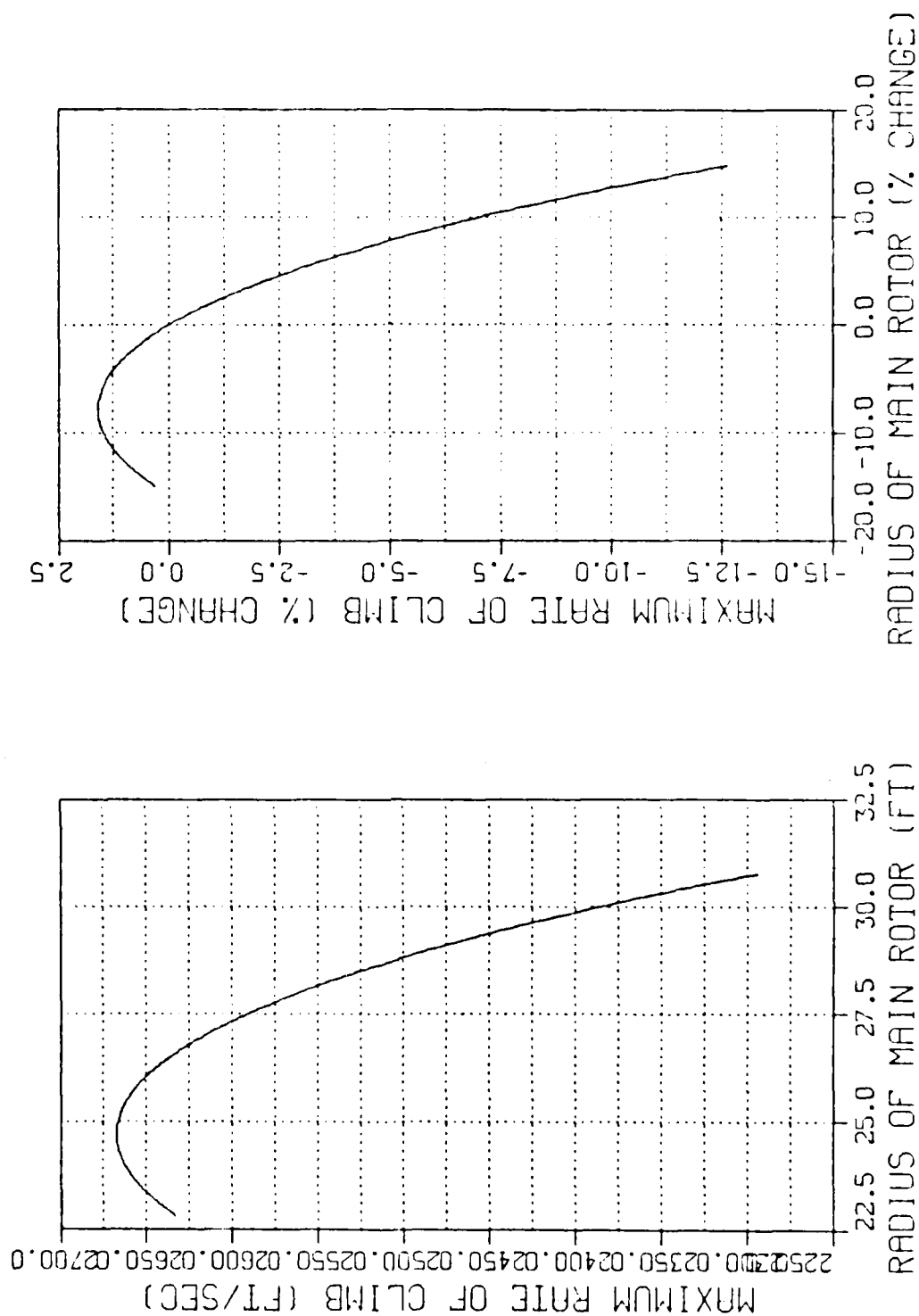


Figure 4.3 Example Maximum Rate of Climb - Case 3

MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE
 CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
 SOLIDITY, ROTATIONAL VELOCITY
 ALLOWED TO VARY WITH RADIUS

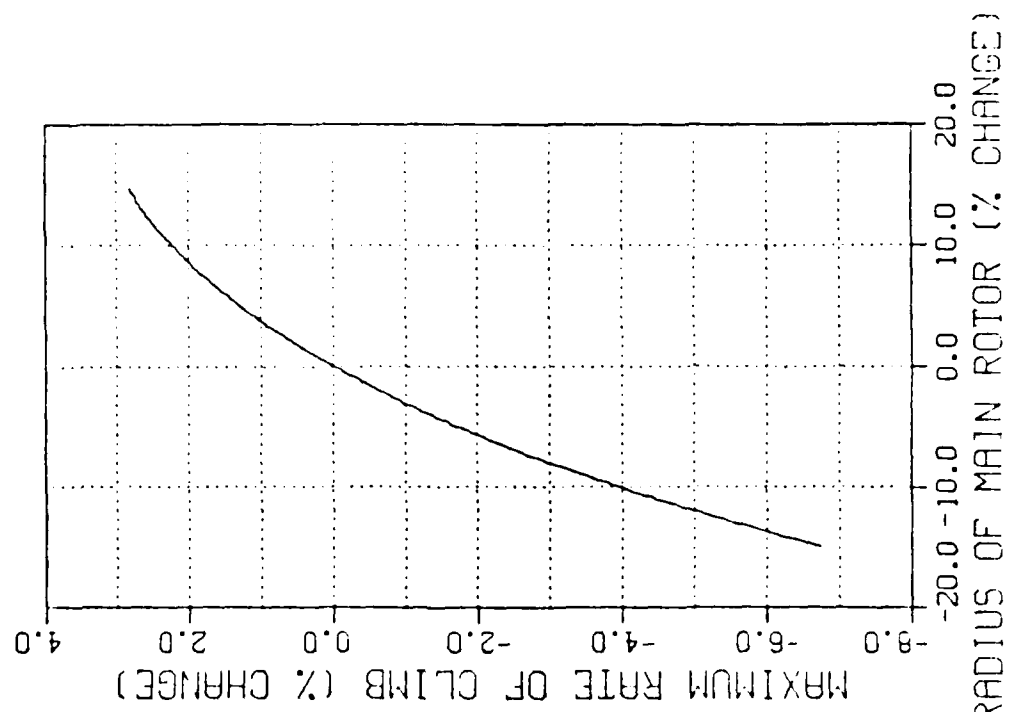
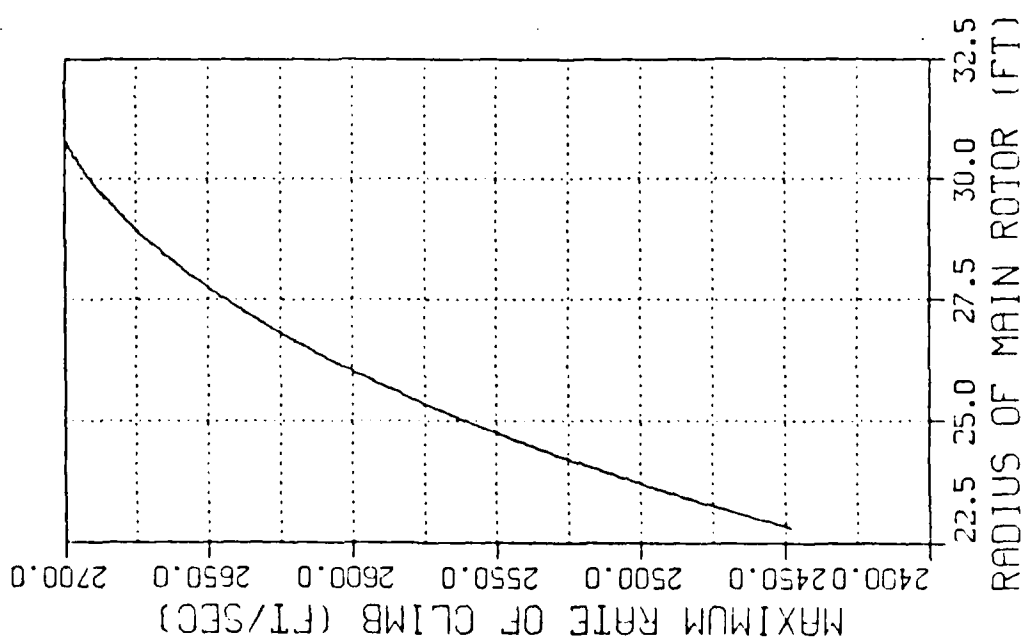


Figure 4.2 Example Maximum Rate of Climb - Case 2

MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE
 CHORD & ROTATIONAL VELOCITY HELD CONSTANT
 SOLIDITY, TIP VELOCITY & ADVANCE RATIO
 ALLOWED TO VARY WITH RADIUS

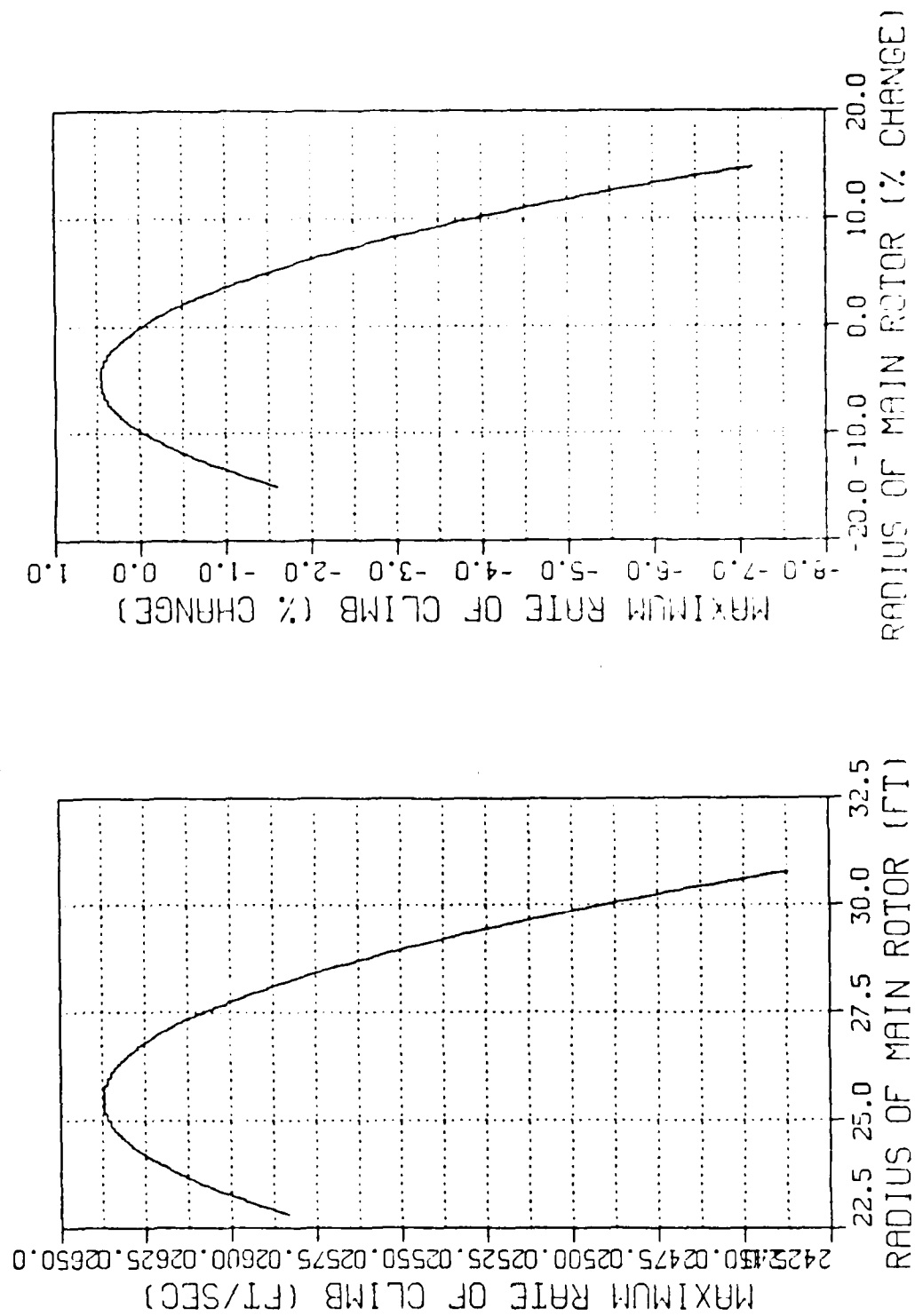


Figure 4.1 Example Maximum Rate of Climb - Case 1

and percent increase of rate of climb. The cases were explained in Table 3 and the Figures follow Table 4.

Table 4

Case Evaluation of Maximum Rate of Climb Plots

Case	Figure Reference	Rate of Climb(ft/sec)	Radius (ft)	Percent Increase
1	4.1	2628	27.8	+ 0.130
2	4.2	2668	24.8	- 0.156
3	4.3	2700	30.63	+ 2.80
4	4.4	2637	25.63	- 5.00

From examining Table 4, it would be best to increase the radius to 30.63 feet while holding the chord to 1.75 feet. the tip velocity to 723.6 ft/sec (radius times rotational velocity), changing the rotational velocity to 23.62 rad/sec, and calculating solidity from radius and chord. This, however, may not be the best answer if there is a limitation to the radius of the blade. For this particular example, the optimum solution may be to leave the design as it is. Increasing or decreasing the radius by small amounts does little to increase or decrease the rate of climb.

One must not only look at one factor in making the final decision. The changes affect the performance factors in different ways, therefore, the final decision must be made after weighting all the changes against all the performance factors.

IV. RESULTS

The program will generate a total of thirty two (32) plots. The first twelve (12) pertain to power requirements. This includes induced, profile, and parasite power for the main rotor; and induced and profile power for the tail rotor. To indicate a relation between the power terms of the main rotor the figure of merit (induced divided by the sum of the induced and profile power) is included. The sum of the terms (total power) for the main and tail rotor was included with the total power required for the aircraft. The remaining twenty (20) plots are for the performance factors of endurance velocity, rate of climb, range velocity, hover ceiling and service ceiling.

The data for this particular model (UH-60A) was used to illustrate the features of the program and the results presented are typical for this example only. The specifications of any model or design can be used as long as the assumption of a single-mast helicopter with rectangular blades is followed.

The plots created by the program represent the results of varying the main rotor radius, chord or rotational velocity. To example how to interrupt the results, the four (4) plots for Maximum Rate of Climb will be analyzed. Table 4 compares the results in relation to rate of climb, radius

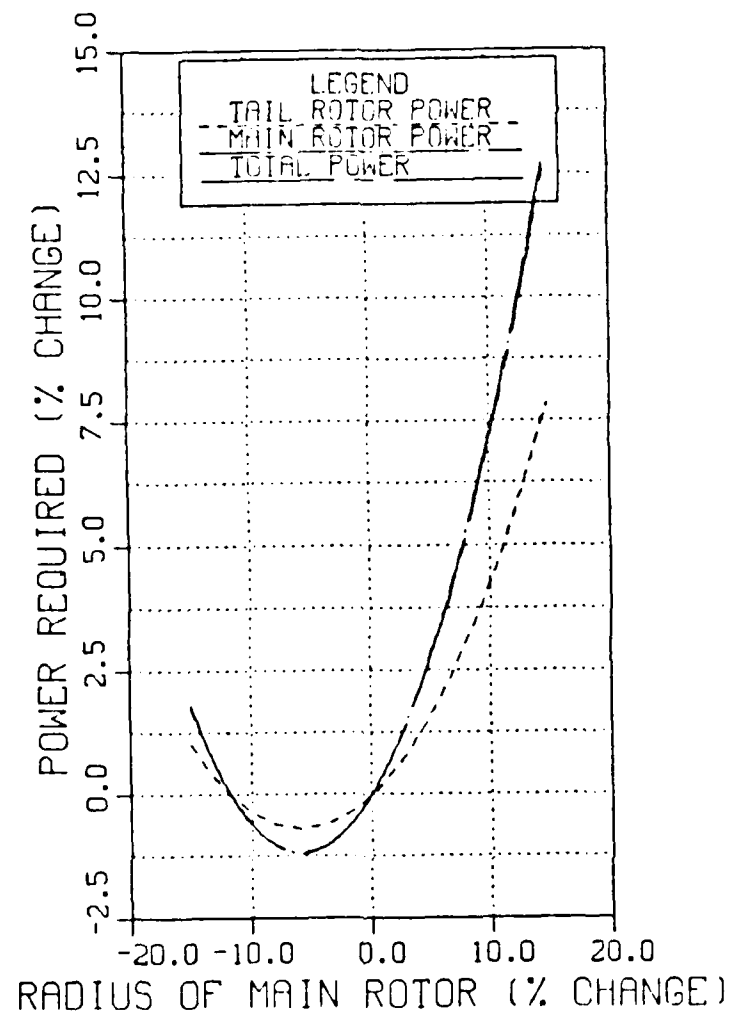


Figure 3.2 Example of Right Hand Plot

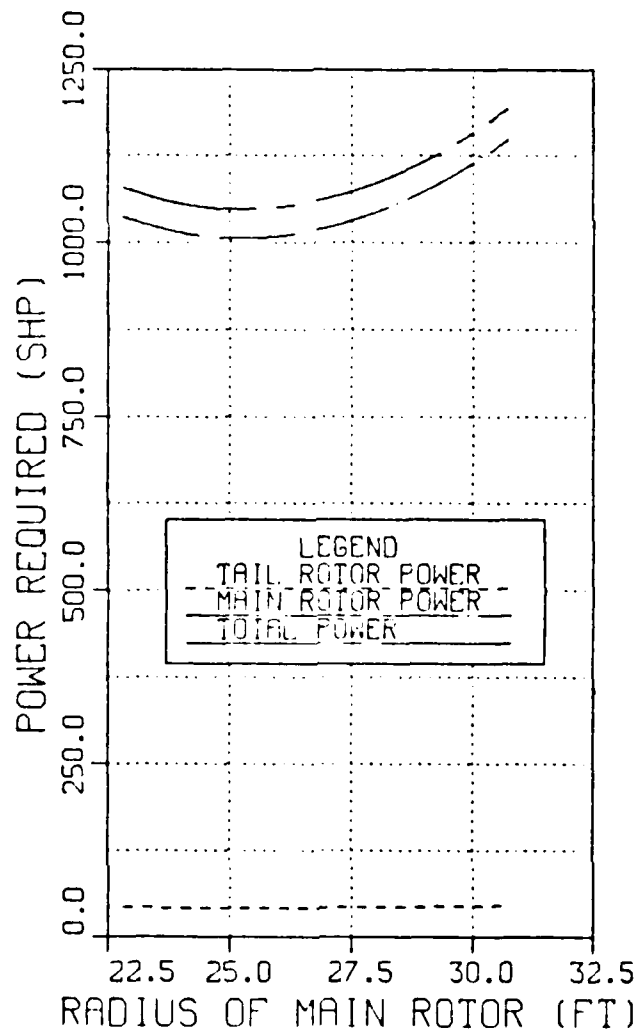


Figure 3.1 Example of Left Hand Plot

The right hand plot shows the values of the performance factors based on a percent difference from the value at the specification conditions versus a percent difference of the radius from the radius specification. Figure 3.2 shows an example of power required (% change) versus radius of the main rotor (% change), Case 1.

percent (3%) for transmission losses (if three engines are used, an additional transmission is required), [Ref. 2].

The secondary goal of the thesis was to examine four (4) different cases involving main rotor Radius, Chord, Rotational Velocity, Tip Velocity, Advance Ratio, and Solidity. Table 3 shows how these variables were used in each of the four (4) cases.

Table 3

Case Use of Variables

	Radius	Chord	Rotational Velocity	Advance Ratio	Tip Velocity	Solidity
Case 1	V	C	C	V	V	V
Case 2	V	C	V	C	C	V
Case 3	V	V	C	V	V	C
Case 4	V	V	V	C	C	C

V : VARIES C : HELD CONSTANT

Advance Ratio, Tip Velocity, and Solidity when held constant keep their values at the specification conditions. The values of Chord and Rotational Velocity were allowed to vary when Advance Ratio, Tip Velocity, or Solidity were held constant as shown in Table 3.

The graphs consist of two (2) plots representing the same data but presented in different ways. The left hand plot shows the actual performance factor value versus main rotor radius. Figure 3.1 shows an example using power required versus main rotor radius, Case 1.

The example uses a value of four (4.0) feet. This value is assigned to a variable called "diff". The program was written to allow the X-axis scale of the plot to be adjusted in accordance with any value you might choose for "diff". The Y-axis scales can also change based on the maximum and minimum values for the performance factor being considered.

Certain assumptions had to be made in order to write the program. Table 2 lists these assumptions and factors used and the line number in the program where the factors can be changed if needed.

Table 2
Assumptions Made in the Program

Assumption	Factor	Line Number
Single Mast Helicopters	NA	NA
Only Rectangular Blades	NA	NA
Profile Power Factor	4.3	1079,1108
Ground Effect Factor	1.6	1138
Transmission & Accessory Losses		
1 engine SHP = (# ENG*ESHP-10.0)*0.97		560,780.887
2 engines SHP = (# ENG*ESHP-10.0)*0.9*0.97		561,781.888
3 engines SHP = (# ENG*ESHP-10.0)*0.9*0.94		562,782.889

ENG: Engine, ESHP: Engine shaft horsepower,
SHP: Rotor Shaft horsepower

The Transmission and Accessory Losses were calculated based on a loss of ten (10) horsepower for the accessories, ten percent (10%) for multiple engine installation, and three

III. SOLUTION TO THE PROBLEM

The program requires input data prior to compiling and execution. To demonstrate how to use the program a set of sample data was used. Table 1 is a listing of the required input data for the program.

Table 1

Sample Data for Use in Program

	Main Rotor	Tail Rotor
Radius	26.8 ft	5.5 ft
Chord	1.75 ft	0.81 ft
Rotational velocity	27.0 rad/sec	124.6 rad/sec
Coefficient of drag	0.008	0.008
Number of blades	4	4

General Helicopter Data

Weight	20000.0 lbs
Tail boom length	31.5 ft
Effective flat plate area (forward)	25.7 sq ft
Effective flat plate area (vertical)	31.8 sq ft
Main rotor height above skids or wheels	11.2 ft
Aircraft velocity (power calculations only)	90.0 kts
Plus/minus (+/-) value for main rotor radius	4.0 ft

Engine Data

Number of engines	2
Shaft horsepower output (military)	1561.0 shp
Shaft horsepower output (normal)	1318.0 shp
Shaft horsepower output (cruise)	989.0 shp
Specific fuel consumption (military)	0.46 lbs/shp/hr
Specific fuel consumption (normal)	0.47 lbs/shp/hr
Specific fuel consumption (cruise)	0.51 lbs/shp/hr

As indicated in Table 1, a value is inputted for the range over which you wish to examine the main rotor radius.

4. POWMR - Calculates power requirements for main rotor only.

5. POWTR - Calculates power requirements for tail rotor and total aircraft.

6. CONST - Calculates general constants to be used in the first five subroutines listed above.

In addition to the subroutines listed above, six more, Sub-group B, [Appendix B], were written to perform the necessary iterations to calculate the performance factors. Some of Sub-group B subroutines were iterated in steps (1000,100,10,1) so that it would not require extensive computer time to obtain the desired results. A final eight subroutines, Sub-group C, [Appendix B], were written to plot the results using DISPLA.

The program takes into account neither compressibility nor blade stall. These considerations were omitted since this program was aimed at facilitating preliminary decision making for a new design, rather than a final production design, and usually the blade selected for a new design is taken from an existing helicopter.

POWER VERSUS RADIUS CHORD & ROTATIONAL VELOCITY HELD CONSTANT SOLIDITY, TIP VELOCITY & ADVANCE RATIO ALLOWED TO VARY WITH RADIUS

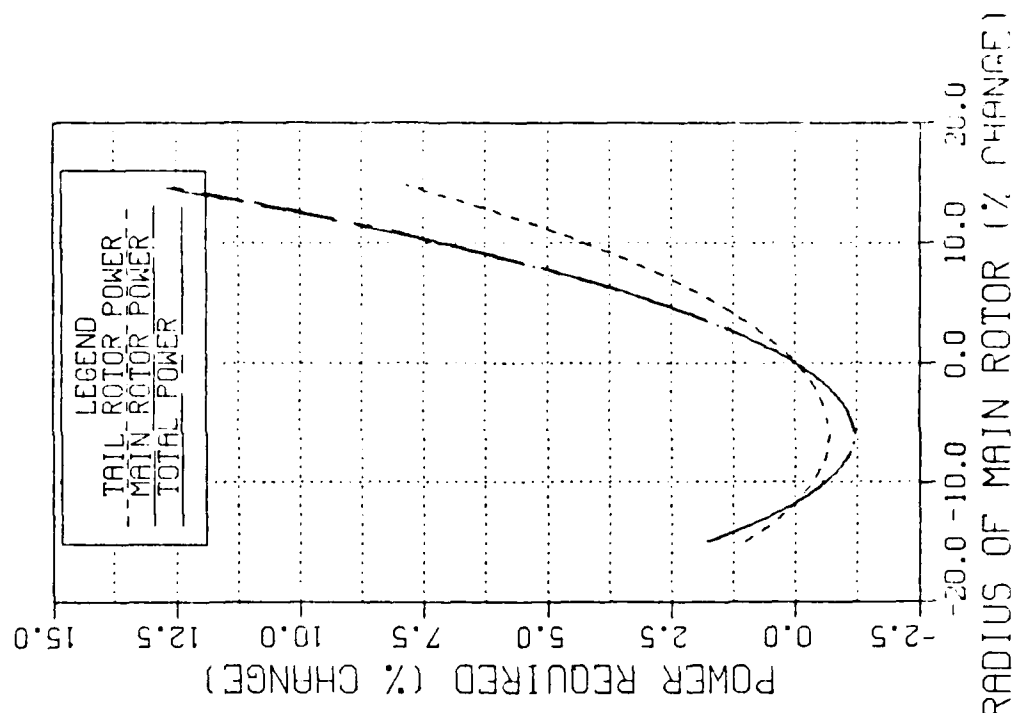
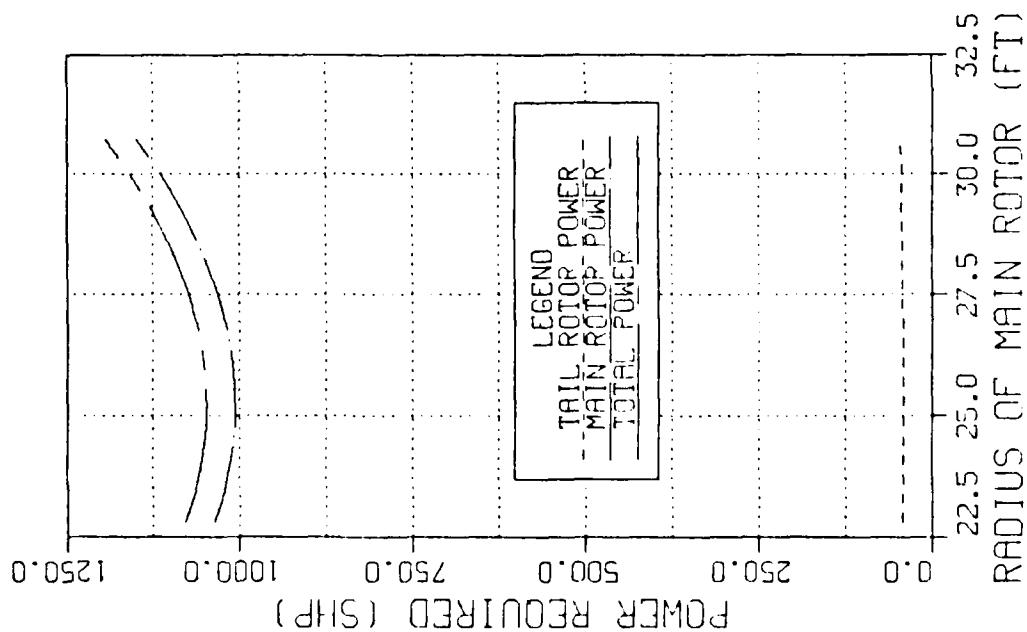
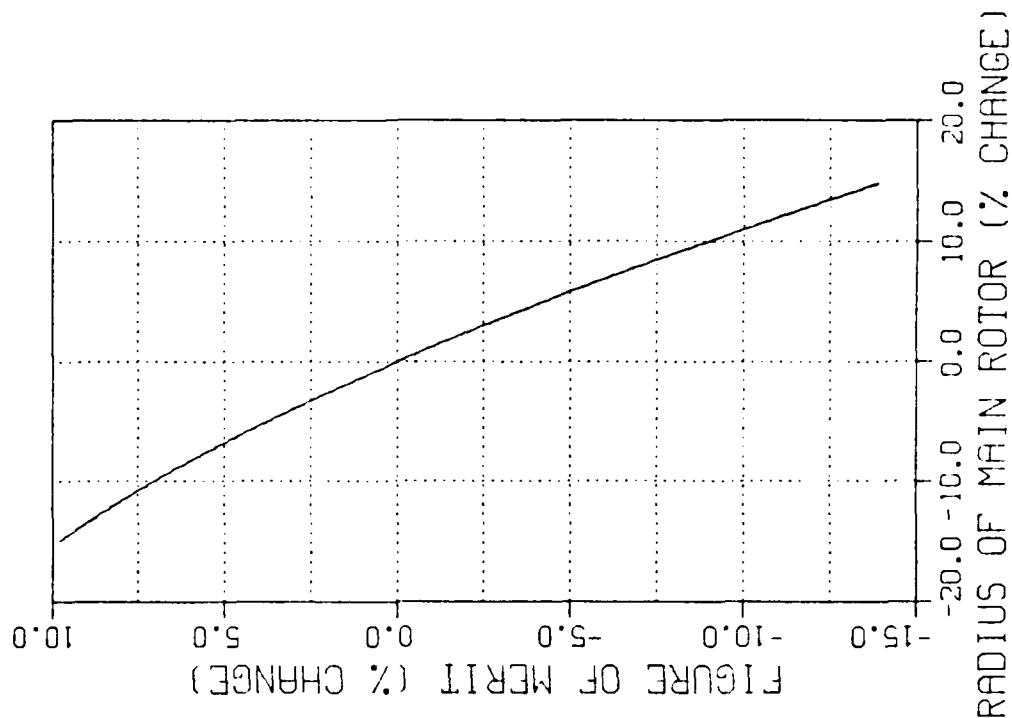
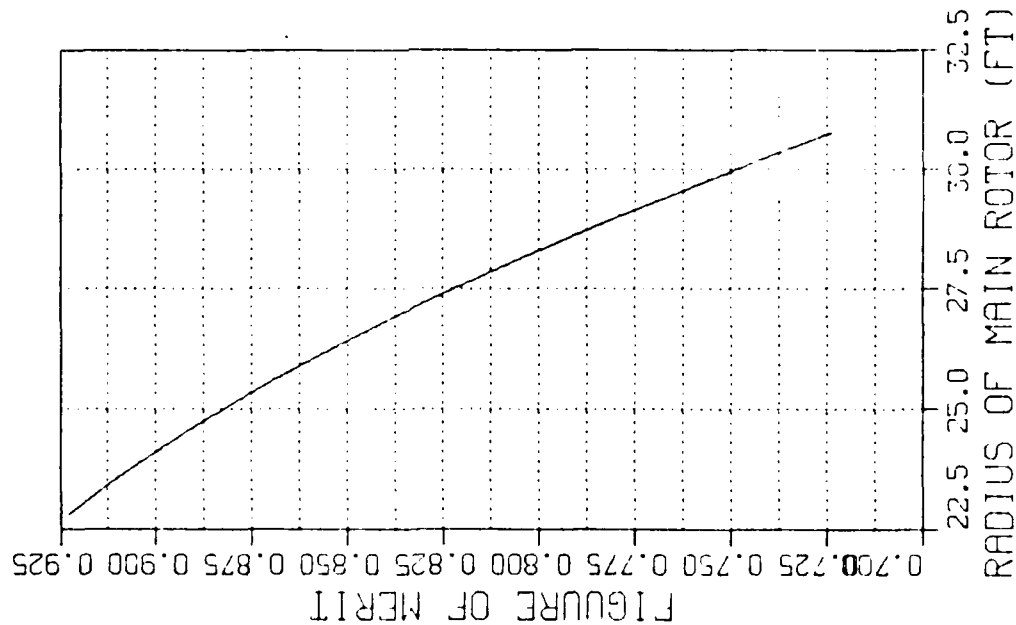
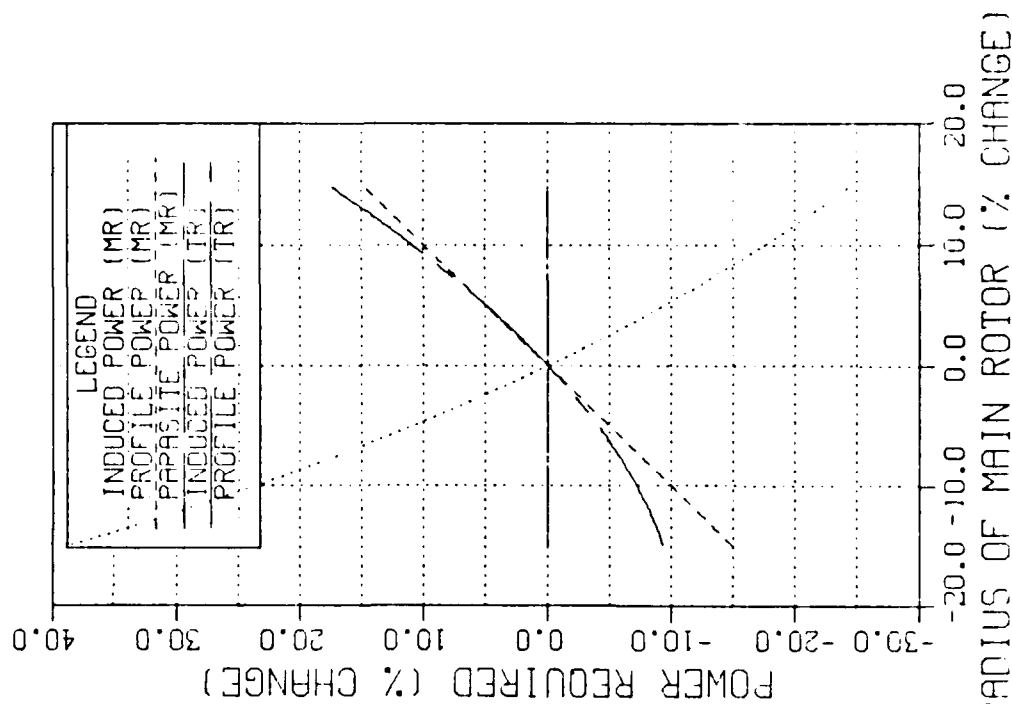
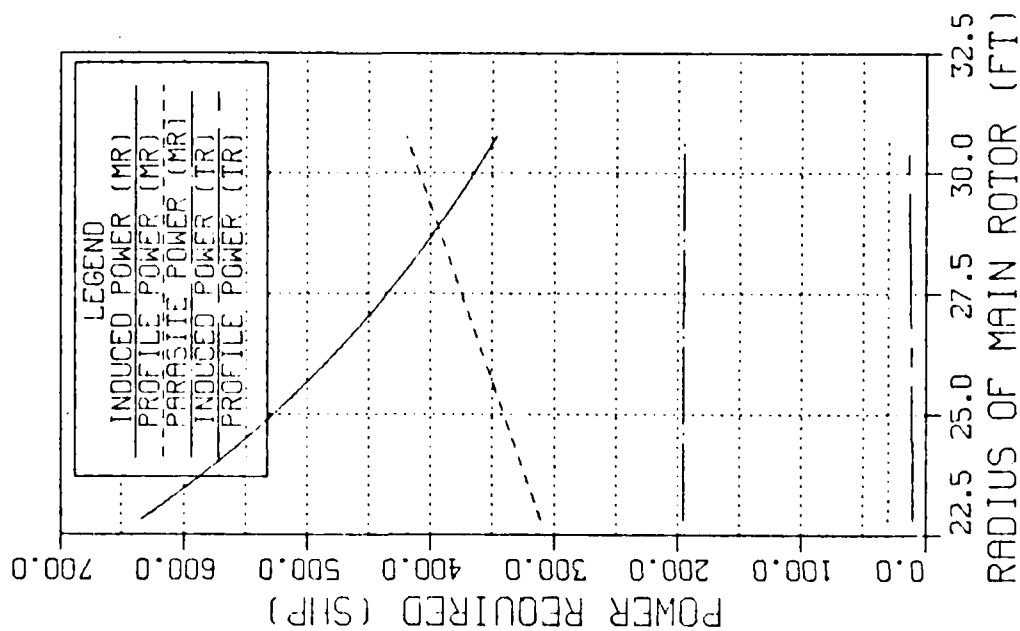


FIGURE OF MERIT VERSUS RADIUS CHANGE
 CHORD & ROTATIONAL VELOCITY HELD CONSTANT
 SOLIDITY, TIP VELOCITY & ADVANCE RATIO
 ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
SOLIDITY, ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS

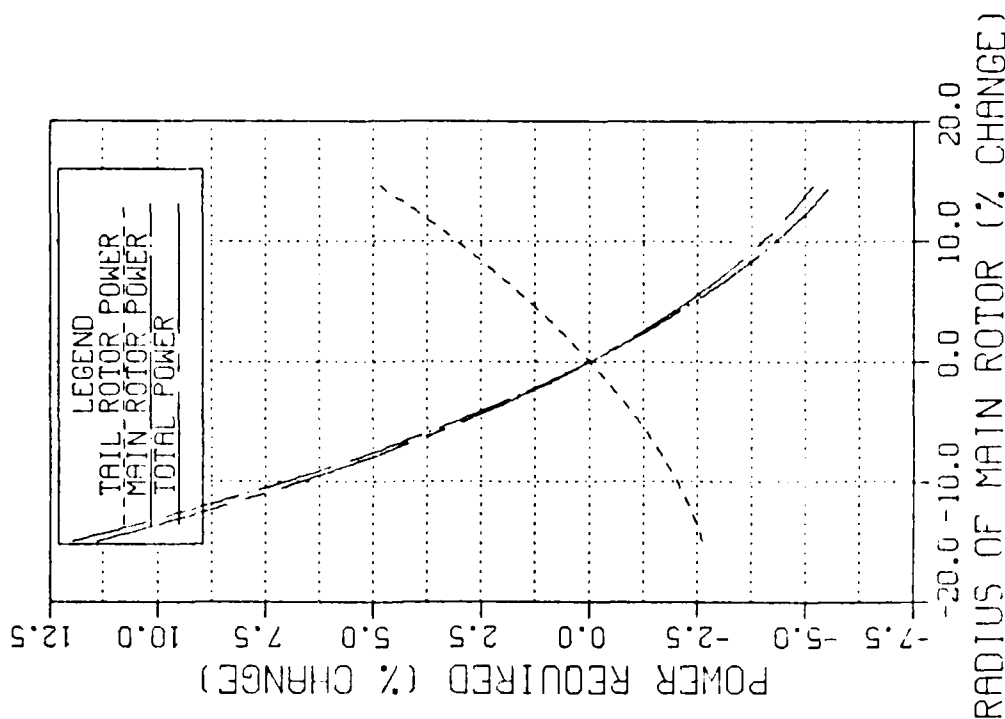
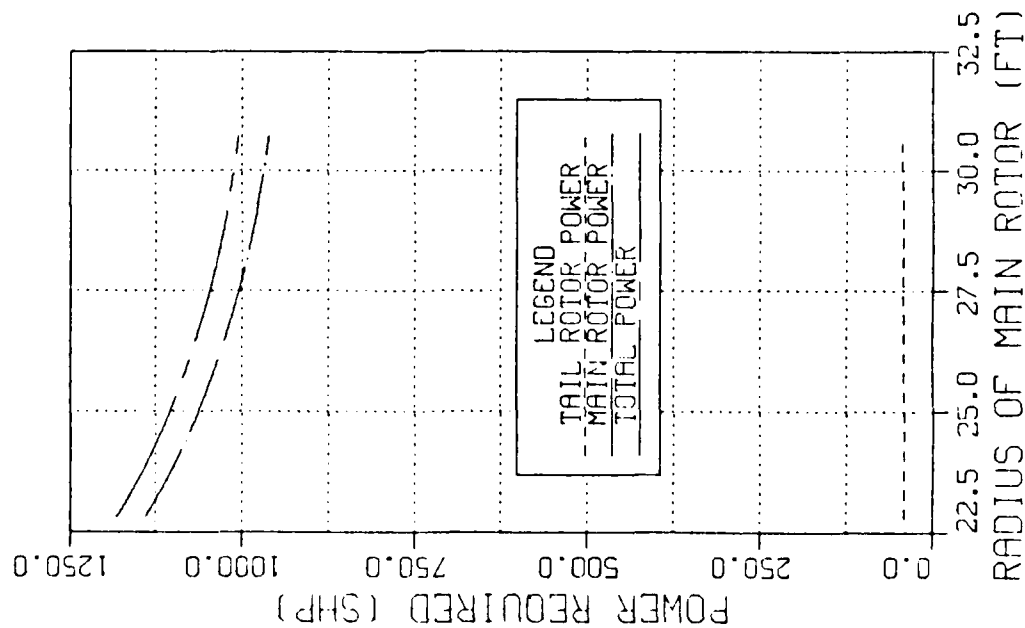
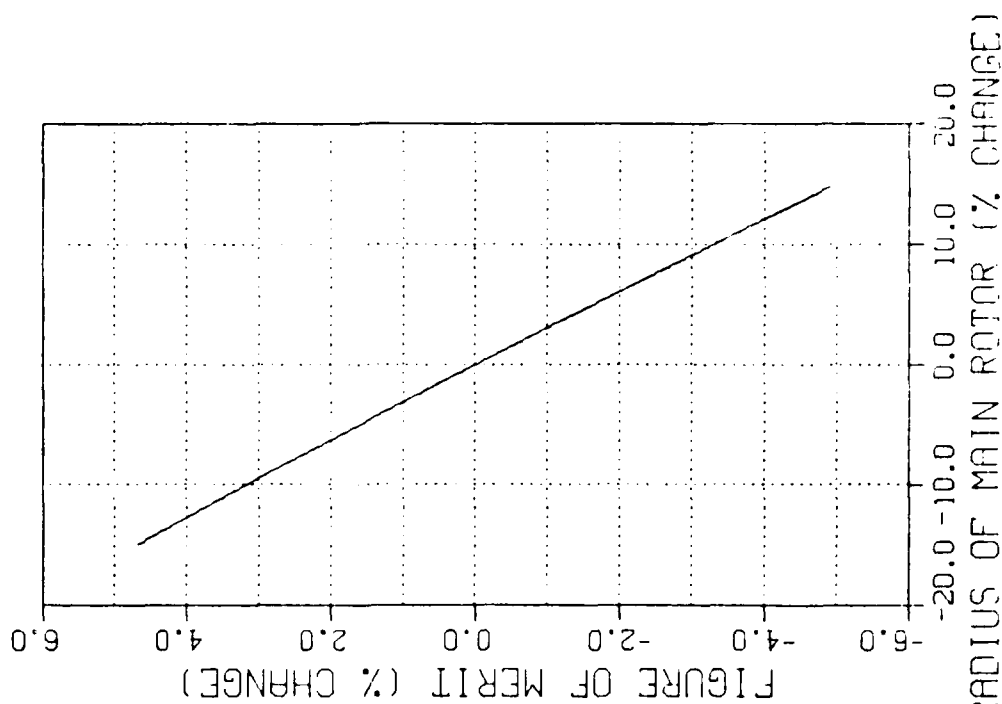
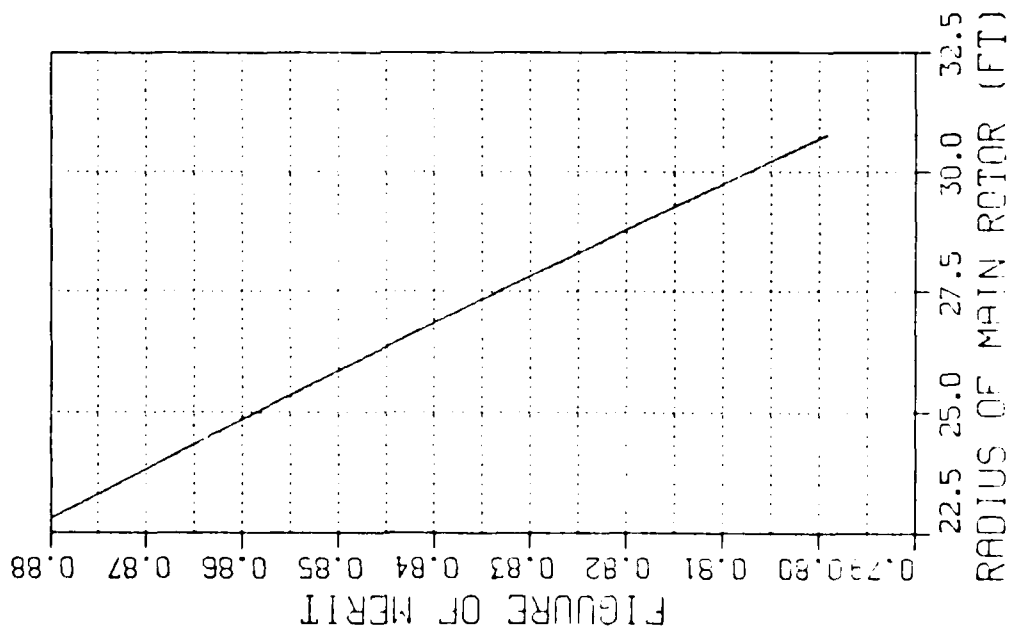
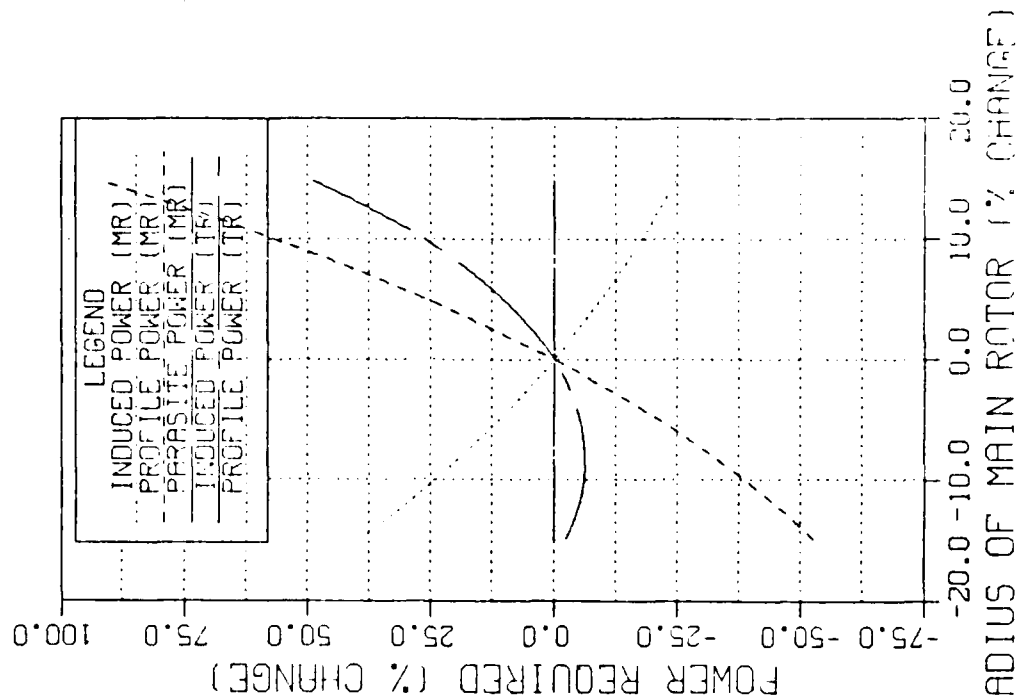
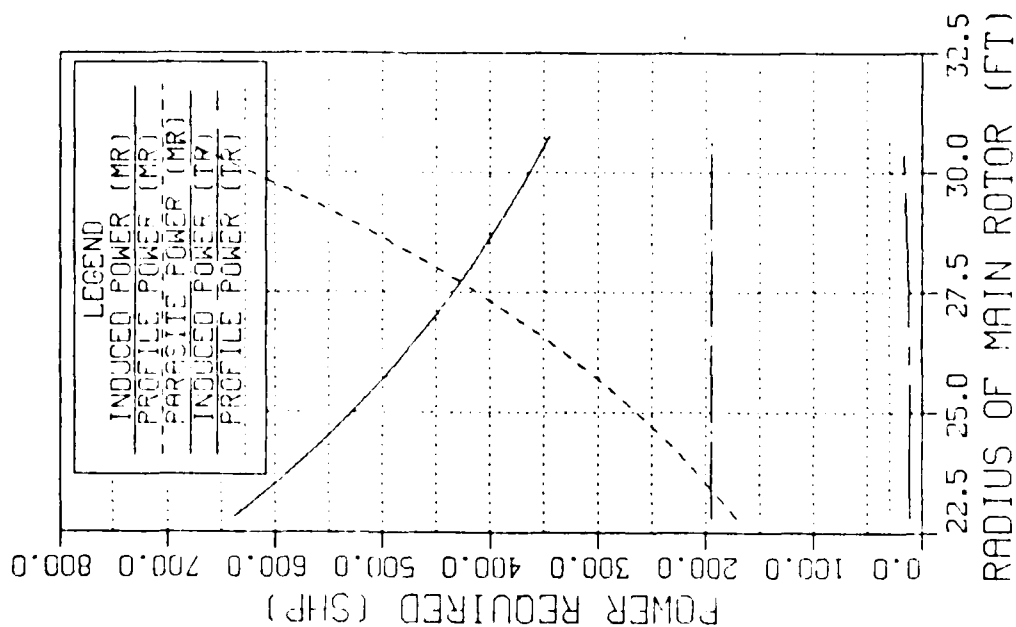


FIGURE OF MERIT VERSUS RADIUS CHANGE
 CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
 SOLIDITY, ROTATIONAL VELOCITY
 ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
CHORD, ADVANCE RATIO & TIP VELOCITY
ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT CHORD, ADVANCE RATIO & TIP VELOCITY ALLOWED TO VARY WITH RADIUS

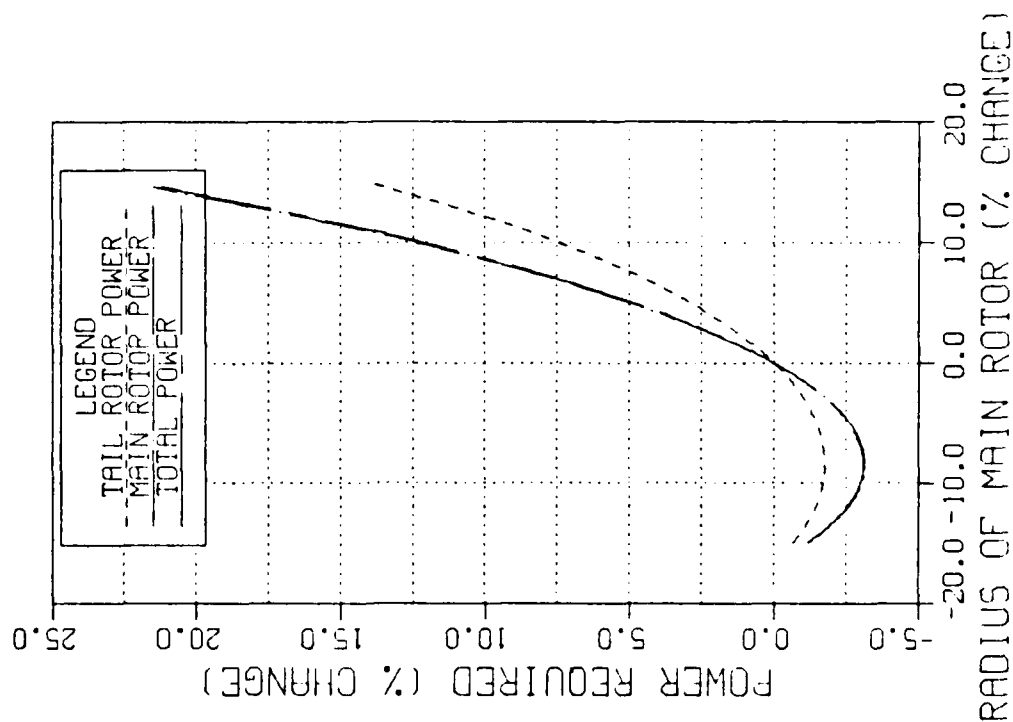
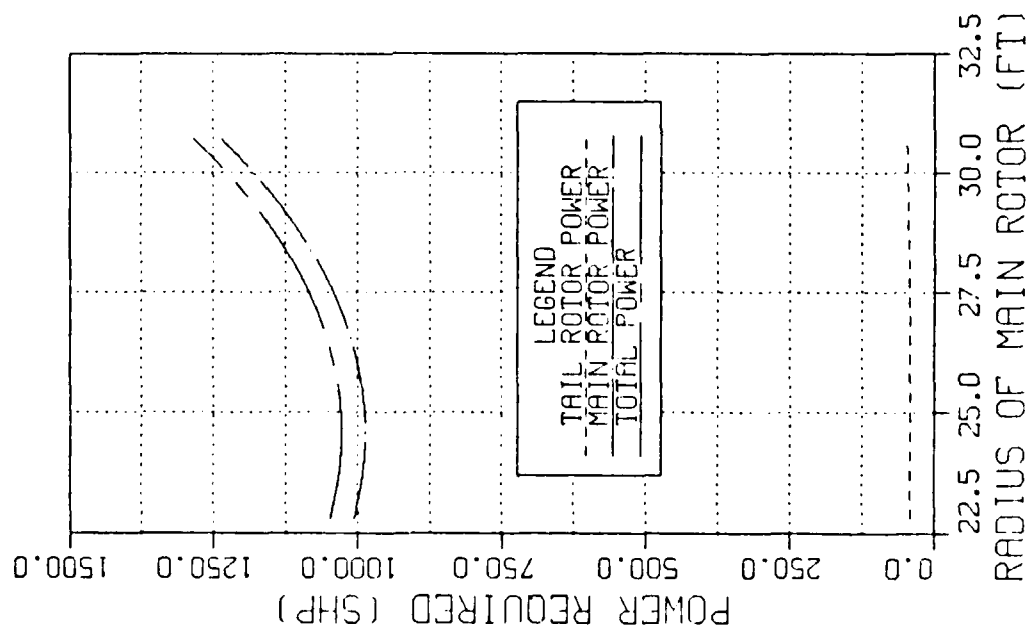
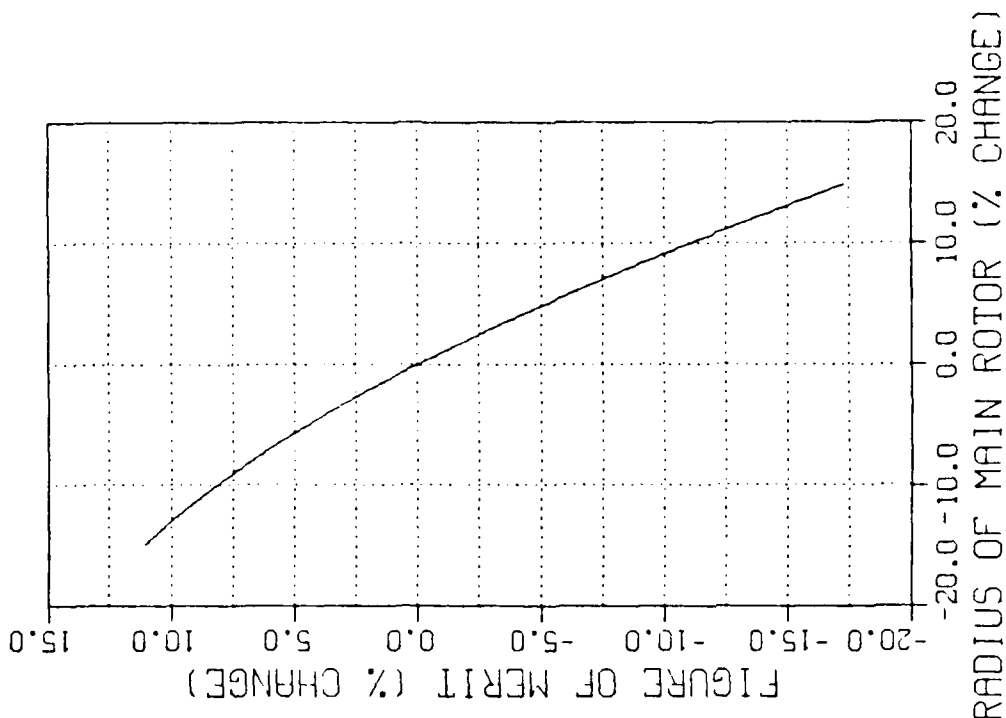
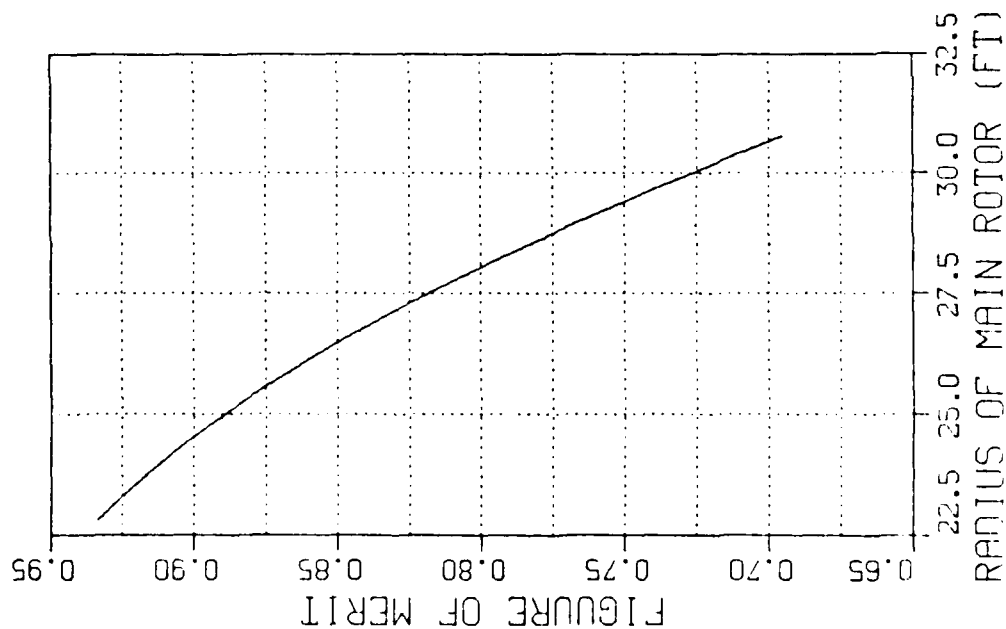
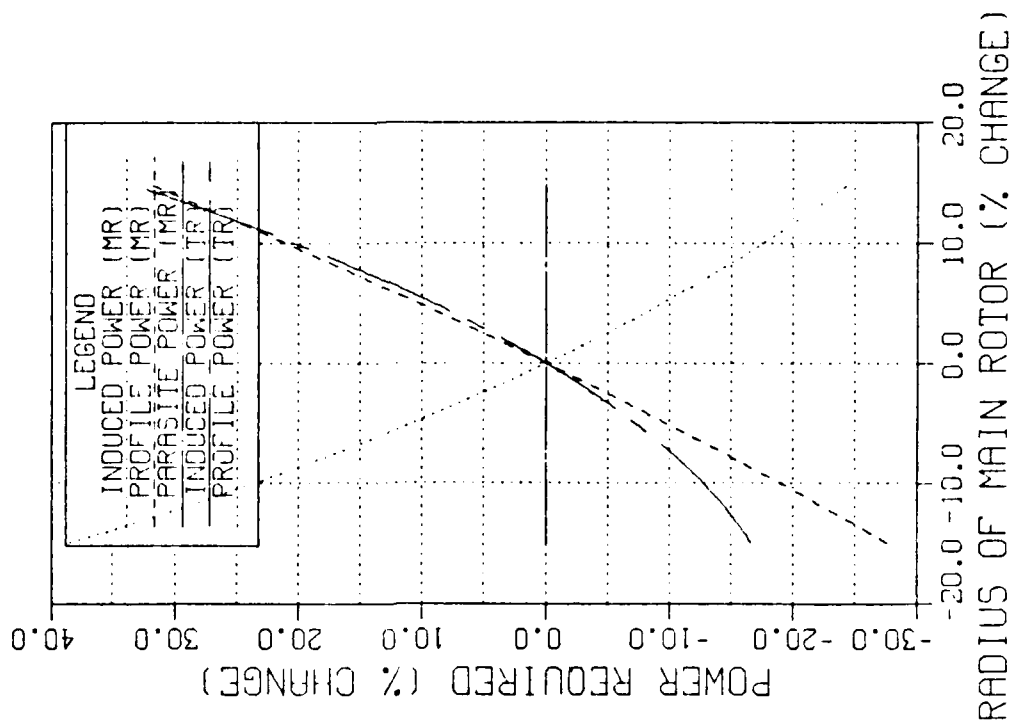
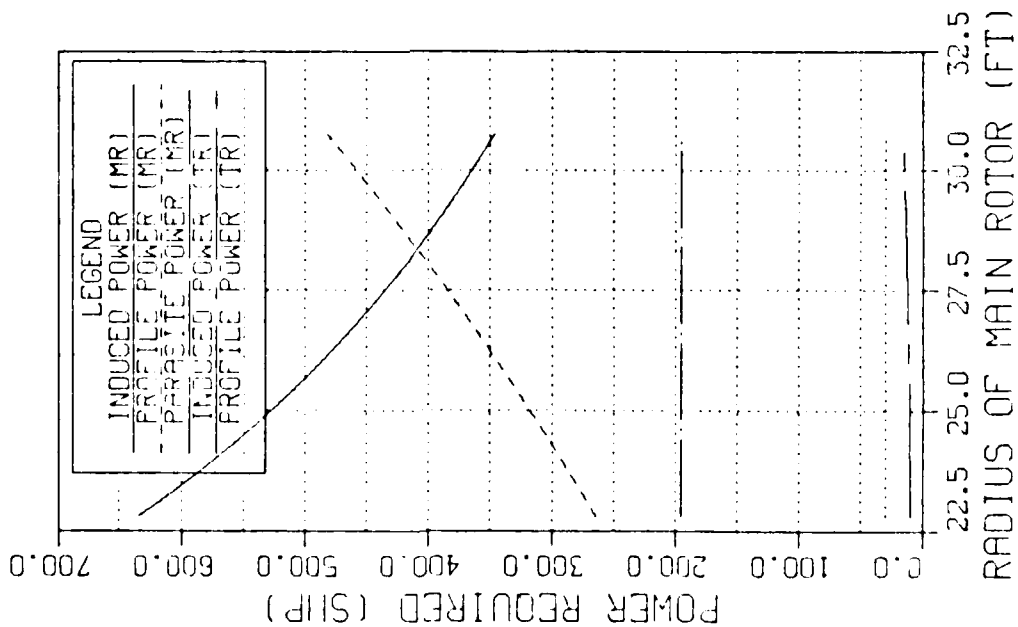


FIGURE OF MERIT VERSUS RADIUS CHANGE
 SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
 CHORD, ADVANCE RATIO & TIP VELOCITY
 ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS



POWER VERSUS RADIUS

SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
CHORD & ROTATIONAL VELOCITY
ALLOWED TO VARY WITH RADIUS

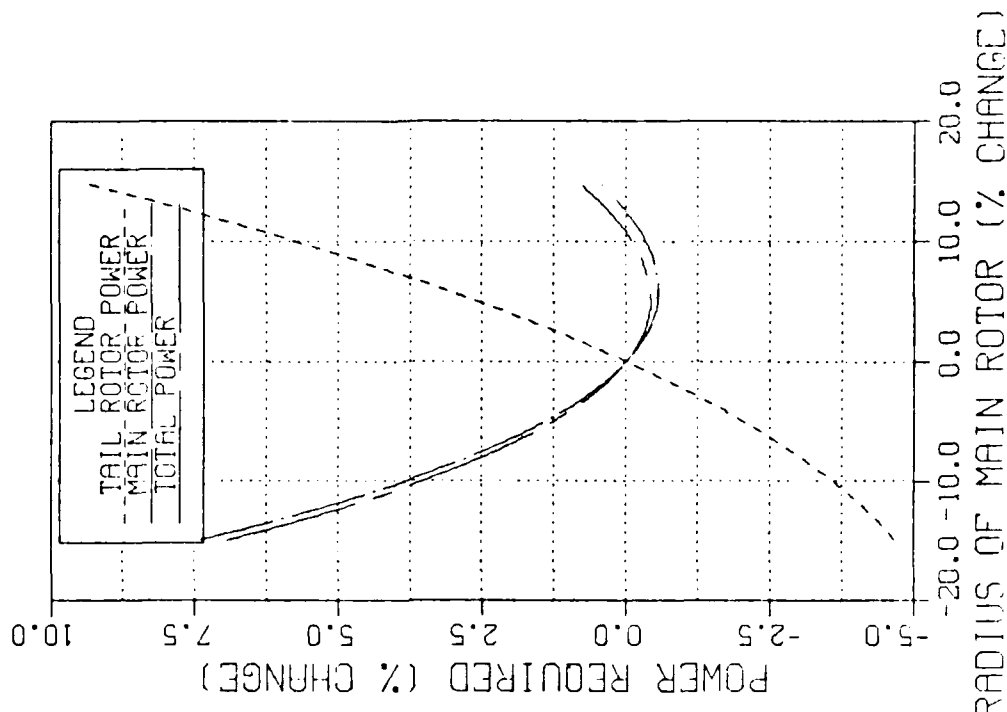
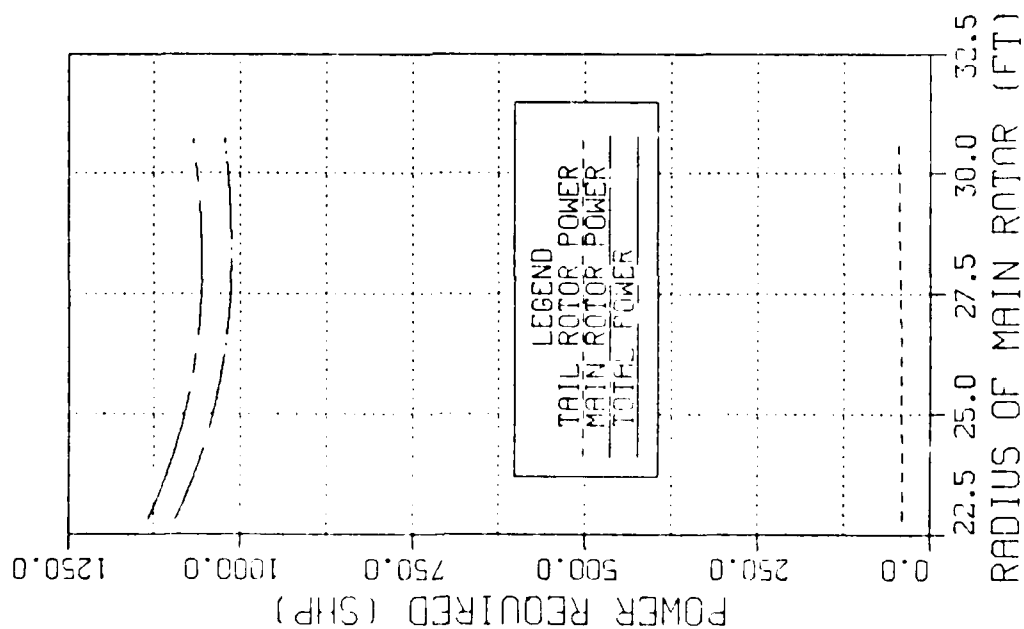
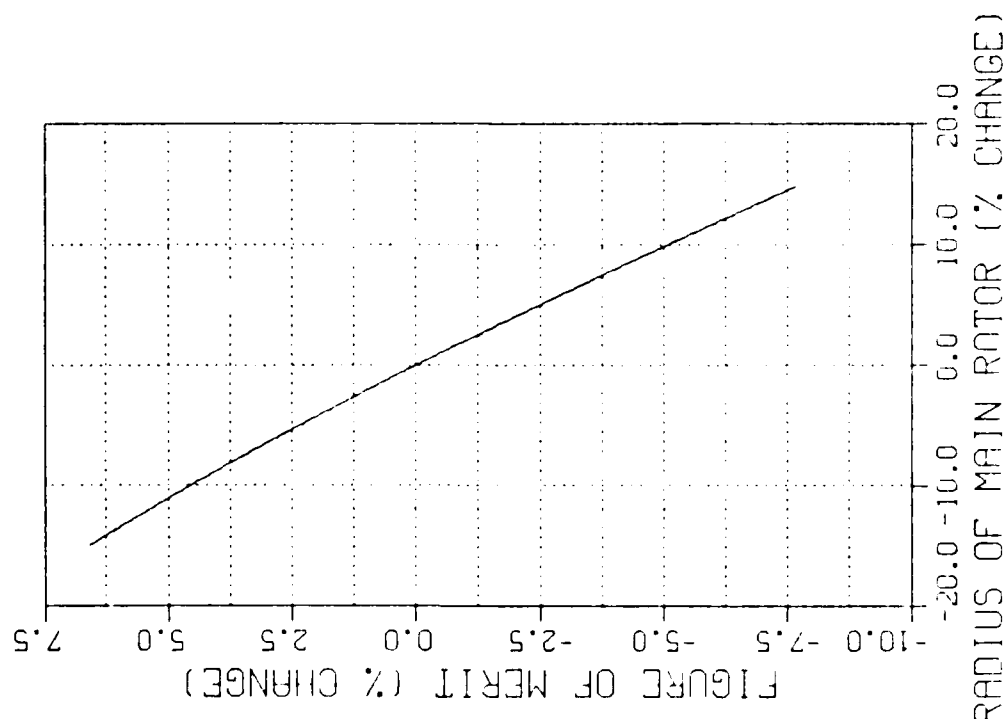
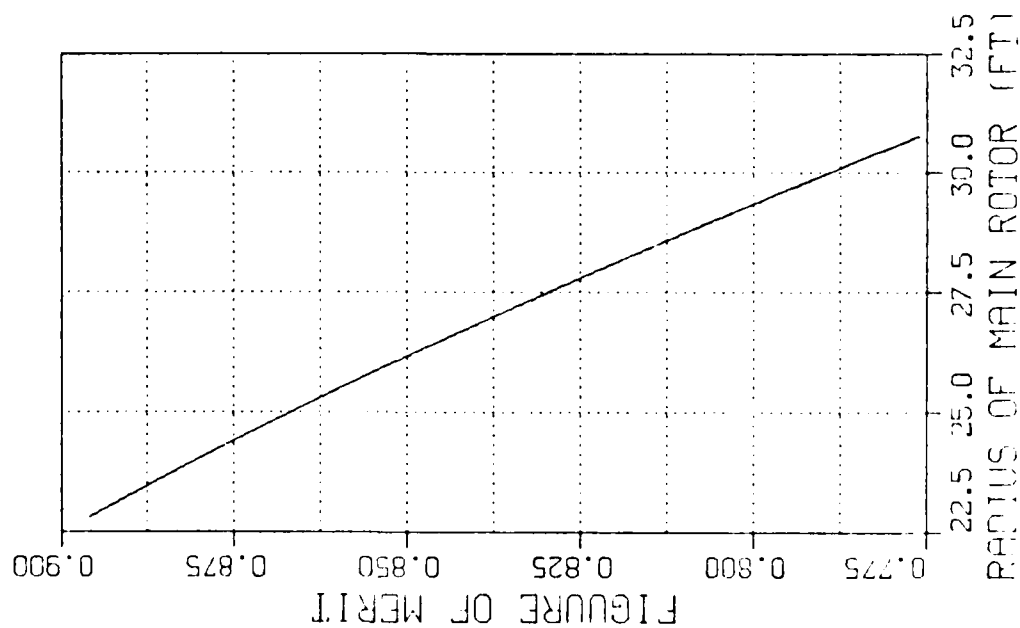


FIGURE OF MERIT VERSUS RADIUS CHANGE
 SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
 CHORD & ROTATIONAL VELOCITY
 ALLOWED TO VARY WITH RADIUS

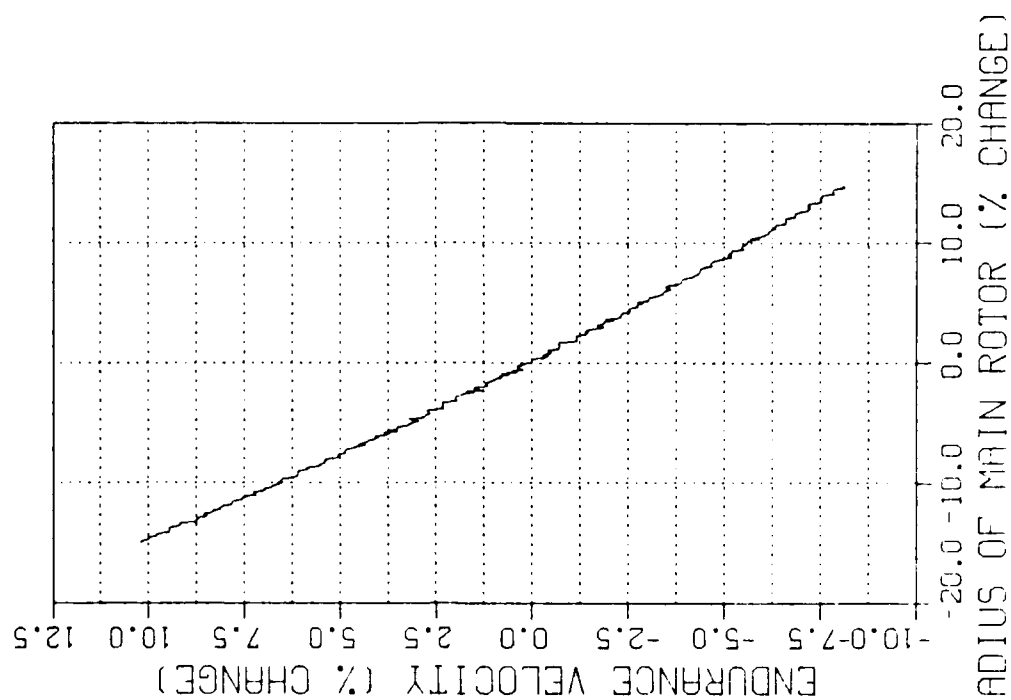
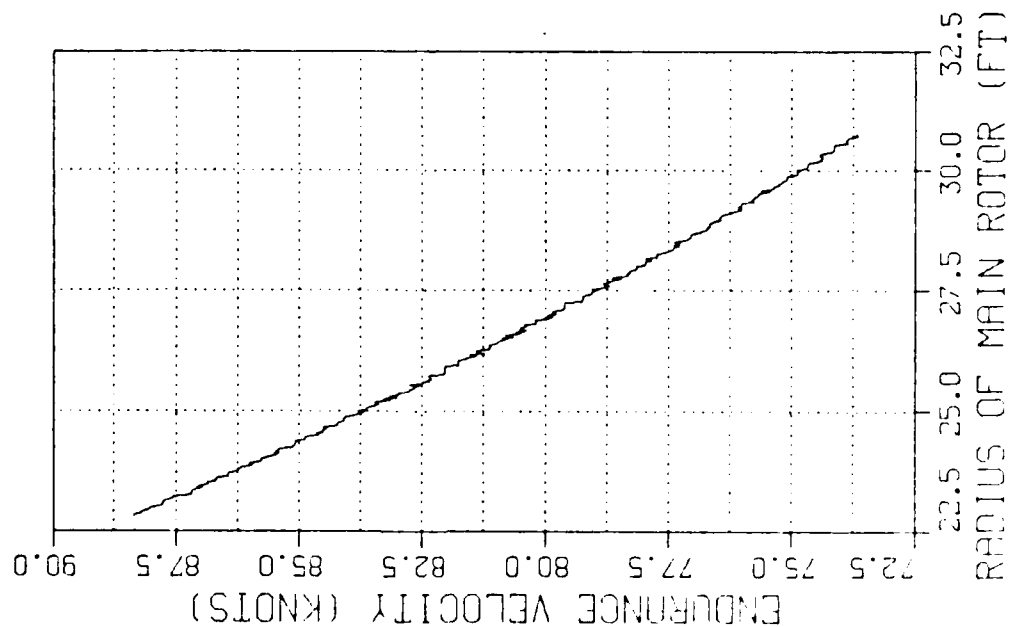


ENDURANCE VELOCITY VERSUS RADIUS

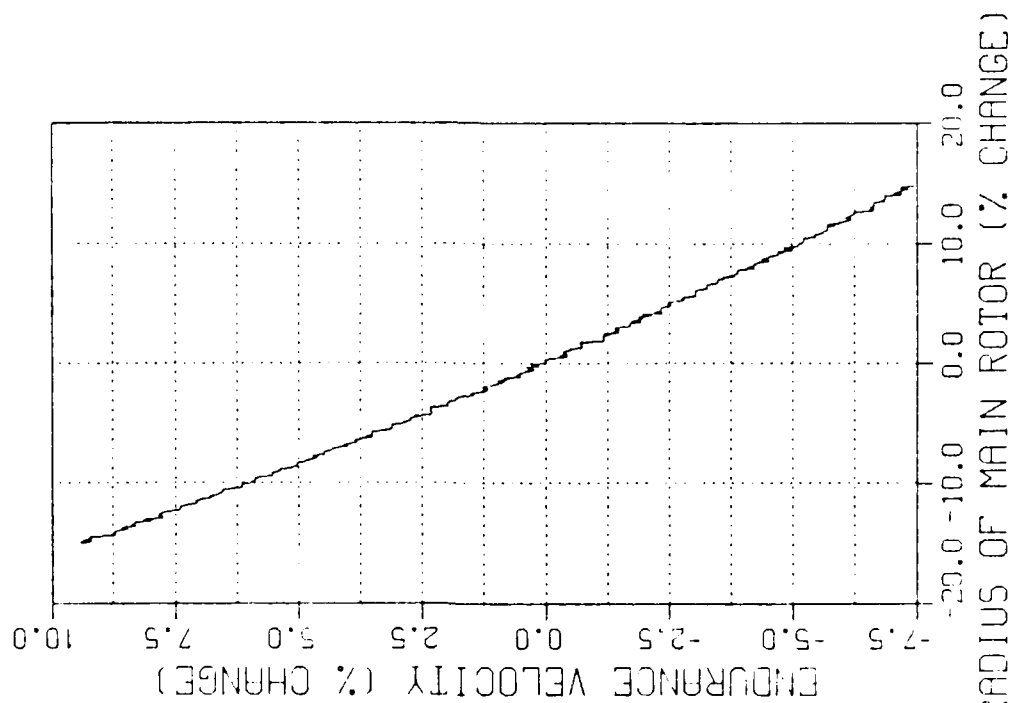
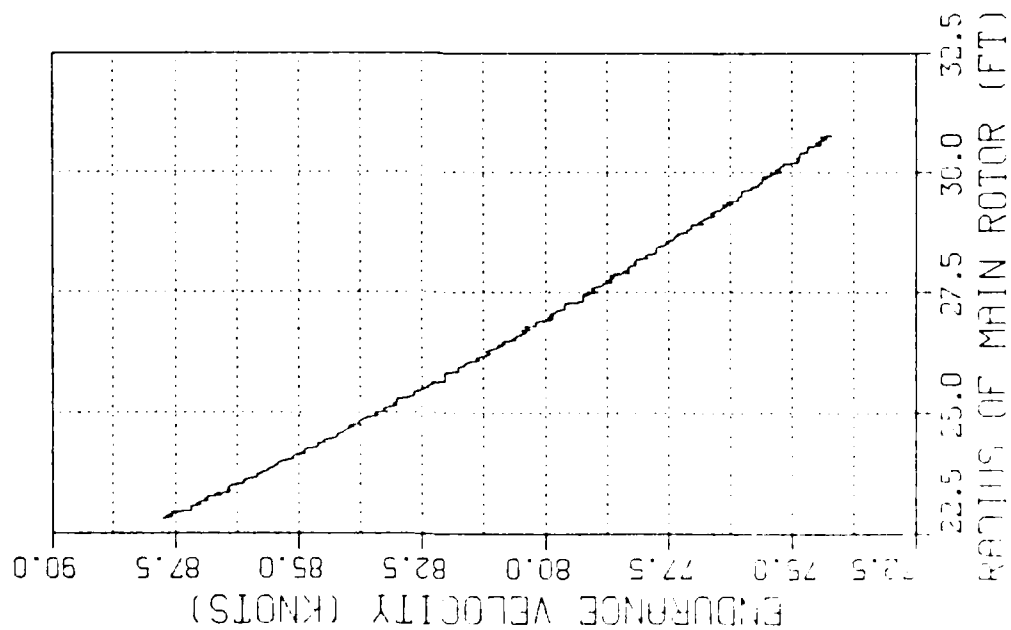
CHORD & ROTATIONAL VELOCITY HELD CONSTANT

SOLIDITY, TIP VELOCITY & ADVANCE RATIO

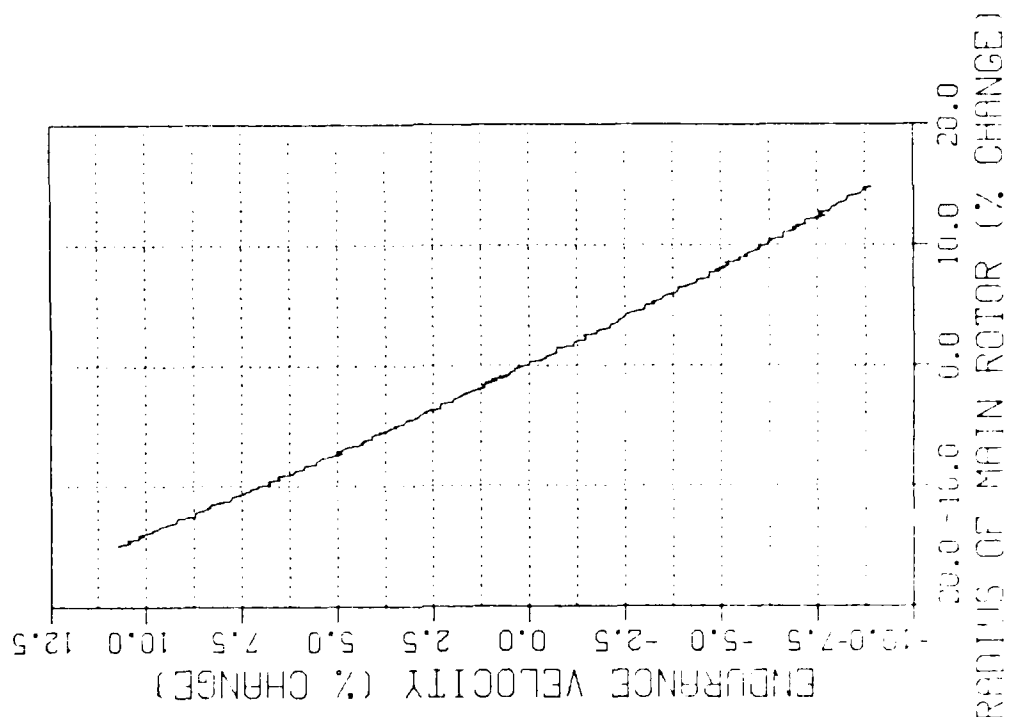
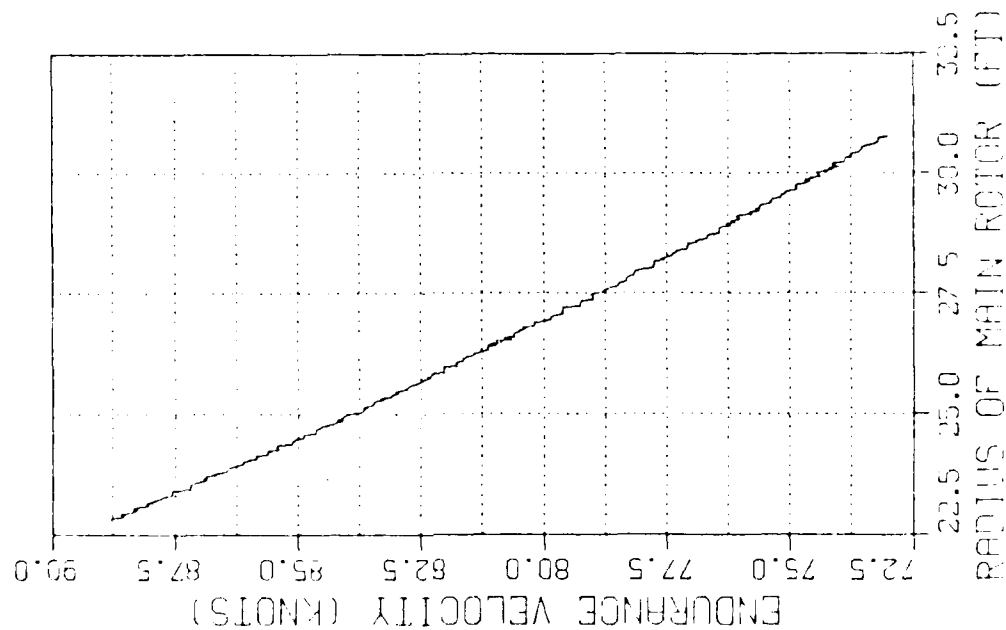
ALLOWED TO VARY WITH RADIUS



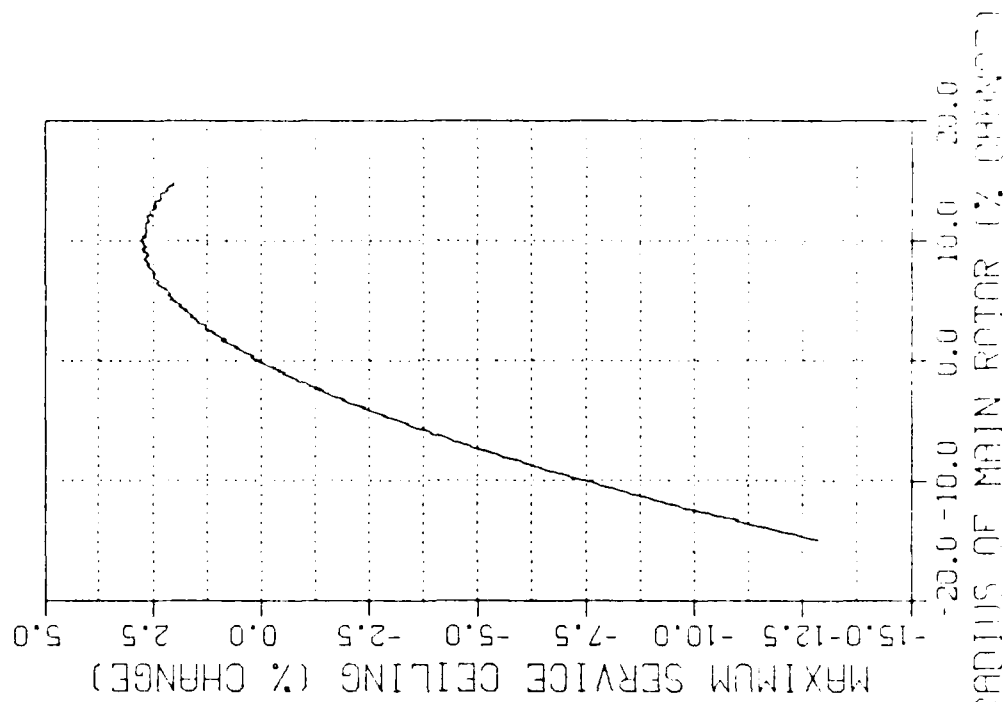
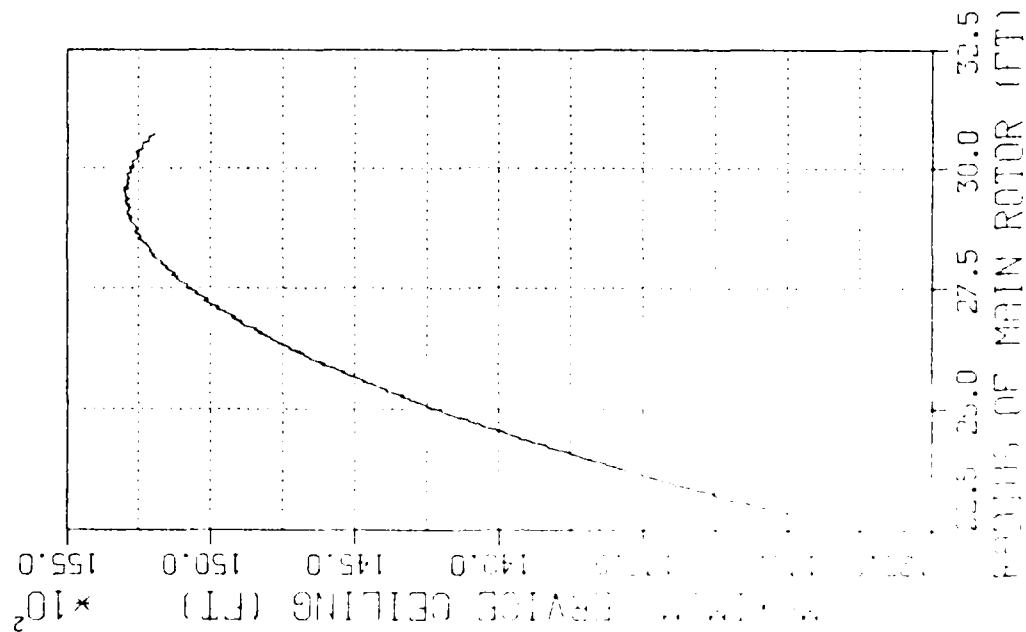
ENDURANCE VELOCITY VERSUS RADIUS CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT SOLIDITY, ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



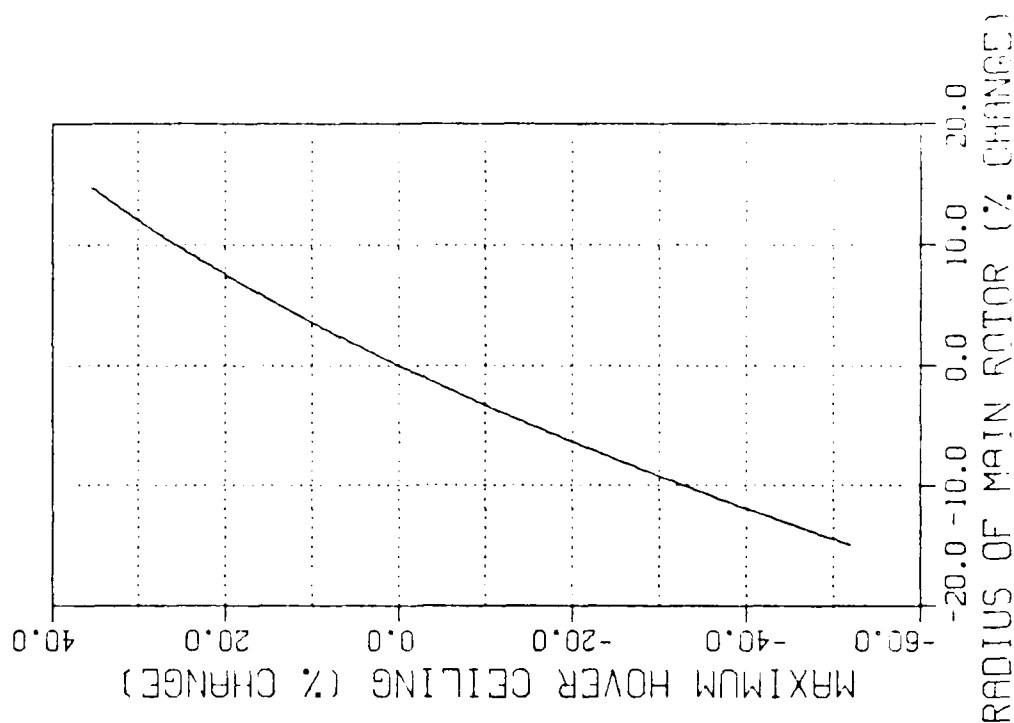
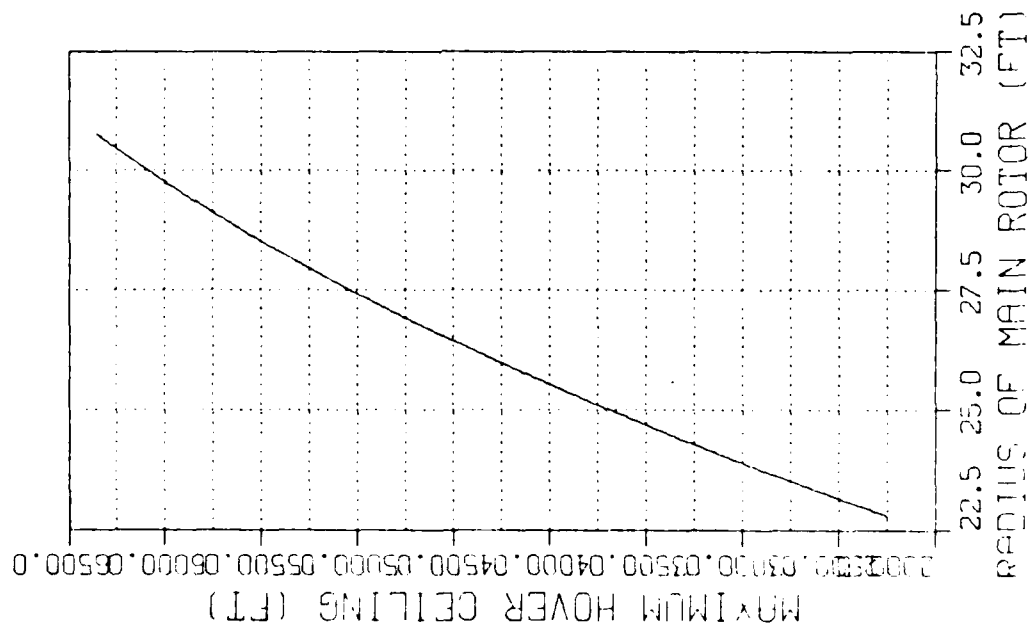
ENDURANCE VELOCITY VERSUS RADIUS SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT CHORD, ADVANCE RATIO & TIP VELOCITY ALLOWED TO VARY WITH RADIUS



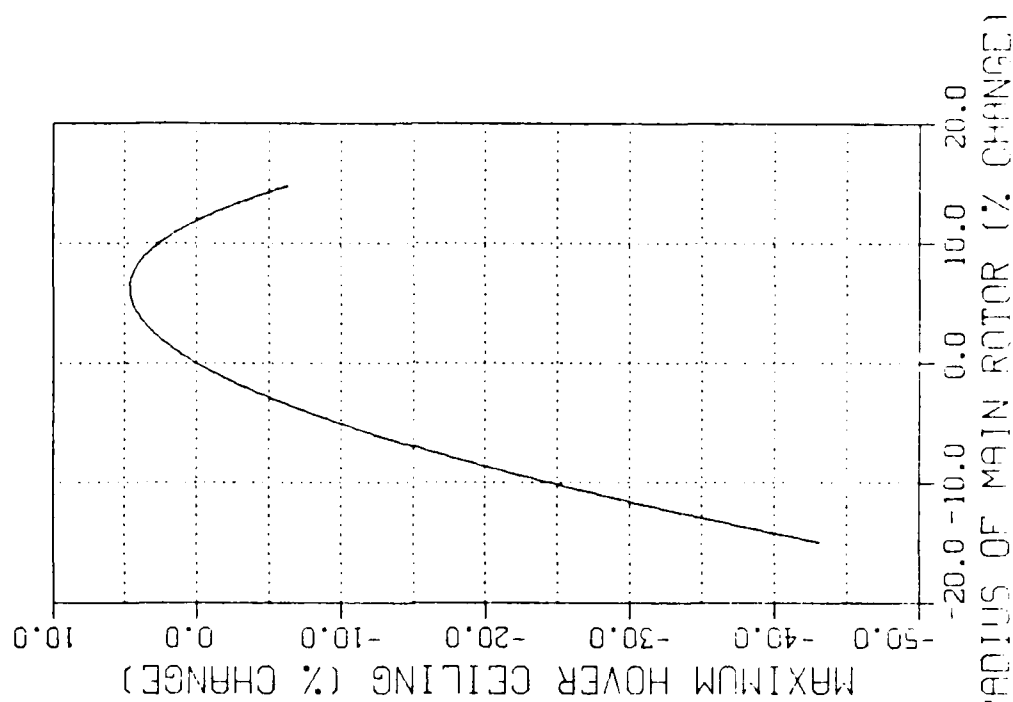
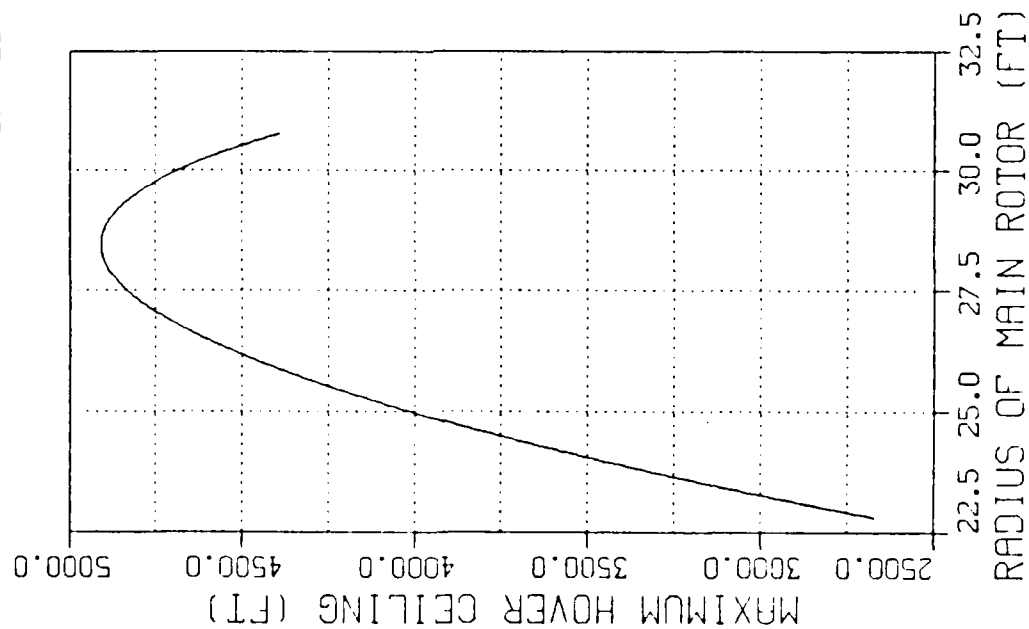
MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE CHORD & ROTATIONAL VELOCITY HELD CONSTANT SOLIDITY, TIP VELOCITY & ADVANCE RATIO ALLOWED TO VARY WITH RADIUS



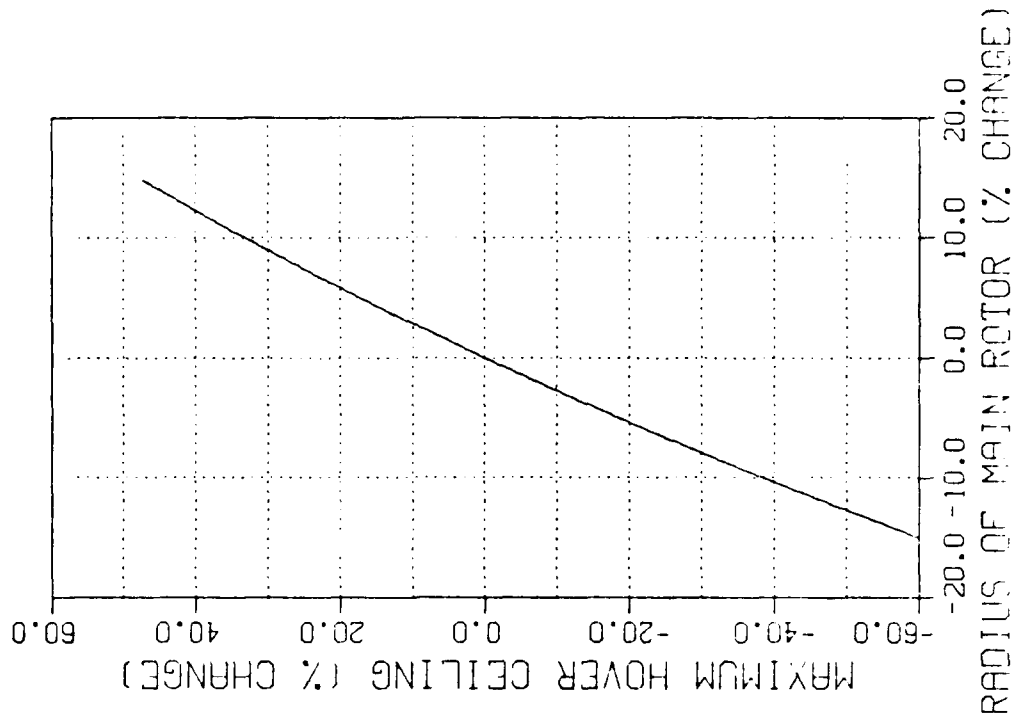
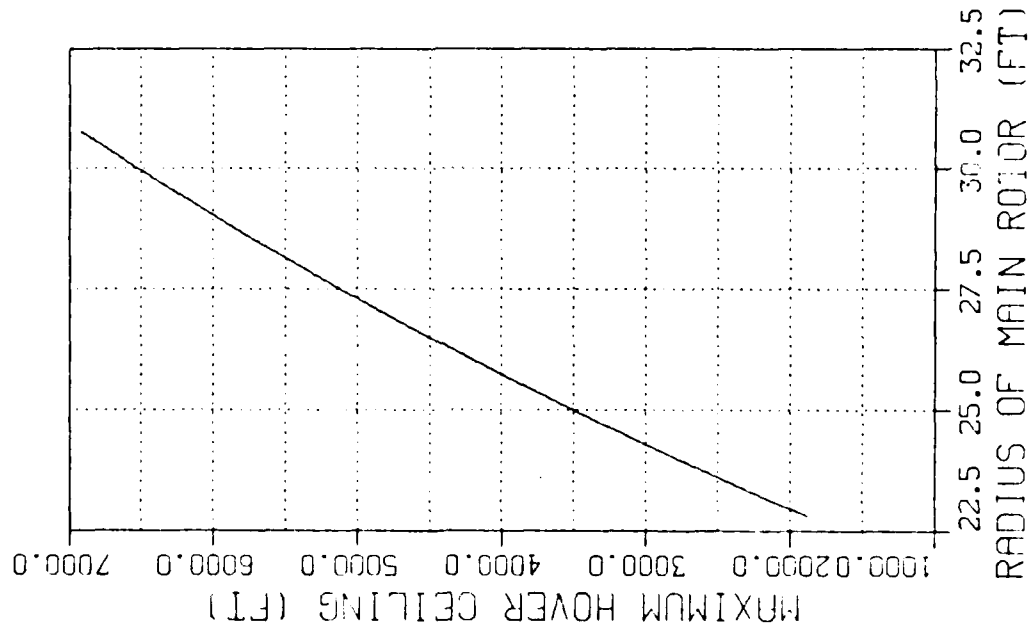
MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS
 SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT
 CHORD & ROTATIONAL VELOCITY
 ALLOWED TO VARY WITH RADIUS



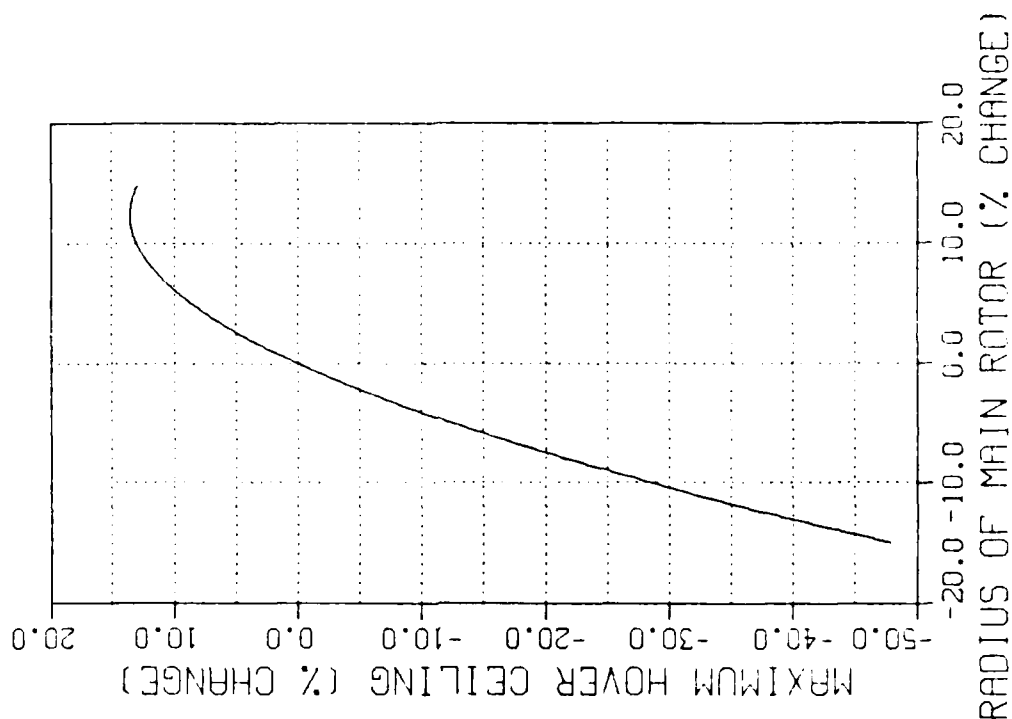
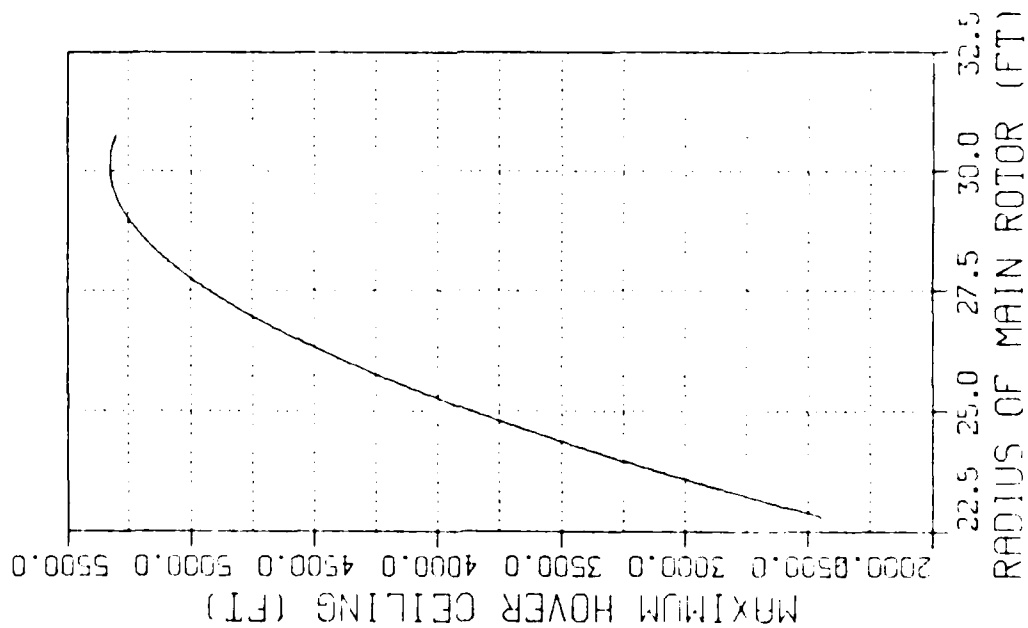
MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS
 SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
 CHORD, ADVANCE RATIO & TIP VELOCITY
 ALLOWED TO VARY WITH RADIUS



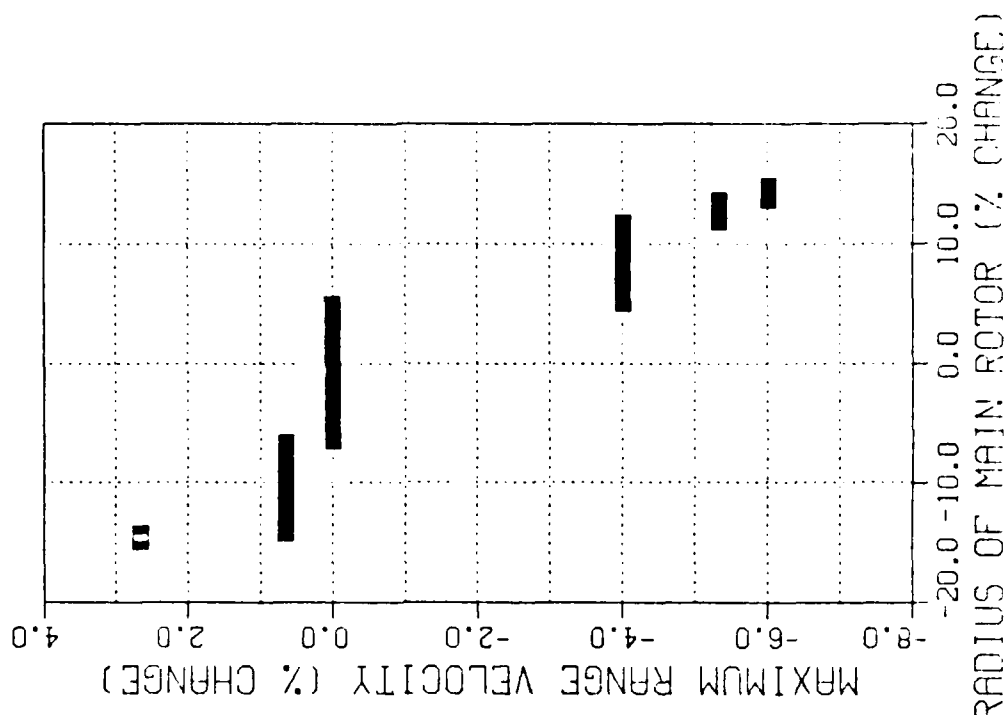
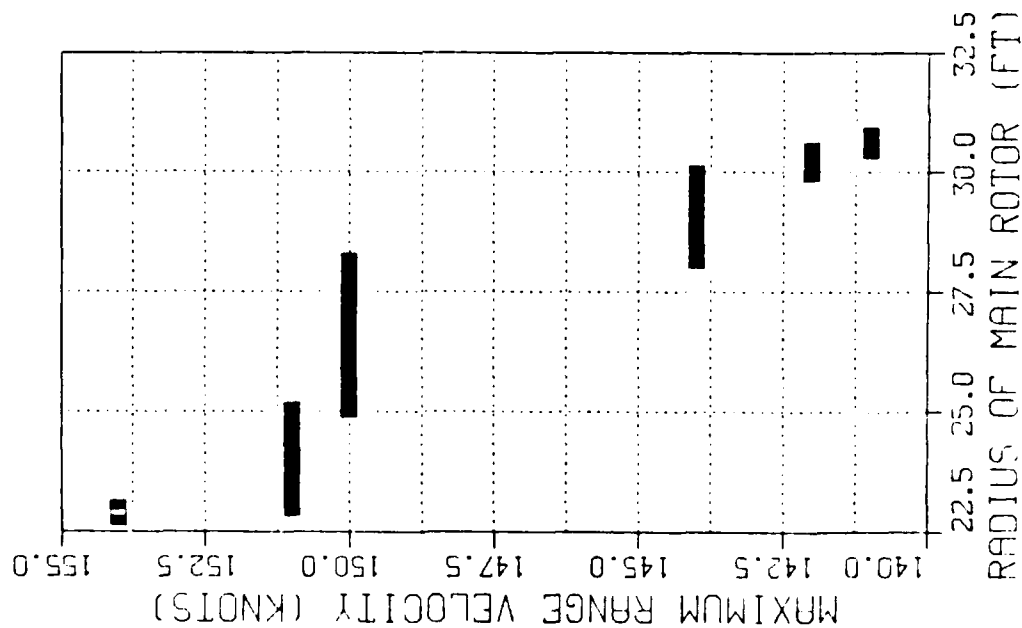
MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS
 CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT
 SOLIDITY, ROTATIONAL VELOCITY
 ALLOWED TO VARY WITH RADIUS



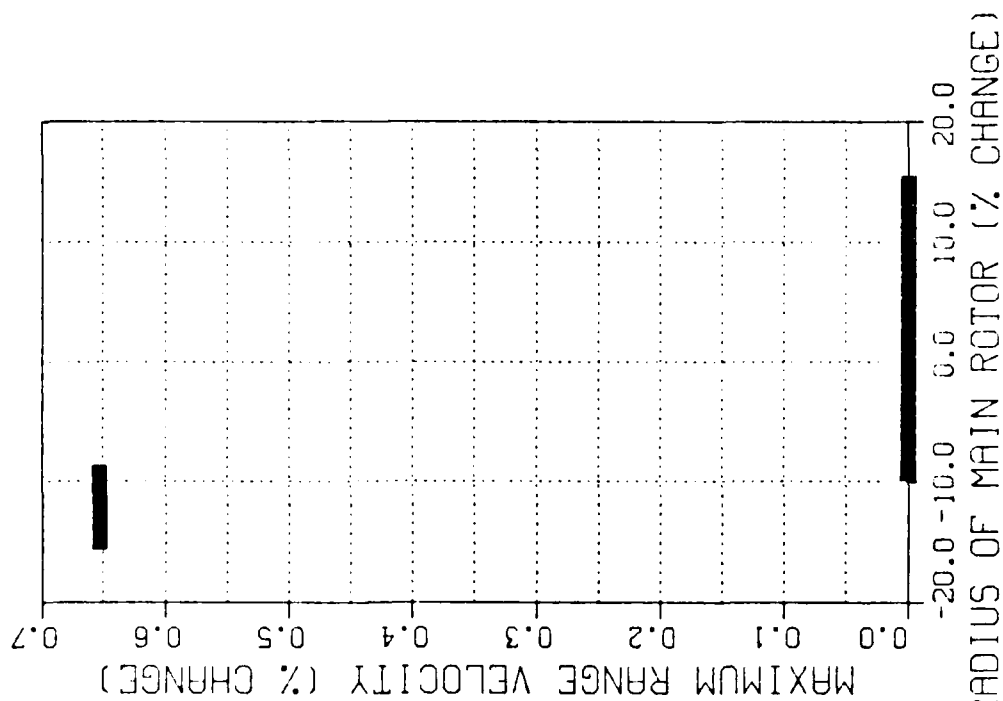
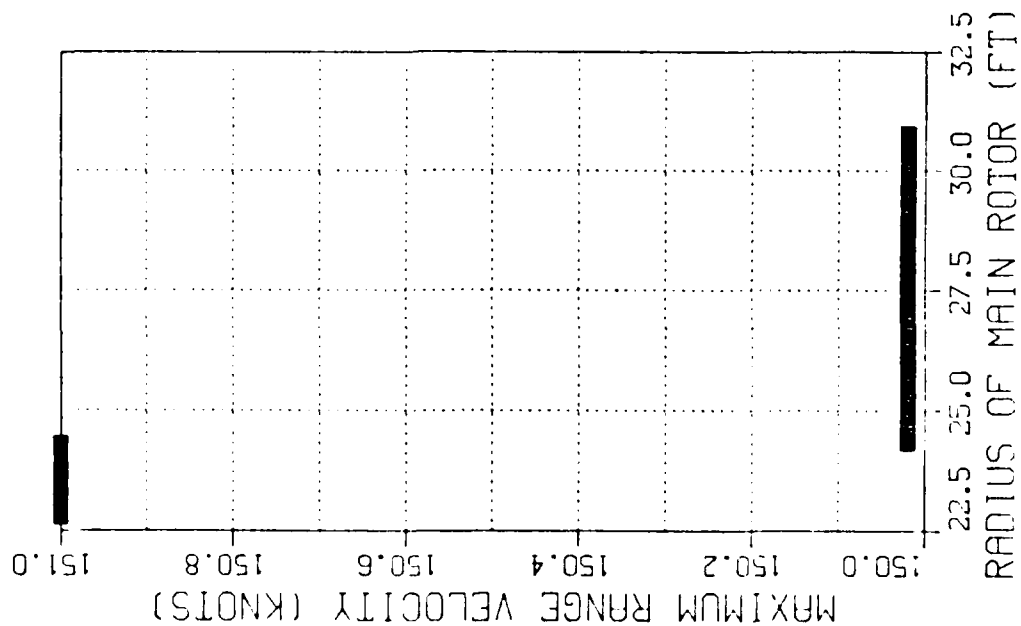
MAXIMUM HOVER CEILING (IGE) VERSUS RADIUS
 CHORD & ROTATIONAL VELOCITY HELD CONSTANT
 SOLIDITY, TIP VELOCITY & ADVANCE RATIO
 ALLOWED TO VARY WITH RADIUS



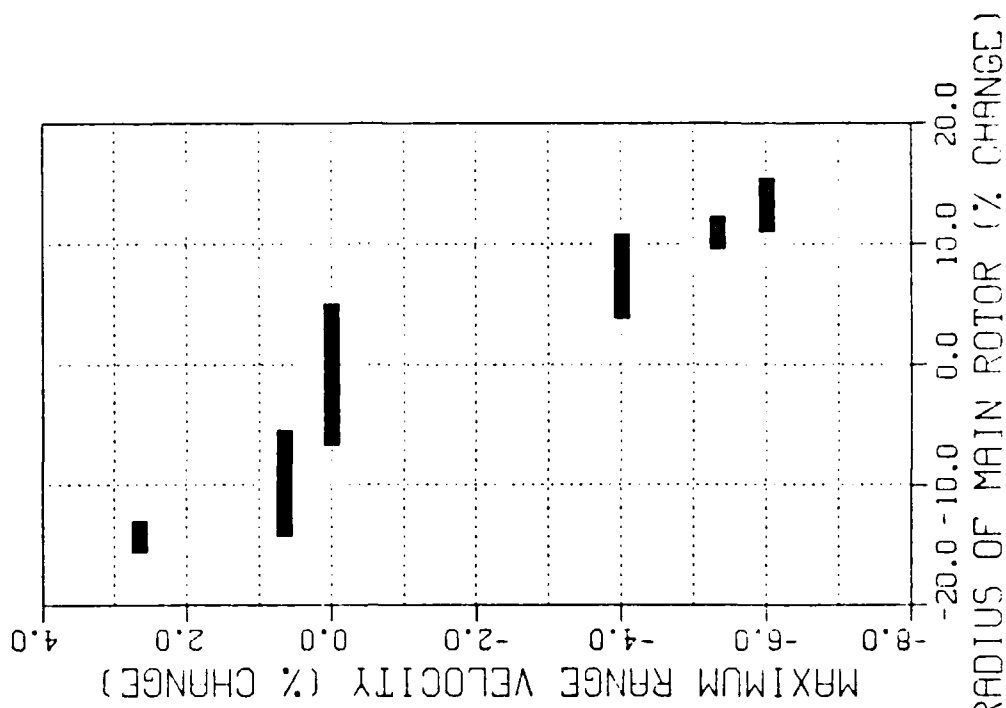
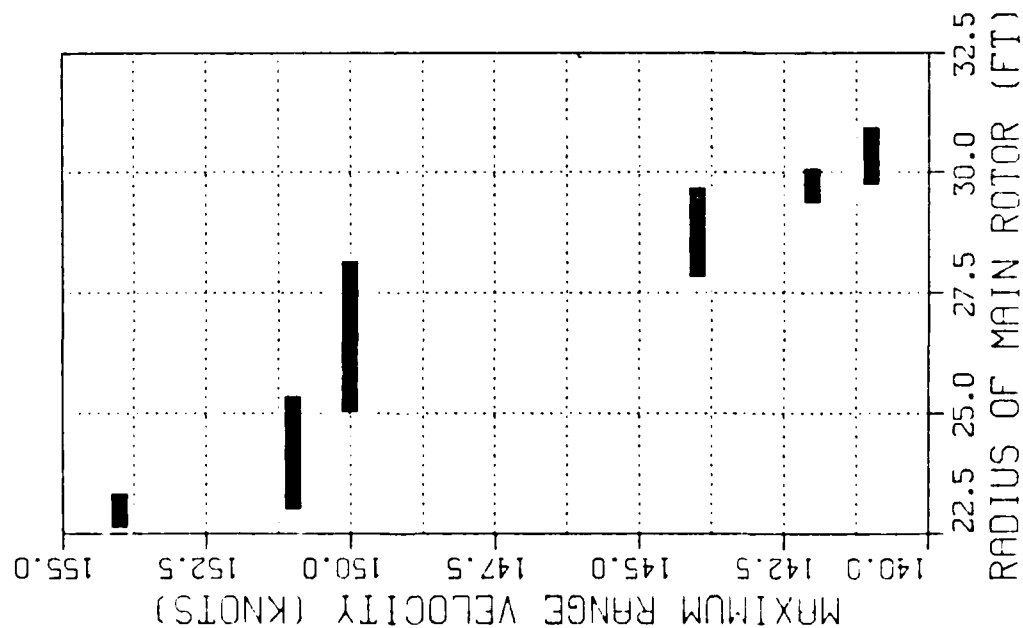
MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT CHORD & ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



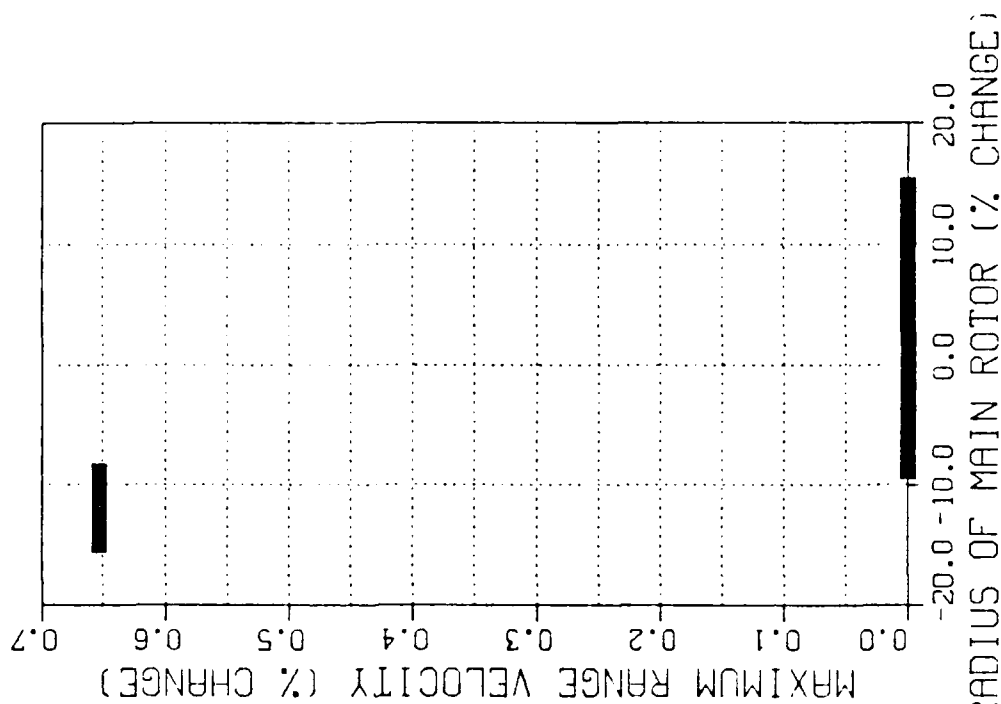
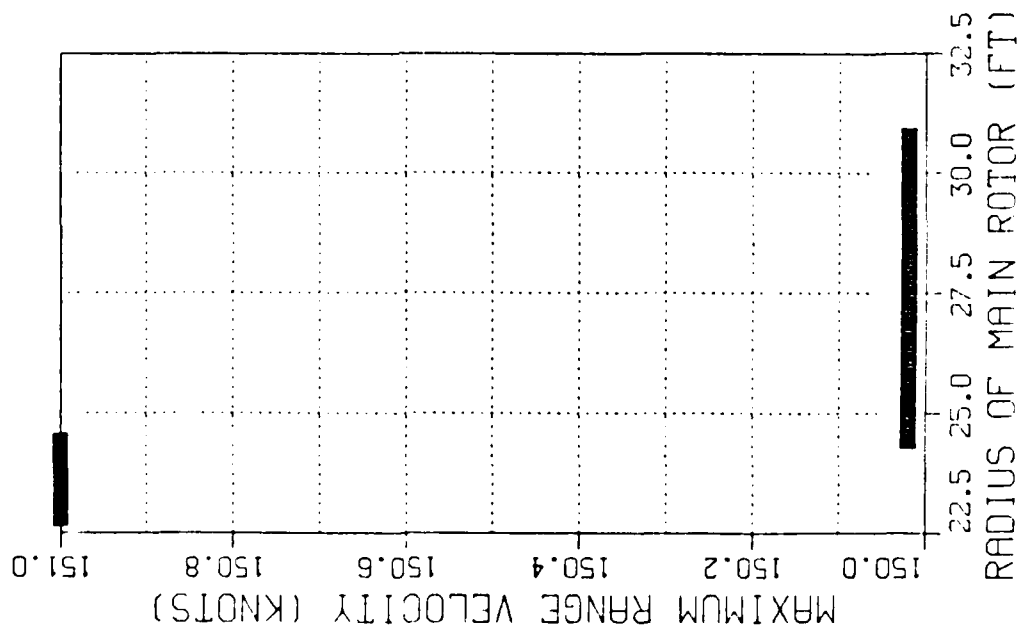
MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT CHORD, ADVANCE RATIO & TIP VELOCITY ALLOWED TO VARY WITH RADIUS



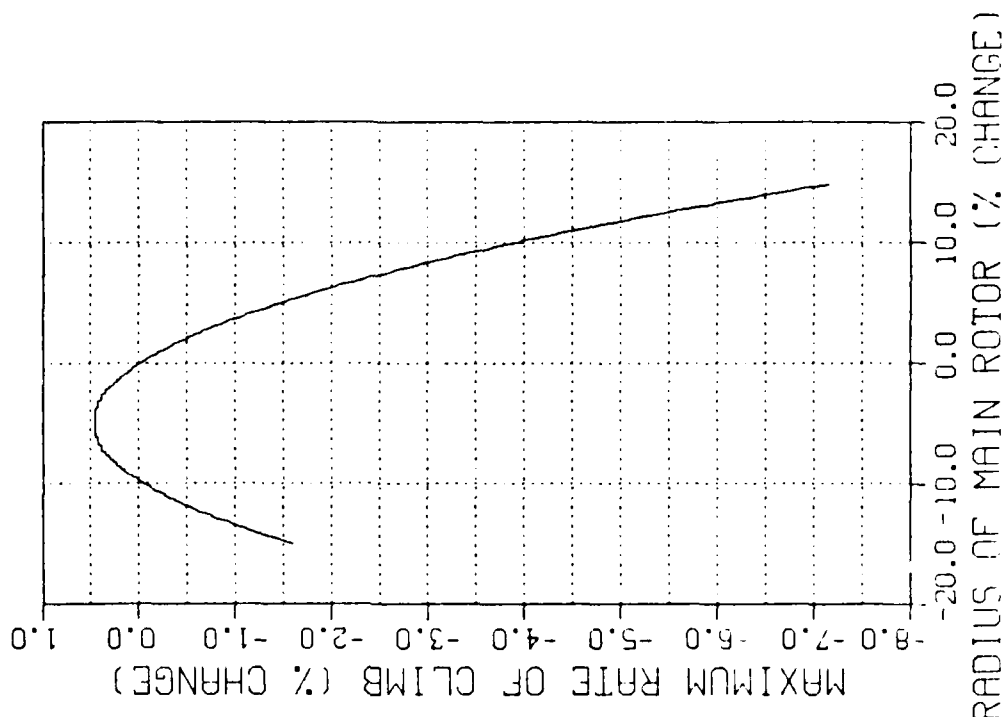
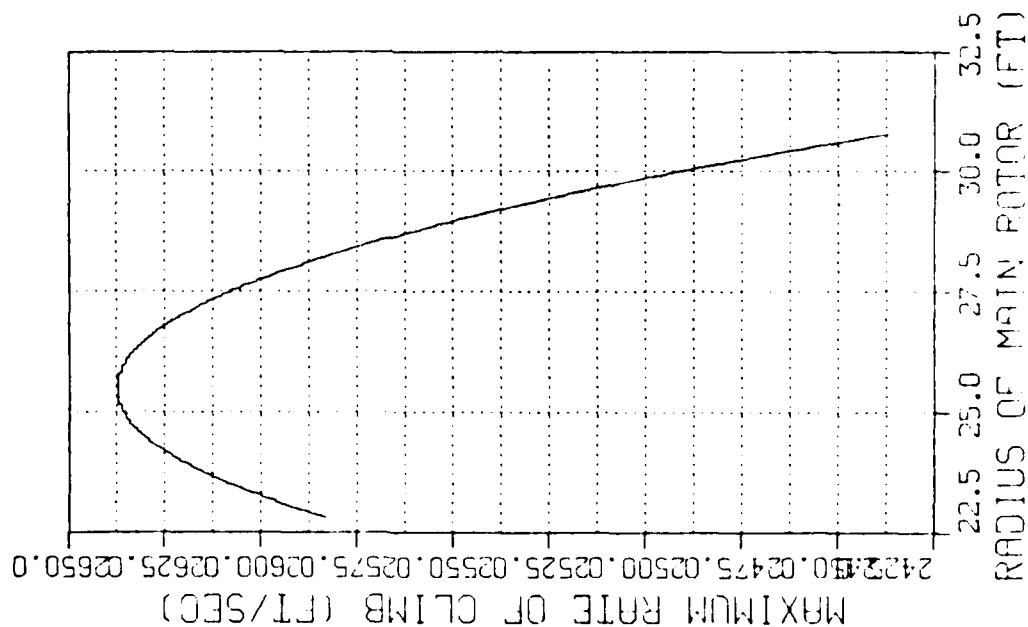
MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT SOLIDITY, ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



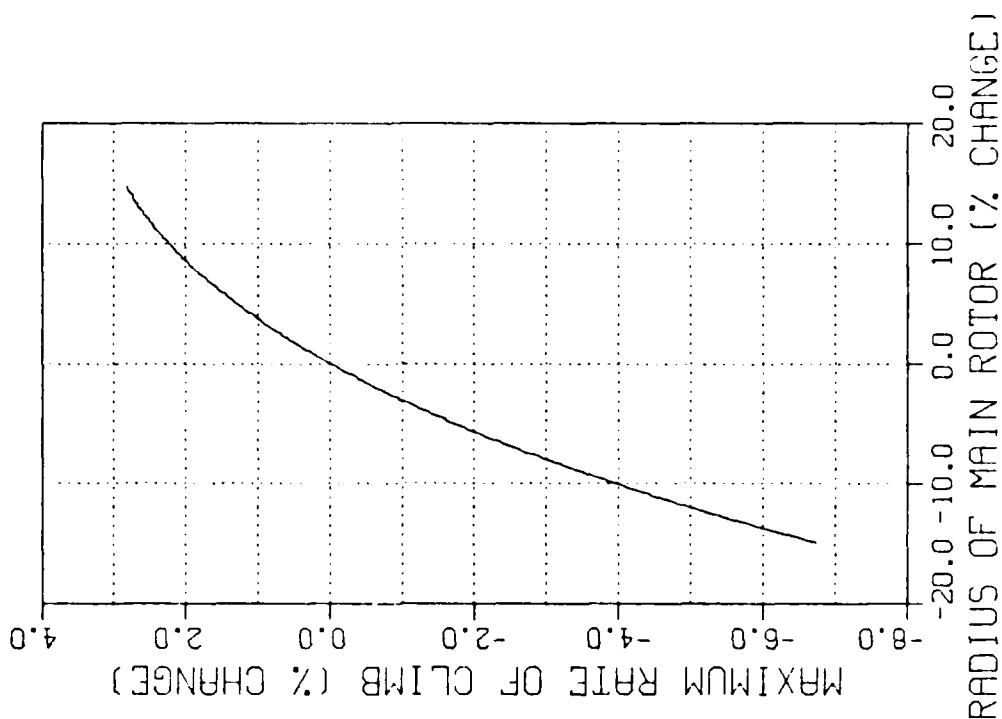
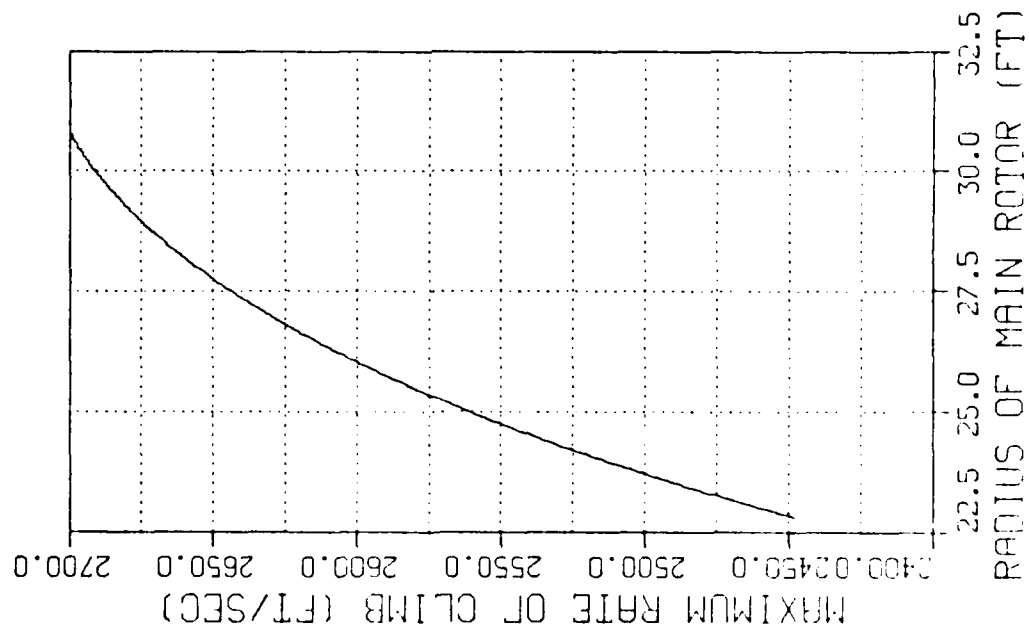
MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE CHORD & ROTATIONAL VELOCITY HELD CONSTANT SOLIDITY, TIP VELOCITY & ADVANCE RATIO ALLOWED TO VARY WITH RADIUS



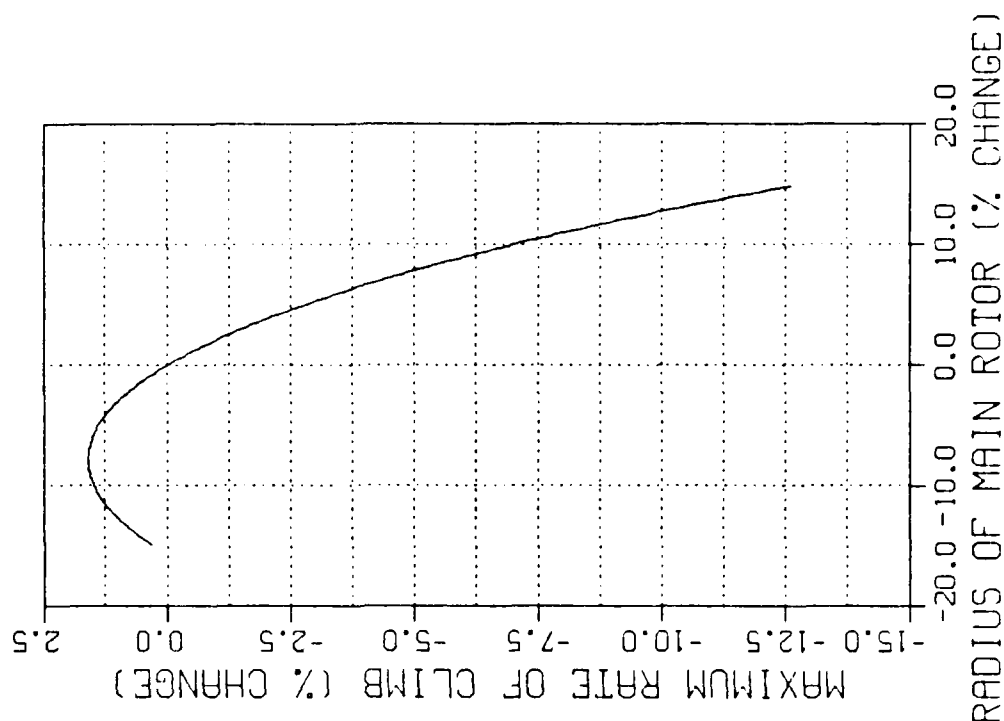
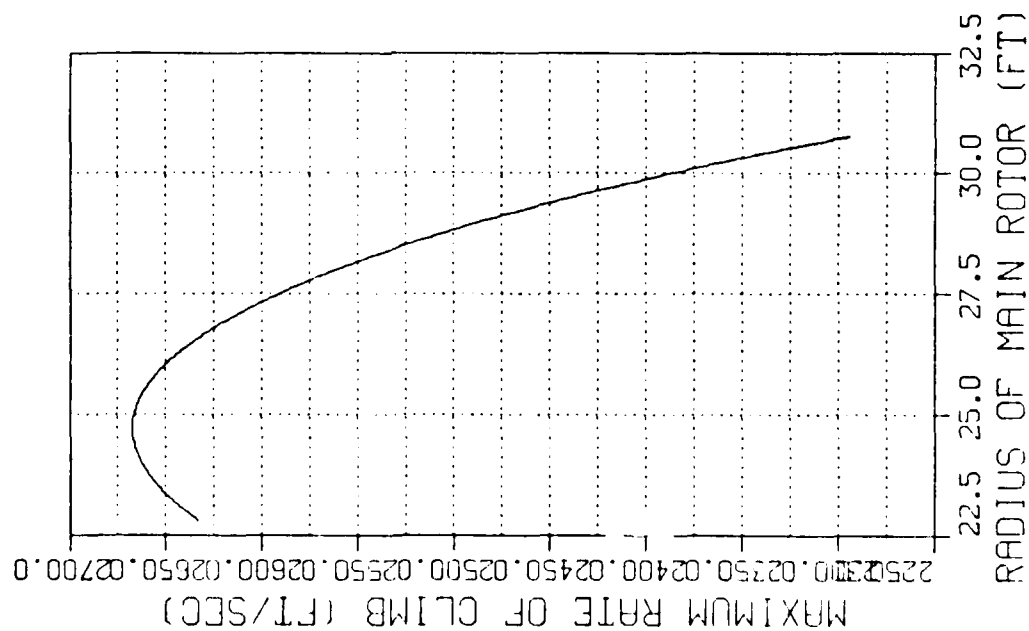
MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE CHORD & ROTATIONAL VELOCITY HELD CONSTANT SOLIDITY, TIP VELOCITY & ADVANCE RATIO ALLOWED TO VARY WITH RADIUS



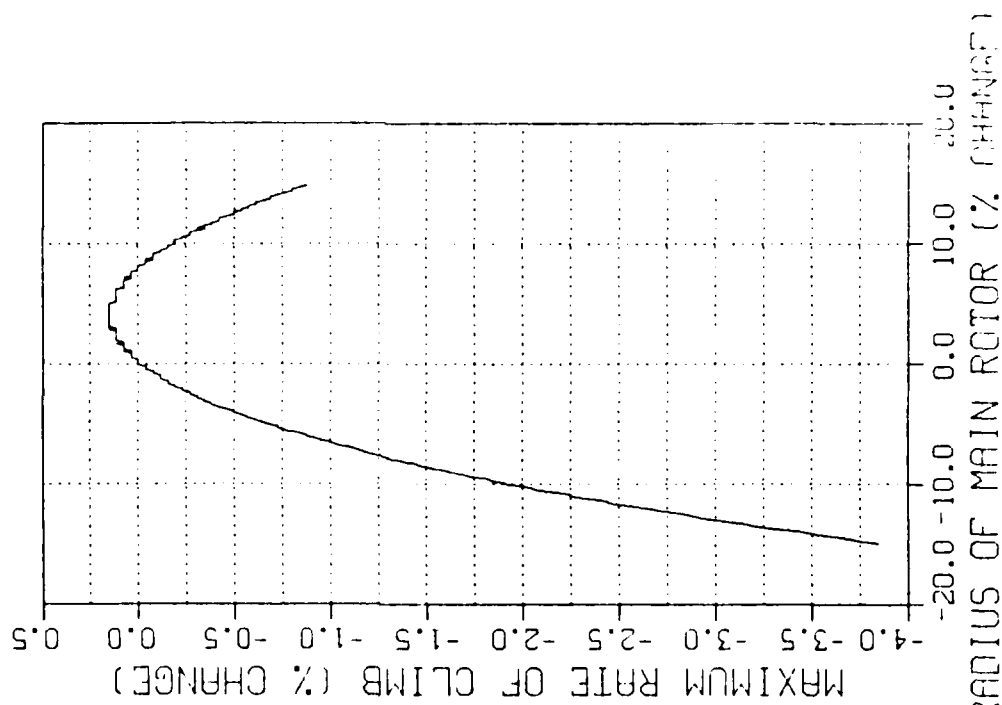
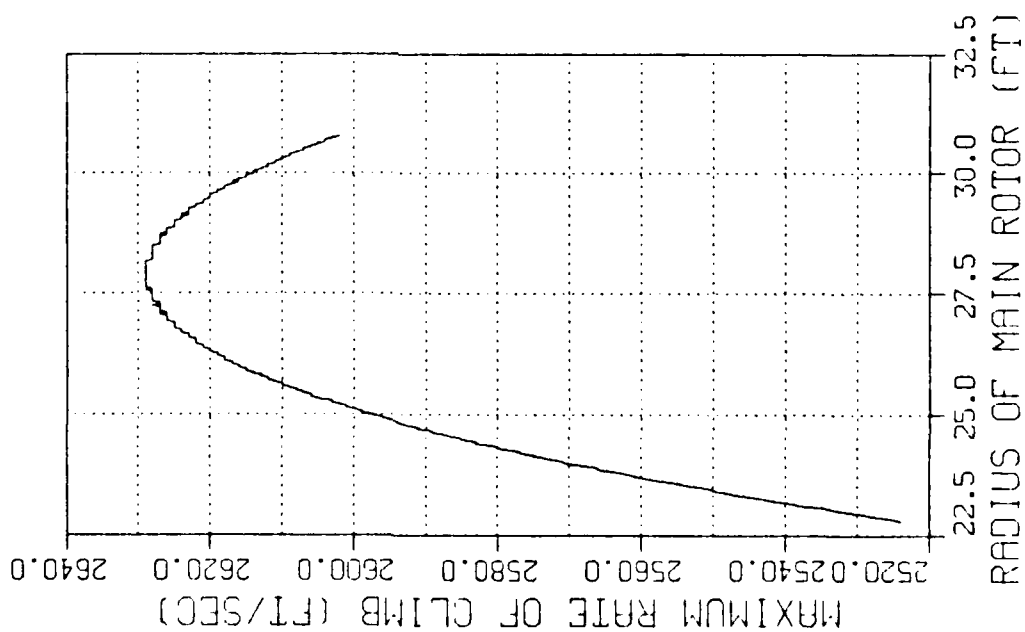
MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT SOLIDITY, ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



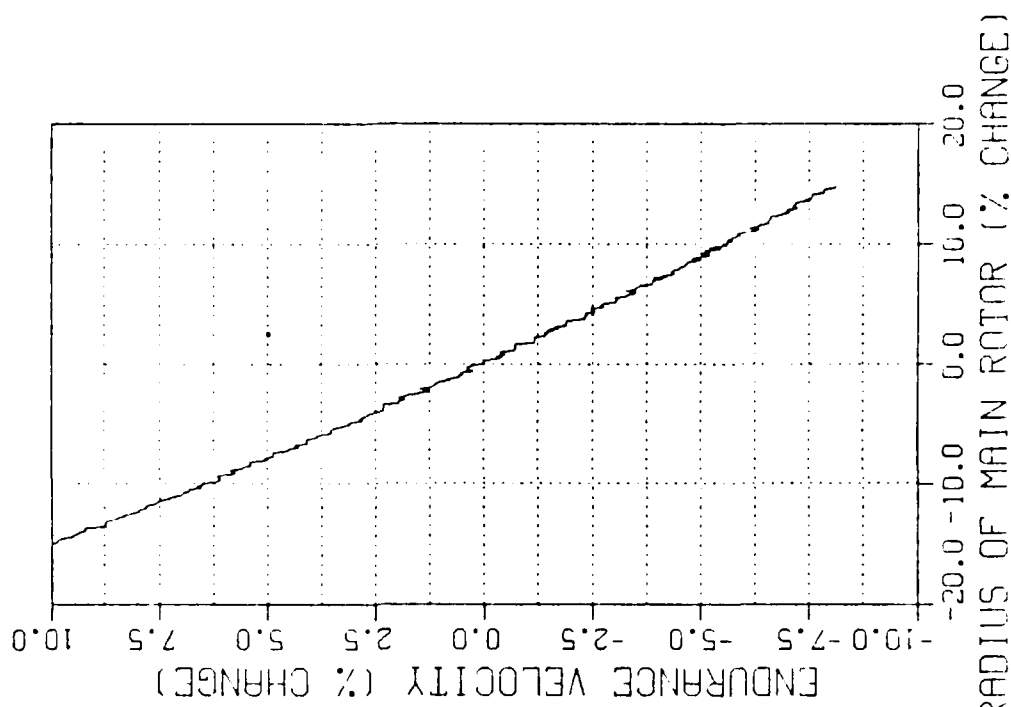
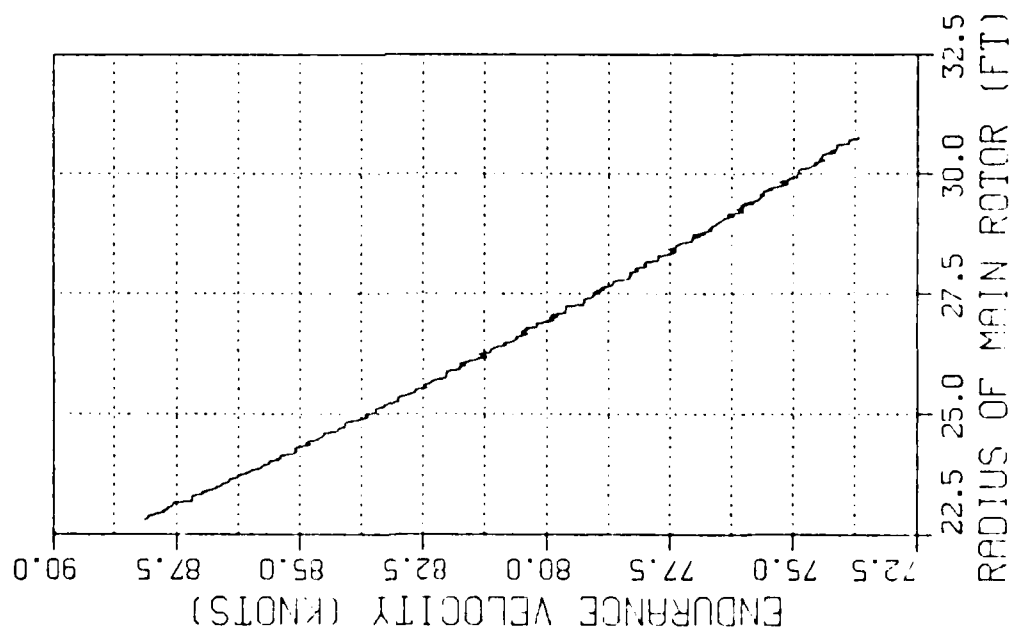
MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE
 SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT
 CHORD, ADVANCE RATIO & TIP VELOCITY
 ALLOWED TO VARY WITH RADIUS



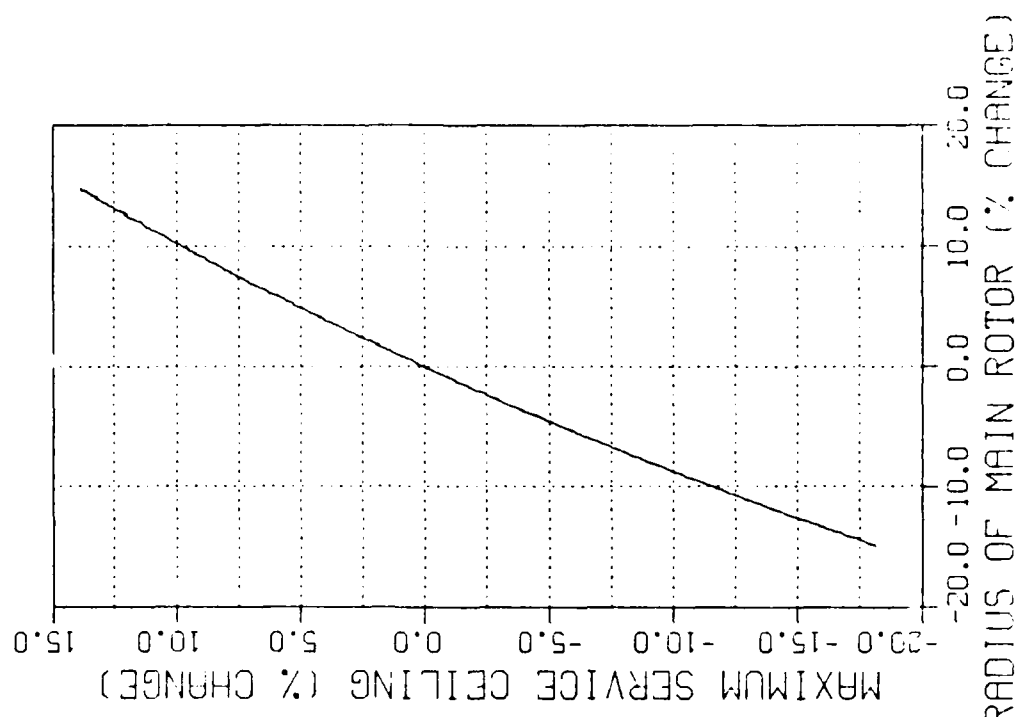
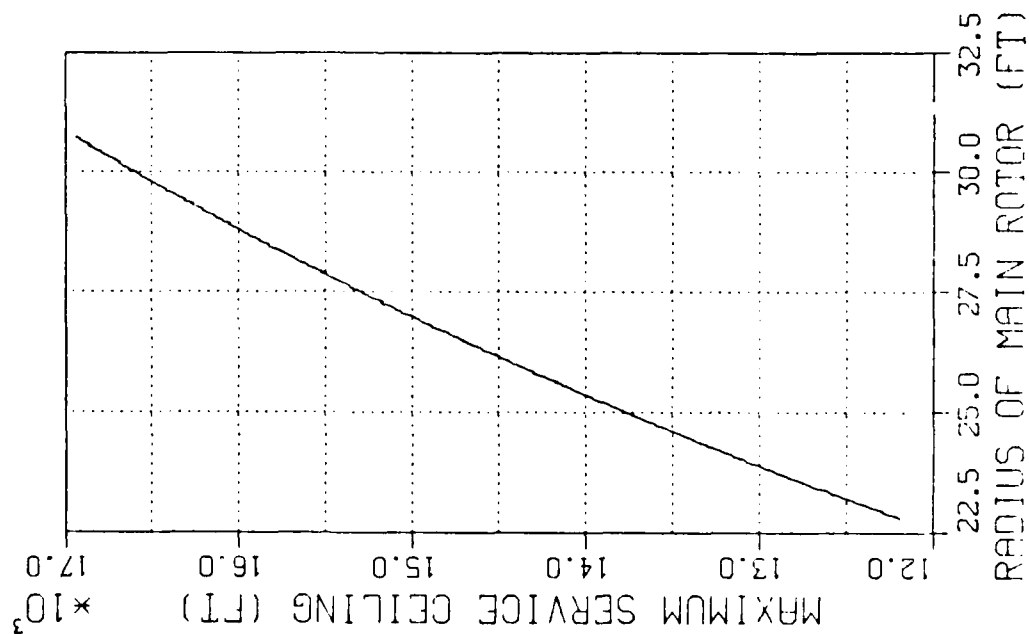
MAXIMUM RATE OF CLIMB VERSUS RADIUS CHANGE SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT CHORD & ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



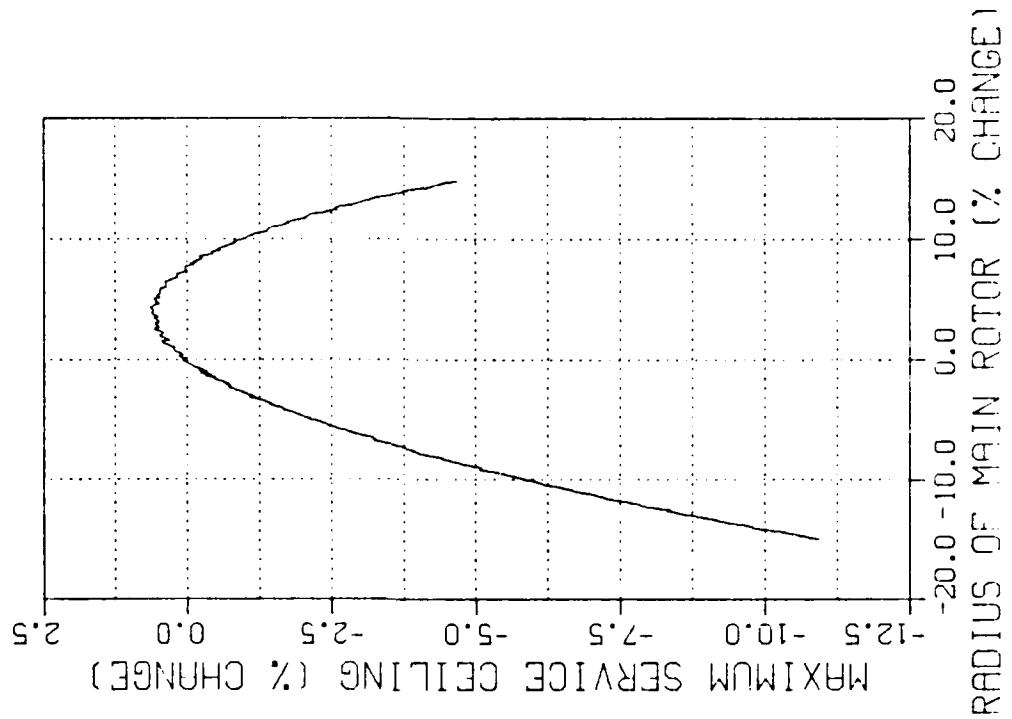
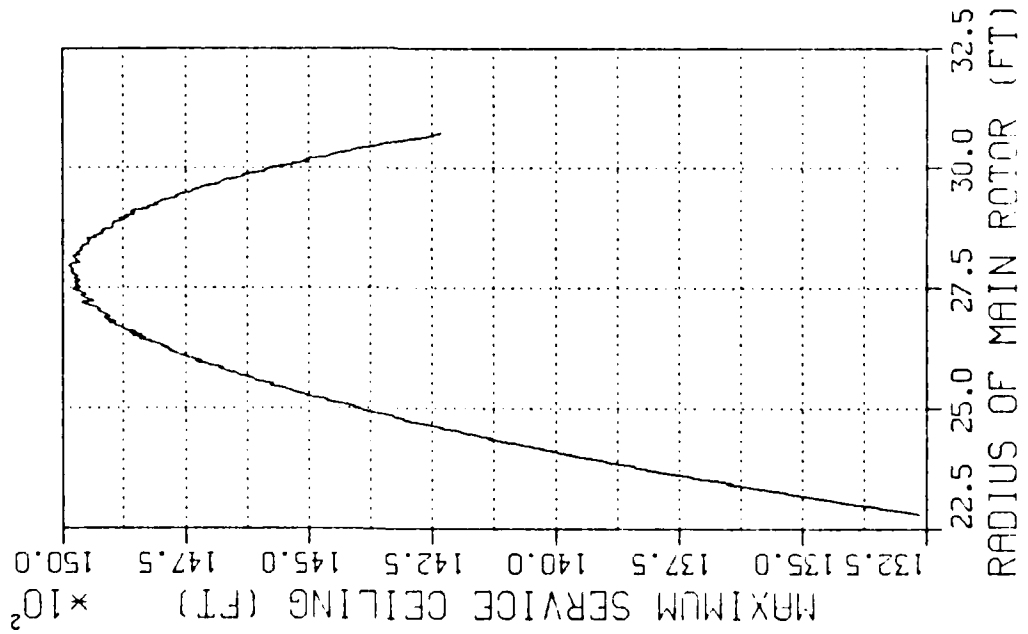
ENDURANCE VELOCITY VERSUS RADIUS SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT CHORD & ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



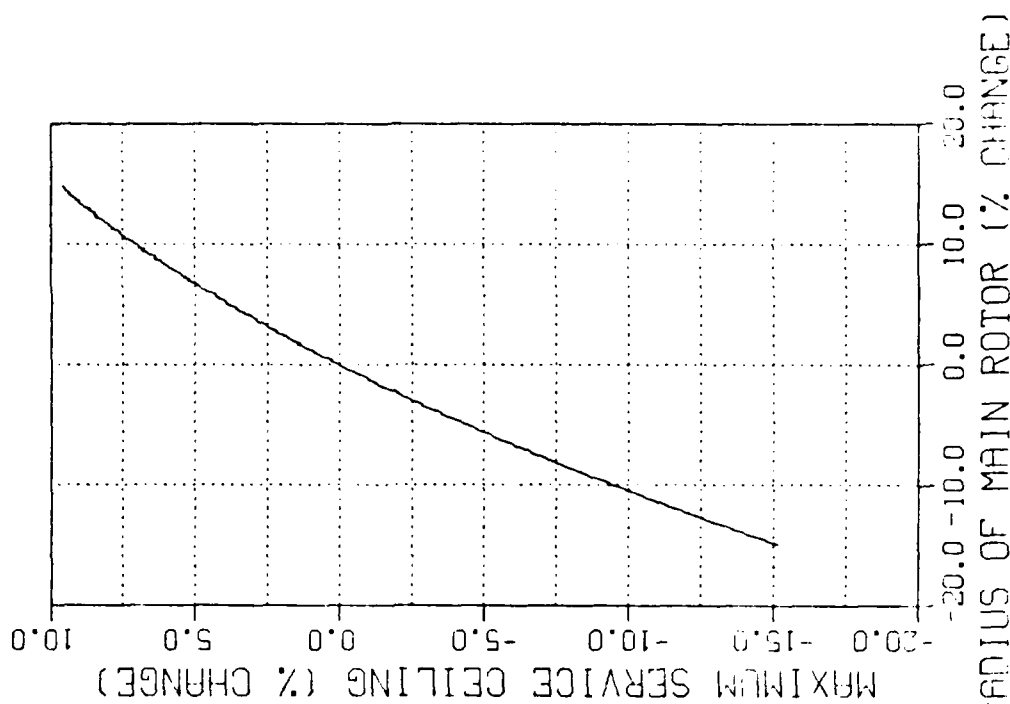
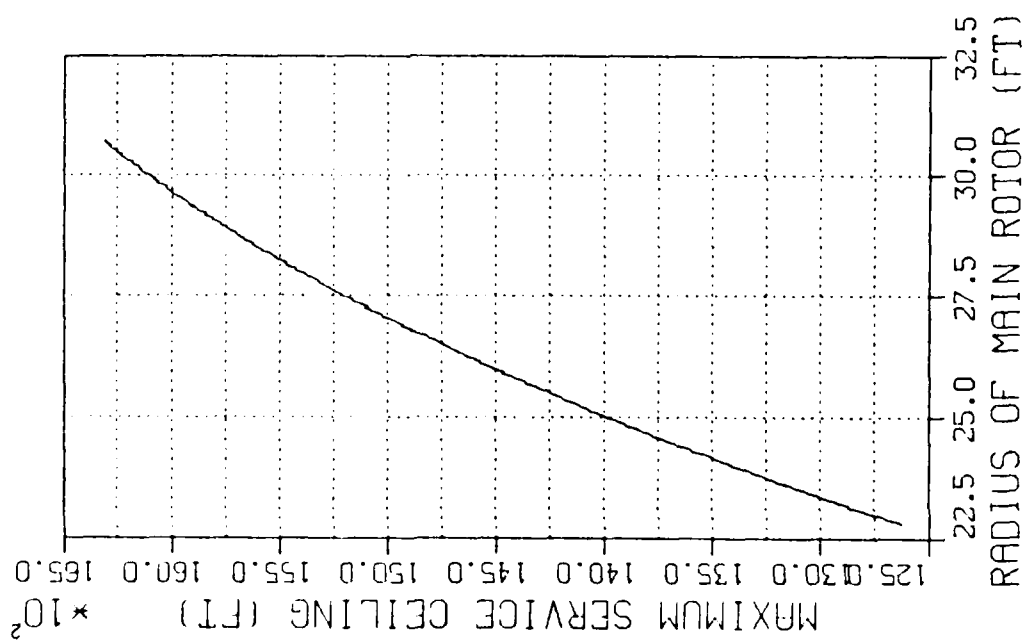
MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE CHORD, TIP VELOCITY & ADVANCE RATIO HELD CONSTANT SOLIDITY, ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT CHORD, ADVANCE RATIO & TIP VELOCITY ALLOWED TO VARY WITH RADIUS



MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT CHORD & ROTATIONAL VELOCITY ALLOWED TO VARY WITH RADIUS



PEROCC10
PEROCC20
PEROCC30
PEROCC40
PEROCC50
PEROCC60
PEROCC70
PEROCC80
PEROCC90
PEROCC100
PEROCC110

APPENDIX B

COMPUTER PROGRAM LISTING

PEROCC170
PEROCC180
PEROCC190
PEROCC200
PEROCC210
PEROCC220
PEROCC230
PEROCC240
PEROCC250
PEROCC260
PEROCC270

PERFORMANCE DESIGN PROGRAM COMPUTER PROGRAM LISTING

THIS PROGRAM WILL TAKE A SET OF HELICOPTER DESIGN DATA, RANGE
CALCULATE POWER REQUIRED, ENDURANCE VELOCITY, RATE OF CLIMB, RANGE
VELOCITY, POWER CEILING AND SERVICE CEILING FOR THAT DATA. IT
ALSO WILL CALCULATE PERFORMANCE FACTORS LISTED ABOVE OVER A
RANGE OF VALUES FOR THE GIVEN DATA. IT THEN WILL PLCT THE RESULTS
VERSUS MAIN ROTOR RADIUS IN TWO FORMS. FORM 1 BEING THE PERFORMANCE
FACTOR AT THE GIVEN SPECIFICATION VALUE VERSUS THE PERCENT
DIFFERENCE OF THE RADIUS TO THE SPECIFICATION RADIUS.

TC RUN THE PROGRAM FLLCW THESE STEPS:

1. INPLT THE REQUIRED DATA IN LINES 172-207
2. COMPILE PROGRAM USING: FORTFX PERFORM (NOMAP
NOTE: IF YOU DO NOT USE (NOMAP, YOU WILL NOT HAVE ENOUGH DISK
SPACE TO RUN THE PROGRAM
3. GRAFF PRGGRAM USING: DISSPLA PERFORM
4. EXAMINE RESULTS BY USING: DISSPCF

***** DEFINITION OF TERMS *****

AHAT - ZERC HORSEPOWER FUEL FLOW INTERCEPT
AMR - DISK AREA OF THE MAIN ROTOR (FT*2)
ARMR - ADVANCE RATIO OF THE MAIN ROTOR SYSTEM
AKTR - ADVANCE RATIO OF THE TAIL ROTOR SYSTEM
ATR - DISK AREA OF THE TAIL ROTOR (FT*2)
EHAT - AVERAGE SLOPE OF THE FUEL FLOW VS HORSEPOWER
EUCMR - AVERAGE COEFFICIENT OF DRAG OF THE MAIN ROTOR BLADES
CDCTR - AVERAGE COEFFICIENT OF DRAG OF THE TAIL ROTOR
CEIL - SUBROUTINE TO CALCULATE MAX SERVICE CEILING
CMA, CMRI - CHORD LENGTH OF MAIN ROTOR (FT)
CONST - SUBROUTINE TO CALCULATE COMMON USED CONSTANTS
CTMR - COEFFICIENT OF THRUST OF THE MAIN ROTOR SYSTEM
CTTR - COEFFICIENT OF THRUST OF THE TAIL ROTOR SYSTEM
C... - COEFFICIENT INDICATE A DIFFERENCE IN A VARIABLE, SEE VARIABLE
DA - DENSITY ALTITUDE (FT)
DELTA - PRESSURE RATIO (ALTITUDE/SSL)


```

EFPAF = 25.7
EFPAV = 31.8
RTHT = 11.2
----- INPUT ENGINE DATA -----
IENG = 2
SFCM = C.46
SFCN = C.47
SFCC = C.51
SHPM = 1561.0
SHPN = 1218.0
SHPC = 589.0
----- INPUT FLIGHT VELOCITY FOR INITIAL POWER CALCULATIONS -----
VF1 = 50.0
===== END OF REQUIRED INPUT INFORMATION =====
DO 3000 K = 1,4
VV1 = C.C
PA = 0.0
TEMP = C.0
DA = 0.0
DENSSL = 0.00237694
SKDHT = 250C.0
VF = VF1
VV = VV1
CMR = CMR1
RVMR = FVMR1
PIE = 3.141593
SDMR3 = 18MR*CMR1 / (PIE*RADIUS)
VTMR2 = RVMR1*RADIUS
CALL RHC
CALL RAC (DIFF, RADIUS, RRMR, RMR, DRMR, MINZ, MAXZ, DMIN, DMAX)
IF (IANS.NE. 1) GO TO 2800
CALL PCWER
GO TO 3000
2800 IF (IANS.NE. 2) GO TO 2810
CALL ENCLR
GO TO 3000
2810 IF (IANS.NE. 3) GO TO 2820
CALL ENCLR
CALL MAXFC
GO TO 3000
2820 IF (IANS.NE. 4) GO TO 2830
CALL RANGE
GO TO 3000
2830 IF (IANS.NE. 5) GO TO 2840
CALL HCVER
GO TO 3000
2840 IF (IANS.NE. 6) GO TO 2850

```

```

PER01530
PER01540
PER01550
PER01560
PER01570
PER01580
PER01590
PER02000
PER02010
PER02020
PER02030
PER02040
PER02050
PER02060
PER02070
PER02080
PER02090
PER02100
PER02110
PER02120
PER02130
PER02140
PER02150
PER02160
PER02170
PER02180
PER02190
PER02200
PER02210
PER02220
PER02230
PER02240
PER02250
PER02260
PER02270
PER02280
PER02290
PER02300
PER02310
PER02320
PER02330
PER02340
PER02350
PER02360
PER02370
PER02380
PER02390
PER02400

```

```

2850 CALL ENCLUR
3000 CALL CEIL
5000 GO TO 3000
      GO TO 5
      CONTINUE
      CALL DCNEPL
      CONTINUE
      STCP
      ENC
C*****
C      SUBROUTINES USED IN PRGRAM FULLCW
C*****
C      SUBROUTINE RAD(DIFF,RADIUS,RRMR,RMR,DRMR,MINZ,MAXZ,DMIN,DMAX)
C*****
C      IMPLICIT REAL (A-H,L-Z)
C      DIMENSION RMR(400),DRMR(400)
C      RMR = RADILS - DIFF
C      DO 10 I = 1,324
C        RMR(I) = RRMR
C        IF (I.EQ.163) RMR(I) = RADIUS
C        RRMR = RRMR + CIFF/162
C      CONTINUE
C      DO 20 I = 1,324
C        DRMF(I) = ((RMR(I) - RMR(163)) / RMR(163)) * 100.0
C      CONTINUE
C      DMIN = DRMR(1)
C      DMAX = DRMR(324)
C      MINZ = RMR(1)
C      MAXZ = RMR(324)
C      RETURN
C      ENC
C*****
C      SUB-GROUP B
C      PERFORMANCE SUBROUTINES
C*****
C      SUBROUTINE TO CALCULATE ALL PARTS OF THE TOTAL POWER REQUIREMENTS
C      AND FIGURE OF MERIT.
C*****
C      SUBROUTINE POWER
C      IMPLICIT REAL (A-H,L-Z)

```

```

PERO2410
PERO2420
PERO2430
PERO2440
PERO2450
PERO2460
PERO2470
PERO2480
PERO2490
PERO2500
PERO2510
PERO2520
PERO2530
PERO2540
PERO2550
PERO2560
PERO2570
PERO2580
PERO2590
PERO2600
PERO2610
PERO2620
PERO2630
PERO2640
PERO2650
PERO2660
PERO2670
PERO2680
PERO2690
PERO2700
PERO2710
PERO2720
PERO2730
PERO2740
PERO2750
PERO2760
PERO2770
PERO2780
PERO2790
PERO2800
PERO2810
PERO2820
PERO2830
PERO2840
PERO2850
PERO2860
PERO2870
PERO2880

```

```

C-----
INTEGER I,J,K
COMMON /A/ AMR, ARTR, ATR, CDOMR, CDOTR, CMR, CTMR, CTR, CTTR,
* DA, DEN, EEPAF, GE, HD, LPA, RADILS, RTR, RIHT, RVMR, RVTR, SLMR,
* SCIR, SKLHT, TEMP, W, VF, VV, VVIR, IBMR, DENSSL, PIE, PTMIN(400), VIMR,
* VITR, VITR, VIMR, RANMAX, THETA, DELTA, SHPRAN(400), VV1, VV1, IENG, SUMR3,
* VIMR2, LIFF, SFCM, SFCN, SFCC, SHPN, SHFN, SFFC, CMRI, RVMRI, RRRMR,
* CGMMCN /B/ PIMR(400), PCMR(400), PPMR(400), PCMR(400), PITR(400),
* POTR(400), PTMR(400), PTTR(400), PTAC(400)
* CGMMCN /C/ RMR(400), I, J, K, DRMR(400), MINX, MAXX, MINY, MAXY, MINZ,
* MAXZ, CMIN, DMAX, MINE, MAXB, MINC, MAXC, MIN, MAX, MAXA, IANS, TEXT(200)
COMMON /K/ RCMAX(400), DRCMAX(400), VFRAN(400), DVFRAN(400),
* HCVALT(400), DHVAL(400), SEVALT(400), DSEVAL(400), FMERT(400),
* OFMERT(400), VFEND(400), DVEND(400)
COMMON /Y/ DPIMR(400), DPOMR(400), DPPMR(400), DPCMR(400), DPTMR(400),
* DPIR(400), DPOTR(400), DPTAC(400), DPTTR(400)
C-----
MAX = 0.0
MAXA = 0.0
DO 40 I = 1, 324
CALL CONST
AMR = RMR(I) * 2.0 * PIE
IF(K .NE. 1) GO TO 8
      VIMR = RVMR * RMR(I)
      ARMR = VF / (RVMR * RMR(I))
      SLMR = IBMR * CMR / (PIE * RMR(I))
8      IF(K .NE. 2) GO TO 10
      RVMR = VTMR2 / RMR(I)
      VIMR = RVMR * RMR(I)
      ARMR = VF / (RVMR * RMR(I))
      SLMR = IBMR * CMRI / (PIE * RMR(I))
10     IF(K .NE. 3) GO TO 12
      CMR = SDMR3 * PIE * RMR(I) / IBMR
      SLMR = IBMR * CMR / (PIE * RMR(I))
      RVMR = RVMRI
      ARMR = VF / (RVMR * RMR(I))
      VIMR = RVMR * RMR(I)
12     IF(K .NE. 4) GO TO 14
      CMR = SDMR3 * PIE * RMR(I) / IBMR
      SLMR = IBMR * CMR / (PIE * RMR(I))
      VIMR = VTMR2 / RMR(I)
      ARMR = RVMR * RMR(I)
      VIMR = VF / (RVMR * RMR(I))
14     CALL VELMR
      CALL FCWMR
      CALL VELTR
      CALL FCWTR
      IF(FIMR(I) .GT. MAXA) MAXA = PIMR(I)
      IF(FCMR(I) .GT. MAXA) MAXA = PCMR(I)
PERO2890
PERO2900
PERO2910
PERO2920
PERO2930
PERO2940
PERO2950
PERO2960
PERO2970
PERO2980
PERO2990
PERO3000
PERO3010
PERO3020
PERO3030
PERO3040
PERO3050
PERO3060
PERO3070
PERO3080
PERO3090
PERO3100
PERO3110
PERO3120
PERO3130
PERO3140
PERO3150
PERO3160
PERO3170
PERO3180
PERO3190
PERO3200
PERO3210
PERO3220
PERO3230
PERO3240
PERO3250
PERO3260
PERO3270
PERO3280
PERO3290
PERO3300
PERO3310
PERO3320
PERO3330
PERO3340
PERO3350
PERO3360

```

```

20      IF (FFMR(I) .GT. MAXA) MAXA = PPMR(I)
      IF (VV .LE. 0.0) GO TO 20
      IF (FCMR(I) .GT. MAXA) MAXA = PCMR(I)
      IF (PTAC(I) .GT. MAX) MAX = PTAC(I)
      VF = VF / 1.6885
      VV = VV * 60.0
40      CONTINUE
      MINV = 500.0
      MAXY = -500.0
      MINX = 500.0
      MAXX = -500.0
      DO 140 I = 1, 324
        CPIMR(I) = ((PIMR(I) - PIMR(163)) / PIMR(163)) * 100.0
        IF (CPIMR(I) .GT. MAXX) MAXX = CPIMR(I)
        IF (CPIMR(I) .LT. MINX) MINX = CPIMR(I)
        DPMF(I) = ((POMR(I) - POMR(163)) / POMR(163)) * 100.0
        IF (DPMF(I) .GT. MAXX) MAXX = DPMF(I)
        IF (DPMF(I) .LT. MINX) MINX = DPMF(I)
        DPPMR(I) = ((PPMR(I) - PPMR(163)) / PPMR(163)) * 100.0
        IF (DPPMR(I) .GT. MAXX) MAXX = DPPMR(I)
        IF (DPPMR(I) .LT. MINX) MINX = DPPMR(I)
        IF (VV .LE. 0.0) GO TO 120
        CPCMR(I) = ((PCMR(I) - PCMR(163)) / PCMR(163)) * 100.0
        IF (CPCMR(I) .GT. MAXX) MAXX = CPCMR(I)
        IF (CPCMR(I) .LT. MINX) MINX = CPCMR(I)
        DPTMR(I) = ((PTMR(I) - PTMR(163)) / PTMR(163)) * 100.0
        DPTTR(I) = ((PTTR(I) - PTTR(163)) / PTTR(163)) * 100.0
        DPTTF(I) = ((PTTF(I) - PTTF(163)) / PTTF(163)) * 100.0
        DPTAC(I) = ((PTAC(I) - PTAC(163)) / PTAC(163)) * 100.0
        IF (DPTAC(I) .GT. MAXY) MAXY = DPTAC(I)
        IF (DPTAC(I) .LT. MINY) MINY = DPTAC(I)
        IF (DPTMR(I) .GT. MAXY) MAXY = DPTMR(I)
        IF (DPTMR(I) .LT. MINY) MINY = DPTMR(I)
        IF (DPTTR(I) .GT. MAXY) MAXY = DPTTR(I)
        IF (DPTTR(I) .LT. MINY) MINY = DPTTR(I)
140      CONTINUE
      CALL GRAFIC
      CALL GRAFID
      VF = 0.0
      MINB = 1.0
      MAXB = 0.0
      DO 150 I = 1, 324
        CALL CCNST
        AMR = RMNR(I)*2.0*PIE
        IF (K .NE. 1) GO TO 142

```

```

PERO3370
PERO3380
PERO3390
PERO3400
PERO3410
PERO3420
PERO3430
PERO3440
PERO3450
PERO3460
PERO3470
PERO3480
PERO3490
PERO3500
PERO3510
PERO3520
PERO3530
PERO3540
PERO3550
PERO3560
PERO3570
PERO3580
PERO3590
PERO3600
PERO3610
PERO3620
PERO3630
PERO3640
PERO3650
PERO3660
PERO3670
PERO3680
PERO3690
PERO3700
PERO3710
PERO3720
PERO3730
PERO3740
PERO3750
PERO3760
PERO3770
PERO3780
PERO3790
PERO3800
PERO3810
PERO3820
PERO3830
PERO3840

```

```

142      VIMR = RVMR*RM(R(I))
      ARMR = VF / (RVMR*RM(R(I)))
      SCMR = IBMR*CMR / (PIE*RM(R(I)))
      IF(K.NE.2) GO TO 144
      VIMR = VTMR2/RMR(I)
      ARMR = RVMR*RM(R(I))
      SCMR = VF / (RVMR*RM(R(I)))
      IF(K.NE.3) GO TO 146
      CMR = SDMR3*PIE*RM(R(I))/IBMR
      SCMR = IBMR*CMR / (PIE*RM(R(I)))
      RVMR = RVMR1
      ARMR = VF / (RVMR*RM(R(I)))
      VIMR = RVMR*RM(R(I))
      IF(K.NE.4) GO TO 148
      CMR = SDMR3*PIE*RM(R(I))/IBMR
      SCMR = IBMR*CMR / (PIE*RM(R(I)))
      RVMR = VTMR2/RMR(I)
      VIMR = RVMR*RM(R(I))
      ARMR = VF / (RVMR*RM(R(I)))
      CALL VELMR
      CALL POWMR
      CALL VELTR
      CALL FCWTR
      FMERT(I) = PIMR(I) / (PIMR(I) + PCMR(I))
      IF(FMERT(I) .GT. MAXB) MAXB = FMERT(I)
      IF(FMERT(I) .LT. MINB) MINB = FMERT(I)
      IF(FMERT(I) .LT. MINC) MINC = FMERT(I)
      CGNTINLE
150-----
      MINC = 500.0
      MAXC = 500.0
      DO 155 I = 1,324
      FMERT(I) = ((FMERT(I) - FMERT(163)) / FMERT(163)) * 100.0
      IF(CFMERT(I) .GT. MAXC) MAXC = CFMERT(I)
      IF(CFMERT(I) .LT. MINC) MINC = CFMERT(I)
155 CONTINUE
      CALL GRAFLE
      RETURN
      ENCL
C*****
C SUBROUTINE TO CALCULATE THE ENDURANCE VELCCITY
C*****
C SUBROUTINE ENDUR
C*****
C IMPLICIT REAL (A-H,I-Z)
C COMMON /A/ AMR, ARMR, ARTR, ATR, CCMR, CDMTR, CMR, CTR, CTRR,
PER03850
PER03860
PER03870
PER03880
PER03890
PER03900
PER03910
PER03920
PER03930
PER03940
PER03950
PER03960
PER03970
PER03980
PER03990
PER04000
PER04010
PER04020
PER04030
PER04040
PER04050
PER04060
PER04070
PER04080
PER04090
PER04100
PER04110
PER04120
PER04130
PER04140
PER04150
PER04160
PER04170
PER04180
PER04190
PER04200
PER04210
PER04220
PER04230
PER04240
PER04250
PER04260
PER04270
PER04280
PER04290
PER04300
PER04310
PER04320

```



```

160 VFEND(I) = VF/1.68889
180 CCNTINUE
CONTINUE
VFEND(I) = VFEND(I) - 10.0
VFF = VFEND(I)
DO 220 J = 1,20
VF = (FLGAT(J)-1.0) + VFF
CALL CONST
ARMR = VF / (RVMR*RM(R))
CALL VELMR
CALL POWMR
CALL VELTR
CALL POWTR
IF (PTAC(I) -GT. PTMIN(I)) GC TC 200
PTMIN(I) = PTAC(I)
VFEND(I) = VF/1.68889
CCNTINUE
200 CONTINUE
VFEND(I) = VFEND(I) - 1.0
VFF = VFEND(I)
DO 260 J = 1,200
VF = ((FLGAT(J)/100)-0.01) + VFF
CALL CONST
ARMR = VF / (RVMR*RM(R))
CALL VELMR
CALL POWMR
CALL VELTR
CALL POWTR
IF (PTAC(I) -GT. PTMIN(I)) GC TC 240
PTMIN(I) = PTAC(I)
VFEND(I) = VF/1.68889
CCNTINUE
240 CONTINUE
VFEND(I) = VFEND(I) - 0.01
IF (K.EQ. 1) VFEND1(I) = VFEND(I)
IF (K.EQ. 2) VFEND2(I) = VFEND(I)
IF (K.EQ. 3) VFEND3(I) = VFEND(I)
IF (K.EQ. 4) VFEND4(I) = VFEND(I)
IF (VFEND(I) .LT. MIN) MIN = VFEND(I)
IF (VFEND(I) .GT. MAX) MAX = VFEND(I)
CCNTINUE
280 CONTINUE
MAXY = -500.0
MINY = 500.0
DO 480 I = 1,324
DVVFEND(I) = ((VFEND(I) - VFEND(163)) / VFEND(163)) * 100.0
IF (CVVFEND(I) .GT. MAXY) MAXY = CVVFEND(I)
IF (CVVFEND(I) .LT. MINY) MINY = CVVFEND(I)

```

```

PER04810
PER04820
PER04830
PER04840
PER04850
PER04860
PER04870
PER04880
PER04890
PER04900
PER04910
PER04920
PER04930
PER04940
PER04950
PER04960
PER04970
PER04980
PER04990
PER05000
PER05010
PER05020
PER05030
PER05040
PER05050
PER05060
PER05070
PER05080
PER05090
PER05100
PER05110
PER05120
PER05130
PER05140
PER05150
PER05160
PER05170
PER05180
PER05190
PER05200
PER05210
PER05220
PER05230
PER05240
PER05250
PER05260
PER05270
PER05280

```


78

PER09130
PER09140
PER09150
PER09160
PER09170
PER09180
PER09190
PER09200
PER09210
PER09220
PER09230
PER09240
PER09250
PER09260
PER09270
PER09280
PER09290
PER09300
PER09310
PER09320
PER09330
PER09340
PER09350
PER09360
PER09370
PER09380
PER09390
PER09400
PER09410
PER09420
PER09430
PER09440
PER09450
PER09460
PER09470
PER09480
PER09490
PER09500
PER09510
PER09520
PER09530
PER09540
PER09550
PER09560
PER09570
PER09580
PER09590
PER09600

```

VF = VFF
LA = FLOAT(J)*1000.0
CALL RHO
VV = 100.0
CALL CONST
ARMR = VF / (RVMR*RRMR(I))
CALL VELMR
CALL POWMR
CALL VELTR
CALL POWTR
FCWAA = PTAVL*DELTA*SQR(THETA)
IF (PTAC(I) .LT. POWAA) GO TC 1420
SEVALT(I) = CA
GO TO 1440
CCNTINUE
CONTINUE
SEV = SEVALT(I) - 1000.0
DO 1460 J = 1,20
VF = VFF
LA = FLOAT(J)*50.0 + SEV
CALL RHC
VV = 100.0
CALL CONST
ARMR = VF / (RVMR*RRMR(I))
CALL VELMR
CALL POWMR
CALL VELTR
CALL POWTR
FCWAA = PTAVL*DELTA*SQR(THETA)
IF (PTAC(I) .LT. POWAA) GO TC 1460
SEVALT(I) = CA
GO TO 1480
CCNTINUE
CONTINUE
SEV = SEVALT(I) - 50.0
DO 1480 J = 1,100
VF = VFF
LA = FLOAT(J)*0.5 + SEV
CALL RHO
VV = 100.0
CALL CONST
ARMR = VF / (RVMR*RRMR(I))
CALL VELMR
CALL POWMR
CALL VELTR
CALL POWTR
FCWAA = PTAVL*DELTA*SQR(THETA)
IF (PTAC(I) .LT. POWAA) GO TC 1500

```

1420
1440

1460
1480

```

C*****
SUBROUTINE CEIL (A-H, L-Z)
IMPLICIT REAL (A-H, L-Z)
COMMON /A/ AMR, ARTR, ATR, CDCMR, CDOIR, CMR, CTMR, CTR, CTTR, SDMR,
DA, DEN, EFAF, EFPV, GE, HD, L, PA, RADILS, RTR, RTHI, RVMR, RVTR, SDMR,
SDTR, SKCHT, TEMP, W, VF, VV, IBTR, IBMR, DENSSL, PIE, PTMIN(400), VIMR,
VITR, VITR, VIMR, RANMAX, THETA, DELTA, SHPRAN(400), VV1, VF1, IENG, SDMR3,
VITR2, CIFF, SFCM, SFCN, SFCR, SHPN, SHPN, SHFC, CPR1, RVMR1, RMR,
CCMMCN /B/ PTMR(400), PTMR(400), PCMR(400), PIRK(400),
PTR(400), PTR(400), PTAC(400),
CCMMCN /C/ RMTR(400), I, J, K, DRMR(400), MINX, MAXX, MINY, MAXY, MINZ,
MAXZ, DMIN, DMAX, MINB, MAXB, PINC, MAXC, MIN, MAX, IANS, TEXT(200),
COMMON /A/ RCMAX(400), DRCMAX(400), VFRAN(400), DVFRAN(400),
HCVALT(400), DHOVAL(400), SEVALT(400), DSEVAL(400), FMERT(400),
FMERT(400), VFEND(400), DVFEND(400),
CCMMCN /Z/ VFEND1(400), VFEND2(400), VFEND3(400), VFEND4(400)
C-----
SKCHT = 2500.0
MIN = 30000.0
MAX = 0.0
PA = 0.0
TEMP = 0.0
IF (IENG .EQ. 1) PTAVAI = (FLOAT(IENG)*SHPN - 10.0) * 0.97
IF (IENG .EQ. 2) PTAVAI = (FLOAT(IENG)*SHPN - 10.0) * 0.9#0.97
IF (IENG .EQ. 3) PTAVAI = (FLOAT(IENG)*SHPN - 10.0) * 0.9#0.94
DO 154C I = 1, 324
AMR = RMR(I)*2.0*PIE
IF (K .NE. 1) GC TC 1408
VFF = VFEND1(I)
VTMR = RVMR1/RMR(I)
SDMR = IBMR*CMR1 / (PIE*RMR(I))
IF (K .NE. 2) GO TO 1412
VFF = VFEND2(I)
RVMR = VTMR2/RMR(I)
VTMR = RVMR/RMR(I)
SDMR = IBMR*CMR1 / (PIE*RMR(I))
IF (K .NE. 3) GO TO 1414
VFF = VFEND3(I)
CMR = SDMR3*PIE/RMR(I)/IBMR
SDMR = IBMR*CMR / (PIE*RMR(I))
VTMR = RVMR1/RMR(I)
IF (K .NE. 4) GO TO 1416
CMR = SDMR3*PIE/RMR(I)/IBMR
SDMR = IBMR*CMR / (PIE*RMR(I))
VTMR = RVMR2/RMR(I)
VFF = VFEND4(I)
CO 1420 J = 1, 30

```

```

PER08650
PER08660
PER08670
PER08680
PER08690
PER08700
PER08710
PER08720
PER08730
PER08740
PER08750
PER08760
PER08770
PER08780
PER08790
PER08800
PER08810
PER08820
PER08830
PER08840
PER08850
PER08860
PER08870
PER08880
PER08890
PER08900
PER08910
PER08920
PER08930
PER08940
PER08950
PER08960
PER08970
PER08980
PER08990
PER09000
PER09010
PER09020
PER09030
PER09040
PER09050
PER09060
PER09070
PER09080
PER09090
PER09100
PER09110
PER09120

```



```

COMMON /W/ RCMAX(400), DRCMAX(400), VFRAN(400), DVFRAN(400),
* HCVALT(400), DHOVAL(400), SEVALT(400), DSEVAL(400), FMERT(400),
* DFMEKT(400), VFEND(400), DVFEND(400)

```

```

C-----
VF = 0.0
VV = 0.0
SKCHT = 10.0
MIN = 3000.0
MAX = 0.0
PA = 0.0
TEMP = 0.0
IF (IENG.EQ.1) PTAVAL = (FLOAT(IENG)*SHPN - 10.0) * 0.97
IF (IENG.EQ.2) PTAVAL = (FLOAT(IENG)*SHPN - 10.0) * 0.9 * C.97
IF (IENG.EQ.3) PTAVAL = (FLOAT(IENG)*SHPN - 10.0) * 0.9 * 0.94
DO 1240 I = 1, 324
  AMR = RMR(I) * 2.0 * PIE
  IF (K.NE.1) GO TO 1108
  VTMR = RMR1 * RMR(I) / (PIE * RMR(I))
  SDMR = IBMR * CMR1 / (PIE * RMR(I))
1108   IF (K.NE.2) GO TO 1112
  VTMR = VTMR2 / RMR(I)
  SDMR = IBMR * CMR1 / (PIE * RMR(I))
1112   IF (K.NE.3) GO TO 1114
  CMR = SDMR3 * PIE * RMR(I) / IBMR
  SLMR = IBMR * CMR / (PIE * RMR(I))
  VTMR = RMR1 * RMR(I)
1114   IF (K.NE.4) GO TO 1116
  CMR = SDMR3 * PIE * RMR(I) / IBMR
  SLMR = IBMR * CMR / (PIE * RMR(I))
  FVMR = VTMR2 / RMR(I)
  VTMR = RMR * RMR(I)
1116   ARMR = VF / (RVMR * RMR(I))
  DO 1120 J = 1, 30
    IA = FLOAT(J) * 1000.0
    CALL RHO
    CALL CUNST
    CALL VELMR
    CALL POWMR
    CALL VELTR
    CALL POWTR
    FCWAA = PTAVAL * DELTA * SQR(THETA)
    IF (PTAC(I) .LT. POWAA) GOTC 1120
    HCVALT(I) = CA
    GC TO 1140
  CCNTINUE
CONTINUE
HOV = HOVALT(I) - 1000.0
1120
1140

```

```

PERO7690
PERO7700
PERO7710
PERO7720
PERO7730
PERO7740
PERO7750
PERO7760
PERO7770
PERO7780
PERO7790
PERO7800
PERO7810
PERO7820
PERO7830
PERO7840
PERO7850
PERO7860
PERO7870
PERO7880
PERO7890
PERO7900
PERO7910
PERO7920
PERO7930
PERO7940
PERO7950
PERO7960
PERO7970
PERO7980
PERO7990
PERO8000
PERO8010
PERO8020
PERO8030
PERO8040
PERO8050
PERO8060
PERO8070
PERO8080
PERO8090
PERO8100
PERO8110
PERO8120
PERO8130
PERO8140
PERO8150
PERO8160

```



```

734      VTMR = RVMR1*RMRR(I)
      IF(K.NE.4) GO TO 736
      CMR = SDMR3*PIE*RMRR(I)/IBMR
      SDMR = IBMR*CMR / (PIE*RMRR(I))
      RVMR = VTMR2/RMR(I)
      VTMR = 1.20
      DO 740 J = 1,20
      VF = FLOCAT(J)*10.0
      CALL CONST
      ARMR = VF / (RVMR*RMRR(I))
      CALL VELMR
      CALL POWMR
      CALL VELTR
      CALL POWTR
      DE = 550.0*(PTAC(I)+PSHP) / VF
      WDE = W / DE
      IF (WDE.LE. RANMAX) GO TO 750
      RANMAX = WDE
      VFRAN(I) = VF/1.68885
      CONTINUE
740      CONTINUE
750      VFF = VFRAN(I)
      RANMAX = 0.0
      DO 820 J = 1,20
      VF = FLOCAT(J)-1.0 + VFF
      CALL CONST
      ARMR = VF / (RVMR*RMRR(I))
      CALL VELMR
      CALL POWMR
      CALL VELTR
      CALL POWTR
      LE = 550.0 * (PTAC(I)+PSHP) / VF
      IF (DE.EQ.0.0) GO TO 780
      WDE = W / DE
      GC TO 800
      WDE = 0.0
      IF (WDE.LT. RANMAX) GO TO 850
      RANMAX = WDE
      VFRAN(I) = VF/1.68885
      CONTINUE
780      CONTINUE
800      VFF = VFRAN(I)
      RANMAX = 0.0
      DO 860 J = 1,201
      VF = (FLOCAT(J-1)/100.0) + VFF
      CALL CONST
      ARMR = VF / (RVMR*RMRR(I))
      CALL VELMR

```

```

PER06730
PER06740
PER06750
PER06760
PER06770
PER06780
PER06790
PER06800
PER06810
PER06820
PER06830
PER06840
PER06850
PER06860
PER06870
PER06880
PER06890
PER06900
PER06910
PER06920
PER06930
PER06940
PER06950
PER06960
PER06970
PER06980
PER06990
PER07000
PER07010
PER07020
PER07030
PER07040
PER07050
PER07060
PER07070
PER07080
PER07090
PER07100
PER07110
PER07120
PER07130
PER07140
PER07150
PER07160
PER07170
PER07180
PER07190
PER07200

```


PER05770
 PER05780
 PER05790
 PER05800
 PER05810
 PER05820
 PER05830
 PER05840
 PER05850
 PER05860
 PER05870
 PER05880
 PER05890
 PER05900
 PER05910
 PER05920
 PER05930
 PER05940
 PER05950
 PER05960
 PER05970
 PER05980
 PER05990
 PER06000
 PER06010
 PER06020
 PER06030
 PER06040
 PER06050
 PER06060
 PER06070
 PER06080
 PER06090
 PER06100
 PER06110
 PER06120
 PER06130
 PER06140
 PER06150
 PER06160
 PER06170
 PER06180
 PER06190
 PER06200
 PER06210
 PER06220
 PER06230
 PER06240

```

492 IF(K .NE. 3) GO TO 494
    VF = VFEND3(I)
    CMR = SDMR3*PIE*RM(R(I))/IBMR
    SDCMR = IBMR*CMR / (PIE*RM(R(I)))
    VTMR = RM(R(I))*RM(R(I))
494 IF(K .NE. 4) GO TO 496
    VF = VFEND4(I)
    CMR = SDMR3*PIE*RM(R(I))/IBMR
    SDCMR = IBMR*CMR / (PIE*RM(R(I)))
    VTMR = RM(R(I))*RM(R(I))
496 DO 500 J = 1,80
    VV = FLOAT(J)*100.0
    CALL CONST
    ARMR = VF / (RVMR*RM(R(I)))
    CALL VELMR
    CALL POWMR
    CALL VELTR
    CALL POWTR
    IF (PTAC(I) .GT. PTAVAL) GO TO 520
    VF = VF/1.68889
    CONTINUE
500 RCMAXX = (VV*60.0) - 100.0
520 VF = VF/1.68889
    DO 540 J = 1,100
    VV = FLOAT(J)*1.0 + RCMAXX
    CALL CONST
    ARMR = VF / (RVMR*RM(R(I)))
    CALL VELMR
    CALL POWMR
    CALL VELTR
    CALL POWTR
    IF (PTAC(I) .GT. PTAVAL) GO TO 560
    VF = VF/1.68889
    CONTINUE
540 RCMAX(I) = (VV*60.0)-1.0
560 IF(RCMAX(I) .LT. MIN) MIN = RCMAX(I)
    IF(RCMAX(I) .GT. MAX) MAX = RCMAX(I)
    CONTINUE
580 CONTINUE
    MAXY = -500.0
    MINY = 500.0
    DO 700 I = 1,324
    DRCMAX(I) = ((RCMAX(I) - RCMAX(163)) / RCMAX(163)) * 100.0
    IF(DRCMAX(I) .GT. MAXY) MAXY = DRCMAX(I)
    IF(DRCMAX(I) .LT. MINY) MINY = DRCMAX(I)
    CONTINUE
700 CALL GRAF3
  
```

```

480 CONTINUE
   IF(IANS.EQ.3.OR.IANS.EQ.6) GO TO 482
482 CALL GRAF2
   CONTINUE
   RETURN
   ENC
C*****
C SUBROUTINE TO CALCULATE THE MAXIMUM RATE OF CLIMB
C*****
C SUEROUTINE MAXRC
C IMPLICIT REAL (A-H,L-Z)
C COMMON /A/ AMR,ARTR,ATR,CDCMR,CDOIR,CMF,CTMR,CTR,CTTR,SDMR,
* DA,DEN,EFPF,EFPAF,GE,HD,L,PA,RADILS,RTR,RTHT,RVMR,RVTR,SDMR,
* SDTR,SKDHT,TEMP,W,VF,VV,IBTR,IBMR,DENSSL,PIE,PTMIN(400),VIMR,
* VITR,VITR,VIMR,RANMAX,THETA,DELTA,SHPRAN(400),VV1,VF1,IENG,SDMR3,
* VIMR2,LIFF,SFCM,SFCN,SFCC,SHPM,SHPN,SHFC,CPRI,RVMR1,RMRMR
* COMMON /B/ PIMR(400),POMR(400),PPMR(400),PCMR(400),PITR(400),
* PCTR(400),PTMR(400),PTTR(400),PTAC(400)
* CCMCN /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
* MAXZ,CMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXA,IANS
* COMMON /W/ RCMAX(400),CRCMAX(400),VFRAN(400),DVFRAN(400),
* HUVALT(400),DHQVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
* DFEMERT(400),VFEND(400),DVEND(400)
* CCMCN /Z/ VFEND1(400),VFEND2(400),VFEND3(400),VFEND4(400)
C-----
PA = 0.0
TEMP = 0.0
DA = 0.0
IF(IENG.EQ.1) PTAVAL = (FLOAT(IENG)*SHPM - 10.0) * 0.97
IF(IENG.EQ.2) PTAVAL = (FLOAT(IENG)*SHPM - 10.0) * 0.9*0.97
IF(IENG.EQ.3) PTAVAL = (FLOAT(IENG)*SHPM - 10.0) * 0.9*0.94
CALL RFC
MAX = 800.0
DO I = 1,324
  AMR = RMR(I)*#2,0*PIE
  IF(K.NE.1) GC TO 490
  VF = VFEND1(I)
  VIMR = RVMR1/RMR(I)
  SCMR = IBMR*CMR1 / (PIE*RMR(I))
  IF(K.NE.2) GO TO 492
  VF = VFEND2(I)
  VIMR = RVMR2/RMR(I)
  VIMR = RVMR*CMR1 / (PIE*RMR(I))
  SCMR = IBMR*CMR1 / (PIE*RMR(I))
490

```

CALL CHNCCT (DRMR,DPTMR,322,0)	PER12010
CALL CLARVE (DRMR,DPTAC,322,0)	PER12020
CALL CLNCDSH (DRMR,DPTAC,322,0)	PER12030
CALL CLFVE (DRMR,DPTAC,322,0)	PER12040
MAXLIN=LENGTH(TEXT,100,30)	PER12050
CALL FLIGHT(0,1)	PER12060
CALL LINES(TAIL RCTOR POWER\$,TEXT,1)	PER12070
CALL LINES(MAIN RCTOR POWER\$,TEXT,2)	PER12080
CALL LINES(TOTAL POWER\$,TEXT,3)	PER12090
CALL LEGEND(TEXT,3,0.4,3.8)	PER12100
CALL BLREC(0.3,3.7,1.55,0.75,1.2)	PER12110
CALL RESET(HEIGHT)	PER12120
CALL DCT	PER12130
CALL GFC(1,2)	PER12140
CALL ENDCR(0)	PER12150
-----	PER12160
CALL PHYSOR(1.5,1.5)	PER12170
CALL AREA2D(6.50,4.70)	PER12180
CALL HEADIN(POWER VERSUS RADIUS\$,19,1.4,3)	PER12190
IF(K NE 1) GO TO 8	PER12200
* CALL HEADIN(CHORD & ROTATIONAL VELOCITY HELD CONSTANT\$,41,	PER12210
HEADIN(SOLIDITY, TIP VELOCITY & ADVANCE RATIO\$,38,0.8,4)	PER12220
CALL MESSAG(ALLOWED TO VARY WITH RADIUS\$,27,0.8,4)	PER12230
CALL MESSAG(SPEC RMR=\$,100,7.4,6.5)	PER12240
CALL REALNO(RMR(1,63),2,ABUT\$,ABLT\$)	PER12250
CALL MESSAG(CASE 1\$,100,8.7,6.0)	PER12260
IF(K NE 2) GO TO 10	PER12270
* CALL HEADIN(CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT\$,	PER12280
48,0.8,4)	PER12290
CALL HEADIN(SOLIDITY, ROTATIONAL VELOCITY\$,29,0.8,4)	PER12300
CALL MESSAG(ALLOWED TO VARY WITH RADIUS\$,27,0.8,4)	PER12310
CALL MESSAG(SPEC RMR=\$,100,7.4,6.5)	PER12320
CALL REALNO(RMR(1,63),2,ABUT\$,ABLT\$)	PER12330
CALL MESSAG(CASE 2\$,100,8.7,6.0)	PER12340
IF(K NE 3) GO TO 12	PER12350
* CALL HEADIN(SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT\$,	PER12360
44,0.8,4)	PER12370
CALL HEADIN(CHORD,ADVANCE RATIO & TIP VELOCITY\$,34,0.8,4)	PER12380
CALL MESSAG(ALLOWED TO VARY WITH RADIUS\$,27,0.8,4)	PER12390
CALL MESSAG(SPEC RMR=\$,100,7.4,6.5)	PER12400
CALL REALNO(RMR(1,63),2,ABUT\$,ABLT\$)	PER12410
CALL MESSAG(CASE 3\$,100,8.7,6.0)	PER12420
IF(K NE 4) GO TO 14	PER12430
* CALL HEADIN(SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT\$,	PER12440
51,0.8,4)	PER12450
CALL HEADIN(CHORD & ROTATIONAL VELOCITY\$,27,0.8,4)	PER12460
CALL HEADIN(ALLOWED TO VARY WITH RADIUS\$,27,0.8,4)	PER12470
	PER12480

```

CALL MESSAGE ('SPEC RMR =',100,7.4,6.5)
CALL REALNO (RMR(163),2,'ABUT','ABUT')
CALL MESSAGE ('CASE 4',100,8.7,6.0)
CALL ENCLPL(0)
RETURN
END

```

14

C*****

```

SUBROUTINE TO CREATE GRAPH OF PCWER (INDUCEL,PROFILE, ETC.)
VS RADII.

```

C*****

C*****

```

SUBROUTINE GRAFIC
IMPLICIT REAL (A-H,L-Z)

```

```

* CCMCN /B/ PIMR(400),POMR(400),PPMR(400),PTTR(400),PI TR(400),
* CCMCN /C/ RMR(400),PTTR(400),PTAC(400)
* MAXZ,LMIN,DMAX,MINB,MAXB,MINC,MAXC,MIN,MAX,MAXX,MINY,MAXY,MINZ,
* COMMON /Y/DPIMR(400),DPOMR(400),DPPMR(400),DPCMR(400),EPTMR(400),
* DPTR(400),DPGTR(400),DPTAC(400)

```

```

CALL CCMPRS
CALL PAGE (11,0,8.5)
CALL PHYSGR (1,5,1.5)
CALL AREA2D (2,50,4.50)
CALL XNAME ('RADIUS OF MAIN ROTOR (FT)',100)
CALL YNAME ('POWER REQUIRED (SHP)',100)
CALL GRAF (MINZ,'SCALE',MAXZ,0.0,'SCALE',MAXX)
CALL LEGLIN
CALL CURVE (RMR,PIMR,324,0)
CALL DASH
CALL CURVE (RMR,POMR,324,0)
CALL CHNDOT
CALL CURVE (RMR,PPMR,324,0)
CALL CFCSH
CALL CURVE (RMR,PI TR,324,0)
CALL DCT
CALL CURVE (RMR,POTR,324,0)
IF (V LE .0) GO TO 20
CALL CURVE (RMR,PCMR,324,0)
MAXLIN=LINEST(TEXT,100,30)
CALL HEIGHT(0.1)

```

20

```

CALL LINES('INDUCEL POWER (MR)',TEXT,1)
CALL LINES('PROFILE POWER (MR)',TEXT,2)
CALL LINES('PARASITE POWER (MR)',TEXT,3)
CALL LINES('INDUCED PCWER (TR)',TEXT,4)
CALL LINES('PROFILE PCWER (TR)',TEXT,5)
CALL LEGEND(TEXT,5,0.4,3.50)

```

PER12490
PER12500
PER12510
PER12520
PER12530
PER12540
PER12550
PER12560
PER12570
PER12580
PER12590
PER12600
PER12610
PER12620
PER12630
PER12640
PER12650
PER12660
PER12670
PER12680
PER12690
PER12700
PER12710
PER12720
PER12730
PER12740
PER12750
PER12760
PER12770
PER12780
PER12790
PER12800
PER12810
PER12820
PER12830
PER12840
PER12850
PER12860
PER12870
PER12880
PER12890
PER12900
PER12910
PER12920
PER12930
PER12940
PER12950
PER12960

PER12570
PER12580
PER12590
PER12600
PER12610
PER12620
PER12630
PER12640
PER12650
PER12660
PER12670
PER12680
PER12690
PER12700
PER12710
PER12720
PER12730
PER12740
PER12750
PER12760
PER12770
PER12780
PER12790
PER12800
PER12810
PER12820
PER12830
PER12840
PER12850
PER12860
PER12870
PER12880
PER12890
PER12900
PER12910
PER12920
PER12930
PER12940
PER12950
PER12960
PER12970
PER12980
PER12990
PER13000
PER13010
PER13020
PER13030
PER13040
PER13050
PER13060
PER13070
PER13080
PER13090
PER13100
PER13110
PER13120
PER13130
PER13140
PER13150
PER13160
PER13170
PER13180
PER13190
PER13200
PER13210
PER13220
PER13230
PER13240
PER13250
PER13260
PER13270
PER13280
PER13290
PER13300
PER13310
PER13320
PER13330
PER13340
PER13350
PER13360
PER13370
PER13380
PER13390
PER13400
PER13410
PER13420
PER13430
PER13440

CALL BLREC(0.3,3.42,2.15,1.00,1.1)
CALL RESET(HEIGHT)
CALL DCT
CALL GRID(1,2)
CALL ENGR(0)

PHYSOR(5.5,1.5)
CALL AREA2D(2.50,4.50)
CALL XNAME('RADIUS OF MAIN ROTCR (% CHANGE)',100)
CALL YNAME('POWER REQUIRED (% CHANGE)',100)
CALL GRAF(EMIN,SCALE,DMAX,MINX,SCALE,MAXX)
CALL LEGLIN
CALL CUFVE(DRMR,DPIMR,324,0)
CALL DASH
CALL CUFVE(DRMR,DPQMR,324,0)
CALL CFNCOT
CALL CURVE(DRMR,DPPMR,324,0)
CALL CFNCOSH
CALL CURVE(DRMR,DPITR,324,0)
CALL DCT
CALL CUFVE(DRMR,DFCTR,324,0)
IF (VV.LE.0.0) GC TC 22
CALL CUFVE(DRMR,DPCMR,322,0)
MAXLIN=LINEST(TEXT,100,30)
CALL HEIGHT(0.1)
CALL LINES(INDUCED POWER (MR)\$,TEXT,1)
CALL LINES(PROFILE POWER (MR)\$,TEXT,2)
CALL LINES(PARASITE POWER (MR)\$,TEXT,3)
CALL LINES(INDUCED POWER (TR)\$,TEXT,4)
CALL LINES(PROFILE POWER (TR)\$,TEXT,5)
CALL LEGEND(TEXT,5,0.4,3.50)
CALL BLREC(0.3,3.42,2.2,1.00,1.2)
CALL RESET(HEIGHT)
CALL DCT
CALL GRID(1,2)
CALL ENGR(0)

PHYSOR(1.5,1.5)
CALL AREA2D(6.50,4.70)
CALL HEACIN('POWER 8
IF(K.NE.1) GO TC 8
CALL HEACIN('CHORD & ROTATIONAL VELOCITY HELD CONSTANT',41,
0.8,3)
*
CALL HEACIN('SOLIDITY, TIP VELOCITY & ADVANCE RATIO',38,0.8,4)
CALL HEACIN('ALLOWED TC VARY WITH RADIUS',27,0.8,4)
CALL MESSAGE('SPEC RMR =',100,7.4,6.5)
CALL REALNO(RMR(1.63),2,ABUT,ABUT)
CALL MESSAGE('CASE 1',100,8.7,6.0)

```

8      IF(K.NE.2) GO TO 10
        CALL HEADIN('CHCRC,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$',
         *    48,0.8,4)
        CALL HEADIN('SOLICITY, ROTATIONAL VELOCITY$, 29,0.8,4')
        CALL MESSAG('ALLOWED TO VARY WITH RADIUS$', 27,0.8,4)
        CALL MESSAG('SPEC RMR =$, 100,7.4,6.5)
        CALL REALNC(RMR(163),2,'ABUT','ABUT:')
        CALL MESSAG('CASE 2$, 100,8.7,6.0)
10     IF(K.NE.3) GO TO 12
        CALL HEADIN('SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$',
         *    44,C.8,4)
        CALL HEADIN('CHCRC,ADVANCE RATIO & TIP VELOCITY$, 34,0.8,4)
        CALL MESSAG('ALLOWED TO VARY WITH RADIUS$', 27,0.8,4)
        CALL MESSAG('SPEC RMR =$, 100,7.4,6.5)
        CALL REALNC(RMR(163),2,'ABUT','ABUT:')
        CALL MESSAG('CASE 3$, 100,8.7,6.0)
12     IF(K.NE.4) GO TC 14
        CALL HEADIN('SOLIDITY,ADVANCE RATIC & TIP VELOCITY HELD CONSTANTS$',
         *    51,0.8,4)
        CALL HEADIN('CHCRC & ROTATIONAL VELOCITY$, 27,0.8,4)
        CALL MESSAG('ALLOWED TO VARY WITH RADIUS$', 27,0.8,4)
        CALL MESSAG('SPEC RMR =$, 100,7.4,6.5)
        CALL REALNC(RMR(163),2,'ABUT','ABUT:')
        CALL MESSAG('CASE 4$, 100,8.7,6.0)
14     CALL ENCLPL(0)
        RETURN
      END
C*****
CC SUBROUTINE TO CREATE GRAPH FOR ENDURANCE VELCCITY VS RADIUS
C*****
CC*****
CC*****
CC*****
CC*****
CC*****
CC*****
SUBROUTINE GRAF2
IMPLICIT REAL (A-H,L-Z)
CCOMMON /C/ RMR(400), I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
* MAXZ,CMIN,DMAX,MINB,MAXB,MINC,MAXC,MINI,MAXI,TRANS,TEXT(200)
CCOMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
* HCVALT(400),DHOVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
* DFMERT(400),VFEND(400),DVFEND(400)
CALL CMMPRS
CALL PAGE(11,0.8,5)
CALL PHYSUR(1.5,1.5)
CALL AREA2C(2.50,4.50)
CALL XNAME('RADIUS OF MAIN ROTCR (FT)$',10C)
CALL YNAME('ENDURANCE VELOCITY (KNOTS)$',10C)
CALL GRAF(MINZ,SCALE,MAXZ,MIN,SCALE,MAX)
CALL CURVE(KNR,VFEND,324,0)

```

```

CALL DCT
CALL GRIC (1,2)
CALL RESET(,DOT)
CALL ENDCR(0)
-----C-----
CALL PFYSOR(5,5,1.5)
CALL AREA2D (2,50,4,50)
CALL XNAME (,RADIUS OF MAIN ROTOR (% CHANGE)$,100)
CALL YNAME (,ENDURANCE VELCCITY (% CHANGE)$,100)
CALL GRAF (DMIN,SCALE,DMAX,MINY,SCALE,MAXY)
CALL CURVE (DRMR,DVFEND,324,0)
CALL DCT
CALL GRIC (1,2)
CALL ENDCR(0)
-----C-----
CALL PFYSOR(1,5,1.5)
CALL AREA2C (6,50,4,70)
CALL HEADIN (,ENDURANCE VELOCITY VERSUS RADIUS$,32,1.4,3)
IF (K .NE. 1) GO TO 8
CALL HEADIN (,CHORD & ROTATIONAL VELCCITY HELD CONSTANT$,41,
0.8,3)
*
CALL HEADIN (,SOLICITY, TIP VELOCITY & ADVANCE RATIO$,38,0.8,4)
CALL HEADIN (,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL MESSAGE (,SPEC RMR =$,100,7.4,6.5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAGE (,CASE 1$,100,8.7,6.0)
IF (K .NE. 2) GO TO 10
CALL HEADIN (,CHORD, TIP VELOCITY & ADVANCE RATIO HELD CCNSTANT$,
48,0.8,4)
*
CALL HEADIN (,SOLICITY, ROTATIONAL VELOCITY$,29,0.8,4)
CALL HEADIN (,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL MESSAGE (,SPEC RMR =$,100,7.4,6.5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAGE (,CASE 2$,100,8.7,6.0)
IF (K .NE. 3) GO TO 12
CALL HEADIN (,SOLICITY & ROTATIONAL VELOCITY HELD CONSTANT$,
44,0.8,4)
*
CALL HEADIN (,CHORD, ADVANCE RATIO & TIP VELOCITY$,34,0.8,4)
CALL HEADIN (,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL MESSAGE (,SPEC RMR =$,100,7.4,6.5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAGE (,CASE 3$,100,8.7,6.0)
IF (K .NE. 4) GO TO 14
CALL HEADIN (,SOLICITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$,
51,0.8,4)
*
CALL HEADIN (,CHORD & ROTATIONAL VELOCITY$,27,0.8,4)
CALL HEADIN (,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL MESSAGE (,SPEC RMR =$,100,7.4,6.5)

```

```

PER13530
PER13540
PER13550
PER13560
PER13570
PER13580
PER13590
PER14000
PER14010
PER14020
PER14030
PER14040
PER14050
PER14060
PER14070
PER14080
PER14090
PER14100
PER14110
PER14120
PER14130
PER14140
PER14150
PER14160
PER14170
PER14180
PER14190
PER14200
PER14210
PER14220
PER14230
PER14240
PER14250
PER14260
PER14270
PER14280
PER14290
PER14300
PER14310
PER14320
PER14330
PER14340
PER14350
PER14360
PER14370
PER14380
PER14390
PER14400

```



```

CALL AREA2C (2.50,4.50)
CALL XNAME ('RADIUS OF MAIN ROTOR (FT) $',100)
CALL YNAME ('MAXIMUM RANGE VELOCITY (KNOTS) $',100)
CALL GRAF ('MINZ',SCALE,MAXZ,MIN,SCALE,MAX)
CALL CUFVE (RMR,VFRAN,324,-1)
CALL CCI
CALL GRID (1,2)
CALL RESET ('DOT')
CALL ENLCR(0)
*****
CALL PHYSOR(5.5,1.5)
CALL AREA2C (2.5,4.5)
CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE) $',100)
CALL YNAME ('MAXIMUM RANGE VELOCITY (% CHANGE) $',100)
CALL GRAF ('DMIN',SCALE,DMAX,MINY,SCALE,MAXY)
CALL CUFVE (DRMR,DVFRAN,324,-1)
CALL DCI
CALL GRID (1,2)
CALL ENLCR(0)
-----
CALL PHYSOR(1.5,1.5)
CALL AREA2C (6.50,4.70)
CALL HEADIN ('MAXIMUM RANGE VELOCITY VERSUS RADIUS CHANGE $',
43,1.4,4)
* IF (K.NE.1) GC TC 8
CALL HEADIN ('CHORD & ROTATIONAL VELOCITY HELD CONSTANT $',41,
0.8,3)
CALL HEADIN ('SOLIDITY, TIP VELOCITY & ADVANCE RATIO $',38,0.8,4)
CALL MESSAG ('ALLOWED TO VARY WITH RADIUS $',27,0.8,4)
CALL MESSAG ('SPEC RMR = $,100,7.4,6.5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAG ('CASE 1 $',100,8.7,6.0)
IF (K.NE.2) GC TO 10
CALL HEADIN ('CHORD,TIP VELOCITY & ADVANCE RATIO HELD CCNSTANT $',
48,3.8,4)
CALL HEADIN ('SOLIDITY, ROTATIONAL VELOCITY $',29,0.8,4)
CALL HEADIN ('ALLOWED TO VARY WITH RADIUS $',27,0.8,4)
CALL MESSAG ('SPEC RMR = $,100,7.4,6.5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAG ('CASE 2 $',100,8.7,6.0)
IF (K.NE.3) GC TC 12
CALL HEADIN ('SOLIDITY & ROTATIONAL VELOCITY HELD CCNSTANT $',
44,3.8,4)
CALL HEADIN ('CHORD,ADVANCE RATIO & TIP VELOCITY $',34,0.8,4)
CALL HEADIN ('ALLOWED TO VARY WITH RADIUS $',27,0.8,4)
CALL MESSAG ('SPEC RMR = $,100,7.4,6.5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAG ('CASE 3 $',100,8.7,6.0)

```

8

10

```

PERI5860 PERI5870 PERI5880 PERI5890 PERI5900 PERI5910 PERI5920 PERI5930 PERI5940 PERI5950 PERI5960 PERI5970 PERI5980 PERI5990 PERI6000 PERI6010 PERI6020 PERI6030 PERI6040 PERI6050 PERI6060 PERI6070 PERI6080 PERI6090 PERI6100 PERI6110 PERI6120 PERI6130 PERI6140 PERI6150 PERI6160 PERI6170 PERI6180 PERI6190 PERI6200 PERI6210 PERI6220 PERI6230 PERI6240 PERI6250 PERI6260 PERI6270 PERI6280 PERI6290 PERI6300 PERI6310 PERI6320
IF( K .NE. 4) GC TO 14
CALL HEADIN('SOLIDITY,ADVANCE RATIC & TIP VELOCITY HELD CONSTANT$',
*,51.0,8.4),
CALL HEADIN ('CHORD & ROTATIONAL VELCCITY$',27,0.8,4)
CALL HEADIN ('ALLOWED TO VARY WITH RADILCS$,27,0.8,4)
CALL MESSAG ('SPEC RMR =$,100,7.4,6.5)
CALL REALNO (RMR(163),2,'ABUT.',,ABLT.)
CALL MESSAG('CASE 4$,,100,8.7,6.0)
CALL ENCLPL(C)
RETURN
ENC
C***** SUBROUTINE TO CREATE GRAPH FOR MAXIMUM HOVER CEILING *****
C
C***** SUBROUPTINE GRAF5 *****
C
IMPLICIT REAL (A-H,L-Z)
COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
* MAXXZ,CNIN,DMAX,MINDB,MAXB,MINC,MAXC,MINT,MAX,TANS,TEXT(200),
* COMMON /W/ RCMMAX(400),CRCMAM(400),VFRAN(400),DVFRAN(400),
* HVALT(400),DHVAL(400),SEVALT(400),DSEVAL(400),FMERT(400),
* FMERT(400),VFEND(400),DVFEND(400)
CALL CCMPRS
CALL PAGE(11.0,8.5)
CALL PHYSOR(1.5,1.5)
CALL AREA2D (2.50,4.50)
CALL XNAME ('RADIUS OF MAIN ROTOR (FT)$,100)
CALL YNAME ('MAXIMUM HOVER CEILING (FT)$,100)
CALL GRAF (MINZ,'SCALE',MAXZ,'SCALE',MAX)
CALL CUFE (RMR,HVALT,324,0)
CALL DCT
CALL GRID (1,2)
CALL RESET('DOT.')
CALL ENDGR(O)
C-----
CALL PHYSOR(5.5,1.50)
CALL AREA2D (2.50,4.50)
CALL XNAME ('RADIUS OF MAIN ROTOR (% CHANGE)$,100)
CALL YNAME ('MAXIMUM HOVER CEILING (% CFANGE)$,100)
CALL GNAME ('CMIN','SCALE',DMAX,MINY,'SCALE',MAXY)
CALL CUFE (DRMR,DHCVL,324,0)
CALL DCT
CALL GRID (1,2)
CALL ENCLR(O)
C-----
CALL PHYSOR(1.5,1.5)

```



```

COMMON /W/ RCMAX(400), DRCMAX(400), VFRAN(400), DSEVAL(400), FDMERT(400),
* HCVALT(400), DHOVAL(400), SEVALT(400), DVFEND(400),
* DFMERT(400), VFEND(400), DVFEND(400),
CALL CCMPRS
CALL PACE(11,0,8.5)
CALL PFYSOR(1.5,1.5)
CALL AREA2D(2.50,4.50)
CALL XNAME(,RADIUS OF MAIN ROTOR (FT),$,100)
CALL YNAME(,MAXIMUM SERVICE CEILING (FT),$,100)
CALL GRAF(MINZ,SCALE,MAXZ,MIN,SCALE,MAX)
CALL CURVE(RMR,SEVALT,322,0)
CALL GCT
CALL GFIC(1,2)
CALL RESET(.DOT.)
CALL ENLGR(0)

C-----
CALL PFYSOR(5.5,1.5)
CALL AREA2D(2.50,4.50)
CALL XNAME(,RADIUS OF MAIN ROTOR (% CHANGE),$,100)
CALL YNAME(,MAXIMUM SERVICE CEILING (% CHANGE),$,100)
CALL GRAF(DMIN,SCALE,DMAX,MIN,SCALE,MAX)
CALL CURVE(DRMR,DSEVAL,324,0)
CALL GCT
CALL GRID(1,2)
CALL ENLGR(0)

C-----
CALL PFYSOR(1.5,1.5)
CALL AREA2D(6.50,4.70)
CALL HEADIN(,MAXIMUM SERVICE CEILING VERSUS RADIUS CHANGE$,
44,1.4,4)
* IF(K.NE.1) GOTO 8
CALL HEADIN(,CHORD & ROTATIONAL VELOCITY HELD CONSTANT$,41,
0.8,3)
* CALL HEADIN(,SOLIDITY, TIP VELOCITY & ADVANCE RATIO$,38,0.8,4)
CALL HEADIN(,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL MESSAGE(,SPEC RMR =$,100,7.4,6.5)
CALL REALNO(,RMR(163),2,ABUT,ABUT)
CALL MESSAGE(,CASE 1$,100,8.7,6.0)
IF(K.NE.2) GOTO 10
CALL HEADIN(,CHORD,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$,
48,0.8,4)
* CALL HEADIN(,SOLIDITY, ROTATIONAL VELOCITY$,29,0.8,4)
CALL HEADIN(,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL MESSAGE(,SPEC RMR =$,100,7.4,6.5)
CALL REALNO(,RMR(163),2,ABUT,ABUT)
CALL MESSAGE(,CASE 2$,100,8.7,6.0)
IF(K.NE.3) GOTO 12
CALL HEADIN(,SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$,

```

```

*      44,C.8,4) CHORD,ADVANCE RATIO & TIP VELCCITY$,34,0.8,4)
CALL HEADIN (,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL MESSAG (,SPEC RMR =$,100,7.4,6.5)
CALL REALNO (RMR(163),2,ABUT,ABUT)
CALL MESSAG (,CASE 3$,100,8.7,6.0)
IF (K.NE.4) GO TO 14
CALL HEADIN (,SOLIDITY,ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$,
,51,0.8,4)
*      CALL HEADIN (,CHORD & ROTATIONAL VELCCITY$,27,0.8,4)
CALL MESSAG (,ALLOWED TO VARY WITH RADIUS$,27,0.8,4)
CALL REALNO (RMR(163),2,ABUT,ABUT)
CALL MESSAG (,CASE 4$,100,8.7,6.0)
CALL ENDPL(C)
RETURN
ENC
12
14
C*****
C      SUBROUTINE TO CREATE GRAPH OF FIGURE OF MERIT VS RADIUS
C*****
C      SUBROUTINE GRAFLE
C      IMPLICIT REAL (A-H,L-Z)
C      COMMON /C/ RMR(400),I,J,K,DRMR(400),MINX,MAXX,MINY,MAXY,MINZ,
COMMON /W/ RCMAX(400),DRCMAX(400),VFRAN(400),DVFRAN(400),
*      HCVALT(400),DHOVAL(400),SEVALT(400),FSEVAL(400),
*      DFMER(400),VFEND(400),DVFEND(400)
CALL CCMPRS
CALL PAGE (11,0.8,5)
CALL PHYSOR (1.5,1.5)
CALL AREA2D (2.50,4.50)
CALL XNAME (,RADIUS OF MAIN ROTOR (FT)$,10C)
CALL YNAME (,FIGURE OF MERIT$,100)
CALL GRAF (MINZ,SCALE,MAXZ,SCALE,MAXB)
CALL CURVE (RMR,FMERT,324,0)
CALL DCT
CALL GRID (1,2)
CALL RESET (,DOT, )
CALL ENGR(C)
C-----
CALL PHYSOR (5.5,1.5)
CALL AREA2D (2.50,4.50)
CALL XNAME (,RADIUS OF MAIN ROTOR (2 CHANGE)$,100)
CALL YNAME (,FIGURE OF MERIT (% CHANGE)$,100)
CALL GRAF (CMIN,SCALE,DMAX,MINC,SCALE,MAXC)

```

PER17290
 PER17300
 PER17310
 PER17320
 PER17330
 PER17340
 PER17350
 PER17360
 PER17370
 PER17380
 PER17390
 PER17400
 PER17410
 PER17420
 PER17430
 PER17440
 PER17450
 PER17460
 PER17470
 PER17480
 PER17490
 PER17500
 PER17510
 PER17520
 PER17530
 PER17540
 PER17550
 PER17560
 PER17570
 PER17580
 PER17590
 PER17600
 PER17610
 PER17620
 PER17630
 PER17640
 PER17650
 PER17660
 PER17670
 PER17680
 PER17690
 PER17700
 PER17710
 PER17720
 PER17730
 PER17740
 PER17750
 PER17760

```

CALL CLFVE (DRMR,DFMERT,324,0)
CALL CCT
CALL GFIL (1,2)
CALL ENCR(0)
-----
CALL PHYSOR(1,5,1,5)
CALL AREA2D (6,50,4,70)
CALL HEACIN (,FIGURE CF MERIT VERSUS RADIUS CHANGE$,
36,1,4,4)
* IF (K .NE. 1) GO TO 8
CALL HEACIN (,CHORD & ROTATIONAL VELOCITY HELD CONSTANT$,41,
0,8,3)
* CALL HEACIN (,SOLIDITY, TIP VELOCITY & ADVANCE RATIO$,38,0,8,4)
CALL HEACIN (,ALLOWED TO VARY WITH RADIUS$,27,0,8,4)
CALL MESSAGE (,SPEC RMR =$,100,7,4,6,5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAGE (,CASE 1$,100,8,7,6,0)
IF (K .NE. 2) GO TO 10
CALL HEACIN (,CHCRC,TIP VELOCITY & ADVANCE RATIO HELD CONSTANT$,
4,0,8,4)
* CALL HEACIN (,SOLIDITY, ROTATIONAL VELOCITY$,29,0,8,4)
CALL HEACIN (,ALLOWED TO VARY WITH RADIUS$,27,0,8,4)
CALL MESSAGE (,SPEC RMR =$,100,7,4,6,5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAGE (,CASE 2$,100,8,7,6,0)
IF (K .NE. 3) GO TO 12
CALL HEACIN (,SOLIDITY & ROTATIONAL VELOCITY HELD CONSTANT$,
4,0,8,4)
* CALL HEACIN (,CHORD, ADVANCE RATIO & TIP VELOCITY$,34,0,8,4)
CALL HEACIN (,ALLOWED TO VARY WITH RADIUS$,27,0,8,4)
CALL MESSAGE (,SPEC RMR =$,100,7,4,6,5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAGE (,CASE 3$,100,8,7,6,0)
IF (K .NE. 4) GO TO 14
CALL HEACIN (,SOLIDITY, ADVANCE RATIO & TIP VELOCITY HELD CONSTANT$,
51,0,8,4)
* CALL HEACIN (,CHORD & ROTATIONAL VELOCITY$,27,0,8,4)
CALL HEACIN (,ALLOWED TO VARY WITH RADIUS$,27,0,8,4)
CALL MESSAGE (,SPEC RMR =$,100,7,4,6,5)
CALL REALNC (RMR(163),2,ABUT,ABUT)
CALL MESSAGE (,CASE 4$,100,8,7,6,0)
CALL ENCLPL(C)
RETURN
END
=====
C SUBROUTINE RDINT --- INTERACTIVELY READS AN INTEGER REPLY
C INTO A FORTRAN PROGRAM. IF THE USER INADVERTENTLY ENTERS AN IMPROPER
PER17770
PER17780
PER17790
PER17800
PER17810
PER17820
PER17830
PER17840
PER17850
PER17860
PER17870
PER17880
PER17890
PER17900
PER17910
PER17920
PER17930
PER17940
PER17950
PER17960
PER17970
PER17980
PER17990
PER18000
PER18010
PER18020
PER18030
PER18040
PER18050
PER18060
PER18070
PER18080
PER18090
PER18100
PER18110
PER18120
PER18130
PER18140
PER18150
PER18160
PER18170
PER18180
PER18190
PER18200
PER18210
PER18220
PER18230
PER18240

```

```

C DATA CHARACTER THE S/R ISSUES A WARNING AND ALLOWS A RECOVERY.
C=====
C SUBROUTINE RJOINT ( IANS)
C INTEGER CCUNT
C-----
C COUNT=C
C CONTINUE
C COUNT=CCUNT+1
C IF (COUNT.LT.3) GO TO 20
C WRITE (5,60)
C GO TO 50
C CONTINUE
C READ (5,*,END=40,ERR=40) IANS
C IF (IANS) 40,40,30
C CONTINUE
C RETURN
C REWIND 5
C WRITE (5,70)
C GO TO 10
C CONTINUE
C STOP
C-----
C 60 FORMAT (//,5X,49HPROGRAM TERMINATION - TWO IMPROPER DATA ENTRIES
C 1)
C 70 FORMAT (1X,56HWARNING: IMPROPER DATA ENTRY ENTER A POSITIVE INTE
C 1GER.)
C END

```

```

=PER18250
PER18260
PER18270
PER18280
PER18290
PER18300
PER18310
PER18320
PER18330
PER18340
PER18350
PER18360
PER18370
PER18380
PER18390
PER18400
PER18410
PER18420
PER18430
PER18440
PER18450
PER18460
PER18470
PER18480
PER18490
PER18500
PER18510
PER18520

```

AD-A151 488

PRELIMINARY HELICOPTER DESIGN DECISION MAKING BASED ON
FLIGHT PERFORMANCE FACTORS(U) NAVAL POSTGRADUATE SCHOOL
MONTEREY CA P V ADANCIK SEP 84

2/2

UNCLASSIFIED

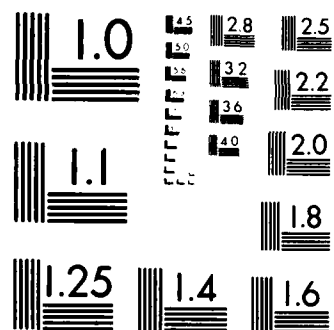
F/G 6/3

NL

END

FILMED

DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

LIST OF REFERENCES

1. Integrated Software Systems Corporation, DISSELA : Pocket Guide, Version 9.0, 1981.
2. Layton, Professor Donald M., Helicopter Performance, Naval Postgraduate School, Monterey California, 1983.
3. Layton, Professor Donald M., AE 4306 Design Design Manual, Naval Postgraduate School, Monterey California, 1983.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Army Aviation Systems Command Attn: Mr Crawford DRAV-GT 4300 Goodfellow Blvd St. Louis, Missouri 63120	1
3. Library, Code 0142 Naval Postgraduate School Monterey, California 93943	3
4. Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93943	1
5. Prof Donald M. Layton, Code 67-LN Department of Aeronautics Naval Postgraduate School Monterey, California 93943	5
6. CPT Patrick V. Adamcik c/o CPT Steve Kee 302 Lakeshore Road Enterprise, Alabama 36330	4

END

FILMED

4-85

DTIC