DESIGN OPTIMIZATION OF A SHAFT ON TWO BEARINGS

DESIGN OPTIMIZATION OF A SHAFT ON TWO BEARINGS

By

LAURENTIU POPA

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Master of Engineering

McMaster University
October 1978

MASTER OF ENGINEERING (1978)
(Design)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: Design Optimization of a Shaft on Two Bearings

AUTHOR: Laurentiu Popa, Electrical-Mechanical Engineer
(Politechnic Institute of Timisoara,
Romania)

SUPERVISOR: Professor J.N. Siddall

NUMBER OF PAGES: Vii, 83

ABSTRACT

The purpose of this project was to develop an aid to designing machinery shafts to find the optimum design for most shaft design applications met in design engineers' practices, to reduce the cost of the shafts.

The result was a computer package capable of handling the design optimization of step shafts supported on two bearings which will find the shaft design with minimum cost.

Additional benefits were obtained such as reducing engineering time and cost, increasing reliability of the design, etc.

ACKNOWLEDGMENTS

I wish to express appreciation to Prof. James N. Siddall for his assistance in my work, to the management of Orenstein and Koppell of Canada Ltd. who have supported part of this work and to my wife Dorothy for her dedicated typing.

TABLE OF CONTENTS

APPENDICES

LIST OF GRAPHS

LIST OF TABLES

1. INTRODUCTION

The design of a step shaft is a laborious process. Starting from assumed diameter sizes, the stresses along the shaft, deflections and the natural frequency would be determined for these diameters. This could be done by hand in which case a graphical method would probably be used to determine deflections and natural frequencies (see references 3, 5, 10 and 11), or it could be done by using computer packages as is the package "Stress" (25) available from McMaster University or "Multi-Span" (26) from Massachusetts Institute of Technology, in which case the shaft will be considered as a piece of structure. Considerable effort will be required from the user to provide the data concerning the shaft to be fed in these vast application packages.

The stresses determined by hand or by computer would be multiplied by stress concentration factors available from handbooks or manuals to obtain intensified stresses at shaft shoulders, keyways, holes and other stress raising irregularities on the shaft.

If the values obtained for stresses, deflections and natural frequency are not satisfactory, the process will be repeated for a different set of diameters. After a number of cycles, all the stresses would be below the allowable stress, the deflections below allowable deflection and natural frequency above allowable value, but it is unlikely that the weight of the shaft will be minimum. More cycles could be repeated at expensive engineering time to obtain a decreased weight or the process could be stopped at the risk of having an overdimensioned shaft.

The object of this project was to create a computer package applicable specifically to a step shaft on two bearings that will determine the optimum design of the shaft at low engineering cost. To the best knowledge of the authors of this package, there are no other packages available to treat the design optimization of a shaft. Compared with the design of a shaft by hand or by hand and some computer package help as described above, the use of this package will present the following advantages:

1. The weight of the shaft will likely be lower. How low the weight of a shaft designed by hand is depends on the ability of the designer. A number of trials done during the progress of this project have shown that the weight could be reduced by roughly 10% as an average. On a hoist shaft weighing 40,000 lbs., this would mean 4,000 lbs. which at a cost of about $1./lb. means a $4,000. savings.

2. The engineering time required to design a shaft with this package will be low. While by hand the time required will be between 8 to 24 hours, this package will only require 2 to 3 hours to prepare the input data and run the program. This would result in an average savings of about 14 hours which at a cost of $25./hr. means a $350. savings.

3. While a design done by hand could be subjected to errors, the result given by this package will be accurate and correct and reliability of the shaft might be greater.

The application of this package is limited to horizontal shafts on two bearings with or without cantilever extension subjected to concentrated and uniformly distributed loads and torsional moments. This covers most of the shaft designs met in practice. It could be developed to cover a larger range of shafts including multi-span shafts and shafts with other cases of loading. The work done so far will be useful for such a development.

The use of the package is especially attractive for shafts for heavy machinery such as mine hoists, naval machinery, steel mill machinery, presses, etc. where a reduction of the weight of the shaft results in substantial savings and for shafts for spacecraft machinery, aircraft machinery, nuclear reactor machinery etc. where the reliability and the weight of the shaft is important.

2. DESCRIPTION OF THE PROBLEM

2.1 GEOMETRICAL CONFIGURATION OF THE SHAFT

The shaft consists of a series of cylindrical sections of various diameter size. It is supported on two bearings and can have a cantilever extension. The bearings permit slopes of the shaft at the bearing point and one of them is free to translate along the axis of the shaft. (See a typical configuration shown in Fig. 1).

The location of the bearings and the location of the shaft shoulder and shoulder radius is known. The diameters of the shaft will be determined as the result of the optimization process with the exception of certain diameters that can be of predetermined sizes that will not be changed during the optimization (as it may be required by the bearing size, coupling size, standard parts mounted on the shaft, etc.).

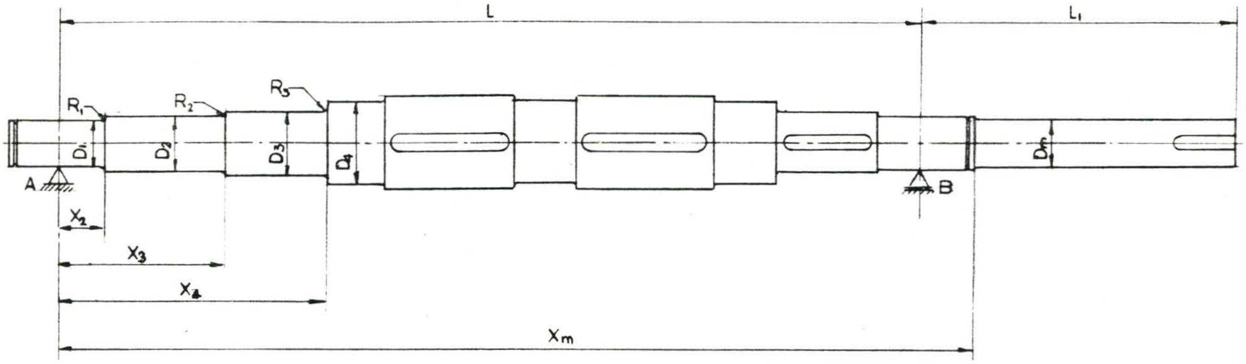

Fig. 1
Typical Configuration of the Shaft

The configuration of the shaft may be restricted by some diameters having to be greater than others for assembly considerations. A minimum shoulder may be required for example adjacent to bearings, a gear, coupling, etc.

The shaft may have keyways, grooves, holes, etc.

2.2 LOADING

The axis of the shaft is situated in the horizontal plane. The shaft can be subjected to any concentrated or uniformly distributed forces perpendicular to axis of shaft, projected in vertical and horizontal plane and any torsional moments. (See Fig. 2.) showing typical loading on the shaft.

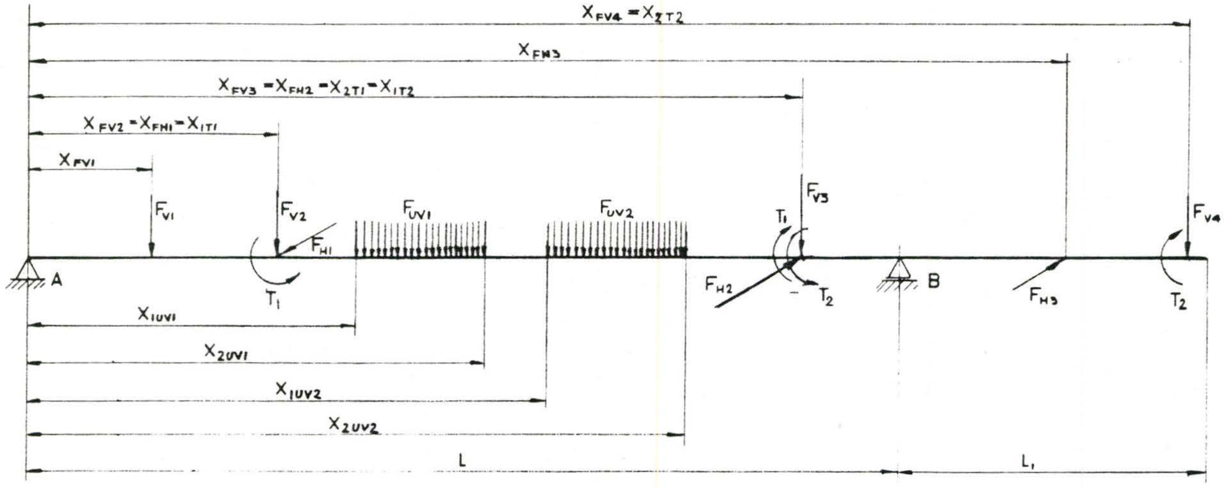

Fig. 2

Typical Loading on the Shaft

2.3 OPTIMIZATION PROBLEM

We wish to optimize the shaft of a geometrical configuration and loading as described above. Our optimization problem is to find the unknown diameter sizes of the shaft for which the weight of the shaft will be minimum and the following constraints will be satisfied:

- the stress at any point along the shaft will be below the allowable value;
- the maximum deflection will be less than an allowable deflection;
- the slope of the shaft in bearings will be below an allowable slope;
- the torsional deflection will be less than the allowable limit;
- the critical speed will be greater than a specified value;
- the shoulder size at specified points along the shaft will be greater than certain given values.

2.4 INDEPENDENT VARIABLES

Of all diameters of the shaft D_1, D_2 ... D_m, some are of predetermined size and cannot be changed through optimization. The unknown diameters form the independent variables d_1, d_2 ... d_n ($n \leqslant m$).

2.5 SPECIFICATIONS

The user will define the geometrical configuration of the shaft and the loading. He will also set the specifications to which the design of the shaft will have to conform. The references 7, 12, 13, 14, 15, 16, 17 and other manuals, codes and standards could be to users' assistance to set some of the following specifications required to be supplied:

1. allowable stress;
2. allowable bending deflection;
3. allowable slope in bearings;
4. allowable torsional deflection;
5. critical speed to be above a minimum value;
6. size and location of diameters that the user wishes to stay unchanged, diameters that will not be optimized;
7. minimum diameter increase or decrease and location for cases in which the shoulder cannot be below a given value;
8. stress concentration factor for stress raising irregularities other than shaft shoulders. For shaft shoulders, the stress concentration factor is selected internally from tables extracted from (6).

2.6 OBJECTIVE FUNCTION

We wish to find the design of the shaft that can be manufactured at a minimum cost. The cost of the shaft could depend on the size of the shaft, on geometrical configuration and could vary from one manufacturer to another.

By decreasing the weight of the shaft, the cost will decrease and for a shaft of defined geometrical configuration manufactured by a certain company, the design of minimum weight will cost the least.

The objective function of our optimization problem is the total weight of the shaft which is to be minimized.

$$W(d_1, d_2 \dots d_n) = \sum_{i=2}^{m} \gamma \frac{\pi D_{i-1}^2}{4}(X_i - X_{i-1})$$

$$= \sum_{j=2}^{n} \gamma \frac{\pi d_{j-1}^2}{4}(X_j - X_{j-1}) + \sum_{k=n}^{m} \gamma \frac{\pi D_{k-1}^2}{4}(X_k - X_{k-1})$$

Where:

D_i, d_i are shaft diameters as described in 2.4;

W $(d_i, d_2 \dots d_n)$ is the weight of the shaft;

X_i is the abscisa at the point where the shaft section of a diameter d_i or D_i starts;

γ is the specific weight for steel.

2.7 CONSTRAINT FUNCTIONS

Our optimization problem has inequality constraints only:

1. The maximum stress on the shaft σ_{cmax} shall be less than the allowable stress σ_{al}. Our first constraint function is:

$$\Phi_1 = \sigma_{al} - \sigma_{cmax} \geqslant 0 \tag{2.2}$$

2. The maximum deflection δ_{max} shall be less than the allowable deflection δ_{al}

$$\Phi_2 = \delta_{al} - \delta_{max} \geqslant 0 \tag{2.3}$$

3. The slope of the shaft in bearings α_A and α_B shall be less than the allowable limit α_{al}

$$\begin{aligned} \Phi_3 &= \alpha_{al} - \alpha_A \geqslant 0 \\ \Phi_4 &= \alpha_{al} - \alpha_B \geqslant 0 \end{aligned} \tag{2.4}$$

4. The maximum rotational deflection φ_{max} shall be less than the allowable torsional deflection φ_{al}

$$\Phi_5 = \varphi_{al} - \varphi_{max} \geqslant 0 \tag{2.5}$$

5. The critical speed of the shaft RPM_{CR} shall be greater than the allowable critical speed RPM_{al}

$$\Phi_6 = RPM_{CR} - RPM_{al} \geqslant 0 \tag{2.6}$$

6. The increase or decrease in diameter size in particular points $d_i - d_{i-1}$ shall be greater than a minimum value Δd_i

$$\Phi_j = (d_i - d_{i-1}) - \Delta d_i \geqslant 0 \qquad j = 7, 8 \ldots p \tag{2.7}$$

for an increase in diameter, or

$$\Phi_j = (d_{i-1} - d_i) - \Delta d_i \geqslant 0 \qquad (2.8)$$

for decrease

p is the number of inequality constraints

The increase or decrease in diameter size is chosen depending whether the starting diameters in that point are increasing or decreasing.

In the above expressions $\sigma_{al}, \tau_{al}, \alpha_{al}, \varphi_{al}, RPM_{al}$ and Δd_i are values specified by user while $\sigma_{cmax}, \tau_{max}, \alpha_A, \alpha_B, \varphi_{max}$, and RPM_{cR} are calculated as shown in § 5.

3. OPTIMIZATION METHOD

To minimize the objective function $W(d_1, d_2 \dots d_n)$ subjected to the constraints $\Phi_j(d_1, d_2 \dots d_n) \geqslant 0$, $j = 1, 2 \dots p$ as defined at 2.6 and 2.7, use is made of a direct search method employing the subroutine "SEEK3" from "OPTISEP" package (Reference 2) developed by Prof. James N. Siddall from McMaster University, Hamilton. The method is described in Ref. 1, 2, 20 & 21. With this method, an artificial, unconstrained objective function is defined as follows:

$$P(d_1, d_2 \dots d_n, r) = W(d_1, d_2 \dots d_n) + r \sum_{j=1}^{p} \frac{1}{\Phi_j(d_1, d_2 \dots d_n)} + \sum_{k=1}^{p'} \frac{\psi_k(d_1, d_2 \dots d_n)^2}{\sqrt{r}} \qquad (3.1)$$

In this function ψ_k $(d_1, d_2 \dots {}_n)$ would be equality constraint function. Since we do not have any equality constraints, the last term of the function is not used and the function is reduced to

$$P(d_1, d_2, \ldots d_n, r) = W(d_1, d_2, \ldots d_n) + r \sum_{j=1}^{k} \frac{1}{\Phi_j(d_1, d_2 \ldots d_n)}$$

r is a parameter that is 1.0 at the beginning. With this value for r, a direct search method is used to minimize $P(d_1, d_2 \ldots d_n, r)$. Starting from an initial base point with $d_1, d_2 \ldots d_n$ having starting values given by user, an exploration search is made. One of the variables is increased a small amount; and if the function P is improved, the new value of the variable is retained. If the move does not improve P, a negative step is tried. If this also fails, then the variable takes its base point value. The process is repeated for each variable and if no improvement of P is obtained, the step lengths are halved and the search is repeated until the step lengths are smaller than a preset small value when an optimum is assumed.

If this search yields a smaller value of P, then a new base point is established and a pattern move equal to the vector joining the original and the new base point is attempted. If the search is successful, the new search starts from that point, otherwise starts from the previous point. The process is repeated until a preset number of cycles have been exceeded or until the step lengths are less than the preset value.

After this cycle, the parameter r is reduced and the artificial function is minimized again. The process continues until no improvement in the function $W(d_1, d_2 \ldots d_n)$ can be reached when it is assumed that the optimum solution has been found.

The subroutine has shown to be powerful, reaching the optimum solution after a few minimizations of the artificial objective function.

No feasible starting point is required. A feasible solution is desirable to reduce the computer time and cost of the run.

A typical cost to run the program for a shaft of average complexity would be between \$10.-\$50. at a computer cost of \$416/hr.

The subroutine "SEEK1" (Reference 1, 2 and 21) that follows the same direct search method but combined with a random search has also been tried. The results have not been satisfactory, the subroutine tending to hang up in constraints. If a sufficient number of random shots are tried, the subroutine would eventually converge but slowly, using more computer time than "SEEK3".

The subroutine "SIMPLEX" (Reference 1,2,22,23 and 24) that uses the same artificial function as "SEEK3" but gropes towards the optimum by contracting and expanding an n dimensional space, called SIMPLEX, has been tried but without success. It hangs up in constraints. An optimum solution has not been found at a reasonable computer time and expense.

4. OUTPUT INFORMATION

4.1 OUTPUT FORMAT

A typical output format is shown in the sample run 8 from Appendix A "User's Manual".

4.2 INFORMATION PROVIDED

The following information is printed out:

1. Input data as supplied by user containing:
    - 1.1 length of first span in inches;
    - 1.2 length of the cantilever span in inches;
    - 1.3 concentrated weights attached to the shaft in lbs. provided separately for calculation of natural frequency. Location in inches measured from the first bearing and the weight in lbs. is shown;
    - 1.4 concentrated vertical forces in lbs. and location in inches;
    - 1.5 concentrated horizontal forces in lbs. and location in inches;
    - 1.6 uniformly distributed weights attached to the shaft in lbs./inches and location of the beginning and the end of the force in inches;
    - 1.7 vertical uniformly distributed forces in lbs./inches and location in inches;
    - 1.8 horizontal uniformly distributed forces in lbs./inches and location in inches;
    - 1.9 torsional moments in lbs./inches and location of the beginning and the end of the portion of the shaft between which it is applied in inches;
    - 1.10 diameter size and the location of the point of the shaft where that diameter size starts in inches;
    - 1.11 stress concentration factor for stress raising irregularities and location in inches;
    - 1.12 shoulder radii and location in inches;
    - 1.13 the table 'Dimensions and Loading' systemizing the above information;
    - 1.14 allowable stress in lbs./sq. inches;

1.15 allowable bending deflection in inches;

1.16 allowable slope in bearings in radians;

1.17 allowable torsional deflection in radians;

1.18 minimum critical speed in RPM;

1.19 diameters of predetermined size and point of beginning in inches;

1.20 diameter increase or decrease and location in inches;

2. Information required by optimization subroutine "SEEK3". No user input is required to supply this information. The following data is printed out:

   2.1 N - number of independent variables X(I) that is the number of shaft diameter sizes d_i to be determined;

   2.2 IPRINT = 1 indicates that after each optimization cycle of a constant parameter R the independent variables X and the objective function $U = W(d_1, d_2 \ldots d_n)$ will be printed out;

   2.3 IDATA = 1 indicates that the input data for SEEK 3 is printed out;

   2.4 NCONS is the number of inequality constraints

   2.5 NEQUS is the number of equality constraints set at 0;

   2.6 RMIN(I) is the estimated lower bound of range of X(I), set at

$$\text{RMIN(I)} = .9\ \text{XSTART(I)} \tag{4.1}$$

   2.7 RMAX(I) is the estimated upper bound of range of X(I), set at

$$\text{RMAX(I)} = 1.1\ \text{XSTART(I)} \tag{4.2}$$

2.8 XSTART(I) is the starting value of X(I) = d_i as given by user;

2.9 G is the step size fraction used to establish the minimum step size;

2.10 M = 20 is the maximum number of cycles permitted;

2.11 R is the penalty multiplier used in SEEK3 (See ch. 3);

2.12 Reduce = .05 is the reduction factor for r after each minimization;

3. Results after each minimization cycle of U including values for R, U and X(I);

4. Optimum solution if found including U, X(I) and PHI(I) = Φ_i

5. 'Output Data' concerning the optimum design of the shaft, as follows:

   5.1 The vertical, horizontal and the geometrical resultant of the reactions in the two bearings in lbs. and the location of the bearings in inches;

   5.2 The vertical, horizontal and geometrical resultant of the bending moment and torsional moment in lbs. ins. at all points on the shaft specified by user in input data;

   5.3 Bending deflections in inches, slopes and rotations of the axis of the shaft in radians in all specified points along the shaft, in vertical and horizontal plane and the geometrical resultant;

   5.4 Vertical, horizontal and the resultant of bending stresses, torsion stresses, combined stresses and intensified stresses as well as the stress intensification factor for bending and torsion in all specified points along the shaft;

5.5 The weight of the shaft in lbs.;

5.6 Angular frequency of dundamental mode of vibrations in rad./sec. and the critical speed in RPM;

4.3 PLOT OF SHAFT

The optimum design of the shaft is plotted on a Benson-Lenner Plotter. Depending on the size of the shaft, a suitable scale is chosen so that the plot of the lengths of the shaft is between 12½ - 31¼ inches for shaft shorter than 500 inches, from the following range of scales commonly used in drafting:

3/4 in. = 1 ft.
1 in. = 1 ft.
1½ in. = 1 ft.
3 in. = 1 ft.
Half Full Scale
Full Scale
Double Scale

5. ENGINEERING MODELLING

5.1 BENDING DEFLECTIONS AND STRESSES

The forces applied on the shaft are kept in three distinct groups that are treated separately:

1. gravitational forces caused by masses attached to the shaft;
2. vertical exterior forces;
3. horizontal forces.

They can initially be concentrated or uniformly distributed forces. The shaft's own weight is added to uniformly distributed gravitational forces. Within each of the three groups of forces, the uniformly distributed forces are replaced by a number of concentrated forces equal in size and spaced at equal intervals.

Each force is taken individually and reactions in bearings and bending moments are calculated for the force acting alone.

To calculate deflection caused by each force, use is made of the area-moment method (Reference 3) as follows:

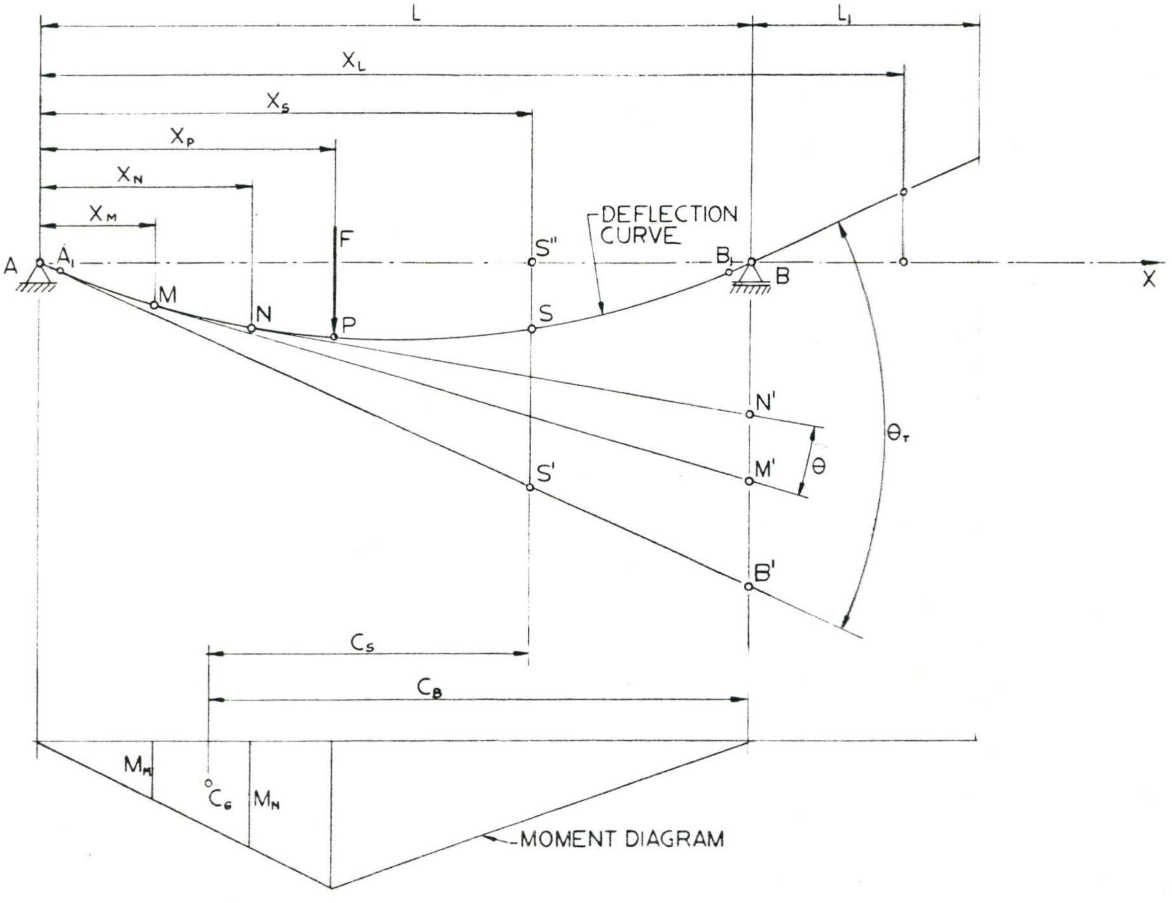

Fig. 3
AREA-MOMENT METHOD

Take first a force F situated in the span AB. For each cylindrical portion of the shaft, MN, between two adjacent discrete points specified in input data, the moment area is calculated:

$$Am = \frac{M_m + M_n}{2} (X_n - X_m) \qquad (5.1)$$

M_m and M_n being bending moments in M and N caused by the force F.

The angle θ between the tangent to the deflection curve at points M and N is

$$\theta = \frac{Am}{EIz} \qquad (5.2)$$

E is modulus of elasticity for shafts material

Iz is the moment of inertia of shafts section between M and N

The total angle θ_T between tangents to the deflection curve at the bearings is

$$\theta_T = \sum_{i=1}^{n_1} \frac{Am_i}{E\,Iz_i} \qquad (5.3)$$

n_1 is the number of discrete points between bearings.

In a point S

$$\theta_S = \sum_{i=1}^{n_S} \frac{Am_i}{E\,Iz_i} \qquad (5.4)$$

n_S is the number of discrete points from bearing A to point S.

Consider the distance $\overline{BB}$' between bearing B and the tangent to the deflection curve in A measured along the vertical. The contribution made by the deformation of the length MN to this distance is $\overline{M'N'}$.

$$\overline{M'N'} = \theta \times C_B = \frac{Am}{E\,Iz} \times C_B \tag{5.5}$$

Where C_B is the distance between the centre of gravity of moment area Am and bearing B. Then the total distance $\overline{BB'}$ is

$$\overline{BB'} = \sum_{i=1}^{n} \frac{Am_i}{E\,Iz_i} \times C_{B_i} \tag{5.6}$$

For any point S on the shaft

$$SS' = \sum_{i=1}^{n_s} \frac{Am_i}{E\,Iz_i} \times C_{S_i} \tag{5.7}$$

The deflection in point S is:

$$\delta_S = \overline{SS''} = \overline{S'S''} - \overline{SS'} = \overline{BB'} \times \frac{X_S}{L} - \overline{SS'} \tag{5.8}$$

Take points A_1 and B_1 close to bearing A and B

$$\overline{AA_1} = \overline{BB_1} = \frac{L}{10\,000.} \tag{5.9}$$

The deflection in A_1 and B_1 is δ_{A_1} and δ_{B_1} calculated as above.

The slope in bearing A and B is:

$$\alpha_A = \frac{\delta_{A_1}}{\overline{AA_1}} \tag{5.10}$$

$$\alpha_B = \frac{\delta_{B_1}}{BB_1} \tag{5.11}$$

The slope in the point S is:

$$\alpha_S = \theta_A - \theta_S \tag{5.12}$$

The deflection in a point T situated in the cantilever span is:

$$\delta_T = (X_T - L) \times \alpha_B \tag{5.13}$$

For a force situated in the cantilever span, the deflections and slopes are calculated in a similar manner.

Applying the superposition principle, the reactions in bearings, moments, deflections and slopes in each point on the shaft are calculated by totalizing the reactions, moments, deflections and slopes caused by each of the gravitational forces, exterior vertical forces and horizontal forces.

The vertical reactions in bearings, moments, deflections and slopes all along the shaft are calculated by adding the effects of gravitational and vertical exterior forces.

The normal stresses produced by bending are calculated then in the vertical and horizontal planes.

$$\sigma_{v_i} = \frac{M_{v_i}}{I_{z_i}} \times \frac{D_i}{2} \tag{5.14}$$

$$\sigma_{H_i} = \frac{M_{H_i}}{I_{z_i}} \times \frac{D_i}{2} \qquad (5.15)$$

The geometrical resultant of the reacions in bearings, moments, deflections, slopes and stresses are computed next:

$$R_A = \sqrt{R_{AV}^2 + R_{AH}^2} \qquad (5.16)$$

$$R_B = \sqrt{R_{BV}^2 + R_{AH}^2} \qquad (5.17)$$

$$M_i = \sqrt{M_{V_i}^2 + M_{H_i}^2} \qquad (5.18)$$

$$\delta_i = \sqrt{\delta_{V_i}^2 + \delta_{H_i}^2} \qquad (5.19)$$

$$\alpha_i = \sqrt{\alpha_{V_i}^2 + \alpha_{H_i}^2} \qquad (5.20)$$

5.2 TORSIONAL DEFLECTIONS AND STRESSES

The torsional deflection at any point on the shaft is calculated as follows:

$$\varphi_k = \sum_{i=1}^{k} \frac{T_i}{G I_{p_i}} (X_i - X_{i-1}) \qquad (5.21)$$

where

T_i is the torsion moment in section i;

G is the modulus of shear;

I_{p_i} is the polar moment of inertia of section i;

X_{i-1}, X_i are the co-ordinates of the beginning and end of section i;

The shearing stress is calculated:

$$\tau_i = \frac{T_i}{I_{Pi}} \times \frac{D_i}{2} \tag{5.22}$$

5.3 COMBINED STRESSES

The combined stresses are calculated according to the maximum strain theory of elastic failure suggested by St. Venant (References 3, 4, 5, 18 and 19) commonly accepted for mechanical shafting::

$$\sigma_{ci} = .35\,\sigma_i + .65\sqrt{\sigma_i^2 + 4\tau_i^2} \tag{5.23}$$

5.4 INTENSIFIED STRESSES

In any point on the shaft where there is a stress concentration due to shaft shoulders, keyways, holes, grooves, etc., the intensified bending or shear stress is calculated by multiplying the stresses calculated as shown above by a stress intensification factor.

For shaft shoulders, the stress intensification factor is calculated within the program. The intensification factor for bending stresses is selected from Table 1 and the intensification factor for torsional shear from Table 2 below by linear interpolation. The tables are reproduced from Ref. (6) pages 386 and 388.

TABLE 5.1 - BENDING STRESS CONCENTRATION FACTORS FOR SHAFT SHOULDERS

$\frac{h}{r}$ \ $\frac{r}{d}$	.05	.10	.20	.27	.50	1.0
.5	1.61	1.49	1.39	1.34	1.22	1.07
1.0	1.91	1.70	1.48	1.38	1.22	1.08
1.5	2.00	1.73	1.50	1.39	1.23	1.08
2.0		1.74	1.52	1.39	1.23	1.09
3.5		1.76	1.54	1.40	1.23	1.10

TABLE 5.2 - TORSION STRESS CONCENTRATION FACTOR FOR SHAFT SHOULDERS

$\frac{D}{d} \backslash \frac{r}{d}$	.005	.01	.02	.03	.04	.06	.08	.10	.12
2.00		3.0	2.25	2.00	1.82	1.65	1.51	1.44	1.39
1.33		2.7	2.16	1.91	1.76	1.60	1.48	1.40	1.35
1.20	3.0	2.5	2.00	1.75	1.62	1.50	1.40	1.34	1.30
1.09	2.20	1.88	1.53	1.40	1.30	1.20	1.16	1.15	1.15

In these tables;

r = shoulder radius

d = the smaller diameter size

D = the greater diameter size

h = (D - d)/2

For other kinds of irregularities on the shaft, the stress intensification factors should be supplied in input data. They could be selected by the user from the same Ref. (6).

5.5 CRITICAL SPEED

The angular frequency of the fundamental mode of transverse vibrations caused by weights attached to the shaft is calculated by applying the energy method suggested by Rayleigh (Refs. 8, 9 and 10).

$$p = \sqrt{\frac{g \sum_{j=1}^{n} W_j y_j}{\sum_{j=1}^{n} W_j y_j^2}} \qquad (5.24)$$

where:

p = angular frequency in rad/sec

g = gravitational acceleration in in/sec^2

W_j = weight attached to the shaft in a point of co-ordinate x_j

y_j = deflection caused by the weights attached to the shaft in a point of co-ordinate x_j

The shaft's own weight is calculated in the program and added to weights attached to the shaft as a number of concentrated forces distributed along the shaft.

The critical speed of the shaft is then calculated:

$$RPM_{cr} = \frac{30p}{\pi} \qquad (5.25)$$

where RPM_{cr} = critical speed in revolutions/minute

6. CONCLUSIONS

The computer package developed by this work is capable of bringing a considerable support to the design of a vast majority of shafts the design engineer would meet in his practice. The engineer's capability will still play the main role in the design. The engineer will establish the geometrical configuration and the loading and will specify allowable stresses, deflections, etc., depending on application, available space, reliability required and other factors.

The main objectives achieved by this package are:

1. The design of the shaft obtained by the package, complying with user specifications, is the optimum design with minimum weight and cost.

2. The package is specifically directed to solve the specific problem of designing a shaft. The use of a large purpose stress analysis package will not only fail to give the user the optimum design but will require user's involvement in areas not relevant to his problem while missing design aspects specific to the shaft as is the stress raisers, minimum shoulder required, etc.
3. User's input is practical and simple. It is limited to a minimum of data determining the design. No theoretical sophistication is required to prepare the input data or to interpret the results printed out.
4. The engineering time required to design a shaft with help of this package is low.
5. The computer time and expense is low.
6. The results are accurate and correct.

The applicability of the package is limited to the design of horizontal step shaft on two bearings with concentrated or uniformly distributed forces and torsional moments.

This would cover most of designer's shaft design problems. The following extensions to this work might be though useful to cover a larger area of applicability and to increase the versatility of the program in other aspects:

1. Design of shafts on more than two bearings;
2. Design of inclined or vertical shafts;
3. Larger loading case coverage such as:
    - 3.1 Axial loads;

   3.2 Moments in planes containing the axis of the shaft (only torsional moments, perpendicular to axis of shaft are handled in present stage);

   3.3 Variable loading along the shaft or on portions of the shaft.

4. Design of shafts with a number of alternatives concerning the loading. The same shaft might at different times be subjected to different loading.

5. Design of shafts that could have sections varying continuously as might be required by application or to enable further reduction of weight.

6. Calculation of natural frequency for rotational vibrations, useful especially for long shafts on several bearings.

7. Interpret user input less restricted to following a format pattern.

8. Adapt package for time sharing system.

7. REFERENCES

1. Siddall, J.N., Analitycal Decision Making in Engineering Design, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1972
2. Siddall, J.N., 'Optisep' Designers' Optimization Subroutines McMaster University, Hamilton, Ontario, 1970
3. Timoshenko, S., Strength of Materials, Huntington, New York, 1976
4. Boyd, J.E., Strength of Materials, McGraw Hill, New York, 1952
5. Duggan, T.V., Stress Analysis and Vibrations of Elastic Bodies, American Elsevier Pub. Co., New York, 1975
6. Roark R.J., Formulas for Stress and Strain, McGraw Hill, New York, 1971
7. Oberg, E., Machinery's Handbook, Industrial Press, Inc., New York, 1972
8. Den Hartog, J.D., Mechanical Vibrations, McGraw Hill, New York, 1976
9. Diementberg, F.M. Flexural Vibrations of Rotating Shafts, Butterworths', London, 1961
10. Timoshenko, S. Vibrations Problems in Engineering, Willey, New York, 1974
11. Hopkins, R.B. Design Analysis of Shafts and beams, McGraw Hill, New York, 1970
12. The Mining Act of Ontario, Part IX and Part XI, J.C. Thatcher, Toronto, Ontario, 1971
13. CSA B167-1964 General Purpose Overhead Travelling Cranes, Canadian Standard Association, Ottawa, Ontario, 1974
14. CSA Z185-1975 Safety Code for Personnel Hoists, Canadian Standard Association, Ottawa, Ontario, 1975
15. CSA B44-1975 Safety Code for Elevators, Canadian Standard Association, Ottawa, Ontario, 1975
16. ASA B17C Code for Design of Transmission Shafting, American Standard Association
17. Mantell, C.L., Engineering Materials Handbook, McGraw Hill,

New York, 1958

18. Fairman, S. & Cutshall C., Mechanics of Materials, Willey, New York, 1953

19. Trathem, H., Statics and Strength of Materials, Willey, New York, 1954

20. Fiacco, A.V., and McCormick, G.P., Nonlinear Programming: Sequential Unconstrained Minimization Techniques, Willey, New York, 1968

21. Hooke, R. and Jeeves, T.A., Direct Search Solution of Numerical & Statistical Problems, J. Assoc. Comput. Mach., Vol 8, 1961, pp. 212-229

22. Garvin, W.W., Introduction to Linear Programming, McGraw Hill, New York, 1960

23. Gass, S.I., Linear Programming, McGraw Hill, New York, 1958

24. Llewellyn, R.W., Linear Programming, Holt, New York, 1964

25. Fenves, S.J., Logcher, R.D., Stress Reference Manual, The Massachusetts Institute of Technology Press, Cambridge, Mass., 1964

26. 'Multi-Span', Version 1.2, Massachusetts Institute of Technology, Cambridge, Mass.

APPENDIX A

PROGRAM SHAFT I

DESIGN OPTIMIZATION OF A SHAFT ON TWO BEARINGS

PROGRAM LISTING

```
      PROGRAM SHAFT1 (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C
C     DESIGN OPTIMIZATION OF A SHAFT ON TWO BEARINGS
C     **********************************************
C
C
C     THE PROGRAM DETERMINES THE OPTIMUM DESIGN OF A SHAFT ON TWO
C     BEARINGS SUBJECTED TO EXTERIOR FORCES AND TORSIONAL MOMENTS FOR
C     THE MINIMUM WEIGHT
C
      DIMENSION
     *RMAX(50),RMIN(50),X(50),XSTRT(50),PHI(20),PSI(1),W1(50),W2(50),
     *W3(50),W4(50),RD(50),WORK1(200),WORK2(200),WORK3(200),WORK4(200),
     *WORKK(200),WORKL(200),WORKM(200),WORKN(200),WORKP(200),
     *XA(50,51),XJ(50),XH(50),XS(50),XL(50),XO(50),XP(50),XE(50),XC(50),
     *STEP(50),FUN(51)
C
      REAL L,L1,L2
C
      COMMON/AA/ ROVDB(8),HOVRB(7),CB(7,8),ROVDT(11),DOVDT(6),CT(6,11),
     *FCT1(20),FCT2(20)
      COMMON/BB/ RA,RB,                 XR(200),X2(200),D2(200),PV2(200)
     *,BMV(200),BMH(200),BM(200),DFLV(200),DFLH(200),SLV(200),SLH(200),
     *SL(200),STRV(200),STRH(200),STR(200),SCB(200),SCT(200),SCF2(200),
     *RAD2(200),STRINT(200),TM2(200),TETA(200),TAU(200),TAUINT(200),
     *STR1(200),SIGMA1(200),PH2(200),DFL(200),PW2(200),BMW(200),
     *DFLW(200),SLW(200),STRW(200),
     *WORKA(200),WORKB(200),WORKC(200),WORKD(200),WORKE(200),WORKF(200),
     *WORKG(200),WORKH(200),WORKI(200)
      COMMON/DD/ E,G,N00,    A,B,N1,N2,N3
      COMMON/EE/XDIA(50),DIA(50),ND
      COMMON/FF/ STRAL,DFLAL,SLAL,TETAAL,RPMMIN,XDI(20),DI(20),NDI
      COMMON/GG/ XDF(20),DF(20),NDF,WORKS(50)
C
C     MODULUS OF ELASTICITY
      E=30000000.
      G=11500000.
      N00=200
      W=1.0
C
            CALL LOAD  (N00,ND,N1,N2,N3,A,B, W,XDIA,DIA,XR,
     *    X2,PW2,PV2,PH2,TM2,D2,SCF2,RAD2,RD,
     *WORKA,WORKB,WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,WORKI,WORKK,WORKL,
     *WORKM,WORKN,WORKP,WORK1,WORK2,WORK3,WORK4)
C
C     READ AND PRINT OUT THE CONSTRAINTS
C
C     ALLOWABLE STRESS
      READ(5,100) STRAL
      WRITE(6,101) STRAL
C
C     ALLOWABLE DEFLECTION
      READ(5,102) DFLAL
      WRITE(6,103) DFLAL
C
C     ALLOWABLE SLOPE IN BEARINGS
      READ(5,102) SLAL
      WRITE(6,104) SLAL
C
C     ALLOWABLE TORSIONAL DEFLECTION
      READ(5,102) TETAAL
      WRITE(6,105) TETAAL
C
C     MINIMUM CRITICAL SPEED
      READ(5,100) RPMMIN
      WRITE(6,106) RPMMIN
C
C     CONSTRAINTS DEFINING SHAFT DIAMETERS
C     DIAMETERS OF FIXED SIZE
      READ(5,107) NDF
      WRITE(6,109)
      DO 1 I=1,NDF
      READ(5,108) XDF(I),DF(I)
```

```
      WRITE(6,110) XDF(I),DF(I)
    1 CONTINUE
C
C     MINIMUM DIAMETER INCREASE
      READ(5,107) NDI
      WRITE(6,112)
      DO 2 I=1,NDI
      READ(5,108) XDI(I),DI(I)
      WRITE(6,110) XDI(I),DI(I)
    2 CONTINUE
C
C     THE PROGRAM USES THE SUBROUTINE SEEK3 FOR OPTIMIZATION
C
      IPRINT=1
      IDATA=1
      NEQUS=0
      MAXM=20
C
      DO 41 I=1,50
      XSTRT(I)=0.0
      RMAX(I)=0.0
      RMIN(I)=0.0
      X(I)=0.0
   41 CONTINUE
C
      N=0
      NND=ND-1
      DO 4 I=1,NND
      N=N+1
      XSTRT(N)=DIA(I)
      DO 3 J=1,NDF
      IF(XDIA(I).EQ.XDF(J)) N=N-1
    3 CONTINUE
      IF(I.EQ.1) GO TO 4
      IF(DIA(I).EQ.DIA(I-1)) N=N-1
    4 CONTINUE
C
      DO 5 I=1,N
      RMAX(I)=1.1*XSTRT(I)
      RMIN(I)=0.9*XSTRT(I)
    5 CONTINUE
C
      NCONS=6+NDI
C
      F=.02
      H=.05
      INDEX=1
      R=1.
      REDUCE=.05
      CALL SEEK3(N,RMAX,RMIN,NCONS,NEQUS,XSTRT,F,H,R,REDUCE,MAXM,INDEX,
     *IPRINT,IDATA,U,X,PHI,PSI,NVIOL,WORK1,WORK2,WORK3,WORK4)
      CALL ANSWER(U,X,PHI,PSI,N,NCONS,NEQUS)
C
      DO 70 I=1,ND
      WORKS(I)=DIA(I)
   70 CONTINUE
C
      NND=ND-1
      K=0
      DO 10 I=1,NND
      K=K+1
      DIA(I)=X(K)
      DO 9 J=1,NDF
      IF(XDIA(I).EQ.XDF(J)) GO TO 8
      GO TO 9
    8 DIA(I)=DF(J)
      K=K-1
    9 CONTINUE
      IF(I.EQ.1) GO TO 10
      IF(WORKS(I).EQ.WORKS(I-1)) GO TO 90
      GO TO 10
   90 DIA(I)=DIA(I-1)
      K=K-1
   10 CONTINUE
```

```
      DIA(ND)=DIA(ND-1)                                                  MAI1490
C                                                                        MAI1500
      CALL       STRESS(E,G,N00,ND,N1,N2,N3,A,B, RA,RB,XDIA,DIA,XR,X2,   MAI1510
     *D2,PW2,PV2,PH2,BMW,BMV,BMH,BM,DFLW,DFLV,DFLH,DFL,SLW,SLV,SLH,SL,   MAI1520
     *STRW,STRV,STRH,STR,RPMCR,SCB,SCT,SCF2,RAD2,STRINT,TM2,TETA,TAU,    MAI1530
     *TAUINT,STR1,SIGMA1,                                                MAI1540
     *      WORKA,WORKB,WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,WORKI,1)     MAI1550
C                                                                        MAI1560
      CALL SHAFTP(XDIA,DIA,RD,  A,ND,WORK1,WORK2,WORK3,WORK4)            MAI1570
C                                                                        MAI1580
      STOP                                                               MAI1590
C                                                                        MAI1600
  100 FORMAT(F10.0)                                                      MAI1610
  101 FORMAT(1H1,/////,                                                  MAI1620
     *10X,40HALLOWABLE STRESS                        ,F10.0,8H LB/SQIN,/ MAI1630
     *//)                                                                MAI1640
  102 FORMAT(F10.5)                                                      MAI1650
  103 FORMAT(10X,40HALLOWABLE BENDING DEFLECTION            ,F10.5,3H IN MAI1660
     *,///)                                                              MAI1670
  104 FORMAT(10X,40HALLOWABLE SLOPE IN BEARINGS             ,F10.5,4H RA MAI1680
     *D,///)                                                             MAI1690
  105 FORMAT(10X,40HALLOWABLE TORSIONAL DEFLECTION          ,F10.5,4H RA MAI1700
     *D,///)                                                             MAI1710
  106 FORMAT(10X,40HMINIMUM CRITICAL SPEED                  ,F10.0,4H RP MAI1720
     *M,///)                                                             MAI1730
  107 FORMAT(I10)                                                        MAI1740
  108 FORMAT(2F10.3)                                                     MAI1750
  109 FORMAT(///,10X,23HDIAMETERS OF FIXED SIZE ,//,10X,20H      X       MAI1760
     *   DIA ,//)                                                        MAI1770
  110 FORMAT(10X,2F10.3,//)                                              MAI1780
  112 FORMAT(///,10X,34HMINIMUM DIAMETER INCREASE/DECREASE ,//,15X,20HX  MAI1790
     *         DIA. INCR. ,//)                                           MAI1800
      END                                                                MAI1810
```

```
      SUBROUTINE CONST(X,NCONST,PHI)
C     ==============================
C
      DIMENSION X(1),PHI(1)
C
      COMMON/AA/ ROVDB(8),HOVRB(7),CB(7,8),ROVDT(11),DOVDT(6),CT(6,11),
     *FCT1(20),FCT2(20)
      COMMON/BB/ RA,RB,                      XR(200),X2(200),D2(200),PV2(200)
     *,BMV(200),BMH(200),BM(200),DFLV(200),DFLH(200),SLV(200),SLH(200),
     *SL(200),STRV(200),STRH(200),STR(200),SCB(200),SCT(200),SCF2(200),
     *RAD2(200),STRINT(200),TM2(200),TETA(200),TAU(200),TAUINT(200),
     *STR1(200),SIGMA1(200),PH2(200),DFL(200),PW2(200),BMW(200),
     *DFLW(200),SLW(200),STRW(200),
     *WORKA(200),WORKB(200),WORKC(200),WORKD(200),WORKE(200),WORKF(200),
     *WORKG(200),WORKH(200),WORKI(200)
      COMMON/DD/ E,G,N00,L1,L2,N1,N2,N
      COMMON/EE/XDIA(50),DIA(50),ND
      COMMON/FF/ STRAL,DFLAL,SLAL,TETAAL,RPMMIN,XDI(20),DI(20),NDI
      COMMON/GG/ XDF(20),DF(20),NDF,WORKS(50)
C
      DO 70 I=1,ND
      WORKS(I)=DIA(I)
 70   CONTINUE
      NND=ND-1
      K=0
      DO 3 I=1,NND
      K=K+1
      DIA(I)=X(K)
      DO 2 J=1,NDF
      IF(XDIA(I).EQ.XDF(J)) GO TO 1
      GO TO 2
 1    DIA(I)=DF(J)
      K=K-1
 2    CONTINUE
      IF(I.EQ.1) GO TO 3
      IF(WORKS(I).EQ.WORKS(I-1)) GO TO 90
      GO TO 3
 90   DIA(I)=DIA(I-1)
 3    CONTINUE
      DIA(ND)=DIA(ND-1)
C
C
      CALL        STRESS(E,G,N00,ND,N1,N2,N,L1,L2,RA,RB,XDIA,DIA,XR,X2,
     *D2,PW2,PV2,PH2,BMW,BMV,BMH,BM,DFLW,DFLV,DFLH,DFL,SLW,SLV,SLH,SL,
     *STRW,STRV,STRH,STR,RPMCR,SCB,SCT,SCF2,RAD2,STRINT,TM2,TETA,TAU,
     *TAUINT,STR1,SIGMA1,
     *       WORKA,WORKB,WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,WORKI,0)
C
      CALL MAXIM(STR1,STRMAX,N)
      CALL MAXIM(DFL,DFLMAX,N)
C
      PHI(1)=STRAL-STRMAX
      PHI(2)=DFLAL-DFLMAX
      PHI(3)=SLAL-SL(1)
      PHI(4)=SLAL-SL(N1)
      PHI(5)=TETAAL-TETA(N)
      PHI(6)=RPMCR-RPMMIN
C
      M=6
      DO 6 I=1,NND
      DO 5 J=1,NDI
      IF(XDIA(I).EQ.XDI(J)) GO TO 4
      GO TO 5
 4    M=M+1
      IF(DIA(I-1).GT.DIA(I)) GO TO 41
      PHI(M)=DIA(I)-DIA(I-1)-DI(J)
      GO TO 5
 41   CONTINUE
      PHI(M)=DIA(I-1)-DIA(I)-DI(J)
 5    CONTINUE
 6    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE UREAL(X,U)
C     =====================
C
      DIMENSION X(1)
C
      COMMON/EE/XDIA(50),DIA(50),ND
      COMMON/GG/ XDF(20),DF(20),NDF,WORKS(50)
C
      DO 70 I=1,ND
      WORKS(I)=DIA(I)
 70   CONTINUE
      NND=ND-1
      K=0
      DO 3 I=1,NND
      K=K+1
      DIA(I)=X(K)
      DO 2 J=1,NDF
      IF(XDIA(I).EQ.XDF(J)) GO TO 1
      GO TO 2
 1    DIA(I)=DF(J)
      K=K-1
 2    CONTINUE
      IF(I.EQ.1) GO TO 3
      IF(WORKS(I).EQ.WORKS(I-1)) GO TO 90
      GO TO 3
 90   DIA(I)=DIA(I-1)
 3    CONTINUE
      DIA(ND)=DIA(ND-1)
      PI=3.1415926
      U=0.0
      DO 4 I=2,ND
      U=U+PI*DIA(I-1)**2/4.*(XDIA(I)-XDIA(I-1))*.283
 4    CONTINUE
      RETURN
      END
```

```
C*S
===
      SUBROUTINE LOAD  (N00,ND,N1,N2,N,L1,L2,W,XDIA,DIA,XR,
     *   X2,PW2,PV2,PH2,TM2,D2,SCF2,RAD2,RD,
     *X1,PW1,PV1,PH1,UNFW1,UNFV1,UNFH1,TM1,D1,SCF1,RAD1,UNFW2,UNFV2,
     *UNFH2,WORK1,WORK2,WORK3,WORK4)
C
      DIMENSION XDIA(ND),DIA(ND),XR(N00),X2(N00),PV2(N00),PH2(N00),
     *TM2(N00),D2(N00),SCF2(N00),RAD2(N00),X1(N00),PV1(N00),PH1(N00),
     *UNFV1(N00),UNFH1(N00),TM1(N00),D1(N00),SCF1(N00),RAD1(N00),
     *UNFV2(N00),UNFH2(N00),WORK1(N00),WORK2(N00),WORK3(N00),WORK4(N00),
     *RD(ND),PW1(N00),PW2(N00),UNFW1(N00),UNFW2(N00)
C
      COMMON/AA/ ROVDB(8),HOVRB(7),CB(7,8),ROVDT(11),DOVDT(6),CT(6,11),
     *FCT1(20),FCT2(20)
C
C
C
C     THIS SUBROUTINE READS INPUT DATA CONSISTING OF DIMENSIONS AND
C     LOADING ON THE SHAFT AND ARRANGES THEM IN WORKING ARRAYS FOR THE
C     OPTIMIZATION PROCESS
C
      REAL L,L1,L2
C
C
C
C
C     DATA FOR CALCULATING STRESS INTENSIFICATION
C
      DATA (ROVDB(J),J=1,8)/0.0,0.05,0.1,0.2,0.27,0.50,1.0,10000./
C
      DATA (HOVRB(I),I=1,7)/0.0,0.5,1.0,1.5,2.0,3.5,10000./
C
      DATA ((CB(I,J),J=1,8),I=1,7)/
     *     1.00,  1.00,  1.00,  1.00,  1.00,  1.00,  1.00,  1.00,
     *     2.00,  1.61,  1.49,  1.39,  1.34,  1.22,  1.08,  1.00,
     *     2.00,  1.91,  1.70,  1.48,  1.38,  1.22,  1.08,  1.00,
     *     2.00,  2.00,  1.73,  1.50,  1.39,  1.23,  1.08,  1.00,
     *     2.00,  2.00,  1.74,  1.52,  1.39,  1.23,  1.09,  1.00,
     *     2.00,  2.00,  1.76,  1.54,  1.40,  1.23,  1.10,  1.00,
     *     2.00,  2.00,  2.00,  2.00,  2.00,  2.00,  2.00,  2.00/
C
      DATA (ROVDT(J),J=1,11)/0.0,0.005,0.01,0.02,0.03,0.04,0.06,0.08,.10
     *,.12,10000./
C
      DATA(DOVDT(I),I=1,6)/10000.,2.00,1.33,1.20,1.09,1.00/
C
      DATA((CT(I,J),J=1,11),I=1,6)/
     * 3.00,  3.00,  3.00,  2.25,  2.00,  1.82,  1.65,  1.51,  1.44,
     * 1.39,  1.00,
     * 3.00,  3.00,  3.00,  2.25,  2.00,  1.82,  1.65,  1.51,  1.44,
     * 1.39,  1.00,
     * 3.00,  3.00,  2.7,  2.16,  1.91,  1.76,  1.60,  1.48,  1.40,
     * 1.35,  1.00,
     * 3.0,  3.00,  2.50,  2.00,  1.75,  1.62,  1.50,  1.40,  1.34,
     * 1.30,  1.00,
     * 3.00,  2.20,  1.88,  1.53,  1.40,  1.30,  1.20,  1.16,  1.15,
     * 1.15,  1.00,
     * 1.00,  1.00,  1.00,  1.00,  1.00,  1.00,  1.00,  1.00,  1.00,
     * 1.00,  1.00/
C
C     SET WORKING ARRAYS TO 0
      DO 1 I=1,N00
      X1(I)=0.0
      X2(I)=0.0
      PW1(I)=0.0
      PW2(I)=0.0
      PV1(I)=0.0
      PV2(I)=0.0
      PH1(I)=0.0
      PH2(I)=0.0
      UNFW1(I)=0.0
      UNFW2(I)=0.0
      UNFV1(I)=0.0
```

```
      UNFV2(I)=0.0
      UNFH1(I)=0.0
      UNFH2(I)=0.0
      TM1(I)=0.0
      TM2(I)=0.0
      D1(I)=0.0
      D2(I)=0.0
      SCF1(I)=1.0
      SCF2(I)=1.0
      RAD1(I)=10000.
      RAD2(I)=10000.
      XR(I)=0.0
      WORK1(I)=0.0
      WORK2(I)=0.0
      WORK3(I)=0.0
      WORK4(I)=0.0
 1    CONTINUE
C
      PW=0.0
      PV=0.0
      PH=0.0
      UNFW=0.0
      UNFV=0.0
      UNFH=0.0
      TM=0.0
      D=0.0
      SCF=1.0
      RAD=10000.
      PI=3.1415926
C
C     PRINT OUT THE TITLE AND INPUT DATA
C
      WRITE(6,150)
C
C     READ INPUT DATA AND PRINT OUT
C
      READ(5,101) L1,L2
      L=L1+L2
      X1(2)=L
C
      WRITE(6,151)L1,L2
C
      NX=2
C
C     CONCENTRATED FORCES
C
      READ(5,110)NPW,NPV,NPH
      IF(NPW.EQ.0) GO TO 69
      WRITE(6,137)
      DO 30 I=1,NPW
      READ(5,111) XPW,PW
      WRITE(6,141) XPW,PW
      X=XPW
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 30   CONTINUE
      PW=0.0
 69   CONTINUE
C
      IF(NPV.EQ.0) GO TO 70
      WRITE(6,140)
      DO 2 I=1,NPV
      READ(5,111) XPV,PV
      WRITE(6,141)XPV,PV
      X=XPV
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 2    CONTINUE
      PV=0.0
C
 70   CONTINUE
      IF(NPH.EQ.0) GO TO 71
```

```
      WRITE(6,142)
C
      DO 3 I=1,NPH
      READ(5,111) XPH,PH
      WRITE(6,141) XPH,PH
      X=XPH
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 3    CONTINUE
      PH=0.0
C
 71   CONTINUE
C
C     UNIFORMLY DISTRIBUTED FORCES
C
      READ(5,110) NUNFW,NUNFV,NUNFH
      IF(NUNFW.EQ.0) GO TO 31
      WRITE(6,138)
      DO 32 I=1,NUNFW
      READ(5,112) X1UNFW,X2UNFW,UNFW
      WRITE(6,144) X1UNFW,X2UNFW,UNFW
      X=X1UNFW
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
      UNFW=-UNFW
      X=X2UNFW
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 32   CONTINUE
      UNFW=0.0
 31   CONTINUE
C
      IF(NUNFV.EQ.0) GO TO 72
      WRITE(6,143)
      DO 4 I=1,NUNFV
      READ(5,112) X1UNFV,X2UNFV,UNFV
      WRITE(6,144)X1UNFV,X2UNFV,UNFV
      X=X1UNFV
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
      UNFV=-UNFV
      X=X2UNFV
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 4    CONTINUE
      UNFV=0.0
C
 72   CONTINUE
      IF(NUNFH.EQ.0) GO TO 73
C
      WRITE (6,145)
      DO 5 I=1,NUNFH
      READ(5,112) X1UNFH,X2UNFH,UNFH
      WRITE(6,144) X1UNFH,X2UNFH,UNFH
      X=X1UNFH
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
      UNFH=-UNFH
      X=X2UNFH
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 5    CONTINUE
      UNFH=0.0
C
 73   CONTINUE
C
C     TORSION MOMENTS
```

```
C
      READ(5,120) NTM
      IF(NTM.EQ.0) GO TO 74
      WRITE(6,146)
      DO 6 I=1,NTM
      READ(5,112) X1TM,X2TM,TM
      WRITE(6,144)X1TM,X2TM,TM
      X=X1TM
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
      X=X2TM
      TM=-TM
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 6    CONTINUE
      TM=0.0
C
 74   CONTINUE
C     DIAMETERS
C
      WRITE(6,147)
      READ(5,120) ND
      DO 7 I=1,ND
      READ(5,113) XD,D
      XDIA(I)=XD
      DIA(I)=D
      WRITE(6,139) XD,D
      X=XD
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 7    CONTINUE
      D=0.0
C
C     STRESS CONCENTRATION FACTORS
C
      READ(5,120) NCF
      IF(NCF.EQ.0) GO TO 75
      WRITE(6,148)
      DO 8 I=1,NCF
      READ(5,101) XSCF,SCF
      WRITE(6,139)XSCF,SCF
      X=XSCF
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 8    CONTINUE
      SCF=1.0
C
 75   CONTINUE
C
C     RADII
      DO 89 I=1,ND
      RD(I)=10000.
 89   CONTINUE
C
      READ(5,120) NRAD
      IF(NRAD.EQ.0) GO TO 76
      WRITE(6,149)
      DO 9 I=1,NRAD
      READ(5,101)XRAD,RAD
      WRITE(6,139) XRAD,RAD
      DO 90 J=1,ND
      IF(XRAD.EQ.XDIA(J)) RD(J)=RAD
 90   CONTINUE
      X=XRAD
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
 9    CONTINUE
C
 76   CONTINUE
```

```
      EPS=0.001
      IF(W.LT.EPS) GO TO 78
      NUNFW=1
C
C     ADD WEIGHT OF SHAFT TO UNIFORM DISTRIBUTED WEIGHTS
      NSS=NX-1
      DO 77 I=1,NSS
      UNFW2(I)=UNFW2(I)+PI*D2(I)**2/4.*.283
 77   CONTINUE
 78   CONTINUE
C
      UNFW1(1)=UNFW2(1)
      UNFV1(1)=UNFV2(1)
      TM1(NX)=TM2(NX-1)
      UNFV1(NX)=UNFV2(NX-1)
      UNFH1(NX)=UNFH2(NX-1)
      TM2(NX)=TM1(NX)
      UNFV2(NX)=UNFV1(NX)
      UNFH2(NX)=UNFH1(NX)
C
C     INCLUDE IN THE RANGE OF X DOUBLE POINTS WHERE THE DIAMETER OR
C     TORSIONAL MOMENT CHANGE VALUE
      NXX=1
C
      DO 11 I=2,NX
      IF(D2(I).NE.D2(I-1)) GO TO 10
      IF(TM2(I).NE.TM2(I-1)) GO TO 10
      NXX=NXX+1
      X1(NXX)=X2(I)
      PW1(NXX)=PW2(I)
      PV1(NXX)=PV2(I)
      PH1(NXX)=PH2(I)
      UNFW1(NXX)=UNFW2(I)
      UNFV1(NXX)=UNFV2(I)
      UNFH1(NXX)=UNFH2(I)
      TM1(NXX)=TM2(I)
      D1(NXX)=D2(I)
      SCF1(NXX)=SCF2(I)
      RAD1(NXX)=RAD2(I)
      GO TO 11
C
 10   NXX=NXX+1
      X1(NXX)=X2(I)
      PW1(NXX)=0.0
      PV1(NXX)=0.0
      PH1(NXX)=0.0
      UNFW1(NXX)=UNFW2(I-1)
      UNFV1(NXX)=UNFV2(I-1)
      UNFH1(NXX)=UNFH2(I-1)
      TM1(NXX)=TM2(I-1)
      D1(NXX)=D2(I-1)
      SCF1(NXX)=1.0
      RAD1(NXX)=RAD2(I)
C
      NXX=NXX+1
      X1(NXX)=X2(I)
      PW1(NXX)=PW2(I)
      PV1(NXX)=PV2(I)
      PH1(NXX)=PH2(I)
      UNFW1(NXX)=UNFW2(I)
      UNFV1(NXX)=UNFV2(I)
      UNFH1(NXX)=UNFH2(I)
      TM1(NXX)=TM2(I)
      D1(NXX)=D2(I)
      SCF1(NXX)=SCF2(I)
      RAD1(NXX)=10000.
C
      K=I+1
      ICOUNT=NXX
      DO 12 J=K,NX
      ICOUNT=ICOUNT+1
      X1(ICOUNT)=X2(J)
      PW1(ICOUNT)=PW2(J)
      PV1(ICOUNT)=PV2(J)
```

```
      PH1(ICOUNT)=PH2(J)
      UNFW1(ICOUNT)=UNFW2(J)
      UNFV1(ICOUNT)=UNFV2(J)
      UNFH1(ICOUNT)=UNFH2(J)
      TM1(ICOUNT)=TM2(J)
      D1(ICOUNT)=D2(J)
      SCF1(ICOUNT)=SCF2(J)
      RAD1(ICOUNT)=RAD2(J)
 12   CONTINUE
C
 11   CONTINUE
C
      NX=ICOUNT
C
      WRITE(6,200)
C
      DO 13 I=1,NXX
      RAD2(I)=RAD1(I)
      IF(RAD1(I).EQ.10000.) RAD1(I)=0.0
 13   CONTINUE
      I=1
      WRITE(6,202)I,X1(I),D1(I),RAD1(I),SCF1(I),PW1(I),PV1(I),PH1(I),
     *UNFW1(I),UNFV1(I),UNFH1(I),TM1(I)
      NLAST=NXX-1
      DO 15 I=2,NLAST
      IF(X1(I).EQ.L1) GO TO 14
      WRITE(6,203)I,X1(I),D1(I),RAD1(I),SCF1(I),PW1(I),PV1(I),PH1(I),
     *UNFW1(I),UNFV1(I),UNFH1(I),TM1(I)
      GO TO 15
 14   WRITE(6,202)I,X1(I),D1(I),RAD1(I),SCF1(I),PW1(I),PV1(I),PH1(I),
     *UNFW1(I),UNFV1(I),UNFH1(I),TM1(I)
 15   CONTINUE
      I=NXX
      WRITE(6,204)I,X1(I),D1(I),RAD1(I),SCF1(I),PW1(I),PV1(I),PH1(I),
     *UNFW1(I),UNFV1(I),UNFH1(I),TM1(I)
C
      DO 131 I=1,NXX
      RAD1(I)=RAD2(I)
 131  CONTINUE
C
      DO 16 I=1,NXX
      XR(I)=X1(I)
 16   CONTINUE
C
C     REPLACE UNIFORMLY DISTRIBUTED FORCES BY CONCENTRATED FORCES
C
      N=NXX
      MCONC=1
C
      KK=0
      NSS=N-1
      IF(NUNFW.EQ.0) GO TO 17
      DO 19 I=1,NSS
      IF(UNFW1(I).EQ.0.0) GO TO 19
      IF((X1(I+1)-X1(I)).EQ.0.0) GO TO 19
      DO 191 J=1,MCONC
      KK=KK+1
      WORK1(KK)=
     *  X1(I)+(X1(I+1)-X1(I))/FLOAT(MCONC*2)+(X1(I+1)-X1(I))/FLOAT(MCONC
     *)*(FLOAT(J)-1)
      WORK2(KK)=
     *   UNFW1(I)*(X1(I+1)-X1(I))/FLOAT(MCONC)
 191  CONTINUE
 19   CONTINUE
      DO 192 I=1,KK
      IF(UNFW1(I).EQ.0.0) GO TO 192
      IF((X1(I+1)-X1(I)).EQ.0.0) GO TO 192
      X=WORK1(I)
      WORK1(I)=0.0
      PW=WORK2(I)
      WORK2(I)=0.0
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,N,N00)
```

```
  192 CONTINUE
      PW=0.0
   17 CONTINUE
C
      KK=0
      NSS=N-1
      IF(NUNFV.EQ.0) GO TO 291
      DO 20 I=1,NSS
      IF(UNFV1( I).EQ.0.0) GO TO 20
      IF((X1( I+1)-X1( I)).EQ.0.0) GO TO 20
      DO 21 J=1,MCONC
      KK=KK+1
      WORK1(KK)=
     *  X1( I)+(X1( I+1)-X1( I))/FLOAT(MCONC*2)+(X1( I+1)-X1( I))/FLOAT(MCONC
     *)*(FLOAT(J)-1)
      WORK2(KK)=
     *    UNFV1( I)*(X1( I+1)-X1( I))/FLOAT(MCONC)
   21 CONTINUE
   20 CONTINUE
      DO 201 I=1,KK
      IF(UNFV1( I).EQ.0.0) GO TO 201
      IF((X1( I+1)-X1( I)).EQ.0.0) GO TO 201
      X=WORK1( I)
      WORK1( I)=0.0
      PV=WORK2( I)
      WORK2( I)=0.0
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,N,N00)
  201 CONTINUE
      PV=0.0
  291 CONTINUE
C
      KK=0
      NSS=N-1
      IF(NUNFH.EQ.0) GO TO 222
      DO 22 I=1,NSS
      IF(UNFH1( I).EQ.0.0) GO TO 22
      IF((X1( I+1)-X1( I)).EQ.0.0) GO TO 22
      DO 23 J=1,20
      KK=KK+1
      WORK3(KK)=
     *  X1( I)+(X1( I+1)-X1( I))/FLOAT(MCONC*2)+(X1( I+1)-X1( I))/FLOAT(MCONC
     *)*(FLOAT(J)-1)
      WORK4(KK)=
     *    UNFH1( I)*(X1( I+1)-X1( I))/FLOAT(MCONC)
   23 CONTINUE
   22 CONTINUE
      DO 221 I=1,KK
      IF(UNFH1( I).EQ.0.0) GO TO 221
      IF((X1( I+1)-X1( I)).EQ.0.0) GO TO 221
      X=WORK3( I)
      WORK3( I)=0.0
      PH=WORK4( I)
      WORK4( I)=0.0
      CALL RANGE(X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,N,N00)
  221 CONTINUE
      PH=0.0
  222 CONTINUE
C
      DO 301 I=1,N00
      X2( I)=X1( I)
      PW2( I)=PW1( I)
      PV2( I)=PV1( I)
      PH2( I)=PH1( I)
      TM2( I)=TM1( I)
      D2( I)=D1( I)
      SCF2( I)=SCF1( I)
      RAD2( I)=RAD1( I)
  301 CONTINUE
C
      DO 24 I=1,N
```

```
      IF(X2(I).NE.L1) GO TO 24
      N1=I
 24   CONTINUE
      N2=N-N1
C
      RETURN
 101  FORMAT(2F10.3)
C
 110  FORMAT(3I10)
C
 111  FORMAT(F10.3,F10.0)
C
 112  FORMAT(2F10.3,F10.)
C
 113  FORMAT(2F10.3)
C
 120  FORMAT(I10)
C
 137  FORMAT(///,10X,25HWEIGHTS ATTACHED TO SHAFT,//,15X,16HX        WE
     *IGHT   )
 138  FORMAT(///,10X,42HUNIFORMLY DISTRIBUTED WEIGHTS              ,//,
     *15X,33HX START      X END       WEIGHT  )
 139  FORMAT(/,10X,F9.3,F13.3)
C
 140  FORMAT(///,10X,20HVERTICAL FORCES     ,//,15X,16HX          FORCE)
C
 141  FORMAT(/,10X,F9.3,F13.0)
C
 142  FORMAT(///,10X,20HHORIZONTAL FORCES   ,//,15X,16HX          FORCE)
C
 143  FORMAT(///,10X,44HVERTICAL UNIFORMLY DISTRIBUTED FORCES      ,//,
     *15X,33HX START      X END       FORCE   )
C
 144  FORMAT(//,6X,2F13.3,F13.0)
C
 145  FORMAT(///,10X,44HHORIZONTAL UNIFORMLY DISTRIBUTED FORCES    ,//,
     *15X,33HX START      X END       FORCE   )
C
 146  FORMAT(//,10X,17HTORSIONAL MOMENTS,//,10X,35H   X START     X END
     *      MOMENT    )
C
 147  FORMAT(//,10X,9HDIAMETERS//,10X,21H    X           DIA )
C
 148  FORMAT(//,10X,28HSTRESS CONCENTRATION FACTORS,//,15X,19HX
     *  FACTOR  )
C
 149  FORMAT(//,10X,5HRADII,//,15X,16HX            RAD   )
C
 150  FORMAT(1H1,////,
     *20X,25HSHAFT DESIGN OPTIMIZATION,/,
     *20X,25H*************************,////,
     *20X,11HINPUT DATA  ,/,
     *20X,10H**********,//)
C
 151  FORMAT(10X,20HLENGTH OF FIRST SPAN,F10.3,3H IN,//,
     *       10X,20HLENGTH OF SC"ND SPAN,F10.3,3H IN,//)
C
 200  FORMAT(1H1,////,,10X,22HDIMENSIONS AND LOADING,//,
     *10X,127H==========================================================
     *==================================================================
     *===                                                        ,/,10X,
     * 123HPOINT      X      DIA     RADII   STRESS         CONC.FORCE
     *                   UNIF.DISTR.FORCE       TORS.MOM.   REMARKS  ,/,
     *10X,
     * 123HNO                                CONC.FCT.  WEIGHT    VERT.
     *   HORIZ.    WEIGHT    VERT.     HORIZ.                        ,/,
     *10X,
     * 123H          IN      IN        IN                  LB        LB
     *       LB     LB/IN     LB/IN     LB/IN     LBIN               ,/,
     *10X,127H==========================================================
     *==================================================================
     *===          )
C
 202  FORMAT(10X,I4,3F9.3,F9.2,7F10.0,3X,13HSUPPORT POINT,/)
```

```
C
 203  FORMAT( 10X, I4,3F9.3,F9.2,7F10.0,3X,/)
C
 204  FORMAT( 10X, I4,3F9.3,F9.2,7F10.0,3X,12HEND OF SHAFT,/)
C
      END
```

```
      SUBROUTINE STRESS(E,G,N00,ND,N1,N2,N,L1,L2,RA,RB,XDIA,DIA,XR,X2,
C     ================================================================
     *D2,PW2,PV2,PH2,BMW,BMV,BMH,BM,DFLW,DFLV,DFLH,DFL,SLW,SLV,SLH,SL,
     *STRW,STRV,STRH,STR,RPMCR,SCB,SCT,SCF2,RAD2,STRINT,TM2,TETA,TAU,
     *TAUINT,STR1,SIGMA1,SI,SIP,Z,ZP,WORKE,WORKF,WORKG,WORKH,WORKI,IPRT)
C
      COMMON/AA/ ROVDB(8),HOVRB(7),CB(7,8),ROVDT(11),DOVDT(6),CT(6,11),
     *FCT1(20),FCT2(20)
C
      DIMENSION XDIA(ND),DIA(ND),XR(N00),D2(N00),X2(N00),SI(N00),
     *SIP(N00),Z(N00),ZP(N00),PV2(N00),PH2(N00),BMV(N00),BMH(N00),
     *BM(N00),DFLV(N00),DFLH(N00),DFL(N00),SLV(N00),SLH(N00),SL(N00),
     *STRV(N00),STRH(N00),STR(N00),SCB(N00),SCT(N00),SCF2(N00),RAD2(N00)
     *,STRINT(N00),TAUINT(N00),TM2(N00),TETA(N00),TAU(N00),STR1(N00),
     *SIGMA1(N00),PW2(N00),BMW(N00),DFLW(N00),SLW(N00),STRW(N00),
     *WORKE(N00),WORKF(N00),WORKG(N00),WORKH(N00),WORKI(N00)
C
      REAL L,L1,L2
C
      PI=3.1415926
C
      DO 10 I=1,N
      WORKE(I)=D2(I)
 10   CONTINUE
C
      D2(1)=DIA(1)
      KK=ND-1
      DO 2 I=2,N
      DO 1 J=1,KK
      IF(X2(I).LE.XDIA(J)) GO TO 1
      IF(X2(I).GT.XDIA(J+1)) GO TO 1
      D2(I)=DIA(J)
      IF(WORKE(I).EQ.WORKE(I-1)) GO TO 1
      IF(X2(I).EQ.X2(I-1)) D2(I)=DIA(J+1)
  1   CONTINUE
  2   CONTINUE
C
C     CALCULATE MOMENT OF INERTIA AND SECTION MODULUS
      DO 25 I=1,N
      IF(D2(I).EQ.0.0) GO TO 25
      SI(I)=PI*D2(I)**4/64.
      SIP(I)=SI(I)*2.
      Z(I)=SI(I)*2./D2(I)
      ZP(I)=SIP(I)*2./D2(I)
 25   CONTINUE
C
      CALL BEND(E,L1,L2,N1,N2,N,X2,SI,Z,PW2,BMW,DFLW,SLW,STRW,RAW,RBW,
     *N00,BM,DFL,SL,STR,TAU,STR1,WORKE,WORKF,WORKG,WORKH,WORKI)
      CALL BEND(E,L1,L2,N1,N2,N,X2,SI,Z,PV2,BMV,DFLV,SLV,STRV,RAV,RBV,
     *N00,BM,DFL,SL,STR,TAU,STR1,WORKE,WORKF,WORKG,WORKH,WORKI)
      CALL BEND(E,L1,L2,N1,N2,N,X2,SI,Z,PH2,BMH,DFLH,SLH,STRH,RAH,RBH,
     *N00,BM,DFL,SL,STR,TAU,STR1,WORKE,WORKF,WORKG,WORKH,WORKI)
C
C     CALCULATE THE RESULTANT BENDING MOMENT,DEFLECTION,SLOPE AND STRESS
      RAV=RAV+RAW
      RA=SQRT(RAV**2+RAH**2)
      RBV=RBV+RBW
      RB=SQRT(RBV**2+RBH**2)
      DO 26 I=1,N
      BMV(I)=BMV(I)+BMW(I)
      BM(I)=SQRT(BMV(I)**2+BMH(I)**2)
      DFLV(I)=DFLV(I)+DFLW(I)
      DFL(I)= SQRT(DFLV(I)**2+DFLH(I)**2)
      SLV(I)=SLV(I)+SLW(I)
      SL(I)=SQRT(SLV(I)**2+SLH(I)**2)
      STRV(I)=STRV(I)+STRW(I)
      STR(I)=SQRT(STRV(I)**2+STRH(I)**2)
 26   CONTINUE
C
      DO 60 K=1,N
      SCB(K)=SCF2(K)
      SCT(K)=SCF2(K)
      IF(D2(K).EQ.0.0) GO TO 60
      IF(D2(K+1).EQ.0.0) GO TO 60
```

```
      IF(RAD2(K).GE.10000.) GO TO 60
      DMIN=D2(K)
      IF(D2(K+1).LT.D2(K)) DMIN=D2(K+1)
      RDX=RAD2(K)/DMIN
C
      HRX=ABS(D2(K+1)-D2(K))/2./RAD2(K)
C
      IF(D2(K+1).GT.D2(K))GO TO 53
      DDX=D2(K)/D2(K+1)
      GO TO 54
 53   CONTINUE
      DDX=D2(K+1)/D2(K)
 54   CONTINUE
C
      DO 55 I=2,7
      IF(HRX.GE.HOVRB(I))GO TO 55
      DO 56 J=1,8
      FCT1(J)= CB(I-1,J)
      FCT2(J)= CB(I,J)
 56   CONTINUE
      FCTR1=FTABLE(ROVDB,FCT1,RDX,8)
      FCTR2=FTABLE(ROVDB,FCT2,RDX,8)
      SCB(K)=ABS((HRX-HOVRB(I-1))/(HOVRB(I)-HOVRB(I-1))*(FCTR2-FCTR1)+
     *FCTR1)
      GO TO 551
 55   CONTINUE
 551  CONTINUE
C
      DO 57 I=2,6
      IF(DDX.LE.DOVDT(I)) GO TO 57
      DO 58 J=1,11
      FCT1(J)= CT(I-1,J)
      FCT2(J)=CT(I,J)
 58   CONTINUE
      FCTR1=FTABLE(ROVDT,FCT1,RDX,11)
      FCTR2=FTABLE(ROVDT,FCT2,RDX,11)
      SCT(K)=ABS((DDX-DOVDT(I-1))/(DOVDT(I)-DOVDT(I-1))*(FCTR2-FCTR1)+
     *FCTR1)
      GO TO 571
 57   CONTINUE
 571  CONTINUE
 60   CONTINUE
C
C     CALCULATE THE INTENSIFIED STRESS
      DO 27 I=1,N
      STRINT(I)=STR(I)*SCB(I)
 27   CONTINUE
C
C     FIND THE ROTATIONS AND SHEAR STRESS DUE TO TORSION
C
      CALL TWIST(G,N,X2,SIP,ZP,TM2,TETA,TAU,N00)
C
C     CALCULATE INTENSIFIED SHEAR STRESSES
      DO 28 I=1,N
      TAUINT(I)=TAU(I)*SCT(I)
 28   CONTINUE
C
C     FIND COMBINED STRESSES
      DO 30 I=1,N
      SIGMA1(I)=.35*STR(I)+.65*SQRT(STR(I)**2+4.*TAU(I)**2)
      STR1(I)=.35*STRINT(I)+.65*SQRT(STRINT(I)**2+4.*TAUINT(I)**2)
 30   CONTINUE
C
C     CALCULATE CRITICAL SPEED
      GRAV=385.92
      SUM1=0.0
      SUM2=0.0
      DO 3 I=1,N
      SUM1=SUM1+PW2(I)*DFLW(I)
       SUM2=SUM2+PW2(I)*DFLW(I)**2
 3    CONTINUE
C
C     ANGULAR FREQUENCY OF FUNDAMENTAL VIBRATIONS
      P=SQRT(GRAV*SUM1/SUM2)
```

```
C                                                                       STR1480
C     CRITICAL SPEED                                                    STR1490
      RPMCR=30.*P/PI                                                    STR1500
C                                                                       STR1510
      IF(IPRT.EQ.0) GO TO 502                                           STR1520
C                                                                       STR1530
C     FIND THE WEIGHT OF SHAFT                                          STR1540
      WEIGHT=0.0                                                        STR1550
      DO 40 I=2,N                                                       STR1560
      WEIGHT=WEIGHT+PI*D2(I-1)**2/4.*(X2(I)-X2(I-1))*.283               STR1570
 40   CONTINUE                                                          STR1580
C                                                                       STR1590
      XA=0.0                                                            STR1600
      XB=L1                                                             STR1610
C     PRINT OUT THE REACTIONS                                           STR1620
      WRITE(6,240) XA,RAV,RAH,RA,XB,RBV,RBH,RB                          STR1630
C                                                                       STR1640
C                                                                       STR1650
      WRITE(6,250)                                                      STR1660
C                                                                       STR1670
      K=0                                                               STR1680
      NXX=N                                                             STR1690
      DO 48 I=1,N                                                       STR1700
      DO 481 J=1,NXX                                                    STR1710
      IF(X2(I).NE.XR(J)) GO TO 481                                      STR1720
      K=K+1                                                             STR1730
      WRITE(6,251)K,X2(I),BMV(I),BMH(I),BM(I),TM2(I)                    STR1740
      GO TO 48                                                          STR1750
 481  CONTINUE                                                          STR1760
 48   CONTINUE                                                          STR1770
C                                                                       STR1780
      WRITE(6,260)                                                      STR1790
C                                                                       STR1800
      K=0                                                               STR1810
      DO 49 I=1,N                                                       STR1820
      DO 491 J=1,NXX                                                    STR1830
      IF(X2(I).NE.XR(J)) GO TO 491                                      STR1840
      K=K+1                                                             STR1850
      WRITE(6,261) K,X2(I),D2(I),                                       STR1860
     *                     DFLV(I),DFLH(I),DFL(I),SLV(I),SLH(I),SL(I),TE STR1870
     *TA(I)                                                             STR1880
      GO TO 49                                                          STR1890
 491  CONTINUE                                                          STR1900
 49   CONTINUE                                                          STR1910
C                                                                       STR1920
      WRITE(6,270)                                                      STR1930
C                                                                       STR1940
      K=0                                                               STR1950
      DO 50 I=1,N                                                       STR1960
      DO 501 J=1,NXX                                                    STR1970
      IF(X2(I).NE.XR(J)) GO TO 501                                      STR1980
      K=K+1                                                             STR1990
      WRITE(6,271)K,X2(I),STRV(I),STRH(I),STR(I),TAU(I),SIGMA1(I),SCB(I) STR2000
     *,SCT(I),STRINT(I),TAUINT(I),STR1(I)                               STR2010
      GO TO 50                                                          STR2020
 501  CONTINUE                                                          STR2030
 50     CONTINUE                                                        STR2040
C                                                                       STR2050
C     PRINT OUT WEIGHT OF SHAFT                                         STR2060
      WRITE(6,280) WEIGHT                                               STR2070
C                                                                       STR2080
C     PRINT OUT FUNDAMENTAL FREQUENCY                                   STR2090
C                                                                       STR2100
      WRITE(6,281)P,RPMCR                                               STR2110
C                                                                       STR2120
 502  CONTINUE                                                          STR2130
C                                                                       STR2140
      RETURN                                                            STR2150
C                                                                       STR2160
 240  FORMAT(1H1,                                                       STR2170
     * //////,10X,11HOUTPUT DATA,/,10X,11H***********,////,10X,         STR2180
     *45H    X      VERTICAL    HORIZONTAL  REZULTANT ,//,10X,          STR2190
     *F9.3,3F12.0,//,10X,                                               STR2200
     *F9.3,3F12.0,//)                                                   STR2210
```

```
C
 250  FORMAT( ////,10X,38H BENDING MOMENTS AND TORSIONAL MOMENTS ,//,10X
     *,72H==============================================================
     *==========  ,/,12X,
     *64HNO      X            BENDING MOMENT                TORSION MOME
     *NT ,/,12X,
     *45H                 VERT        HORIZ       RES  ,/,12X,
     *62H                     LBIN                                 LBIN
     *,/,10X,
     * 72H==============================================================
     *==========  )
C
 251  FORMAT( /,10X,I4,F10.3,2X,4F12.0)
C
 260  FORMAT(1H1,
     *   ////,10X,32HDEFLECTIONS,SLOPES AND ROTATIONS ,//,10X,
     * 98H==============================================================
     *====================================  ,/,11X,                    5
     *2HNR         X       DIA           DEFLECTION             ,35H
     * SLOPE                 ROTATION ,/,22X,
     *45H                VERT       HORIZ     RES     ,26H VERT       HO
     *RIZ     RES  ,/,12X,
     *52H         IN       IN       IN        IN        IN      ,44H RAD
     *        RAD       RAD       RAD           ,/,10X,
     * 98H==============================================================
     *====================================  )
C
 261  FORMAT(/,10X,I4,2F10.3,7F10.6)
C
 270  FORMAT(1H1,////,10X,8HSTRESSES,//,10X,
     *116H==============================================================
     *======================================================,/,10X,  116
     *H NR       X             BENDING STRESSES        TORSION    COMBIN
     *ED    INT FCT             INTENSIFIED STRESS          ,/,10X,  118
     *H                   VERT      HORIZ     RES       STRESS     STRES
     *S    BEND      TORS      BEND      TORS      COMB     ,/,10X,  116
     *H         IN        PSI       PSI       PSI        PSI       PSI
     *                         PSI       PSI       PSI      ,/,10X,
     *116H==============================================================
     *======================================================  )
C
 271  FORMAT( /,10X,I4,11F10.3)
C
 280  FORMAT(////,10X,16HWEIGHT OF SHAFT= ,F10.0,4H LBS,//)
 281  FORMAT( 10X,52HANGULAR FREQUENCY OF FUNDAMENTAL MODE OF VIBRATIONS
     *=,F10.3,7HRAD/SEC,//,10X,15HCRITICAL SPEED=,F10.3,3HRPM)
C
      END
```

```
      SUBROUTINE MAXIM(A,AMAX,M)                                        MAX  10
C     ==========================
C                                                                       MAX  20
      DIMENSION A(1)                                                    MAX  30
C                                                                       MAX  40
      AMAX=A(1)                                                         MAX  50
      DO 1 I=2,M                                                        MAX  60
      IF(A(I).GT.AMAX) AMAX=A(I)                                        MAX  70
 1    CONTINUE                                                          MAX  80
C                                                                       MAX  90
      RETURN                                                            MAX 100
      END                                                               MAX 110
```

```
      SUBROUTINE RANGE
C     ================
     *         (X,PW,PV,PH,UNFW,UNFV,UNFH,TM,D,SCF,X1,PW1,PV1,PH1,UNFW1
     *,UNFV1,UNFH1,TM1,D1,SCF1,X2,PW2,PV2,PH2,UNFW2,UNFV2,UNFH2,TM2,D2,
     *SCF2,RAD,RAD1,RAD2,NX,N00)
C
      DIMENSION X1(N00),X2(N00),PV1(N00),PV2(N00),PH1(N00),PH2(N00),
     *UNFV1(N00),UNFV2(N00),UNFH1(N00),UNFH2(N00),TM1(N00),TM2(N00),
     *PW1(N00),PW2(N00),UNFW1(N00),UNFW2(N00),D1(N00),D2(N00),SCF1(N00),
     *SCF2(N00),RAD1(N00),RAD2(N00)
C
C     THE SUBROUTINE PLACES THE INPUT DATA IN ARRAYS IN INCREASING ORDER
C     OF X
C
      DO 6 I=1,N00
      IF(X.LE.X1(I)) GO TO 1
      X2(I)=X1(I)
      PW2(I)=PW1(I)
      PV2(I)=PV1(I)
      PH2(I)=PH1(I)
      UNFW2(I)=UNFW1(I)
      UNFV2(I)=UNFV1(I)
      UNFH2(I)=UNFH1(I)
      TM2(I)=TM1(I)
      D2(I)=D1(I)
      SCF2(I)=SCF1(I)
      RAD2(I)=RAD1(I)
      GO TO 6
 1    IF(X.LT.X1(I)) GO TO 3
      X2(I)=X
      PW2(I)=PW1(I)+PW
      PV2(I)=PV1(I)+PV
      PH2(I)=PH1(I)+PH
      UNFW2(I)=UNFW1(I)+UNFW
      UNFV2(I)=UNFV1(I)+UNFV
      UNFH2(I)=UNFH1(I)+UNFH
      TM2(I)=TM1(I)+TM
      IF(D.EQ.0.0) GO TO 10
      D2(I)=D
      GO TO 11
 10   D2(I)=D1(I)
 11   CONTINUE
      SCF2(I)=SCF
      RAD2(I)=RAD
C
      K=I+1
      DO 2 J=K,N00
      PW2(J)=PW1(J)
      PV2(J)=PV1(J)
      PH2(J)=PH1(J)
      UNFW2(J)=UNFW1(J)+UNFW
      UNFV2(J)=UNFV1(J)+UNFV
      UNFH2(J)=UNFH1(J)+UNFH
      TM2(J)=TM1(J)+TM
      IF(D.EQ.0.0) GO TO 100
      D2(J)=D
      GO TO 101
 100  D2(J)=D1(J)
 101  CONTINUE
      SCF2(J)=SCF1(J)
      RAD2(J)=RAD1(J)
 2    CONTINUE
      GO TO 7
C
 3    X2(I)=X
      PW2(I)=PW
      PV2(I)=PV
      PH2(I)=PH
      UNFW2(I)=UNFW1(I-1)+UNFW
      UNFV2(I)=UNFV1(I-1)+UNFV
      UNFH2(I)=UNFH1(I-1)+UNFH
      TM2(I)=TM1(I-1)+TM
      IF(D.EQ.0.0) GO TO 30
      D2(I)=D
```

```
      GO TO 31                                    RAN 740
 30   D2(I)=D1(I-1)                               RAN 750
 31   CONTINUE                                    RAN 760
      SCF2(I)=SCF                                 RAN 770
      RAD2(I)=RAD                                 RAN 780
      NX=NX+1                                     RAN 790
      K=I+1                                       RAN 800
C                                                 RAN 810
      DO 4 J=K,N00                                RAN 820
      X2(J)=X1(J-1)                               RAN 830
      PW2(J)=PW1(J-1)                             RAN 840
      PV2(J)=PV1(J-1)                             RAN 850
      PH2(J)=PH1(J-1)                             RAN 860
      UNFW2(J)=UNFW1(J-1)+UNFW                    RAN 870
      UNFV2(J)=UNFV1(J-1)+UNFV                    RAN 880
      UNFH2(J)=UNFH1(J-1)+UNFH                    RAN 890
      TM2(J)=TM1(J-1)+TM                          RAN 900
      IF(D.EQ.0.0) GO TO 300                      RAN 910
      D2(J)=D                                     RAN 920
      GO TO 301                                   RAN 930
 300  D2(J)=D1(J-1)                               RAN 940
 301  CONTINUE                                    RAN 950
      SCF2(J)=SCF1(J-1)                           RAN 960
      RAD2(J)=RAD1(J-1)                           RAN 970
  4   CONTINUE                                    RAN 980
      GO TO 7                                     RAN 990
C                                                 RAN1000
  6   CONTINUE                                    RAN1010
  7   CONTINUE                                    RAN1020
C                                                 RAN1030
      DO 8 I=1,N00                                RAN1040
      X1(I)=X2(I)                                 RAN1050
      PW1(I)=PW2(I)                               RAN1060
      PV1(I)=PV2(I)                               RAN1070
      PH1(I)=PH2(I)                               RAN1080
      UNFW1(I)=UNFW2(I)                           RAN1090
      UNFV1(I)=UNFV2(I)                           RAN1100
      UNFH1(I)=UNFH2(I)                           RAN1110
      TM1(I)=TM2(I)                               RAN1120
      D1(I)=D2(I)                                 RAN1130
      SCF1(I)=SCF2(I)                             RAN1140
      RAD1(I)=RAD2(I)                             RAN1150
  8   CONTINUE                                    RAN1160
C                                                 RAN1170
      RETURN                                      RAN1180
      END                                         RAN1190
```

```
      SUBROUTINE TWIST(G,N,X,SIP,ZP,TM,TETA,TAU,N00)                    TWI  10
C     ==============================================
C                                                                       TWI  20
C     THE SUBROUTINE CALCULATES THE ANGLE OF TWIST ON A CIRCULAR SHAFT  TWI  30
C     AND SHEAR STRESS DUE TO TORSION                                   TWI  40
C                                                                       TWI  50
      DIMENSION X(N00),SIP(N00),TM(N00),TETA(N00),TAU(N00),ZP(N00)      TWI  60
C                                                                       TWI  70
      TETA(1)=0.0                                                       TWI  80
      DO 1 I=2,N                                                        TWI  90
      TETA(I)=TM(I)/(G*SIP(I))*(X(I)-X(I-1))+TETA(I-1)                  TWI 100
    1 CONTINUE                                                          TWI 110
      DO 2 I=1,N                                                        TWI 120
      TAU(I)=TM(I)/ZP(I)                                                TWI 130
    2 CONTINUE                                                          TWI 140
C                                                                       TWI 150
      RETURN                                                            TWI 160
      END                                                               TWI 170
```

```
      SUBROUTINE    BEND(E,L1,L2,N1,N2,N,X,SI,Z,P,BM,DFL,SL,STR,RA,RB,
C     ==============================================================
     *N00,BMOM,DEFL,SLOPE,WORKB,WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,
     *WORKI)
C
C     THE SUBROUTINE CALCULATES BENDING STRESSES IN ONE PLANE ALONG THE
C     SHAFT
C
      REAL L1,L2,L
      DIMENSION X(N00),SI(N00),Z(N00),P(N00),BM(N00),DFL(N00),SL(N00),
     *STR(N00),BMOM(N00),DEFL(N00),SLOPE(N00),WORKB(N00),WORKC(N00),
     *WORKD(N00),WORKE(N00),WORKF(N00),WORKG(N00),WORKH(N00),WORKI(N00)
C
      DO 1 I=1,N
      BM(I)=0.0
      DFL(I)=0.0
      SL(I)=0.0
 1    CONTINUE
C
      RA=0.0
      RB=0.0
C
      L=L1+L2
      DO 6 I=1,N
      IF(P(I).EQ.0.0) GO TO 6
      IF(X(I).GT.L1) GO TO 4
      A=X(I)
      PA=P(I)
      CALL FORCE(E,L1,N1,X,PA,A,SI,BMOM,DEFL,SLOPE,RRA,RRB,N00,
     *STR,WORKB,WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,WORKI)
C
      RA=RA+RRA
      RB=RB+RRB
      DO 2 J=1,N1
      BM(J)=BM(J)+BMOM(J)
      DFL(J)=DFL(J)+DEFL(J)
      SL(J)=SL(J)+SLOPE(J)
 2    CONTINUE
C
      N11=N1+1
      DO 3 K=N11,N
      DFL(K)=(X(K)-L1)*SLOPE(N1)+DFL(K)
      SL(K)=SLOPE(N1)+SL(K)
 3    CONTINUE
      GO TO 6
 4    CONTINUE
C
      C=X(I)-L1
      PC=P(I)
      CALL FORCEX(E,L,L1,N,N1,X,PC,C,SI,BMOM,DEFL,SLOPE,RRA,RRB,N00,
     *STR,WORKB,WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,WORKI)
C
      RA=RA+RRA
      RB=RB+RRB
      DO 5 J=1,N
      BM(J)=BM(J)+BMOM(J)
      DFL(J)=DFL(J)+DEFL(J)
      SL(J)=SL(J)+SLOPE(J)
 5    CONTINUE
 6    CONTINUE
C
      DO 7 I=1,N
      STR(I)=BM(I)/Z(I)
 7    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FORCE (E,L,N,X,P,A,SI,BMOM,DEFL,SLOPE,RA,RB,N00,
C     ===========================================================
     *      WORKA,WORKB,WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,WORKI)
C
C     THE SUBROUTINE CALCULATES MOMENTS AND DEFLECTIONS OF A BEAM OF
C     VARIABLE MOMENT OF INERTIA,SIMPLY SUPPORTED AT BOTH ENDS,WITH
C     CONCENTRATED FORCE AT ANY POINT BETWEEN THE SUPPORTS
C
      REAL L
      DIMENSION X(N00),SI(N00),BMOM(N00),DEFL(N00),SLOPE(N00),
     *                 WORKA(N00),WORKB(N00),WORKC(N00),WORKD(N00),
     *WORKE(N00),WORKF(N00),WORKG(N00),WORKH(N00),WORKI(N00)
C
      DO 10 I=1,N
      BMOM(I)=0.0
      DEFL(I)=0.0
      SLOPE(I)=0.0
   10 CONTINUE
C
      B=L-A
C     CALCULATE REACTIONS IN SUPPORT POINTS
      RA=P*B/L
      RB=P*A/L
C
C     CALCULATE THE DISTRIBUTION OF THE BENDING MOMENT ALONG THE SHAFT
      DO 2 I=1,N
      IF(X(I).GT.A) GO TO 1
      BMOM(I)=RA*X(I)
      GO TO 2
    1 BMOM(I)=RA*X(I)-P*(X(I)-A)
    2 CONTINUE
C
C
      NN=N+2
      CALL DEFLEC(E,L,N,N00,X,SI,BMOM,DEFL,SLOPE,WORKA,WORKB,WORKC,
     *WORKD,WORKE,WORKF,WORKG,WORKH,WORKI,NN)
C
      RETURN
      END
```

```
      SUBROUTINE FORCEX(E,L,L1,N,N1,X,PC,C,SI,BMOM,DEFL,SLOPE,RA,RB,N00,
C     ==================================================================
     *DD,SS,            WORKC,WORKD,WORKE,WORKF,WORKG,WORKH,WORKI)
C
C     THE SUBROUTINE CALCULATES MOMENTS AND DEFLECTIONS OF A BEAM OF
C     VARIABLE MOMENT OF INERTIA,SIMPLY SUPPORTED,WITH CONCENTRATED
C     FORCE AT ANY POINT OUTSIDE OF SUPPORTS
C
      REAL L,L1
      DIMENSION X(N00),SI(N00),BMOM(N00),DEFL(N00),SLOPE(N00)
     *,DD(N00),SS(N00),WORKC(N00),WORKD(N00),
     *WORKE(N00),WORKF(N00),WORKG(N00),WORKH(N00),WORKI(N00)
C
      DO 10 I=1,N
      BMOM(I)=0.0
      DEFL(I)=0.0
      SLOPE(I)=0.0
      DD(I)=0.0
      SS(I)=0.0
 10   CONTINUE
C
C     CALCULATE REACTIONS IN SUPPORT POINTS
      RA=-PC*C/L1
      RB=PC*(C+L1)/L1
C
C     CALCULATE THE DISTRIBUTION OF THE BENDING MOMENT ALONG THE SHAFT
      F=L1+C
      DO 2 I=1,N
      IF(X(I).GT.L1) GO TO 1
      BMOM(I)=RA*X(I)
      GO TO 2
  1   CONTINUE
      IF(X(I).GT.F) GO TO 2
      BMOM(I)=RA*X(I)+RB*(X(I)-L1)
  2   CONTINUE
      NN=N+2
C
      CALL DEFLEC(E,L,N,N00,X,SI,BMOM,DD,SS,DEFL,SLOPE,WORKC,WORKD,
     *WORKE,WORKF,WORKG,WORKH,WORKI,NN)
C
      DO 3 I=1,N
      DEFL(I)=DD(I)-DD(N1)/L1*X(I)
      SLOPE(I)=SS(I)-DD(N1)/L1
  3   CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE DEFLEC(E,L,N,N00,X,SI,BMOM,DEFL,SLOPE,XX,SSI,BBMOM,TETA
C     ==================================================================
     *,CG,DDEFL,XDEFL,BMOMA,SSLOPE,NN)
C
C     THE SUBROUTINE CALCULATES DEFLECTIONS AND SLOPES ALONG A BEAM WITH
C     KNOWN DISTRIBUTION OF BENDING MOMENT AND MOMENT OF INERTIA
C
      REAL L
      DIMENSION X(N00),SI(N00),BMOM(N00),DEFL(N00),SLOPE(N00),
     *           XX(N00),SSI(N00),BBMOM(N00),TETA(N00),CG(N00),DDEFL(N00
     *),XDEFL(N00),BMOMA(N00),SSLOPE(N00)
C
      DO 100 I=1,NN
      TETA(I)=0.0
      CG(I)=0.0
      DDEFL(I)=0.0
      XDEFL(I)=0.0
      BMOMA(I)=0.0
      SSLOPE(I)=0.0
 100  CONTINUE
C
C     ADD TWO STATION ON THE SHAFT CLOSE TO ENDS
      DO 1 I=3,N
      XX(I)=X(I-1)
      SSI(I)=SI(I-1)
      BBMOM(I)=BMOM(I-1)
 1    CONTINUE
C
      XX(1)=X(1)
      XX(2)=L/10000.
      XX(N+1)=L-L/10000.
      XX(N+2)=X(N)
      SSI(1)=SI(1)
      SSI(2)=SI(1)
      SSI(N+1)=SI(N-1)
      SSI(N+2)=SI(N)
C
      BBMOM(1)=BMOM(1)
      BBMOM(2)=(BMOM(2)-BMOM(1))/(X(2)-X(1))*(XX(2)-XX(1))+BMOM(1)
      BBMOM(N+1)=
     *          -(BMOM(N)-BMOM(N-1))/(X(N)-X(N-1))*(XX(N+2)-XX(N+1))+
     *BMOM(N)
      BBMOM(N+2)=BMOM(N)
C
C     DETERMINE AREA-MOMENT AND ROTATIONS FOR THE N+1 INTERVALS
      NNN=N+1
C
      DO 2 I=1,NNN
      IF((XX(I+1)-XX(I)).EQ.0.0) GO TO 3
      BMOMA(I)=(BBMOM(I)+BBMOM(I+1))/2.*(XX(I+1)-XX(I))
      GO TO 4
 3    CONTINUE
      BMOMA(I)=0.0
 4    CONTINUE
      TETA(I+1)=TETA(I)+BMOMA(I)/(E*SSI(I))
 2    CONTINUE
C
C     DETERMINE THE DEFLECTIONS ALONG THE SHAFT
      TDEFL=0.0
      DO 7 I=1,NNN
C
C     CENTER OF GRAVITY OF AREA MOMENT
      XA=XX(I)
      XB=XX(I+1)
      IF((XB-XA).EQ.0.0) GO TO 60
      IF(BBMOM(I).GT.BBMOM(I+1))GO TO 5
      Y1=(XB-XA)/2.
      A1=BBMOM(I)*(XB-XA)
      Y2=(XB-XA)*2./3.
      A2=(BBMOM(I+1)-BBMOM(I))*(XB-XA)/2.
      GO TO 6
 5    Y1=(XB-XA)/2.
      A1=BBMOM(I+1)*(XB-XA)
      Y2=(XB-XA)/3.
```

```
      A2=-(BBMOM(I+1)-BBMOM(I))*(XB-XA)/2.
 6    CONTINUE
      AAA=A1+A2
      IF(AAA.EQ.0.0) GO TO 60
      CG(I)=(A1*Y1+A2*Y2)/(A1+A2)
      GO TO 61
 60   CG(I)=0.0
 61   CONTINUE
      CGB=L-(XA+CG(I))
      TDEFL=BMOMA(I)*CGB/(E*SSI(I))+TDEFL
 7    CONTINUE
C
      DO 9 I=2,NN
      DEFLJ=0.0
      II=I-1
C
      DO 8 J=1,II
      CGI=XX(I)-(CG(J)+XX(J))
      DEFLJ=BMOMA(J)*CGI/(E*SSI(J))+DEFLJ
 8    CONTINUE
      XDEFL(I)=TDEFL/L*XX(I)
      DDEFL(I)=XDEFL(I)-DEFLJ
 9    CONTINUE
C
C
C     DETERMINE THE END SLOPES
      TETAA=DDEFL(2)/XX(2)
      TETAB=DDEFL(N+1)/(L-XX(N+1))
C
C     DETERMINE SLOPES ALONG THE BEAM
      DO 10 I=1,NN
      SSLOPE(I)=TETAA-TETA(I)
 10   CONTINUE
C
C     ELIMINATE THE TWO POINTS ADDED
      DEFL(1)=DDEFL(1)
      SLOPE(1)=SSLOPE(1)
      III=N-1
      DO 11 I=2,III
      DEFL(I)=DDEFL(I+1)
      SLOPE(I)=SSLOPE(I+1)
 11   CONTINUE
      DEFL(N)=DDEFL(NN)
      SLOPE(N)=SSLOPE(NN)
C
      RETURN
      END
```

```
      FUNCTION FTABLE(VAR,FUNC,XX,M)
C     ==============================
C
      DIMENSION VAR(1),FUNC(1)
C
      NEND=M-1
      DO 10 I=1,NEND
      INT=I
      IF(XX.GT.VAR(I).AND.XX.LE.VAR(I+1))GO TO 11
 10   CONTINUE
C
 11   FTABLE=FUNC(INT)+(XX-VAR(INT))*(FUNC(INT+1)-FUNC(INT))/(VAR(INT+1)
     *-VAR(INT))
C
      RETURN
      END
```

```
      SUBROUTINE SHAFTP(XDIA,DIA,RAD,A,ND,XX,YY,WORK3,WORK4)
C     ======================================================
C
      DIMENSION XDIA(ND),DIA(ND),RAD(ND),XX(1),YY(1),WORK3(1),WORK4(1)
C
      CALL DATE(THEDATE)
      CALL LETTER(8,.24,90.,.5,1.0,6HL.POPA)
      CALL LETTER(10,.125,90.,.75,1.0,THEDATE)
      CALL PLOT(1.5,0.0,-3)
C
      XN=18.
      YN=1.
      CALL LETTER(30,.2,0.0,XN,YN,27HOPTIMUM DESIGN OF THE SHAFT   )
      XN=19.5
      YN=.75
      IF(XDIA(ND).GT.500.) GO TO 11
      IF(XDIA(ND).GT.200.) GO TO 12
      IF(XDIA(ND).GT.100.) GO TO 13
      IF(XDIA(ND).GT. 50.) GO TO 14
      IF(XDIA(ND).GT. 25.) GO TO 15
      IF(XDIA(ND).GT.12.5) GO TO 16
      XSCALE=.5
      CALL LETTER(18,.1,0.0,XN,YN,18HSCALE DOUBLE SIZE   )
      GO TO 17
C
   11 XSCALE=16.
      CALL LETTER(18,.1,0.0,XN,YN,18HSCALE 3/4IN=1FT     )
      GO TO 17
   12 XSCALE=12.
      CALL LETTER(18,.1,0.0,XN,YN,18HSCALE 1IN=1FT       )
      GO TO 17
   13 XSCALE=8.
      CALL LETTER(18,.1,0.0,XN,YN,18HSCALE 1-1/2IN=1FT   )
      GO TO 17
   14 XSCALE=4.
      CALL LETTER(18,.1,0.0,XN,YN,18HSCALE 3IN=1FT       )
      GO TO 17
   15 XSCALE=2.
      CALL LETTER(18,.1,0.0,XN,YN,18HSCALE HALF FULL     )
      GO TO 17
   16 XSCALE=1.
      CALL LETTER(18,.1,0.0,XN,YN,18HSCALE FULL          )
   17 CONTINUE
C
      YSCALE=XSCALE
      V=-2.*XSCALE
      W=-7.*YSCALE
      XMIN=-1.8*XSCALE
      XMAX=60.*XSCALE
      YMIN=-6.5*YSCALE
      YMAX=3.*YSCALE
C
      CALL PLTIN(XSCALE,YSCALE,V,W,XMIN,XMAX,YMIN,YMAX)
C
C     PLOT THE CENTER LINE
      X1=0.0
      X2=XDIA(ND)
      Y1=0.0
      Y2=0.0
      CALL UNITTO(X1,Y1,XP1,YP1)
      XX(1)=XP1-.5
      CALL UNITTO(X2,Y2,XP2,YP2)
      XX(2)=XP2+.5
      YY(1)=YP1
      YY(2)=YP2
      NDIM=2
      DASH1=1.0
      DASH2=.1
      GAP=.15
      UNIT=.3
      N=1
      CALL DDASHM(XX,YY,NDIM,DASH1,DASH2,N,GAP,UNIT,IE,WORK3,WORK4)
C
C     PLOT THE SHAFT
```

```
C     DRAW VERTICAL LINES MARKING SHAFT SHOULDERS                       SHA 740
C                                                                       SHA 750
      X1=XDIA(1)                                                        SHA 760
      Y1=DIA(1)/2.                                                      SHA 770
      X2=X1                                                             SHA 780
      Y2=-Y1                                                            SHA 790
      CALL PLTLN(X1,Y1,X2,Y2)                                           SHA 800
      DO 1 I=2,ND                                                       SHA 810
      IF(I.EQ.ND) GO TO 10                                              SHA 820
      IF(DIA(I).EQ.DIA(I-1)) GO TO 1                                    SHA 830
 10   X1=XDIA(I)                                                        SHA 840
      Y1=DIA(I)/2.                                                      SHA 850
      IF(DIA(I).LT.DIA(I-1)) Y1=DIA(I-1)/2.                             SHA 860
      X2=X1                                                             SHA 870
      Y2=-Y1                                                            SHA 880
      CALL PLTLN(X1,Y1,X2,Y2)                                           SHA 890
 1    CONTINUE                                                          SHA 900
C                                                                       SHA 910
C     DRAW HORIZONTAL LINES AND QUARTER CIRCLES FOR SHOULDER RADIUS     SHA 920
      DEV=.005                                                          SHA 930
      NND=ND-1                                                          SHA 940
      DO 4 I=1,NND                                                      SHA 950
      X1=XDIA(I)                                                        SHA 960
      Y1=DIA(I)/2.                                                      SHA 970
      X2=XDIA(I+1)                                                      SHA 980
      Y2=Y1                                                             SHA 990
      IF(RAD(I).GE.10000.) GO TO 2                                      SHA1000
      IF(DIA(I).GT.DIA(I-1)) GO TO 2                                    SHA1010
      XA=XDIA(I)                                                        SHA1020
      YA=DIA(I)/2.+RAD(I)                                               SHA1030
      XB=XDIA(I)+RAD(I)                                                 SHA1040
      YB=DIA(I)/2.                                                      SHA1050
      XC=XB                                                             SHA1060
      YC=YA                                                             SHA1070
      CALL PLTARC(XA,YA,XB,YB,XC,YC,DEV)                                SHA1080
      CALL UNITTO(XC,YC,XCP,YCP)                                        SHA1090
      X3=XCP+.15                                                        SHA1100
      Y3=YCP+.15                                                        SHA1110
      X4=XC-RAD(I)/SQRT(2.)                                             SHA1120
      Y4=YC-RAD(I)/SQRT(2.)                                             SHA1130
      CALLUNITTO(X4,Y4,X4P,Y4P)                                         SHA1140
      CALL ARROW(X3,Y3,X4P,Y4P,3)                                       SHA1150
      RADM=RAD(I)                                                       SHA1160
      XN=X3+.03                                                         SHA1170
      CALL NUMBER(XN,Y3,.100,RADM,0.0,6H(F5.3))                         SHA1180
      XN=X3+.55                                                         SHA1190
      CALL LETTER(10,.100,0.0,XN,Y3,3HRAD)                              SHA1200
      X1=XB                                                             SHA1210
      Y1=YB                                                             SHA1220
      YA=-YA                                                            SHA1230
      YB=-YB                                                            SHA1240
      YC=-YC                                                            SHA1250
      CALL PLTARC(XB,YB,XA,YA,XC,YC,DEV)                                SHA1260
C                                                                       SHA1270
 2    CONTINUE                                                          SHA1280
      IF(RAD(I+1).GE.10000.) GO TO 3                                    SHA1290
      IF(DIA(I).GT.DIA(I+1)) GO TO 3                                    SHA1300
      XA=XDIA(I+1)-RAD(I+1)                                             SHA1310
      YA=DIA(I)/2.                                                      SHA1320
      XB=XDIA(I+1)                                                      SHA1330
      YB=DIA(I)/2.+RAD(I+1)                                             SHA1340
      XC=XA                                                             SHA1350
      YC=YB                                                             SHA1360
      CALL PLTARC(XA,YA,XB,YB,XC,YC,DEV)                                SHA1370
      CALL UNITTO(XC,YC,XCP,YCP)                                        SHA1380
      X3=XCP-.15                                                        SHA1390
      Y3=YCP+.15                                                        SHA1400
      X4=XC+RAD(I+1)/SQRT(2.)                                           SHA1410
      Y4=YC-RAD(I+1)/SQRT(2.)                                           SHA1420
      CALL UNITTO(X4,Y4,X4P,Y4P)                                        SHA1430
      CALL ARROW(X3,Y3,X4P,Y4P,3)                                       SHA1440
      RADM=RAD(I+1)                                                     SHA1450
      XN=X3-.85                                                         SHA1460
      CALL NUMBER(XN,Y3,.100,RADM,0.0,6H(F5.3))                         SHA1470
```

```
        XN=X3-.32                                                       SHA1480
      CALL LETTER(10,.100,0.0,XN,Y3,3HRAD)                              SHA1490
      X2=XA                                                             SHA1500
      Y2=YA                                                             SHA1510
      YA=-YA                                                            SHA1520
      YB=-YB                                                            SHA1530
      YC=-YC                                                            SHA1540
      CALL PLTARC(XB,YB,XA,YA,XC,YC,DEV)                                SHA1550
 3    CONTINUE                                                          SHA1560
      CALL PLTLN(X1,Y1,X2,Y2)                                           SHA1570
      Y1=-Y1                                                            SHA1580
      Y2=-Y2                                                            SHA1590
      CALL PLTLN(X1,Y1,X2,Y2)                                           SHA1600
 4    CONTINUE                                                          SHA1610
C                                                                       SHA1620
C     DRAW DIMENSION LINES                                              SHA1630
      DO 5 I=1,ND                                                       SHA1640
      X1=XDIA(I)                                                        SHA1650
      Y1=-DIA(I)/2.                                                     SHA1660
      X2=X1                                                             SHA1670
      Y2=-DIA(1)/2.-(FLOAT(I)*.38+.2)*YSCALE                            SHA1680
      IF(XDIA(I).EQ.0.0) GO TO 40                                       SHA1690
      IF(XDIA(I).EQ.A) GO TO 40                                         SHA1700
      CALL PLTLN(X1,Y1,X2,Y2)                                           SHA1710
      GO TO 5                                                           SHA1720
 40   CONTINUE                                                          SHA1730
      CALL UNITTO(X1,Y1,XP1,YP1)                                        SHA1740
      XX(1)=XP1                                                         SHA1750
      YY(1)=YP1                                                         SHA1760
      Y2=-DIA(1)/2.                                                     SHA1770
      CALL UNITTO(X2,Y2,XP2,YP2)                                        SHA1780
      YP2=YP2-FLOAT(ND)*.38-.2                                          SHA1790
      XX(2)=XP2                                                         SHA1800
      YY(2)=YP2                                                         SHA1810
      CALL DDASHM(XX,YY,NDIM,DASH1,DASH2,N,GAP,UNIT,IE,WORK3,WORK4)     SHA1820
      XN=XP1-.06                                                        SHA1830
      YN=YP2+1.                                                         SHA1840
      CALL LETTER(10,.1,90.,XN,YN,10HCL.BEARING)                        SHA1850
 5    CONTINUE                                                          SHA1860
C                                                                       SHA1870
      DO 6 I=1,NND                                                      SHA1880
      X1=XDIA(1)                                                        SHA1890
      Y1=-DIA(1)/2.-(FLOAT(I)*.38+.5)*YSCALE                            SHA1900
      CALL UNITTO(X1,Y1,XP1,YP1)                                        SHA1910
      X2=XDIA(I+1)                                                      SHA1920
      Y2=Y1                                                             SHA1930
      CALL UNITTO(X2,Y2,XP2,YP2)                                        SHA1940
      CALL ARROW(XP1,YP1,XP2,YP1,1)                                     SHA1950
      DIST=XDIA(I+1)-XDIA(1)                                            SHA1960
      XN=DIST/2.5                                                       SHA1970
      YN=Y1+.06*YSCALE                                                  SHA1980
      CALL UNITTO(XN,YN,XPN,YPN)                                        SHA1990
      CALL NUMBER(XPN,YPN,.100,DIST,0.0,6H(F7.3))                       SHA2000
      X1=XDIA(I)+(XDIA(I+1)-XDIA(I))/1.8                                SHA2010
      Y1=DIA(I)/2.                                                      SHA2020
      X2=X1                                                             SHA2030
      Y2=-Y1                                                            SHA2040
      CALL UNITTO(X1,Y1,XP1,YP1)                                        SHA2050
      CALL UNITTO(X2,Y2,XP2,YP2)                                        SHA2060
      CALL ARROW(XP1,YP1,XP2,YP2,1)                                     SHA2070
      XN=X1-.06*XSCALE                                                  SHA2080
      YN=-.4*DIA(I)                                                     SHA2090
      DIAM=DIA(I)                                                       SHA2100
      CALL UNITTO(XN,YN,XPN,YPN)                                        SHA2110
      CALL NUMBER(XPN,YPN,.100,DIAM,90.,6H(F7.3))                       SHA2120
 6    CONTINUE                                                          SHA2130
      RETURN                                                            SHA2140
      END                                                               SHA2150
```

```
      SUBROUTINE NUMBER(X,Y,HEIGHT,ANUM,THETA,FMT)                      NUM  10
C     ============================================
      DIMENSION BCD(1)                                                  NUM  20
      ENCODE(10,FMT,BCD)ANUM                                            NUM  30
      CALL SYMBOL(X,Y,HEIGHT,BCD,THETA,10)                              NUM  40
      RETURN                                                            NUM  50
      END                                                               NUM  60
```

APPENDIX B

USERS' MANUAL

DESIGN OPTIMIZATION
OF A SHAFT ON TWO BEARINGS

USERS' MANUAL

PROGRAM SHAFT 1
McMaster University
Mechanical Engineering Dept.

Laurentiu Popa
Prof. James N. Siddall

1. PURPOSE

Given a shaft of known geometrical configuration and loading, this program will determine diameter sizes along the shaft for which the weight of the shaft will be minimum and the following constraints will be satisfied:

- the stress at any point along the shaft will be below a user set allowable value;
- the maximum deflection will be less than an allowable deflection specified by user;
- the slope of the shaft in bearings will be below an admissible value;
- the torsional deflection will be less than the allowable value;
- the critical speed will be greater than a user specified value;
- the shoulder size at specified points along the shaft will be greater than a specified value (for shoulders adjacent to gear hubs, couplings, bearings, etc.).

Certain diameters will remain of fixed size specified by user (as it may be required by the bearing size, standard parts on shaft, etc.).

2. GEOMETRICAL CONFIGURATION

The shaft is supported on two bearings and can have a cantilever extension (see the example shown on Fig. 1).

The bearings permit slopes of the shaft at bearing point and one of them is free to translate along the axis of the shaft.

The location of the bearings and the points where diameter changes size have to be specified by user. Shoulder radius should also be specified by user.

The user can further restrict the configuration of the shaft by specifying a diameter increase or decrease of a minimum value at any point along the shaft. (e.g. He may wish to specify a minimum shoulder size adjacent to bearing, a gear, coupling, etc.)

3. LOADING

The program can take any concentrated or uniformly distributed forces perpendicular to axis or shaft projected in the vertical and horizontal planes and any torsional moments (torques). Gravitational forces attached to the shaft shall be specified separately, these forces being the only ones considered in calculating of the critical speed.

4. METHOD

The input data consisting of information describing geometrical configuration, loading, allowable stresses, deflections, etc. are read and information pertaining to different points of the shaft is arranged in arrays in an order following the axis of the shaft.

The optimization process begins with starting diameter sizes supplied by the user in input data. Deflections and stresses are calculated in all the points along the shaft that are specified in input data. The natural frequency is also calculated.

An unconstrained artificial objective function is calculated. It is defined as follows:

$$P(d_1, d_2 \ldots d_n, r) = W(d_1, d_2 \ldots d_n) + r \sum_{k=1}^{p} \frac{1}{\Phi_k(d_1, d_2 \ldots d_n)}$$

where

$d_1, d_2 \ldots d_n$ are the unknown diameters

$W(d_1, d_2 \ldots d_n)$ is the weight at the shaft

$\Phi_k(d_1, d_2 \ldots d_n) \geqslant 0$ $(k = 1, 2 \ldots p)$ are inequality constraints limiting stresses, slopes, critical speed and diameter increase or decrease

r is a positive parameter that is 1.0 at the beginning of the optimization and decreases after each cycle.

A series of steps are taken with varying diameter size following a direct search method and the objective function is minimized until no further improvement can be found.

When calculating deflections, stresses and natural frequencies, the following procedure is applied:

Using the superposition principle, each force is taken alone and reactions in bearings and moments along the shaft are calculated.

The area-moment method is applied to calculate deflections in all required points along the shaft.

The effects of individual forces are totalized in all specified points for all gravitational forces, exterior vertical forces and horizontal forces.

Using the deflections caused by gravitational forces attached to the shaft, the frequency of fundamental mode is calculated for transverse vibrations following Rayleigh's method.

Stresses are calculated in the horizontal and vertical planes and then the geometrical resultant of moments, deflections and stresses are calculated all along the shaft.

The torsional deflection and the shear stress caused by torsion is calculated next and then the combined stress is calculated using the maximum strain theory:

$$\sigma_c = .35\ \sigma + .65 \sqrt{\sigma^2 + \tau^2}$$

where:

σ = normal stress (bending)

τ = shear stress (torsion)

σ_c = combined stress

The intensified stress that appears in some points due to shaft shoulders, keyways, grooves, etc. is calculated by multiplying the stresses calculated as shown above by a stress concentration factor. The stress concentration factor at shaft shoulders is selected by this program from Tables in (6), pages 386 and 388, making use of linear interpolation. Other stress concentration factors are supplied by user. They could also be selected from (6).

The optimum design of the shaft is plotted on a Benson-Lehner Plotter at a suitable scale.

5. INPUT DATA

It is useful to draw a sketch as in the example in Fig. 1 showing the geometrical configuration supports and loading.

Table 1 shows the data cards used to run the program Shaft 1 for the optimization of the shaft shown in Fig. 1. Only the first 40 columns on the cards are shown, their being the only ones used. On the side of the cards is described what the figures represent.

All data fields are 10 columns long. The data should be right justified in their fields. Put the decimal point of real numbers in same field as in example. Put integers right justified without decimal point.

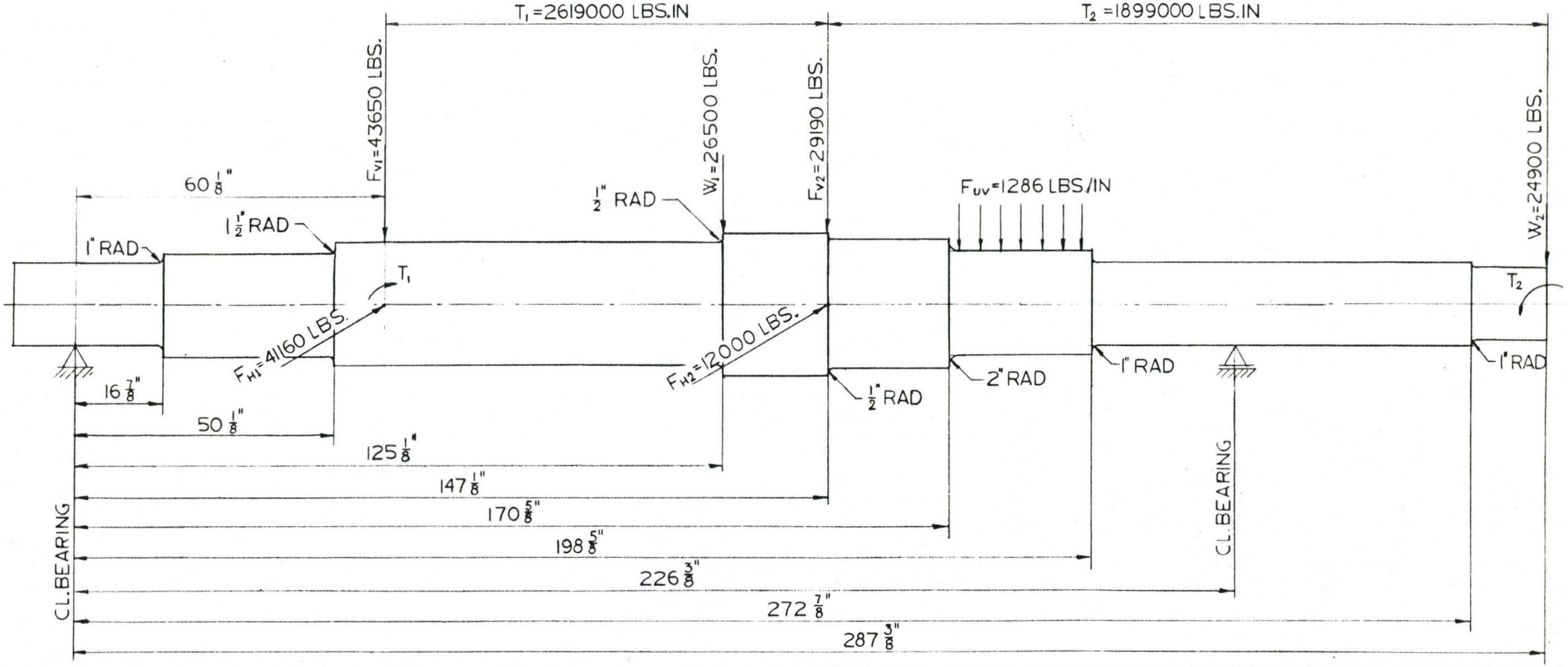

Fig. 1

SHAFT SKETCH FOR SAMPLE RUN

TABLE 1 - INPUT DATA FOR SAMPLE RUN (SHAFT SHOWN IN FIG. 1)

DATA CARDS (Beginning Shown Only)			
1 10	20	30	DESCRIPTION OF DATA
226.375	61.000		Length of first span and length of second span
2	2	2	Number of readings for weights attached to shaft, vertical and horizontal forces
125.125	26500.		Co-ordinate, Force
287.375	24900.		
60.625	41160.		
147.125	12000.		
0	1	0	Number of readings for uniformly distributed weights attached to shaft, vertical and horizontal uniformly distributed forces
170.625	198.625	1286.	Co-ordinate of beginning, co-ordinate of end uniformly distributed force
2			Number of readings for torsional moments
60.625	147.125	2619000.	Co-ordinate of beginning, co-ordinate of the end, torsional moment
147.125	287.375	1899000.	
10			Number of readings for diameters
0.000	16.000		Co-ordinate of point where given diameter starts, diameter size
16.875	20.000		
50.125	24.000		
125.125	26.000		
147.125	24.000		
170.625	20.000		
198.625	16.000		
226.375	16.000		
272.875	14.000		
287.375	14.000		
1			Number of readings for stress concentration factor
181.625	1.300		Co-ordinate, stress concentration factor
7			Number of readings for shoulder radii
16.875	1.000		Co-ordinate, Radius
50.125	1.500		
125.125	0.500		
147.125	0.500		
170.625	2.000		
198.625	1.000		
272.875	1.000		
8000.			Allowable stress

0.05000		Allowable bending deflection
0.00100		Allowable slope in bearings
0.01000		Allowable torsional deflection
60.		Minimum critical speed
3		Number of readings for diameters of fixed size
0.000	16.000	Co-ordinate, Diameter
198.625	16.000	
272.875	14.000	
1		Number of readings for minimum diameter increase/decrease
16.875	1.000	Co-ordinate of shoulder, minimum diameter increase/decrease

All X co-ordinates of points on shaft are measured from the first bearing (bearing A in our example). Within each group of data, e.g. weights attached to the shaft, vertical forces, etc. keep the cards in a sequence following the X co-ordinates of points of the shaft in an increasing order. Below is shown the complete set of input data and their sequence:

1. Length of first span and length of second span (in.).
2. Number of weights attached to shaft, number of vertical forces and number of horizontal forces.
3. X co-ordinate (in.) and weight attached to shaft (lbs.). Do not show own weight of shaft. This is going to be calculated internally by the program.
4. X co-ordinate and exterior vertical force.
5. X co-ordinate and horizontal force.
6. Number of uniformly distributed weights attached to shaft, number of exterior vertical uniformly distributed forces and number of horizontal uniformly distributed forces.
7. X co-ordinate for beginning of uniformly distributed weight, X co-ordinate for end of uniformly distributed weight and uniformly distributed weight (lbs./in.).
8. Same as 7 but for exterior vertical uniformly distributed forces.
9. Same as 7 but for horizontal uniformly distributed forces.
10. Number of torsion moments.
11. X co-ordinate for beginning of torsion moment, X co-ordinate for end of torsion moment and torsion moment in lbs./in.

12. Number of diameter readings.

13. X co-ordinate where diameter of given size starts and diameter size (ins.).

14. Number of readings for stress concentration factor.

15. X co-ordinate and stress concentration factor. Do not give stress concentration factor for shoulder. This will be calculated internally.

16. Number of readings for radii.

17. X co-ordinate and shoulder radius.

18. Allowable stress (lbs./sq. in.).

19. Allowable bending deflection (in.).

20. Allowable slope in bearings (rad.).

21. Allowable torsional deflection (rad.).

22. Minimum critical speed (rpm.).

23. Number of diameters of fixed size.

24. X co-ordinate and diameter size for sections where diameter must remain of given value.

25. Number of readings for diameter increase or decrease.

26. X co-ordinate and minimum diameter increase, or decrease.

6. OUTPUT DATA

The following data are printed out:

- input data as supplied by the user;
- data supplied to the direct search optimization Subroutine SEEK3;

- the objective function and the independent variables (unknown diameter sizes) after each optimization cycle;
- the optimum solution consisting of the independent variables and inequality constraints;
- reactions in bearings;
- bending moments and torsional moments;
- diameters;
- bending deflections, slopes and torsional deflections;
- bending stresses, torsion stresses, combined stresses, intensification factors and intensified stresses;
- weight of shaft;
- critical speed.

7. COMMENTS

If a feasible solution is not found during the first minimization cycle, an error message is printed out. In this case, another set of starting diameters should be tried.

If the method has not converged, the current solution is printed out. A new run should be tried with these current diameters as start diameters.

8. SAMPLE RUN

The program has been run for a shaft of geometrical configuration and loading as shown on Sketch in Fig. 1.

Below is the computer print-out for this run and the plot of the optimum design (Fig.2)

```
            SHAFT DESIGN OPTIMIZATION
            *************************


            INPUT DATA
            **********

LENGTH OF FIRST SPAN    226.375 IN
LENGTH OF SC≠ND SPAN     61.000 IN



WEIGHTS ATTACHED TO SHAFT
      X            WEIGHT
  125.125           26500.
  287.375           24900.

VERTICAL FORCES
      X             FORCE
   60.625           43650.
  147.125           29190.

HORIZONTAL FORCES
      X             FORCE
   60.625           41160.
  147.125           12000.

VERTICAL UNIFORMLY DISTRIBUTED FORCES
      X START        X END          FORCE

  170.625       198.625           1286.

TORSIONAL MOMENTS
    X START       X END        MOMENT

   60.625       147.125       2619000.
```

```
   147.125          287.375       1899000.

DIAMETERS
      X                 DIA
     0.000            16.000
    16.875            20.000
    50.125            25.000
   125.125            26.000
   147.125            24.000
   170.625            20.000
   198.625            16.000
   226.375            16.000
   272.875            14.000
   287.375            14.000

STRESS  CONCENTRATION  FACTORS
      X                FACTOR
   181.625             1.300

RADII
      X                 RAD
    16.875             1.000
    50.125             1.500
   125.125              .500
   147.125              .500
   170.625             2.000
   198.625             1.000
   272.875             1.000
```

```
DIMENSIONS AND LOADING

==============================================================================================================================================
POINT       X        DIA       RADII    STRESS           CONC.FORCE                               UNIF.DISTR.FORCE           TORS.MOM.   REMARKS
NO                                      CONC.FCT.  WEIGHT      VERT.       HORIZ.     WEIGHT       VERT.        HORIZ.
            IN        IN          IN                   LB          LB          LB     LB/IN        LB/IN        LB/IN         LBIN
==============================================================================================================================================
   1      0.000    16.000      0.000      1.00         0.          0.          0.        57.          0.           0.           0.   SUPPORT POINT
   2     16.875    16.000      1.000      1.00         0.          0.          0.        57.          0.           0.           0.
   3     16.875    20.000      0.000      1.00         0.          0.          0.        89.          0.           0.           0.
   4     50.125    20.000      1.500      1.00         0.          0.          0.        89.          0.           0.           0.
   5     50.125    25.000      0.000      1.00         0.          0.          0.       139.          0.           0.           0.
   6     60.625    25.000      0.000      1.00         0.          0.          0.       139.          0.           0.           0.
   7     60.625    25.000      0.000      1.00         0.      43650.      41160.      139.          0.           0.     2619000.
   8    125.125    25.000       .500      1.00         0.          0.          0.       139.          0.           0.     2619000.
   9    125.125    26.000      0.000      1.00     26500.          0.          0.       150.          0.           0.     2619000.
  10    147.125    26.000       .500      1.00         0.          0.          0.       150.          0.           0.     2619000.
  11    147.125    24.000      0.000      1.00         0.      29190.      12000.      128.          0.           0.     1899000.
  12    170.625    24.000      2.000      1.00         0.          0.          0.       128.          0.           0.     1899000.
  13    170.625    20.000      0.000      1.00         0.          0.          0.        89.       1286.           0.     1899000.
  14    181.625    20.000      0.000      1.00         0.          0.          0.        89.       1286.           0.     1899000.
  15    198.625    20.000      1.000      1.00         0.          0.          0.        89.       1286.           0.     1899000.
  16    198.625    16.000      0.000      1.00         0.          0.          0.        57.          0.           0.     1899000.
  17    226.375    16.000      0.000      1.00         0.          0.          0.        57.          0.           0.     1899000.   SUPPORT POINT
  18    272.875    16.000      1.000      1.00         0.          0.          0.        57.          0.           0.     1899000.
  19    272.875    14.000      0.000      1.00         0.          0.          0.        44.          0.           0.     1899000.
  20    287.375    14.000      0.000      1.00     24900.          0.          0.         0.          0.           0.     1899000.   END OF SHAFT
```

```
ALLOWABLE STRESS                                  8000. LB/SQIN

ALLOWABLE BENDING DEFLECTION                     .05000 IN

ALLOWABLE SLOPE IN BEARINGS                      .00100 RAD

ALLOWABLE TORSIONAL DEFLECTION                   .01000 RAD

MINIMUM CRITICAL SPEED                              60. RPM


DIAMETERS OF FIXED SIZE
     X          DIA

     0.000    16.000

   198.625    16.000

   272.875    14.000


MINIMUM DIAMETER INCREASE/DECREASE
     X          DIA.INCR.

    16.875     1.000
```

```
OPTIMIZATION USING DIRECT SEARCH METHOD SEEK3

NUMBER OF INDEPENDENT VARIABLES . . . . . . . . . .          N   =      5
INTERMEDIATE OUTPUT EVERY IPRINT(TH) CYCLE. . . .      IPRINT   =      1
INPUT DATA IS PRINTED OUT FOR IDATA=1 ONLY. . . .       IDATA   =      1
NUMBER OF INEQUALITY (.GE.) CONSTRAINTS . . . . .       NCONS   =      7
NUMBER OF EQUALITY CONSTRAINTS. . . . . . . . . .       NEQUS   =      0
ESTIMATED LOWER BOUND ON RANGE OF X(I). . . . . . RMIN(I)       =
   .18000000E+02   .22500000E+02   .23400000E+02   .21600000E+02   .18000000E+02
ESTIMATED UPPER BOUND ON RANGE OF X(I). . . . . . RMAX(I)       =
   .22000000E+02   .27500000E+02   .28600000E+02   .26400000E+02   .22000000E+02

STARTING VALUES OF X(I) . . . . . . . . . . . . . XSTRT(I)      =
   .20000000E+02   .25000000E+02   .26000000E+02   .24000000E+02   .20000000E+02
FRACTION OF RANGE USED AS STEP SIZE . . . . . . .           F   =      .20000000E-01
STEP SIZE FRACTION USED AS CONVERGENCE CRITERION.           G   =      .10000000E+00
MAXIMUM NUMBER OF MOVES PERMITTED . . . . . . . .        MAXM   =     20
PENALTY MULTIPLIER USED IN SEEK3. . . . . . . . .           R   =      .10000000E+01
REDUCTION FACTOR FOR (R) AFTER EACH MINIMIZATION.      REDUCE   =      .50000000E-01
```

```
                                                          INDEPENDENT VARIABLES  X(I)

R  =    .10000000E+01
U  =    .27694325E+05     .23120000E+02    .24918750E+02    .23920000E+02    .22170000E+02
                          .13850000E+02

R  =    .50000000E-01
U  =    .25850680E+05     .20160000E+02    .24118750E+02    .24050000E+02    .22074000E+02
                          .17790000E+02

R  =    .25000000E-02
U  =    .25723025E+05     .19830000E+02    .24018750E+02    .24154000E+02    .22026000E+02
                          .17910000E+02

R  =    .12500000E-03
U  =    .25707684E+05     .19830000E+02    .24018750E+02    .24199500E+02    .22014000E+02
                          .17805000E+02

R  =    .62500000E-05
U  =    .25703289E+05     .19815000E+02    .24018750E+02    .24199500E+02    .22014000E+02
                          .17805000E+02

R  =    .31250000E-06
U  =    .25701925E+05     .19810000E+02    .24018750E+02    .24199500E+02    .22014000E+02
                          .17805000E+02

R  =    .15625000E-07
U  =    .25701825E+05     .19810000E+02    .24018750E+02    .24199500E+02    .22014000E+02
                          .17805000E+02
```

```
                                  OPTIMUM SOLUTION FOUND
                               MINIMUM   U =     .25701825E+05

                                     X( 1) =     .19810000E+02
                                     X( 2) =     .24018750E+02
                                     X( 3) =     .24199500E+02
                                     X( 4) =     .22014000E+02


                                     X( 5) =     .17805000E+02

INEQUALITY CONSTRAINTS
                                   PHI( 1) =     .58651522E+02
                                   PHI( 2) =     .26106735E-05
                                   PHI( 3) =     .10094692E-03
                                   PHI( 4) =     .41729940E-03
                                   PHI( 5) =     .62241782E-02
                                   PHI( 6) =     .10518570E+04
                                   PHI( 7) =     .28100009E+01
```

```
OUTPUT DATA
***********

REACTIONS
      X        VERTICAL     HORIZONTAL   REZULTANT
     0.000       57409.        34338.       66895.
   226.375       88216.        18822.       90201.



 BENDING MOMENTS AND TORSIONAL MOMENTS
=========================================================================
  NO        X               BENDING MOMENT                     TORSION MOMEN
                      VERT             HORIZ           RES
                             LBIN                                   LBIN
=========================================================================

    1     0.000              0.              0.              0.              0.
    2    16.875         960678.         579454.        1121904.              0.
    3    16.875         960678.         579454.        1121904.              0.
    4    50.125        2788459.        1721193.        3276891.              0.
    5    50.125        2788459.        1721193.        3276891.              0.
    6    60.625        3350134.        2081743.        3944242.              0.
    7    60.625        3350134.        2081743.        3944242.        2619000.
    8   125.125        3696029.        1641725.        4044242.        2619000.
    9   125.125        3696029.        1641725.        4044242.        2619000.
   10   147.125        3096085.        1491642.        3436676.        2619000.
   11   147.125        3096085.        1491642.        3436676.        1899000.
   12   170.625        1730431.        1049325.        2023728.        1899000.
   13   170.625        1730431.        1049325.        2023728.        1899000.
   14   181.625        1085810.         842283.        1374200.        1899000.
   15   198.625          81265.         522310.         528594.        1899000.
   16   198.625          81265.         522310.         528594.        1899000.
   17   226.375       -1580417.             -0.        1580417.        1899000.
   18   272.875        -361050.              0.         361050.        1899000.
   19   272.875        -361050.              0.         361050.        1899000.
   20   287.375             -0.              0.              0.        1899000.
```

```
DEFLECTIONS,SLOPES AND ROTATIONS

==================================================================================================================
 NP         X         DIA                  DEFLECTION                    SLOPE                      ROTATION
                                 VERT        HORIZ        RES        VERT         HORIZ       RES
           IN          IN        IN          IN           IN     RAD            RAD         RAD           RAD
==================================================================================================================

  1      0.000      16.000   0.000000   0.000000   0.000000    .000776     .000454     .000899   0.000000
  2     16.875      16.000    .012617    .007380    .014617    .000692     .000404     .000801   0.000000
  3     16.875      19.810    .012617    .007380    .014617    .000692     .000404     .000801   0.000000
  4     50.125      19.810    .031754    .018459    .036729    .000415     .000235     .000477   0.000000
  5     50.125      24.019    .031754    .018459    .036729    .000415     .000235     .000477   0.000000
  6     60.625      24.019    .035776    .020719    .041342    .000349     .000194     .000400   0.000000
  7     60.625      24.019    .035776    .020719    .041342    .000349     .000194     .000400   0.000000
  8    125.125      24.019    .043280    .025031    .049997   -.000124    -.000051     .000134    .000450
  9    125.125      24.200    .043280    .025031    .049997   -.000124    -.000051     .000134    .000450
 10    147.125      24.200    .038872    .023151    .045244   -.000272    -.000119     .000297    .000598
 11    147.125      22.014    .038872    .023151    .045244   -.000272    -.000119     .000297    .000598
 12    170.625      22.014    .030363    .019279    .035967   -.000436    -.000205     .000482    .000767
 13    170.625      17.805    .030363    .019279    .035967   -.000436    -.000205     .000482    .000767
 14    181.625      17.805    .024943    .016619    .029973   -.000541    -.000276     .000607    .000951
 15    198.625      17.805    .015012    .011214    .018738   -.000608    -.000354     .000704    .001235
 16    198.625      16.000    .015012    .011214    .018738   -.000608    -.000354     .000704    .001235
 17    226.375      16.000   -.000000   -.000000    .000000   -.000394    -.000429     .000583    .001948
 18    272.875      16.000   -.005350   -.019956    .020660    .000066    -.000429     .000434    .003141
 19    272.875      14.000   -.005350   -.019956    .020660    .000066    -.000429     .000434    .003141
 20    287.375      14.000   -.003944   -.026178    .026474    .000112    -.000429     .000444    .003776
```

```
STRESSES

=====================================================================================================================================
 NR         X                    BENDING STRESSES              TORSION       COMBINED       INT FCT                     INTENSIFIED STRESS
                      VERT          HORIZ          RES         STRESS         STRESS       BEND          TORS          BEND          TORS          COMB
            IN        PSI           PSI            PSI          PSI            PSI                                      PSI           PSI           PSI
=====================================================================================================================================

   1      0.000       0.000         0.000         0.000        0.000          0.000        1.000         1.000         0.000         0.000         0.000
   2     16.875    2389.010      1440.984      2789.947        0.000       2789.947        1.935         1.516      5397.222         0.000      5397.222
   3     16.875    1258.706       759.217      1469.950        0.000       1469.950        1.000         1.000      1469.950         0.000      1469.950
   4     50.125    3653.516      2255.155      4293.472        0.000       4293.472        1.850         1.429      7941.348         0.000      7941.348
   5     50.125    2049.808      1265.257      2408.856        0.000       2408.856        1.000         1.000      2408.856         0.000      2408.856
   6     60.625    2462.697      1530.298      2899.429        0.000       2899.429        1.000         1.000      2899.429         0.000      2899.429
   7     60.625    2462.697      1530.298      2899.429      962.619       3277.065        1.000         1.000      2899.429       962.619      3277.065
   8    125.125    2716.966      1206.839      2972.939      962.619       3342.750        1.303         1.043      3873.151      1004.423      4191.627
   9    125.125    2656.539      1179.998      2906.819      941.210       3268.405        1.000         1.000      2906.819       941.210      3268.405
  10    147.125    2225.327      1072.125      2470.128      941.210       2883.215        2.000         1.532      4940.256      1441.584      5447.112
  11    147.125    2956.080      1424.189      3281.269      906.563       3585.220        1.000         1.000      3281.269       906.563      3585.220
  12    170.625    1652.181      1001.875      1932.215      906.563       2398.577        1.676         1.330      3238.169      1206.106      3757.984
  13    170.625    3122.688      1893.583      3651.963     1713.441       4533.406        1.000         1.000      3651.963      1713.441      4533.406
  14    181.625    1959.423      1519.961      2479.842     1713.441       3617.463        1.300         1.300      3223.795      2227.473      4702.702
  15    196.625     146.648       942.546       953.886     1713.441       2646.017        1.803         1.256      1720.226      2151.500      3614.251
  16    198.625     202.089      1298.878      1314.505     2361.213       3646.352        1.000         1.000      1314.505      2361.213      3646.352
  17    226.375   -3930.174         -.000      3930.174     2361.213       5369.099        1.000         1.000      3930.174      2361.213      5369.099
  18    272.875    -897.858         0.000       897.858     2361.213       3438.614        1.820         1.305      1634.101      3080.967      4715.639
  19    272.875   -1340.242         0.000      1340.242     3524.609       5133.157        1.000         1.000      1340.242      3524.609      5133.157
  20    287.375       -.000         0.000          .000     3524.609       4581.992        1.000         1.000          .000      3524.609      4581.992


WEIGHT OF SHAFT=     25702. LBS

ANGULAR FREQUENCY OF FUNDAMENTAL MODE OF VIBRATIONS=    116.433RAD/SEC
CRITICAL SPEED=  1111.857RPM
```

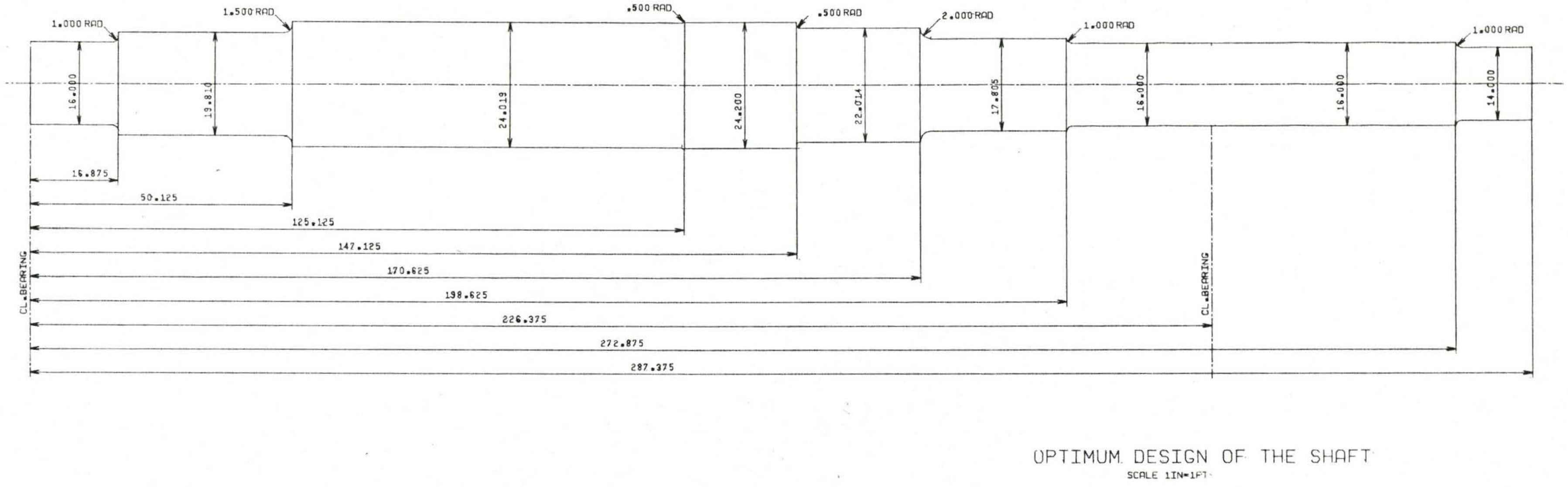

Fig. 2

PLOT OF THE SHAFT
(Reduced Plot Size)