T.&A.M. REPORT NO. 281

A METHOD FOR DETERMINING AN OPTIMUM SHAPE OF A CLASS OF THIN SHELLS OF REVOLUTION

GPO PRICE $ ____________

CSFTI PRICE(S) $ ____________

Hard copy (HC) 3.00

Microfiche (MF) .50

ff 653 July 65

by

Morris Stern

Han-chung Wang

Will J. Worley

NsG-434

Prepared Under Grant No. (NGR-14-005-010)

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

WASHINGTON, D.C.

DEPARTMENT OF THEORETICAL AND APPLIED MECHANICS

UNIVERSITY OF ILLINOIS

URBANA, ILLINOIS

A METHOD FOR DETERMINING AN OPTIMUM SHAPE OF A CLASS OF THIN SHELLS OF REVOLUTION

by

Morris Stern
Han-chung Wang
Will J. Worley

Prepared under Grant No. NGR 14-005-010

by the

Department of Theoretical and Applied Mechanics
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS

for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D.C.
July, 1965

TABLE OF CONTENTS

A METHOD FOR DETERMINING AN OPTIMUM SHAPE OF A CLASS OF THIN SHELLS OF REVOLUTION

by

Morris Stern, Han-chung Wang, Will J. Worley

Department of Theoretical and Applied Mechanics
University of Illinois
Urbana, Illinois

SUMMARY

33503

This third report under the current grant is concerned with a method for determining an optimum shape of a convex shell of revolution with respect to volume, weight and length.

The technique used depends on replacing the class of functions, over which the shape may range, by the parameters b/a, α and β in the equation

$$\left|\frac{x}{a}\right|^{\alpha} + \left|\frac{y}{b}\right|^{\beta} = 1$$

where a, b, α and β are positive constants not necessarily integers, with α and β equal to or greater than unity. The bodies of revolution are generated by revolving the line, described by the above equation, about the x-axis.

The procedure is illustrated for a thin shell which will fit within the space defined by a circular cylinder of radius b and length 2a. The shell is optimized, in terms of α and β, with respect to volume and weight. The FORTRAN program used to achieve these results is presented in Appendix B.

Author

INTRODUCTION

1. Statement of the Problem.

The previous reports under the current grant, [1,2] * stated a future objective of the project as being the optimum contour design of a class of shells. This third report is directed toward achieving that objective in terms of enclosed volume and shell weight for thin shells of revolution.

Optimization can be treated in several ways. A general formulation of the optimization of the design of thin shells of revolution might include the determination of the shell shape as well as the variation of the shell thickness along meridional lines. A less general approach involves assigning the shape and varying the shell thickness [3, 4]. The current report treats an alternate approach. Here a uniform thickness is maintained, but the meridional lines which define the geometry are permitted to vary in accordance with the relation

$$\left|\frac{x}{a}\right|^{\alpha} + \left|\frac{y}{b}\right|^{\beta} = 1 \qquad (3.1)^{**}$$

where a, b, α and β are positive constants, not necessarily integers.

The use of Eq. (3.1) permits an optimization of shape which is limited to the choice of the parameters α and β for a shell of length 2a and of radius b. The body of this report is limited to the variation of α and β for fixed length and fixed diameter, but Appendix A presents a mathematical formulation which permits the length to vary as well as α and β.

The achievement of the stated objective depends on a suitable failure criterion. One criterion could involve a complete stress analysis of the shell including varying thickness. Others could include thick walled shells or buckling. However, in illustrating the method, the shells have been restricted to thin, constant thickness walls with internal pressure loading. Further the failure is assumed to occur either on the central plane circle normal to the x-axis at x = 0 or along a meridian. Thus separate computer programs which involve the complete stress analysis of the shell have not been used.

---

*Numbers in brackets refer to the References.

**The notation (3.1) is adopted to aid in cross-referencing equations from the first two reports under the grant [1, 2].

The techniques described can be applied in a manner which would permit the direct inclusion of one of the existing computer programs on the stress analysis of shells [5, 6, 7] . These auxilliary computer programs would provide the thickness requirement or the variation in thickness of the shell when incorporated into the proper location within the FORTRAN program presented in this report. In this way the optimized shell would be based on a more realistic failure criterion than is actually reported.

2. Symbols

a	half length of the shell, $[L]$*
b	radius of the shell in the equatorial plane, $[L]$
x	horizontal coordinate of the first quadrant of Eq. (3.1), $[L]$
y	vertical coordinate of the first quadrant of Eq. (3.1), $[L]$
g	acceleration due to gravity, $[LT^{-2}]$
V_{xa}	volume of the shell, $[L^3]$
W	weight of the shell, $[MLT^{-2}]$
A	surface area of the shell, $[L^2]$
A_a	area enclosed by first quadrant of Eq. (3.1), $[L^2]$
L	arc length in the first quadrant of Eq. (3.1), $[L]$
t	thickness of the shell, $[L]$
V_{min}	preassigned minimum allowable volume of the shell, $[L^3]$
W_{max}	preassigned maximum allowable weight of the shell, $[MLT^{-2}]$
a_{max}	preassigned maximum half length of the shell, $[L]$
V_{cyl}	volume of cylinder with radius b, length 2a, $[L^3]$
W_{cyl}	weight of cylindrical shell with radius b, length 2a, $[MLT^{-2}]$
v	ratio of V_{xa}/V_{min}, $[1]$
w	ratio of W/W_{max}, $[1]$
ℓ	ratio of a/a_{max}, $[1]$
h^2	$\left(\frac{b\alpha}{a\beta}\right)^2$, $[1]$

*The dimensional notation $[L]$ indicates a length while $[M]$ indicates mass, $[T]$ indicates time and $[1]$ indicates a dimensionless quantity.

p_o uniform internal pressure on shells, $[ML^{-1} T^{-2}]$

k_o preselected limiting value for the ratio $\Delta\alpha/F_\alpha$ or $\Delta\beta/F_\beta$ of iteration, [1]

α exponent of the absolute value of x/a, [1]

β exponent of the absolute value of y/b, [1]

α (as a subscript) indicates partial differentiation with respect to α, [1]

β (as a subscript) indicates partial differentiation with respect to β, [1]

ρ mass density, $[ML^{-3}]$

λ non-negative weighting function of v, [1]

μ non-negative weighting function of w, [1]

ν non-negative weighting function of ℓ, [1]

σ_o yield stress of the shell material, $[ML^{-1} T^{-2}]$

η_o^2 preselected limiting value for the maximum change in $(\Delta v^2 + \Delta w^2)$ to be allowed in one iteration step, [1]

J_1 through J_7 integrals as defined in Eqs. (3.33)

$I(\epsilon)$, $K(\epsilon)$ improper integrals as defined in Eqs. (3.35) and (3.36)

3. Acknowledgment

This project was sponsored by the National Aeronautics and Space Administration, Office of Advanced Research and Technology, Applied Mathematics Branch, of which Dr. Raymond H. Wilson is Chief.

The investigation was part of the work of the Engineering Experiment Station of which Professor Ross J. Martin is Director and was conducted in the Department of Theoretical and Applied Mechanics of which Professor Thomas J. Dolan is Head, with Will J. Worley as Principal Investigator.

The authors wish to acknowledge the assistance of Charles Cecil Fretwell, formerly Instructor in Theoretical and Applied Mechanics, University of Illinois, in the early stages of the numerical programming and the assistance of undergraduate students: Messrs. Tom E. Breuer and Edward H. Stredde with various phases of the project.

The suggestion that the length variation be included as a parameter in the optimization procedure was made by Melvin G. Rosche, Space Vehicle Structures Program, NASA, Washington, D. C.

Both the ILLIAC II and the IBM 7094 computer facilities were used. The ILLIAC II was constructed in the Digital Computer Laboratory, now known as the Department of Computer Science, University of Illinois with support from the Atomic Energy Commission, grant USAEC AT(11-1)-415, and from the Office of Naval Research, grant NONOR-1832 (15). The IBM 7094 computer facility is partially supported by the National Science Foundation under grant NSF GP700.

DISCUSSION OF THE METHOD

Otpimization with respect to enclosed volume and shell weight, for a shell of revolution defined by the meridian curve Eq. (3.1) is achieved by considering the exponents α and β as parameters.

Then the volume and weight may be expressed as

$$V_{xa} = V_{xa}(\alpha, \beta) \tag{3.2}$$

$$W = \rho\, g\, A(\alpha, \beta)\, t(\alpha, \beta) \tag{3.3}$$

where ρ is the mass density of the material of the shell, g is the gravitational acceleration, while A represents the area of the middle surface of the shell and t is the thickness. The thickness is maintained constant over the entire shell and is small compared to the radius b and to the length a.

Both the volume V_{xa} and the surface area A depend only on the geometrical shape of the shell, which is controlled by the parameters α and β for the fixed cylindrical volume. The thickness t depends on the geometrical shape of the shell, on the load condition and on the failure criteria. Therefore the mode of the failure of the shell, under a specified load condition, must be defined for the evaluation of the thickness t, before optimization can be achieved.

Let the primary design requirements, to be fulfilled for the shell, be

$$V_{xa} \geq V_{min} \quad , \qquad W \leq W_{max}$$

where V_{min} and W_{max} are preassigned limits. It is further assumed that at least one set of values (α, β) will satisfy the primary requirements. Otherwise the material of the shell, the assumed mode of failure, the load conditions, or the dimensions a and b have to be modified in order to determine an optimum shape.

To facilitate the calculations, the ratios of the volume and weight are introduced as

$$v = \frac{V_{xa}}{V_{min}} \ , \qquad w = \frac{W}{W_{max}} \tag{3.4}$$

The differential of a function is then defined as

$$dF = \lambda\, dv - \mu\, dw \tag{3.5}$$

where λ and μ are non-negative weighting functions of v and w, which define the relative importance of increasing in volume and of decreasing in weight. These weighting functions are defined in terms of current volume and weight. As long as it is possible to select dv and dw, consistent with the constraints of the problem, such that dF is positive, one has not achieved the optimum shape. Thus one seeks the values of α and β for which the differential dF is either zero or negative.

While superior shapes may exist, the above criteria will assure an optimum shape within the limitations of Eq. (3.1) and with the imposed constraints on volume and weight.

To determine the values of α and β for which F yields the extreme value, one may write Eq. (3.5) in the form

$$dF = F_\alpha \, d\alpha + F_\beta \, d\beta \tag{3.6}$$

with

$$F_\alpha = \lambda \, v_\alpha - \mu \, w_\alpha \tag{3.7}$$

$$F_\beta = \lambda \, v_\beta - \mu \, w_\beta \tag{3.8}$$

where the subscripts α and β indicate the partial differentiation with respect to α and β.

If Eq. (3.6) is an exact differential, then in principle one need only look among the solutions of $F_\alpha = F_\beta = 0$ for the optimum shape. Because of the complex nature of the equations for v_α, v_β, w_α and w_β, it is difficult to determine whether Eq. (3.6) is exact. Even if Eq. (3.6) were exact, the analytical solution of $F_\alpha = F_\beta = 0$ would be extremely difficult to obtain. The following iterative procedure is therefore used in the evaluation of $F_\alpha = F_\beta = 0$.

A shape defined by a set of α and β consistent with the primary requirements is selected first. This defines the shape of the shell middle surface. Therefore, the volume and the surface area of the shell can be calculated and the required thickness computed consistent with the assumed mode failure of the shell. Once the volume and weight are computed, values of λ and μ, which were defined by the design criterion, are established. Hence the values of F_α and F_β are determined by Eqs. (3.7) and (3.8). The shape is then modified by incrementing α and β in accordance with the path of the steepest ascent

$$d\alpha : d\beta = F_\alpha : F_\beta \tag{3.9}$$

The iterative procedure is repeated until F_α and F_β are both essentially zero.

To determine the incremental size $\Delta\alpha$ and $\Delta\beta$ for the steps in the iteration, let a constant k be defined from Eq. (3.9) as

$$\frac{\Delta\alpha}{F_\alpha} = \frac{\Delta\beta}{F_\beta} = k \tag{3.10}$$

Therefore

$$dv = v_\alpha\, d\alpha + v_\beta\, d\beta = k(v_\alpha F_\alpha + v_\beta F_\beta) \tag{3.11}$$

$$dw = w_\alpha\, d\alpha + w_\beta\, d\beta = k(w_\alpha F_\alpha + w_\beta F_\beta) \tag{3.12}$$

In order to limit the size of the increments of Δv and Δw, and of $\Delta\alpha$ and $\Delta\beta$, the constant k is selected in the following way

$$k = \begin{cases} k_1 \text{ if } k_1 < k_o \\ k_o \text{ if } k_1 > k_o \end{cases} \tag{3.13}$$

The constant k_1 is determined from Eqs. (3.11) and (3.12) consistent with the assigned increments of Δv and Δw, and is evaluated as follows

$$\eta_o^2 = \Delta v^2 + \Delta w^2 = k_1^2 \left[(v_\alpha F_\alpha + v_\beta F_\beta)^2 + (w_\alpha F_\alpha + w_\beta F_\beta)^2 \right]$$

from which

$$k_1 = \eta_o \Big/ \left[(v_\alpha F_\alpha + v_\beta F_\beta)^2 + (w_\alpha F_\alpha + w_\beta F_\beta)^2 \right]^{1/2} \tag{3.14}$$

where η_o^2 is a preselected limiting value for the maximum change of $(\Delta v^2 + \Delta w^2)$ to be allowed in one iteration step, while k_o is a preselected limiting value for the ratio $\Delta\alpha/F_\alpha$ or $\Delta\beta/F_\beta$ for each step of iteration. The process is then repeated with a new set of values of α and β formed by adding the increments $\Delta\alpha$ and $\Delta\beta$ to the previous values. The iteration process terminates when the value $(F_\alpha^2 + F_\beta^2)$ is less than a preassigned accuracy parameter.

The mathematical formulation of the more general problem which permits the length to vary as well as α and β is presented in Appendix A.

MATHEMATICAL FORMULATION

In the process of iteration, as described in the previous sections, the values of v, v_α, v_β, w, w_α and w_β for a given set of values of α and β must be calculated. From Eqs. (3.3) and (3.4), w_α and w_β may be written as

$$w_\alpha = \frac{\rho g}{W_{max}} (A \; t_\alpha + A_\alpha \; t) \qquad (3.15)$$

$$w_\beta = \frac{\rho g}{W_{max}} (A \; t_\beta + A_\beta \; t) \qquad (3.16)$$

The symbols used in the iteration procedure, described earlier in the report, are defined by the following integrals. The notation in these integrals is consistent with that used in the previous reports under the current research grant [1, 2] .

$$v = \frac{2 \pi b^2}{V_{min}} \int_0^a \left[1 - \left(\frac{x}{a}\right)^\alpha \right]^{2/\beta} dx$$

$$= \frac{2 \pi ab^2}{V_{min}} \int_0^1 (1 - X^\alpha)^{2/\beta} \; dX \qquad (3.17)$$

$$v_\alpha = \frac{-2 \pi ab^2}{V_{min}} \left(\frac{2}{\beta}\right) \int_0^1 (1 - X^\alpha)^{(2-\beta)/\beta} X^\alpha \log X \, dX \qquad (3.18)$$

$$v_\beta = \frac{-2 \pi ab^2}{V_{min}} \left(\frac{2}{\beta^2}\right) \int_0^1 (1 - X^\alpha)^{2/\beta} \log (1 - X^\alpha) \, dX \qquad (3.19)$$

$$A = 4 \pi b \int_0^a \left[1 - \left(\frac{x}{a}\right)^\alpha\right]^{1/\beta} \left\{1 + \left(\frac{b\alpha}{a\beta}\right)^2 \left(\frac{x}{a}\right)^{2(\alpha-1)} \left[1 - \left(\frac{x}{a}\right)^\alpha\right]^{2(1-\beta)/\beta}\right\}^{1/2} dx$$

$$= 4 \pi ab \int_0^1 (1 - X^\alpha)^{1/\beta} \left[1 + \left(\frac{b\alpha}{a\beta}\right)^2 X^{2(\alpha - 1)} (1 - X^\alpha)^{2(1-\beta)/\beta}\right]^{1/2} dX \qquad (3.20)$$

Let $F(X, \alpha, \beta) = 1 + \left(\frac{b\alpha}{a\beta}\right)^2 X^{2(\alpha - 1)} (1 - X^\alpha)^{2(1-\beta)/\beta}$

and $h^2 = \left(\frac{b\alpha}{a\beta}\right)^2$

then

$$A_\alpha = 4\pi ab\left\{\left(\frac{h^2}{\alpha}\right)\int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^\alpha)^{(3-2\beta)/\beta} dX\right.$$

$$- \frac{1}{\beta}\int_0^1 F(X, \alpha, \beta)^{-1/2} X^\alpha (1 - X^\alpha)^{(1-\beta)/\beta} \log X \, dX$$

$$+ h^2\int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^\alpha)^{(3-2\beta)/\beta} \log X \, dX$$

$$\left. - h^2 \left(\frac{2 - \beta}{\beta}\right)\int_0^1 F(X, \alpha, \beta)^{-1/2} X^{(3\alpha-2)} (1 - X^\alpha)^{3(1-\beta)/\beta} \log X \, dX\right\} \quad (3.21)$$

and

$$A_\beta = -4\pi ab\left\{\left(\frac{h^2}{\beta}\right)\int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^\alpha)^{(3-2\beta)/\beta} dX\right.$$

$$+ \frac{1}{\beta^2}\int_0^1 F(X, \alpha, \beta)^{-1/2} (1 - X^\alpha)^{1/\beta} \log (1 - X^\alpha) \, dX$$

$$\left. + 2\left(\frac{h^2}{\beta^2}\right)\int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^\alpha)^{(3-2\beta)/\beta} \log (1 - X^\alpha) \, dX\right\} \quad (3.22)$$

The next step consists of the determination of the thickness t and the values of t_α and t_β. These values should ideally be determined from a limit analysis, but since this would constitute a major undertaking in itself [6, 7], the following simple failure criterion is adopted. It is assumed that under a uniform internal pressure, p_o, the shell will fail by general yielding either along a longitudinal plane or around the equatorial plane. If σ_o is the yield stress for the shell material, failure along a longitudinal plane requires a thickness given by

$$t_1 = \left(\frac{p_o}{\sigma_o}\right) \frac{A_a}{L} \tag{3.23}$$

while failure around the equatorial plane requires a thickness given by

$$t_2 = \frac{1}{2}\left(\frac{p_o}{\sigma_o}\right) b \tag{3.24}$$

where A_a is the area enclosed by the first quadrant of Eq. (3.1) and L is the complete arc length in the first quadrant of Eq. (3.1). The design thickness t should be either t_1 or t_2, whichever is larger. If $t_2 \geq t_1$ then t equals t_2, a constant; therefore $t_\alpha = t_\beta = 0$. For $t_2 < t_1$, then by Eq. (3.23).

$$t_\alpha = \frac{p_o}{\sigma_o}\left[\frac{(A_a)_\alpha}{L} - \frac{A_a L_\alpha}{L^2}\right] \tag{3.25}$$

$$t_\beta = \frac{p_o}{\sigma_o}\left[\frac{(A_a)_\beta}{L} - \frac{A_a L_\beta}{L^2}\right] \tag{3.26}$$

where

$$A_a = b\int_0^a \left[1 - \left(\frac{x}{a}\right)^\alpha\right]^{1/\beta} dx = ab\int_0^1 (1 - X^\alpha)^{1/\beta}\, dX \tag{3.27}$$

$$(A_a)_\alpha = \frac{\partial A_a}{\partial \alpha} = -\left(\frac{ab}{\beta}\right)\int_0^1 (1 - X^\alpha)^{(1-\beta)/\beta} X^\alpha \log X\, dX \tag{3.28}$$

$$(A_a)_\beta = \frac{\partial A_a}{\partial \beta} = -\left(\frac{ab}{\beta^2}\right)\int_0^1 (1 - X^\alpha)^{1/\beta} \log(1 - X^\alpha)\, dX \tag{3.29}$$

$$L = \int_0^a \left\{ 1 + \left(\frac{b\alpha}{a\beta}\right)^2 \left(\frac{x}{a}\right)^{2(\alpha-1)} \left[1 - \left(\frac{x}{a}\right)^{\alpha}\right]^{2(1-\beta)/\beta} \right\}^{1/2} dX$$

$$= a \int_0^1 F(X, \alpha, \beta)^{1/2} \, dX \qquad (3.30)$$

$$L_\alpha = a \left\{ \frac{h^2}{\alpha} \int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^{\alpha})^{2(1-\beta)/\beta} dX \right.$$

$$+ h^2 \int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^{\alpha})^{(2-3\beta)/\beta} \log X \, dX$$

$$\left. - \frac{h^2}{\beta} \int_0^1 F(X, \alpha, \beta)^{-1/2} X^{3\alpha-2} (1 - X^{\alpha})^{(2-3\beta)\beta} \log X \, dX \right\} \qquad (3.31)$$

$$L_\beta = a \left\{ - \frac{h^2}{\beta} \int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^{\alpha})^{2(1-\beta)/\beta} dX \right.$$

$$\left. - \frac{h^2}{\beta^2} \int_0^1 F(X, \alpha, \beta)^{-1/2} X^{2(\alpha-1)} (1 - X^{\alpha})^{2(1-\beta)/\beta} \log (1 - X^{\alpha}) \, dX \right\} \qquad (3.32)$$

All of the integrals which appear in the above equations may be collected into seven groups, by defining the following convergent but sometimes improper integrals in notations as

$$J_1(p, q) = \int_0^1 (1 - u^p)^q \, du$$

$$J_2(p, q) = \int_0^1 (1 - u^p)^q \log (1 - u^p) \, du$$

$$J_3(p, q) = \int_0^1 (1 - u^p)^{q-1} u^p \log u \, du$$

$$J_4(p, q, s) = \int_0^1 (1 - u^p)^s \left[1 + h^2 u^{2(p-1)} (1 - u^p)^{2(q-1)}\right]^{1/2} du$$

$$J_5(p, q, s) = \int_0^1 \left[1 + h^2 u^{2(p-1)} (1 - u^p)^{2(q-1)}\right]^{-1/2} u^{2(p-1)} (1 - u^p)^{s-2} du \qquad (3.33)$$

$$J_6(p, q, r, s) = \int_0^1 \left[1 + h^2 u^{2(p-1)} (1-u^p)^{2(q-1)}\right]^{-1/2} u^{2(r-1)} (1 - u^p)^{s-2} \log (1-u^p) \, du$$

$$J_7(p, q, s, m) = \int_0^1 \left[1 + h^2 u^{2(p-1)} (1 - u^p)^{2(q-1)}\right]^{-1/2} u^m (1 - u^p)^{s-3} \log u \, du$$

with $h^2 = \left(\frac{b\alpha}{a\beta}\right)^2$ and $p, r \geq 1$, $q, s, m \geq 0$.

Then the equations used in the calculation of v, w and their derivatives can be expressed by Eqs. (3.33) as

$$v = \frac{2 \pi ab^2}{V_{min}} J_1(\alpha, \frac{2}{\beta})$$

$$v_\alpha = - \frac{2 \pi ab^2}{V_{min}} (\frac{2}{\beta}) J_3(\alpha, \frac{2}{\beta})$$

$$v_\beta = - \frac{2 \pi ab^2}{V_{min}} (\frac{2}{\beta^2}) J_2(\alpha, \frac{2}{\beta})$$

$$
\left.
\begin{aligned}
A &= 4\,\pi\, ab\ \ J_4(\alpha,\ \tfrac{1}{\beta},\ \tfrac{1}{\beta}) \\
A_\alpha &= 4\,\pi\, ab \left[\frac{h^2}{\alpha}\ J_5(\alpha,\ \tfrac{1}{\beta},\ \tfrac{3}{\beta}) - \frac{1}{\beta}\ J_7(\alpha,\ \tfrac{1}{\beta},\ \tfrac{1}{\beta}+2,\alpha) \right. \\
&\quad \left. + h^2\ J_7(\alpha,\ \tfrac{1}{\beta},\ \tfrac{3}{\beta}+1,\ 2\alpha - 2) - h^2\,(\tfrac{2}{\beta} - 1)\ J_7(\alpha,\ \tfrac{1}{\beta},\ \tfrac{3}{\beta},\ 3\alpha - 2) \right] \\
A_\beta &= -\,4\,\pi\, ab \left[\frac{h^2}{\beta}\ J_5(\alpha,\ \tfrac{1}{\beta},\ \tfrac{3}{\beta}) + \frac{1}{\beta^2}\ J_6(\alpha,\ \tfrac{1}{\beta},\ 1,\ \tfrac{1}{\beta}+2) \right. \\
&\quad \left. + 2\left(\frac{h}{\beta}\right)^2\ J_6(\alpha,\ \tfrac{1}{\beta},\ \alpha,\ \tfrac{3}{\beta}) \right] \\
A_a &= ab\ \ J_1(\alpha,\ \tfrac{1}{\beta}) \\
(A_a)_\alpha &= ab(-\tfrac{1}{\beta})\ \ J_3(\alpha,\ \tfrac{1}{\beta}) \\
(A_a)_\beta &= ab\left(-\frac{1}{\beta^2}\right)\ J_2(\alpha,\ \tfrac{1}{\beta}) \\
L &= a\ \ J_4(\alpha,\ \tfrac{1}{\beta},\ 0) \\
L_\alpha &= a\left[\frac{h^2}{\alpha}\ J_5(\alpha,\ \tfrac{1}{\beta},\ \tfrac{2}{\beta}) + h^2\ J_7(\alpha,\ \tfrac{1}{\beta},\ \tfrac{2}{\beta},\ 2\alpha - 2) \right. \\
&\quad \left. - \frac{h^2}{\beta}\ J_7(\alpha,\ \tfrac{1}{\beta},\ \tfrac{2}{\beta},\ 3\alpha - 2) \right] \\
L_\beta &= -a\left[\frac{h^2}{\beta}\ J_5(\alpha,\ \tfrac{1}{\beta},\ \tfrac{2}{\beta}) + \frac{h^2}{\beta^2}\ J_6(\alpha,\ \tfrac{1}{\beta},\ \alpha,\ \tfrac{2}{\beta}) \right]
\end{aligned}
\right\} \quad (3.34)
$$

NUMERICAL INTEGRATION

The analytical expressions for the integrals in Eqs.(3.33) are not available except for J_1, J_2 and J_3 which may be expressed in terms of gamma functions and the derivatives of gamma functions, the psi functions. Since the above functions also involve series expansions or numerical integration , all of the integrals in Eqs.(3.33) are evaluated numerically using Simpson's Rule. In this process, special consideration is given to improper integrals of the following two types.

$$I(\epsilon) = \int_0^{\epsilon} f(\xi)\, \xi^{\delta}\, d\xi \qquad \delta > -1 \qquad (3.35)$$

$$K(\epsilon) = \int_0^{\epsilon} f(\xi)\, \xi^{\delta} \log \xi \, d\xi \qquad \delta > -1 \qquad (3.36)$$

with $f(\xi)$ continuous in the interval $0 \leq \xi \leq \epsilon$. For ϵ small enough, replace $f(\xi)$ with a parabola

$$f(\xi) = b_o + b_1 \left(\frac{\xi}{\epsilon}\right) + b_2 \left(\frac{\xi}{\epsilon}\right)^2 \qquad (3.37)$$

where

$$b_o = f(0)$$
$$b_1 = 4\, f(\tfrac{\epsilon}{2}) - f(\epsilon) - 3\, f(0)$$
$$b_2 = 2\, f(\epsilon) - 4\, f(\tfrac{\epsilon}{2}) + 2\, f(0)$$

Then the improper integrals $I(\epsilon)$ and $K(\epsilon)$ can be approximated as

$$I(\epsilon) = \int_0^{\epsilon} \left[b_o + b_1 \left(\frac{\xi}{\epsilon}\right) + b_2 \left(\frac{\xi}{\epsilon}\right)^2 \right] \xi^{\delta}\, d\xi$$

$$= \frac{\epsilon^{\delta+1}}{\delta+1} \left[b_o + \frac{\delta+1}{\delta+2} b_1 + \frac{\delta+1}{\delta+3} b_2 \right] \qquad (3.38)$$

$$K(\epsilon) = \int_0^{\epsilon} \left[b_o + b_1 \left(\frac{\xi}{\epsilon}\right) + b_2 \left(\frac{\xi}{\epsilon}\right)^2 \right] \xi^{\delta} \log \xi \, d\xi$$

$$= \frac{\epsilon^{\delta+1}}{\delta+1} \left\{ \log \epsilon \left[b_o + \frac{\delta+1}{\delta+2} b_1 + \frac{\delta+1}{\delta+3} b_2 \right] - \frac{1}{\delta+1} \left[b_o + \left(\frac{\delta+1}{\delta+2}\right)^2 b_1 + \left(\frac{\delta+1}{\delta+3}\right)^2 b_2 \right] \right\} \qquad (3.39)$$

Therefore the improper integrals J_2 to J_7 listed in Eqs.(3.33) can be expressed in terms of $I(\epsilon)$, $K(\epsilon)$ and a proper integral. They are derived as follows.

$$J_2(p, q) = \int_0^{1-\eta} (1 - u^p)^q \log (1 - u^p) \, du + \int_{1-\eta}^{1} (1 - u^p)^q \log (1 - u^p) \, du$$

Let $\xi = 1 - u^p$

then $du = -\frac{1}{p} (1 - \xi)^{(1-p)/p} \, d\xi$

and $J_2(p, q) = \int_0^{1-\eta} (1-u^p)^q \log (1 - u^p) \, du + \frac{1}{p} \int_0^{1-(1-\eta)^p} (1 - \xi)^{(1-p)/p} \xi^q \log \xi \, d\xi$

Using Eq. (3.39) along with the definitions of Eq. (3.37), J_2 is approximated as

$$J_2(p, q) = \int_0^{1-\eta} (1 - u^p)^q \log (1 - u^p) \, du + \frac{1}{p} K(\epsilon) \Bigg|^{*}_{\substack{\epsilon = 1-(1-\eta)^p \\ f(\xi) = (1-\xi)^{(1-p)/p} \\ \delta = q}} \qquad (3.40)$$

In a similar manner, by the approximations of $I(\epsilon)$ and $K(\epsilon)$, other integrals yield;

$$J_3(p, q) = \int_{\epsilon}^{1-\eta} (1 - u^p)^{q-1} u^p \log u \, du + \frac{1}{p^2} I(\epsilon) \Bigg|_{\substack{\epsilon = 1-(1-\eta)^p \\ f(\xi) = (1-\xi)^{1/p} \log (1-\xi) \\ \delta = q-1}} + K(\epsilon) \Bigg|_{\substack{\epsilon = \epsilon \\ f(\xi) = (1-\xi^p)^{q-1} \\ \delta = p}}$$

*This notation indicates that the integral is evaluated at the indicated values of ϵ, $f(\xi)$ and δ.

$$J_4(p, q, s) = \int_0^{1-\eta} (1 - u^p)^s \left[1 + h^2 u^{2(p-1)} (1 - u^p)^{2(q-1)}\right]^{1/2} du$$

$$+ \frac{h}{p} I(\epsilon) \left| \begin{array}{l} \epsilon = 1-(1-\eta)^p \\ f(\xi) = \left[\frac{1}{h^2} (1 - \xi)^{2(1-p)/p} \xi^{2(1-q)} + 1\right]^{1/2} \\ \delta = q + s - 1 \end{array} \right.$$

$$J_5(p, q, s) = \int_0^{1-\eta} \left[1 + h^2 u^{2(p-1)} (1 - u^p)^{2(q-1)}\right]^{-1/2} u^{2(p-1)} (1 - u^p)^{s-2} du$$

$$+ \frac{1}{p} I(\epsilon) \left| \begin{array}{l} \epsilon = 1-(1-\eta)^p \\ f(\xi) = \left[\xi^{2(1-q)} + h^2 (1-\xi)^{2(p-1)/p}\right]^{-1/2} (1 - \xi)^{(p-1)/p} \\ \delta = s - q - 1 \end{array} \right. \qquad (3.41)$$

$$J_6(p, q, r, s) = \int_0^{1-\eta} \left[1 + h^2 u^{2(p-1)} (1-u^p)^{2(q-1)}\right]^{-1/2} u^{2(r-1)} (1-u^p)^{s-2}$$

$$\log(1-u^p)\, du$$

$$+ \frac{1}{p} K(\epsilon) \left| \begin{array}{l} \epsilon = 1-(1-\eta)^p \\ f(\xi) = \left[\xi^{2(1-q)} + h^2 (1-\xi)^{2(p-1)/p}\right]^{-1/2} (1-\xi)^{(2r-p-1)/p} \\ \delta = s - q - 1 \end{array} \right.$$

$$J_7(p, q, s, m) = \int_{\epsilon}^{1-\eta} \left[1 + h^2 u^{2(p-1)} (1-u^p)^{2(q-1)}\right]^{-1/2} (1-u^p)^{s-3} u^m \log u \, du$$

$$+ K(\epsilon) \Bigg|_{\substack{\epsilon = \epsilon \\ f(\xi) = \left[(1-\xi^p)^{2(1-q)} + h^2 \xi^{2(p-1)}\right]^{-1/2} (1-\xi^p)^{s-q-2} \\ \delta = m}}$$

$$+ \frac{1}{p^2} I(\epsilon) \Bigg|_{\substack{\epsilon = 1-(1-\eta)^p \\ f(\xi) = \left[\xi^{2(1-q)} + h^2 (1-\xi)^{2(p-1)/p}\right]^{-1/2} (1-\xi)^{(m+1-p)/p} \frac{1}{\xi} \log(1-\xi) \\ \delta = s-q-1}}$$

The integrals J_1 to J_7 therefore involve only proper integrals; thus numerical integration by Simpson's rule can be applied. The FORTRAN programs for the evaluation of J_1 through J_7 by means of a digital computer are written in subfunction form as listed in Appendix B.

NUMERICAL COMPUTATIONS

Several characteristics of the design problem have to be defined before the numerical iterations can be performed. One is the magnitude of the required values for V_{min} and W_{max}, the others include the weighting functions λ and μ.

1. The Ranges of the Volumes and Weights of Shells

Among the shells of revolution which may be generated by revolving the meridian curve, Eq. (3.1), about the x-axis, the range of shapes of interest lie between the cylindrical shell for which the exponents α and β are both large, and the conical shell for which the exponents α and β both equal unity. The volume of the cylinder with radius b and length 2a is $V_{cyl} = 2\pi ab^2$, and the volume of the double cone, having apexes at -a and at +a, with corresponding base diameter, 2b, is $V_{cone} = \frac{2}{3}\pi ab^2$. If the required minimum volume V_{min} is written as $V_{min} = C_1(2\pi ab^2)$, then C_1 must lie between 1/3 and 1.

In order to determine the weight for the two limiting cases, the thickness variation must be considered as well as the surface area. The surface areas for the cylindrical and conical shells are

$$A_{cyl} = 4\pi ab\left[1 + \frac{1}{2}\left(\frac{b}{a}\right)\right]$$

$$A_{cone} = 4\pi ab\left[\frac{1}{2}\sqrt{1 + \left(\frac{b}{a}\right)^2}\right]$$

Using Eqs. (3.23) and (3.24), the thicknesses required for the shell of cylindrical type, based on two different failure criteria, are expressed as

$$t_1 = \frac{1}{1 + \left(\frac{b}{a}\right)}\left(\frac{p_o}{\sigma_o}\right)b$$

$$t_2 = \frac{1}{2}\left(\frac{p_o}{\sigma_o}\right)b$$

therefore

$$t = \begin{cases} \frac{1}{2}\left(\frac{p_o}{\sigma_o}\right)b & \text{for } \left(\frac{b}{a}\right) \geq 1 \\ \frac{1}{1 + \left(\frac{b}{a}\right)}\left(\frac{p_o}{\sigma_o}\right)b & \text{for } \left(\frac{b}{a}\right) < 1 \end{cases}$$

Similarly, the thicknesses required for the conical type shell are

$$t_1 = \frac{1}{2} \frac{1}{\sqrt{1 + (\frac{b}{a})^2}} \left(\frac{p_o}{\sigma_o}\right) b$$

$$t_2 = \frac{1}{2} \left(\frac{p_o}{\sigma_o}\right) b$$

Since $\frac{1}{\sqrt{1 + (\frac{b}{a})^2}} \leq 1$ for any value of $(\frac{b}{a})$, it follows that

$$t = \frac{1}{2} \left(\frac{p_o}{\sigma_o}\right) b \qquad \text{for any ratio } (\frac{b}{a}) .$$

Then the weights of the cylindrical and the conical shells become

$$W_{cyl} = \begin{cases} \frac{1}{2} \left[1 + \frac{1}{2} (\frac{b}{a})\right] \left[4 \pi ab^2 \left(\frac{p_o}{\sigma_o}\right) \rho g\right] & \text{for } \frac{b}{a} \geq 1 \\ \frac{1}{1 + (\frac{b}{a})} \left[1 + \frac{1}{2} (\frac{b}{a})\right] \left[4 \pi ab^2 \left(\frac{p_o}{\sigma_o}\right) \rho g\right] & \text{for } \frac{b}{a} < 1 \end{cases} \qquad (3.42)$$

$$W_{cone} = \frac{1}{4} \sqrt{1 + (\frac{b}{a})^2} \left[4 \pi ab^2 \left(\frac{p_o}{\sigma_o}\right) \rho g\right]$$

By writing the primary required limiting weight as

$$W_{max} = C_2 \left[4 \pi ab^2 \left(\frac{p_o}{\sigma_o}\right) \rho g\right]$$

then C_2 has to be in the range

$$\frac{1}{4} \sqrt{1 + (\frac{b}{a})^2} \leq C_2 \leq \frac{1}{2} \left[1 + \frac{1}{2} (\frac{b}{a})\right] \quad \text{for } (\frac{b}{a}) \geq 1$$

$$\frac{1}{4} \sqrt{1 + (\frac{b}{a})^2} \leq C_2 \leq \frac{1}{1 + (\frac{b}{a})} \left[1 + \frac{1}{2} (\frac{b}{a})\right] \quad \text{for } (\frac{b}{a}) < 1$$

2. Weighting Functions λ and μ

The functions λ and μ which define the relative importance of the variation of the volume and the weight of the shells, are preassigned according to the design criterion. Any functions in terms of V and W can be assigned in the problem. One such set of functions is defined as

$$\lambda_1 = \frac{(V_{cyl} - V)^p}{(V - V_{min})^q}, \qquad \mu_1 = \frac{W^m}{(W_{max} - W)^n} \tag{3.43}$$

The shapes of the functions in Eq. (3.43) appear in Fig. 1, for $p = q = m = n = 1$. From the characteristics of the functions λ and μ, one can predict that when the volume is close to V_{min} or when the weight is close to W_{max}, a small increment of V or W will produce a large change of dF as defined in Eq. (3.6). If dF is considered as the slope of a surface F, then the surface has a positive slope along the edge where V is close to V_{min} and has a negative slope along the edge where W is close to W_{max}. Thus it follows that there must exist a maximum value of F, that is $dF = 0$, in the assigned range $V > V_{min}$ and $W < W_{max}$.

The functions λ and μ may also be defined as

$$\lambda_2 = \frac{\lambda_1}{\sqrt{\lambda_1^2 + \mu_1^2}} \qquad \mu_2 = \frac{\mu_1}{\sqrt{\lambda_1^2 + \mu_1^2}} \tag{3.44}$$

where λ_1 and μ_1 are obtained from Eq. (3.43). If one divides λ_1 and μ_1 by $\sqrt{\lambda_1^2 + \mu_1^2}$, the magnitude of λ_2 and μ_2 will be limited to the range of 0 to 1. The ratios λ_2/μ_2 and λ_1/μ_1, remain the same. The characteristics of the functions are as shown in Fig. 2; the values of λ_2 and μ_2 outside the range of V_{min} and W_{max} are arbitrary set equal to 0 and to 1 respectively.

Another form of λ and μ consists of straight lines, which define a linear variation of V and W as

$$\lambda_3 = \frac{V_{cyl} - V}{V_{cyl} - V_{min}} \qquad \mu_3 = \frac{W}{W_{max}} \tag{3.45}$$

The above three definitions of λ and μ are applied in the numerical examples of this report. Subroutine programs for the calculation of λ and μ are attached to the main iteration program as listed in Appendix B.

3. Numerical Examples

The functions of λ and μ for the first example are chosen as in Fig. 1, that is

$$\lambda = \frac{V_{cyl} - V}{V - V_{min}}, \qquad \mu = \frac{W}{W_{max} - W}$$

Let the required minimum volume of the shell be 0.6 of the volume of cylindrical shell, thus $V_{min} = 0.6\ (2\pi ab^2)$, and let the required maximum weight be 0.8 that of the cylindrical shell under the same load condition. Since the thickness required for the cylindrical shell is dependent on the ratio of b/a, the weights for the cylindrical shells with different ratios b/a are given by Eq. (3.42) as

$$W_{cyl} = \begin{Bmatrix} 0.75 \\ 1.00 \\ 0.90 \end{Bmatrix} (4\pi\rho g p_o a b^2/\sigma_o) \quad \begin{matrix} \text{for } b/a = 1.0 \\ \text{for } b/a = 2.0 \\ \text{for } b/a = 0.25 \end{matrix}$$

therefore the values of W_{max} are chosen as $0.6(4\pi\rho g p_o ab^2/\sigma_o)$, $0.8(4\pi\rho g p_o ab^2/\sigma_o)$ and $0.72\ (4\pi\rho g p_o ab^2/\sigma_o)$ for the ratios b/a = 1.0, 2.0, and 0.25 respectively.

With all the requirements set, the iterative calculations are performed with the aid of digital computers. The FORTRAN programs for the iteration procedure are listed in Appendix B.

Choosing the starting values $\alpha = 1.5$, $\beta = 1.5$ with the fineness ratio a/b = 1.0, and the limiting value $\eta_o = 0.1$, $k_o = 0.3$, results in the output listed in Table I. From steps 1 to 7 in the Table, the results listed are presented for each iteration; from step 8 on, the results are presented for every other iteration. From these results it is seen that the values of α and β increase rapidly in each of the first six iterations and then change slowly. The same pattern is apparent for the slopes F_α and F_β.

Table II shows the iteration results with the same parameters as in Table I, but with the starting condition $\alpha = 2.0$, $\beta = 3.0$. The results in steps 1 to 8 are listed for each iteration while after step 8, they are listed for every fourth iteration. The iterated values of α and β decrease rapidly in the first three steps and then change slowly.

The results in Tables I and II, indicate that the shape defined by $\alpha = 1.5$, $\beta = 1.5$ lies on one side of the ridge, Fig. 3, while the shape defined by $\alpha = 2.0$, $\beta = 3.0$ lies on the other side of the ridge. During the iteration process, the successively improving values of α and β climb to the ridge rapidly according to the path of steepest ascent, and then progress slowly along the ridge due to the small variation of slope along the ridge.

The phenomena observed in the above results may be verified or described more clearly by the exact integration of the function dF of Eq. (3.6) using the assigned functions λ and μ. Rewriting the weighting functions λ and μ in dimensionless terms v and w, one obtains

$$\lambda = \frac{c - v}{v - 1} \qquad \mu = \frac{w}{1 - w}$$

where $$c = \frac{V_{cyl}}{V_{min}}$$

Then $$dF = \frac{c - v}{v - 1}\, dv - \frac{w}{1 - w}\, dw$$

which, after integration, yields

$$F = (c - 1) \log (v - 1) + \log (1 - w) - (v + w)$$

The function F is in terms of v and w, which can be represented by the integrals with parameters α and β. The relative variation of F with respect to α and β is plotted as a three dimensional surface in Fig. 3. The surface has the shape of a mountain range with the projection of the ridge shown in the α - β plane in Fig. 3. The peak of the ridge is located near the point $\alpha = 2.65$, $\beta = 1.55$.

Changing the values of V_{min}, W_{max} and the reciprocal of fineness ratio, b/a, results in little change in the shape of the surface F, but does produce a slight shift in the location of the ridge. The projections of the ridges on the α - β plane, with different combinations of V_{min}, W_{max}, and b/a, are plotted in Fig. 4. The shift in the ridge is in the same sense as the change in V_{min} or in W_{max}.

The results in Tables III and IV show the iterative calculations for the case $V_{min} = 0.6\ (2\ \pi\ ab^2)$, $W_{max} = 0.8\ (4\ \pi\ \rho\ g\ p_o\ ab^2/\sigma_o)$ with the ratio b/a = 2.0 and $k_o = 0.3$. While the initial values for α and β are different in Tables III and IV, it is noted that they converge to the same values of α and β after successive iterations.

As a second example, λ_2 and μ_2 of Fig. 2 are chosen as the weighting functions. The results of each iterated calculation with three different starting values are listed in Table V. The preassigned values for computations are $V_{min} = 0.6\ (2\ \pi\ ab^2)$, $W_{max} = 0.6\ (4\ \pi\ \rho\ g\ p_o\ ab^2/\sigma_o)$ and b/a = 1.0. The values of α and β reach the ridge rapidly after several iterations regardless of the starting point.

Table VI gives the iterated results for the weighting functions λ_3 and μ_3, which vary linearly with V and W, as defined by Eq. (3.45). The other preassigned values for computations are the same as for Table V. The results indicate that both the slopes F_α and F_β are within the limit 0.005 after ten iterations for all of the different starting values.

Since the iteration procedure is controlled by the slope of the function F, the rate of convergence is mainly dependent on the weighting functions λ and μ. For the currently assigned functions, the results in the above tables indicate that the values α and β converge rapidly to the region where the ordered pair (α, β) lies near the projection of the ridge and then change slowly along the ridge. Due to the small variation of the slope along the ridge, any point located on the projection of the ridge on the α - β plane constitutes a good shape with respect to volume and weight.

As another example, the functions λ and μ may be considered as constants. In this case, the problem becomes one of determining the relative maximum of the function $F = v - w$. Since the variation of the thickness is very small due to the change of values α, β, a shape which is nearly optimum may be achieved by assigning a specific value of volume in determining the values of α, β for minimum shell surface or by assigning a specific value of surface area in determining the values of α, β for maximum volume.

The shapes to fulfill the above requirement can be determined with the aid of data from previous reports [1, 2]. The surface in Fig. 5 represents the volume variation with respect to α and β. The heavy curve on this surface represents the volumes of shells for which the surface area is equal to a preassigned value. From the projection of this curve on a vertical plane, the values of α and β for the maximum volume for the defined surface area can be established. In a similar manner, the surface in Fig. 6 represents the area variation with respect to α and β. The heavy curve on the surface represents the areas of shells for which the shell volume is equal to a preassigned value. The projection of this curve on a vertical plane indicates the area variation among shells having a constant volume.

APPENDIXES

A. Iteration Procedures with Varying Shell Length

Similar to the Eqs. (3.2) and (3.3), the volume and weight of the shells of revolution may be taken as the functions of three parameters α, β and a,

$$V_{xa} = V_{xa}(\alpha, \beta, a)$$

$$W = \rho g A(\alpha, \beta, a)\ t(\alpha, \beta, a)$$

Here the shape requirements to be fulfilled for the shell are

$$V_{xa} \geq V_{min} \qquad W \leq W_{max} \qquad a \leq a_{max}$$

In this case the dimensionless forms are defined as

$$v = \frac{V_{xa}}{V_{min}} \qquad w = \frac{W}{W_{max}} \qquad \ell = \frac{a}{a_{max}}$$

and the differential of a function is formed as

$$dF = \lambda\, dv - \mu\, dw - \nu\, d\ell$$

where λ, μ and ν are the functions defining the relative importance of increases in volume and decreases in weight and length. The differential dF also can be written as

$$dF = F_\alpha\, d\alpha + F_\beta\, d\beta + F_\ell\, d\ell$$

with

$$F_\alpha = \lambda v_\alpha - \mu w_\alpha$$

$$F_\beta = \lambda v_\beta - \mu w_\beta$$

$$F_\ell = \lambda v_\ell - \mu w_\ell - \nu$$

The iteration steps will then follow path of steepest ascent, as defined by

$$d\alpha : d\beta : d\ell = F_\alpha : F_\beta : F_\ell$$

The iterative procedure is repeated until F_α, F_β and F_ℓ are essentially zero.

B. FORTRAN Programs

1. Programs for FUNCTIONS FJ1 through FJ7

Since the integrals J_1 through J_7 listed in Eqs. (3.33) appear in the calculations of the main iteration program many times, they are computed in separate FUNCTIONS attached to the main program. Simpson's rule is used to evaluate the above integrals with the approximation techniques discussed in the section on Numerical Integration.

Among the input arguments for the FUNCTION programs, the values P, Q, R, S and T are the exponents in the integrals. They are dependent on the values of α and β. The quantities ETAI and EPS are two small numbers assigned in the calculation of the two improper integrals $I(\epsilon)$ and $K(\epsilon)$ of Eqs. (3.35) and (3.36). The value FK2 represents the term $\left(\frac{b\alpha}{a\beta}\right)^2$ and the value ACC is the accuracy required for the relative difference between two successive approximations in the Simpson's rule integration routine. In the previous numerical examples, the value assigned to ETAI and to EPS is 0.01 while the value assigned to ACC is 0.0001.

2. Program for SUBROUTINE FMULAM

The SUBROUTINE FMULAM is written to compute either the values λ_1, μ_1, as Eq. (3.43) or the values λ_2, μ_2 as Eq. (3.44), which is controlled by the number NC. The outputs defined by FLAM and FMU represent λ_1 and μ_1 for NC = 1 and λ_2 and μ_2 for NC = 2. The input arguments P, Q, FM and FN are the same as the exponents p, q, m and n of Eq. (3.43).

3. The Main Iteration Program

The main purpose of the program is to compute the increments of $\Delta\alpha$ and $\Delta\beta$ along the path of the steepest ascent from the current assigned values α and β. The computations are repeated for the new calculated α and β until they reach a point where the absolute values F_α and F_β, as in Eqs. (3.7) and (3.8), are less than a preassigned small number QEPS.

The input data of FP, FQ, FM, FN and NC listed on the first data card are supplied for the calculation of functions λ and μ. The constants EPS, ETAI and ACC on the second data card are the numbers assigned to the FUNCTIONS J_1 through J_7 in order to compute the integrals. The values PO and SIGO represent internal pressure p_0 and the yield stress σ_0 and are used to calculate the thickness t. Input data ETAO and FKV are assigned to limit the step size of α and

β in each iteration, and represent η_0 and k_0 in Eqs. (3.14) and (3.13). The two integers NRVWP and NRAB are the number of the sets of V_{min}, W_{max} and the number of sets of the ratio BOA (b/a) to be calculated in the program.

The input values VMIN and WMAX are two dimensionless numbers which represent the preassigned allowable minimum volume and maximum weight. The true value of the minimum volume is VMIN $\cdot$ $(2 \pi ab^2)$ and the true value of maximum weight is WMAX $\cdot$ $(4 \pi \rho g p_0 ab^2/\sigma_0)$.

For the output, the results of each iteration are printed using the symbols DA, DB, ATIL and BTIL to represent $\Delta\alpha$, $\Delta\beta$, F_α and F_β respectively.

PROGRAM FOR EXAMPLES

```
$      FORTRAN IBM
$      PUNCH OBJECT
$      GO
      READ INPUT TAPE 7, 1, FP, FQ, FM, FN, NC, EPS, ETAI, ACC, NRVWP   SHELL001
    1 FORMAT (4F15.5, I10 / 3E20.8 / I10)                               SHELL002
      READ INPUT TAPE 7, 2, PO, SIGO, QEPS, ETAO, FKV                   SHELL003
    2 FORMAT (2E20.8/ 3F20.5)                                           SHELL004
    3 DO 69 INRVWP = 1, NRVWP                                           SHELL005
    4 READ INPUT TAPE 7, 5, VMIN, WMAX, NRAB                            SHELL006
    5 FORMAT ( 2E20.8 /I10)                                             SHELL007
    9 DO 50 INRAB = 1, NRAB                                             SHELL008
   10 READ INPUT TAPE 7, 11, ALPHA, BETA, BOA                           SHELL009
   11 FORMAT ( 3E20.8)                                                  SHELL010
      WRITE OUTPUT TAPE 6, 12, VMIN, WMAX, BOA                          SHELL011
   12 FORMAT (8H1VMIN = ,F7.3,2X,7HWMAX = ,F7.3,3X, 4HBOA=, F6.3, //)   SHELL012
       NCONT =0                                                         SHELL013
C                                                                       SHELL014
C   TO LIMIT ALPHA AND BETA BOTH LARGER THAN ONE                        SHELL015
C                                                                       SHELL016
   14 IF (ALPHA - 1.0) 15, 17, 17                                       SHELL017
   15 ALPHA = 1.0                                                       SHELL018
   17 IF (BETA - 1.0) 18, 19, 19                                        SHELL019
   18 BETA = 1.0                                                        SHELL020
   19 FK2 = (BOA*ALPHA/BETA)**2                                         SHELL021
       ONOB = 1.0/BETA                                                  SHELL022
       TWOB = 2.0/BETA                                                  SHELL023
       THOB = 3.0/BETA                                                  SHELL024
       NCONT = NCONT +1                                                 SHELL025
C                                                                       SHELL026
C   TO DETERMINE THE VALUE OF T= T1 OR T2                               SHELL027
C                                                                       SHELL028
      AR= FJ1(ALPHA,ONOB,ACC)                                           SHELL029
      SL = FJ4(ALPHA,ONOB,0.0,FK2,ETAI,ACC)                             SHELL030
      RR = AR/ SL                                                       SHELL031
      IF ( RR-0.5 ) 20, 20, 21                                          SHELL032
C                                                                       SHELL033
C   T2 LARGER THAN T1                                                   SHELL034
   20 T = 0.5 *PO/SIGO                                                  SHELL035
      TA = 0.                                                           SHELL036
      TB = 0.                                                           SHELL037
      GO TO 24                                                          SHELL038
C                                                                       SHELL039
C   T1 LARGER THAN T2                                                   SHELL040
   21 T = RR *PO /SIGO                                                  SHELL041
      ARA= -(ONOB)*FJ3(ALPHA,ONOB,EPS,ETAI,ACC)                         SHELL042
      ARB= -(1.0/(BETA**2))*FJ2(ALPHA,ONOB,ETAI,ACC)                    SHELL043
      TE = FJ5(ALPHA,ONOB,TWOB,FK2,ETAI,ACC)                            SHELL044
      SLA = FK2*(FJ7(ALPHA,ONOB,TWOB,2.*(ALPHA-1.),FK2,EPS,ETAI,ACC) -  SHELL045
     1   FJ7(ALPHA,ONOB,TWOB,3.*ALPHA-2.,FK2,EPS,ETAI,ACC)/BETA +       SHELL046
     2   TE/ALPHA)                                                      SHELL047
     SLB = -(FK2/(BETA**2))*(FJ6(ALPHA,ONOB,ALPHA,TWOB,FK2,ETAI,ACC) +  SHELL048
     1   BETA*TE)                                                       SHELL049
      TA = (PO/SIGO) * ( ARA- AR*SLA/SL ) / SL                          SHELL050
      TB = (PO/SIGO) * (ARB- AR * SLB/ SL) /SL                          SHELL051
   24 WRITE OUTPUT TAPE 6, 25, ALPHA, BETA, AR, SL, RR, T, TA, TB       SHELL052
   25 FORMAT ( 1H0, 5X, 8F12.6)                                         SHELL053
```

```
C
C   TO CALCULATE V, W, AND THEIR DERIVATIVES
C
      SU= FJ4(ALPHA,ONOB,ONOB,FK2,ETAI,ACC)
      TE = FK2*FJ5(ALPHA,ONOB,THOB,FK2,ETAI,ACC)
      SUA= TE/ALPHA - FJ7(ALPHA,ONOB, ONOB+2.   ,ALPHA,FK2,EPS,ETAI,ACC)
     1  /BETA + FK2*FJ7(ALPHA,ONOB, THOB+1.0, 2.*(ALPHA-1.),FK2,EPS,
     2    ETAI,ACC) - FK2*(TWOB-1.)*FJ7(ALPHA,ONOB,THOB,3.*ALPHA-2.,
     3    FK2,EPS,ETAI,ACC)
      SUB= -(1./(BETA**2))*(FJ6(ALPHA,ONOB,1.,ONOB+2.,FK2,ETAI,ACC) +
     1    2.*FK2*FJ6(ALPHA,ONOB,ALPHA,THOB,FK2,ETAI,ACC) + BETA*TE)
      V = FJ1(ALPHA,TWOB,ACC)
      VA = -(TWOB)*FJ3(ALPHA,TWOB,EPS,ETAI,ACC)
      VB = -(2.0/(BETA**2))*FJ2(ALPHA,TWOB,ETAI,ACC)
      W = SU *T
      WA = SU*TA + T*SUA
      WB = SU*TB + T*SUB
C
C   TO FIND THE VALUES OF MU AND LAMDA
C
      CALL FMULAM ( FP, FQ, FM, FN, VMIN, WMAX, V, W, FMU, FLAM, NC )
C
      WRITE OUTPUT TAPE 6, 26, V, VA, VB, W, WA, WB, FMU, FLAM
   26 FORMAT (10X,  8F11.6 )
      IF (FMU) 50, 30, 30
   30 IF (FLAM) 50, 31, 31
C
C   TO CALCULATE FALPHA AND FBETA (THAT IS, ATIL AND BTIL)
C
   31 ATIL = FLAM * VA/ VMIN - FMU * WA/ WMAX
      BTIL = FLAM * VB/ VMIN - FMU * WB/ WMAX
      ATILS = ( ATIL*VA + BTIL*VB ) /VMIN
      BTILS = ( ATIL*WA + BTIL*WB ) / WMAX
      VO = V/VMIN
      WO = W/WMAX
C
C   PROGRAM TERMINATES WHEN QQ LESS THAN QEPS
C
      QQ = SQRT (ATIL**2 + BTIL**2)
      IF ( QQ-QEPS ) 48, 48, 35
C
C   TO DETERMINE STEP SIZE, CONTROL ON FKAP
C
   35 FKAPS = ETAO / SQRT (ATILS**2 + BTILS**2)
      IF (FKAPS - FKV) 36, 36, 37
   36 FKAP = FKAPS
      GO TO 40
   37 FKAP = FKV
   40 DA = FKAP * ATIL
      DB = FKAP * BTIL
      WRITE OUTPUT TAPE 6, 42, ALPHA, BETA, DA, DB, VO, WO, FKAP, ATIL,
     1    BTIL, ATILS, BTILS
   42 FORMAT (6F9.5, 5E12.5)
      ALPHA = ALPHA+ DA
      BETA = BETA + DB
      IF ( NCONT - 50  ) 14, 50, 50
   48 WRITE OUTPUT TAPE 6, 49, ALPHA, BETA, VO, WO, ATIL, BTIL, ATILS,
     1    BTILS
   49 FORMAT ( 1H-, 8F13.5 /1H2 )
   50 CONTINUE
   69 CONTINUE
      CALL SYSTEM
      END
```

```
$        FORTRAN IBM
$        PUNCH OBJECT
       SUBROUTINE FMULAM(P,Q,FM,FN,VMIN,WMAX,V,W,FMU,FLAM,NC)              SUBFML01
     1 VV = V/VMIN                                                         SUBFML02
     2 WW = W/WMAX                                                         SUBFML03
     3 IF (VV - 1.0) 4, 4, 10                                              SUBFML04
     4 IF (WW - 1.0) 10, 5, 5                                              SUBFML05
     5 WRITE OUTPUT TAPE 6, 6, VMIN, WMAX, VV, WW                          SUBFML06
     6 FORMAT(7H VMIN = ,E12.6,8H WMAX = ,E12.6,6H  V = ,E14.8,6H  W = ,   SUBFML07
      1E14.8,// 5X, 43HTHE PRIMARY REQUIREMENTS CANNOT BE REACHED )        SUBFML08
     7 FMU = -1.0                                                          SUBFML09
     8 FLAM = -1.0                                                         SUBFML10
     9 RETURN                                                              SUBFML11
    10 IF (V-1.0) 11, 5, 5                                                 SUBFML12
    11 IF (VV - 1.0) 12, 12, 15                                            SUBFML13
    12 FMU = 0.0                                                           SUBFML14
    13 FLAM = 1.0                                                          SUBFML15
    14 IF (WW) 5, 5, 9                                                     SUBFML16
    15 IF (WW) 5, 5, 16                                                    SUBFML17
    16 IF (WW - 1.0) 20, 17, 17                                            SUBFML18
    17 FLAM = 0.0                                                          SUBFML19
    18 FMU = 1.0                                                           SUBFML20
    19 RETURN                                                              SUBFML21
    20 FLAM = (( 1./VMIN - VV)**P)/ ((VV-1. )**Q)                          SUBFML22
    21 FMU = (WW**FM)/((1.0 - WW)**FN)                                     SUBFML23
    22 GO TO (23,26), NC                                                   SUBFML24
    23 TEMP = SQRT(FLAM**2 + FMU**2)                                       SUBFML25
    24 FLAM = FLAM/TEMP                                                    SUBFML26
    25 FMU  = FMU/TEMP                                                     SUBFML27
    26 RETURN                                                              SUBFML28
       END                                                                 SUBFML29
```

```
$       FORTRAN IBM
$PUNCH OBJECT
       FUNCTION FJ1(P,Q,ACC)                                               FJ1 01
     1 ODD = 0.0                                                           FJ1 02
     2 INT = 1                                                             FJ1 03
     3 V = 1.0                                                             FJ1 04
     4 EVEN = 0.0                                                          FJ1 05
       AREA1 = 0.0                                                         FJ1 06
       IF (Q) 19, 5, 6                                                     FJ1 07
     5 ENDS = 2.0                                                          FJ1 08
       GO TO 7                                                             FJ1 09
     6 ENDS = 1.0                                                          FJ1 10
     7 H = 1.0/V                                                           FJ1 11
     8 ODD = EVEN + ODD                                                    FJ1 12
     9 X = H/2.                                                            FJ1 13
    10 EVEN = 0.0                                                          FJ1 14
    11 DO 13 I = 1, INT                                                    FJ1 15
    12 EVEN = EVEN + ((1.0 - X**P)**Q)                                     FJ1 16
    13 X = X + H                                                           FJ1 17
    14 AREA = (ENDS + 4.0*EVEN + 2.0*ODD)*H/6.0                            FJ1 18
    15 R = ABSF(AREA1/AREA - 1.0) - ACC                                    FJ1 19
    16 IF (R) 25, 25, 17                                                   FJ1 20
    17 IF (INT - 16384) 21, 19, 19                                         FJ1 21
    18 FORMAT (23H J1(P,Q) NOT CONVERGENT)                                 FJ1 22
    19 WRITE OUTPUT TAPE 6, 18                                             FJ1 23
    20 CALL SYSERR                                                         FJ1 24
    21 AREA1 = AREA                                                        FJ1 25
    22 INT = 2*INT                                                         FJ1 26
    23 V = 2.0*V                                                           FJ1 27
    24 GO TO 7                                                             FJ1 28
    25 FJ1 = AREA                                                          FJ1 29
    26 RETURN                                                              FJ1 30
       END                                                                 FJ1 31
```

```
$      FORTRAN IBM
$PUNCH OBJECT
      FUNCTION FJ2(P,Q,ETAI, ACC)
    1 DP = P
      E=  1.-(1.0-ETAI)**P
      OME = 1.0 -ETAI
    3 DEL = 1.0 - 1.0/DP
    5 F0 = 1.0
    6 F1 = (1.0 - 0.5*E)**(-DEL)
    7 F2 = (1.0 - E)**(-DEL)
    8 DQ = Q
    9 A = 4.0*F1 - F2 - 3.0*F0
   10 B = 2.0*F2 - 4.0*F1 + 2.0*F0
   11 T1 = DQ + 1.0
   12 T2 = DQ + 2.0
   13 T3 = DQ + 3.0
   140EN = (ELOG (E)*(F0 + T1*(A/T2 + B/T3)) - (F0 + T1*T1*(A/(T2*T2)
     1  + B/(T3*T3)))/T1)*(E**T1)/(T1*DP)
   15 ODD = 0.0
   16 INT = 1
   17 V = 1.0
   18 EVEN = 0.0
      AREA1 = 0.0
   19 ENDS = ((1.0 - OME**DP)**DQ)*ELOG (1.0 - OME**DP)
   20 H = OME/V
   21 ODD = EVEN + ODD
   22 X = H/2.0
   23 EVEN = 0.0
   24 DO 26 I = 1, INT
   25 EVEN = EVEN + ((1.0 - X**DP)**DQ)*ELOG (1.0 - X**DP)
   26 X = X + H
   27 AREA = (ENDS + 4.0*EVEN + 2.0*ODD)*H/6.0
   28 R = ABSF(AREA1/AREA - 1.0) - ACC
   29 IF (R) 38, 38, 30
   30 IF (INT - 16384) 34, 31, 31
   31 WRITE OUTPUT TAPE 6, 32
   32 FORMAT (23H J2(P,Q) NOT CONVERGENT)
   33 CALL SYSERR
   34 AREA1 = AREA
   35 INT = 2*INT
   36 V = 2.0*V
   37 GO TO 20
   38 FJ2 = AREA + EN
   39 RETURN
      END
```

```
$     FORTRAN IBM
$PUNCH OBJECT
      FUNCTION FJ3(P,Q,EPS, ETAI, ACC)                                  FJ3 01
    1 DP = P                                                            FJ3 02
    2 DQ = Q - 1.0                                                      FJ3 03
    3 T1 = DP + 1.0                                                     FJ3 04
    4 T2 = T1/(DP + 2.0)                                                FJ3 05
    5 T3 = T1/(DP + 3.0)                                                FJ3 06
    6 F0 = 1.0                                                          FJ3 07
    7 E = EPS                                                           FJ3 08
    8 F1 = (1.0 - (0.5*E)**DP)**DQ                                      FJ3 09
    9 F2 = (1.0 - E**DP)**DQ                                            FJ3 10
   10 A = 4.0*F1 - F2 - 3.0*F0                                          FJ3 11
   11 B = 2.0*F2 - 4.0*F1 + 2.0*F0                                      FJ3 12
   120EN = (ELOG (E)*(F0 + T2*A + T3*B) - (F0 + T2*T2*A + T3*T3*B)/T1)*(FJ3 13
     1E**T1)/T1                                                         FJ3 14
   13 DEL = 1.0/DP                                                      FJ3 15
      OME = 1.0-ETAI                                                    FJ3 16
      E= 1.0- OME**DP                                                   FJ3 17
   15 T1 = DQ + 1.0                                                     FJ3 18
   16 T2 = T1/(DQ + 2.0)                                                FJ3 19
   17 T3 = T1/(DQ + 3.0)                                                FJ3 20
   18 F0 = 0.0                                                          FJ3 21
   19 F1 = ((1.0 - 0.5*E)**DEL)*ELOG (1.0 - 0.5*E)                      FJ3 22
   20 F2 = ((1.0 - E)**DEL)*ELOG (1.0 - E)                              FJ3 23
   21 A = 4.0*F1 - F2 - 3.0*F0                                          FJ3 24
   22 B = 2.0*F2 - 4.0*F1 + 2.0*F0                                      FJ3 25
   23 EN = EN + (F0 + T2*A + T3*B)*(E**T1)/(T1*DP*DP)                   FJ3 26
      E = EPS                                                           FJ3 27
   240ENDS = ((1.0 - E**DP)**DQ)*(E**DP)*ELOG (E)                       FJ3 28
     1 + ((1.0 - OME**DP)**DQ)*(OME**DP)*ELOG (OME)                     FJ3 29
   25 ODD = 0.0                                                         FJ3 30
   26 INT = 1                                                           FJ3 31
   27 V = 1.0                                                           FJ3 32
   28 EVEN = 0.0                                                        FJ3 33
   29 AREA1 = 0.0                                                       FJ3 34
   30 H = (OME - E)/V                                                   FJ3 35
   31 ODD = EVEN + ODD                                                  FJ3 36
   32 X = E + H/2.                                                      FJ3 37
   33 EVEN = 0.0                                                        FJ3 38
   34 DO 36 I = 1, INT                                                  FJ3 39
   35 EVEN = EVEN + ((1.0 - X**DP)**DQ)*(X**DP)*ELOG (X)                FJ3 40
   36 X = X + H                                                         FJ3 41
   37 AREA = (ENDS + 4.0*EVEN + 2.0*ODD)*H/6.0                          FJ3 42
   38 R = ABSF(AREA1/AREA - 1.0) - ACC                                  FJ3 43
   39 IF (R) 48, 48, 40                                                 FJ3 44
   40 IF (INT - 16384) 44, 42, 42                                       FJ3 45
   41 FORMAT (23H J3(P,Q) NOT CONVERGENT)                               FJ3 46
   42 WRITE OUTPUT TAPE 6, 41                                           FJ3 47
   43 CALL SYSERR                                                       FJ3 48
   44 AREA1 = AREA                                                      FJ3 49
   45 INT = 2*INT                                                       FJ3 50
   46 V = 2.0*V                                                         FJ3 51
   47 GO TO 30                                                          FJ3 52
   48 FJ3 = AREA + EN                                                   FJ3 53
   49 RETURN                                                            FJ3 54
      END                                                               FJ3 55
```

```
$      FORTRAN IBM
$PUNCH OBJECT
       FUNCTION FJ4(P,Q,S,FK2, ETAI, ACC)                                 FJ4 01
     1 DG = S + Q - 1.0                                                   FJ4 02
     2 T1 = DG + 1.0                                                      FJ4 03
     3 T2 = T1/(DG + 2.0)                                                 FJ4 04
     4 T3 = T1/(DG + 3.0)                                                 FJ4 05
       OME = 1.0 - ETAI                                                   FJ4 06
       E = 1.0- OME**P                                                    FJ4 07
     6 DP = P                                                             FJ4 08
     8 DEL = -2.0*(1.0 - 1.0/DP)                                          FJ4 09
       DQ = 2.0*(1.0 -  Q)                                                FJ4 10
       IF (Q - 1.0) 11, 10, 36                                            FJ4 11
    10 FO = SQRT (1.0/FK2 + 1.0)                                          FJ4 12
       GO TO 12                                                           FJ4 13
    11 FO = 1.0                                                           FJ4 14
    12 DF = FK2                                                           FJ4 15
    13 F1 = SQRT (((1.0 - 0.5*E)**DEL)*((0.5*E)**DQ)/FK2 + 1.0)          FJ4 16
    14 F2 = SQRT (((1.0 - E)**DEL)*(E**DQ)/FK2 + 1.0)                     FJ4 17
    15 A = 4.0*F1 - F2 - 3.0*FO                                           FJ4 18
    16 B = 2.0*F2 - 4.0*F1 + 2.0*FO                                       FJ4 19
    17 EN = (FO + T2*A + T3*B)*(E**T1)*SQRT (DF)/(T1*DP)                  FJ4 20
    18 ODD = 0.0                                                          FJ4 21
    19 INT = 1                                                            FJ4 22
    20 V = 1.0                                                            FJ4 23
    21 EVEN = 0.0                                                         FJ4 24
       AREA1 = 0.0                                                        FJ4 25
       EE = 1.0                                                           FJ4 26
       IF (P - 1.0) 36, 22, 23                                            FJ4 27
    22 EE = SQRT (1.0 + FK2)                                              FJ4 28
    230ENDS = EE  + ((1.0 - OME**DP)**DG)*SQRT ((1.0 - OME**DP)**DQ + DF*FJ4 29
     1(OME**(2.0*DP - 2.0)))                                              FJ4 30
    24 DDP = 2.0*(DP - 1.0)                                               FJ4 31
    25 H = OME/V                                                          FJ4 32
    26 ODD = EVEN + ODD                                                   FJ4 33
    27 X = H/2.0                                                          FJ4 34
    28 EVEN = 0.0                                                         FJ4 35
    29 DO 31 I = 1, INT                                                   FJ4 36
    300EVEN = EVEN + ((1.0 - X**DP)**DG)*SQRT ((1.0 - X**DP)**DQ + DF*(X*FJ4 37
     1*DDP))                                                              FJ4 38
    31 X = X + H                                                          FJ4 39
    32 AREA = (ENDS + 4.0*EVEN + 2.0*ODD)*H/6.0                           FJ4 40
    33 R = ABSF(AREA1/AREA - 1.0) - ACC                                   FJ4 41
    34 IF (R) 43, 43, 35                                                  FJ4 42
    35 IF (INT - 16384) 39, 36, 36                                        FJ4 43
    36 WRITE OUTPUT TAPE 6, 37                                            FJ4 44
    37 FORMAT (25H J4(P,Q,S) NOT CONVERGENT)                              FJ4 45
    38 CALL SYSERR                                                        FJ4 46
    39 AREA1 = AREA                                                       FJ4 47
    40 INT = 2*INT                                                        FJ4 48
    41 V = 2.0*V                                                          FJ4 49
    42 GO TO 25                                                           FJ4 50
    43 FJ4 = AREA + EN                                                    FJ4 51
    44 RETURN                                                             FJ4 52
       END                                                                FJ4 53
```

```
$      FORTRAN IBM
$PUNCH OBJECT
      FUNCTION FJ5(P,Q,S,FK2, CTAI, ACC)                                FJ5 01
      OME = 1.0 - ETAI                                                  FJ5 02
    2 G = S - Q - 1.0                                                   FJ5 03
    3 QQ = 2.0*(1.0 - Q)                                                FJ5 04
    4 DEL = 1.0 - 1.0/P                                                 FJ5 05
      TDEL = 2.0*DEL                                                    FJ5 06
      FK = FK2                                                          FJ5 07
      IF (Q - 1.0) 6, 5, 35                                             FJ5 08
    5 FK = 1.0 + FK                                                     FJ5 09
    6 F0 = 1.0/SQRT (FK)                                                FJ5 10
      EPS = 1.0 -OME**P                                                 FJ5 11
    7 TPS = 0.5*EPS                                                     FJ5 12
    8 F1 = ((1.0 - TPS)**DEL)/SQRT (TPS**QQ + FK2*((1.0 - TPS)**TDEL))  FJ5 13
    9 F2 = ((1.0 - EPS)**DEL)/SQRT (EPS**QQ + FK2*((1.0 - EPS)**TDEL))  FJ5 14
   10 A = 4.0*F1 - F2 - 3.0*F0                                          FJ5 15
   11 B = 2.0*F2 - 4.0*F1 + 2.0*F0                                      FJ5 16
   12 T1 = G + 1.0                                                      FJ5 17
   13 T2 = T1/(G + 2.0)                                                 FJ5 18
   14 T3 = T1/(G + 3.0)                                                 FJ5 19
   15 EN = (F0 + T2*A + T3*B)*(EPS**T1)/(T1*P)                          FJ5 20
   16 ODD = 0.0                                                         FJ5 21
   17 INT = 1                                                           FJ5 22
   18 V = 1.0                                                           FJ5 23
   19 EVEN = 0.0                                                        FJ5 24
      AREA1 = 0.0                                                       FJ5 25
      EE = 0.0                                                          FJ5 26
      IF (P - 1.0) 35, 20, 21                                           FJ5 27
   20 EE = 1.0/SQRT (1.0 + FK2)                                         FJ5 28
   21 ENDS = SQRT ((1.0 - OME**P)**QQ + FK2*(OME**(2.0*P - 2.0)))       FJ5 29
   22 PP = 2.0*(P - 1.0)                                                FJ5 30
   23 ENDS = (OME**PP)*((1.0 - OME**P)**G)/ENDS + EE                    FJ5 31
   24 H = OME/V                                                         FJ5 32
   25 ODD = EVEN + ODD                                                  FJ5 33
   26 X = H/2.0                                                         FJ5 34
   27 EVEN = 0.0                                                        FJ5 35
   28 DO 30 I = 1, INT                                                  FJ5 36
   290EVEN = EVEN + (X**PP)*((1.0 - X**P)**G)/SQRT ((1.0 - X**P)**QQ +  FJ5 37
     1 FK2*(X**PP))                                                     FJ5 38
   30 X = X + H                                                         FJ5 39
   31 AREA = (ENDS + 4.0*EVEN + 2.0*ODD)*H/6.0                          FJ5 40
   32 R = ABSF(AREA1/AREA - 1.0) - ACC                                  FJ5 41
   33 IF (R) 42, 42, 34                                                 FJ5 42
   34 IF (INT - 16384) 38, 35, 35                                       FJ5 43
   35 WRITE OUTPUT TAPE 6, 36                                           FJ5 44
   36 FORMAT (25H J5(P,Q,S) NOT CONVERGENT)                             FJ5 45
   37 CALL SYSERR                                                       FJ5 46
   38 AREA1 = AREA                                                      FJ5 47
   39 INT = 2*INT                                                       FJ5 48
   40 V = 2.0*V                                                         FJ5 49
   41 GO TO 24                                                          FJ5 50
   42 FJ5 = AREA + EN                                                   FJ5 51
   43 RETURN                                                            FJ5 52
      END                                                               FJ5 53
```

```
$     FORTRAN IBM
$PUNCH OBJECT
      FUNCTION FJ6(P,Q,R,S,FK2, ETAI, ACC)                              FJ6 01
    1 TQ = 2.0*(1.0 - Q)                                                FJ6 02
    2 TP = 2.0*(P - 1.0)                                                FJ6 03
    3 TR = 2.0*(R - 1.0)                                                FJ6 04
    4 G = S - Q - 1.0                                                   FJ6 05
    5 TD = 2.0*(1.0 - 1.0/P)                                            FJ6 06
      EPS = 1.0- (1.-ETAI)**P                                           FJ6 07
    6 TPS = 0.5*EPS                                                     FJ6 08
    7 TT = 2.0*R/P - 1.0 - 1.0/P                                        FJ6 09
    8 T1 = G + 1.0                                                      FJ6 10
    9 T2 = T1/(G + 2.0)                                                 FJ6 11
      T3 = T1/(G + 3.0)                                                 FJ6 12
      FK = FK2                                                          FJ6 13
      IF (Q - 1.0) 11, 10, 39                                           FJ6 14
   10 FK = 1.0 + FK                                                     FJ6 15
   11 F0 = 1.0/SQRT (FK)                                                FJ6 16
   12 OMW = 1.0 - TPS                                                   FJ6 17
   13 F1 = (OMW**TT)/SQRT (TPS**TQ + FK2*(OMW**TD))                     FJ6 18
   14 OMW = 1.0 - EPS                                                   FJ6 19
   15 F2 = (OMW**TT)/SQRT (EPS**TQ + FK2*(OMW**TD))                     FJ6 20
      OME = 1.0 - ETAI                                                  FJ6 21
   17 A = 4.0*F1 - F2 - 3.0*F0                                          FJ6 22
   18 B = 2.0*F2 - 4.0*F1 + 2.0*F0                                      FJ6 23
   190EN = (ELOG (EPS)*(F0 + T2*A + T3*B) - (F0 + T2*T2*A + T3*T3*B)/T1)FJ6 24
     1*(EPS**T1)/(T1*P)                                                 FJ6 25
   20 ODD = 0.0                                                         FJ6 26
   21 INT = 1                                                           FJ6 27
   22 V = 1.0                                                           FJ6 28
   23 EVEN = 0.0                                                        FJ6 29
   24 AREA1 = 0.0                                                       FJ6 30
   25 OMW = 1.0 - OME**P                                                FJ6 31
   260ENDS =((OME**TR)*(OMW**G)/SQRT (OMW**TQ + FK2*(OME**TP)))*ELOG (OMFJ6 32
     1W)                                                                FJ6 33
   27 H = OME/V                                                         FJ6 34
   28 ODD = EVEN + ODD                                                  FJ6 35
   29 X = H/2.0                                                         FJ6 36
   30 EVEN = 0.0                                                        FJ6 37
   31 DO 34 I = 1, INT                                                  FJ6 38
   32 OMW = 1.0 - X**P                                                  FJ6 39
   330EVEN = EVEN +((X**TR)*(OMW**G)/SQRT (OMW**TQ + FK2*(X**TP)))*ELOG FJ6 40
     1(OMW)                                                             FJ6 41
   34 X = X + H                                                         FJ6 42
   35 AREA = (ENDS + 4.0*EVEN + 2.0*ODD)*H/6.0                          FJ6 43
   36 RR = ABSF(AREA1/AREA - 1.0) - ACC                                 FJ6 44
   37 IF (RR) 46, 46, 38                                                FJ6 45
   38 IF (INT - 16384) 42, 39, 39                                       FJ6 46
   39 WRITE OUTPUT TAPE 6, 40                                           FJ6 47
   40 FORMAT (27H J6(P,Q,R,S) NOT CONVERGENT)                           FJ6 48
   41 CALL SYSERR                                                       FJ6 49
   42 AREA1 = AREA                                                      FJ6 50
   43 INT = 2*INT                                                       FJ6 51
   44 V = 2.0*V                                                         FJ6 52
   45 GO TO 27                                                          FJ6 53
   46 FJ6 = AREA + EN                                                   FJ6 54
   47 RETURN                                                            FJ6 55
      END                                                               FJ6 56
```

```
$      FORTRAN IBM
$PUNCH OBJECT
      FUNCTION FJ7(P,Q,S,T,FK2,EPS, ETAI, ACC)
    1 TQ = 2.0*(1.0 - Q)
    2 TP = 2.0*(P - 1.0)
    3 G = S - Q - 2.0
    4 T1 = T + 1.0
    5 T2 = T1/(T + 2.0)
      T3 = T1/(T + 3.0)
      IF (P - 1.0) 53, 6, 7
    6 FO = 1.0/SQRT (1.0 + FK2)
      GO TO 8
    7 FO = 1.0
    8 TPS = 0.5*EPS
    9 OMW = 1.0 - TPS**P
   10 F1 = (OMW**G)/SQRT (OMW**TQ + FK2*(TPS**TP))
   11 OMW = 1.0 - EPS**P
   12 F2 = (OMW**G)/SQRT (OMW**TQ + FK2*(EPS**TP))
   13 A = 4.0*F1 - F2 - 3.0*FO
   14 B = 2.0*F2 - 4.0*F1 + 2.0*FO
   15 TT = FO + T2*A + T3*B
   16 TU = FO + T2*T2*A + T3*T3*B
   17 EN = (ELOG (EPS)*TT - TU/T1)*(EPS**T1)/T1
      OME = 1.0 - ETAI
   19 T4 = 2.0*(1.0 - 1.0/P)
   20 T5 = T/P - 1.0 + 1.0/P
   21 T1 = G + 2.0
   22 T2 = T1/(G + 3.0)
      T3 = T1/(G + 4.0)
      FK = FK2
      IF (Q - 1.0) 24, 23, 53
   23 FK = 1.0 + FK
   24 FO = -1.0/SQRT (FK)
      E = 1.0- OME**P
      TPS = 0.5 * E
      OMW = 1.- TPS
   27 F1 =  (OMW**T5)/SQRT (TPS**TQ + FK2*(OMW**T4)) * ELOG(OMW) /TPS
   28 OMW = 1.0 - E
      F2 =  (OMW**T5)/SQRT ( E **TQ + FK2*(OMW**T4)) * ELOG(OMW) /E
   30 A = 4.0*F1 - F2 - 3.0*FO
   31 B = 2.0*F2 - 4.0*F1 + 2.0*FO
   32 EN = EN + (FO + T2*A + T3*B)*( E **T1)/(T1*P*P)
   33 ODD = 0.0
   34 INT = 1
   35 V = 1.0
   36 EVEN = 0.0
   37 AREA1 = 0.0
      OMW = 1.0 - EPS**P
   38 ENDS = (EPS**T)*(OMW**G)*ELOG (EPS)/SQRT (OMW**TQ + FK2*(EPS**TP))
   39 O = 1.0 - OME**P
   40 ENDS=ENDS+(OME**T)*(O**G)*ELOG (OME)/SQRT (O**TQ+FK2*(OME**TP))
   41 H = (OME - EPS)/V
   42 ODD = EVEN + ODD
   43 X = EPS + H/2.0
   44 EVEN = 0.0
   45 DO 48 I = 1, INT
   46 O = 1.0 - X**P
   47 EVEN = EVEN + (X**T)*(O**G)*ELOG (X)/SQRT (O**TQ + FK2*(X**TP))
   48 X = X + H
   49 AREA = (ENDS + 4.0*EVEN + 2.0*ODD)*H/6.0
   50 R = ABSF(AREA1/AREA - 1.0) - ACC
   51 IF (R) 60, 60, 52
   52 IF (INT - 16384) 56, 53, 53
   53 WRITE OUTPUT TAPE 6, 54
   54 FORMAT (27H J7(P,Q,S,T) NOT CONVERGENT)
   55 CALL SYSERR
   56 AREA1 = AREA
   57 INT = 2*INT
   58 V = 2.0*V
   59 GO TO 41
   60 FJ7 = AREA + EN
   61 RETURN
      END
```

REFERENCES

1. "Geometrical and Inertial Properties of a Class of Thin Shells of Revolution", by
   Will J. Worley and Han-chung Wang
   National Aeronautics and Space Administration, Grant No. NsG-434, N.A.S.A. Contractor Report CR-89, September, 1964, 208 pages.

2. "Geometrical and Inertial Properties of a Class of Thin Shells of a General Type" by
   Will J. Worley and Han-chung Wang
   National Aeronautics and Space Administration, Grant No. NsG-434, Supplement No. 1, NASA Contractor Report CR-271, Aug. 1965, 67 pages.

3. "Minimum Weight Design of Cylindrical Shells" by
   Walter Freiberger
   Journal of Applied Mechanics, Vol. 23, No. 4., Trans. ASME, Dec. 1956, pp. 576-580.

4. "On the Optimum Design of Shells" by
   R. T. Shield
   Journal of Applied Mechanics, Vol. 27, No. 2, Trans. ASME, June 1960, pp. 316-322.

5. "Numerical Analysis of Unsymmetrical Bending of Shells of Revolution" by
   B. Budiansky and P. P. Radkowski
   AIAA Journal, Vol. 1, No. 8, Aug. 1963, pp. 1833-1842.

6. "Numerical Analysis of Equations of Thin Shell of Revolution" by
   P. P. Radkowski, R. M. Davis and M. R. Bolduc,
   American Rocket Society Journal, Vol. 32, No. 1, Jan. 1962, pp. 36-41.

7. "Analysis of Shells of Revolution Subjected to Symmetrical and Nonsymmetrical Loads" by
   A. Kalnins
   Journal of Applied Mechanics, Vol. 31, No. 3, Trans. ASME Sept. 1964, pp. 467-476.

TABLE I ITERATION RESULTS FOR $\lambda=\lambda_1$, $\mu=\mu_1$, b/a=1, $\alpha_o=1.5$, $\beta_o=1.5$

Step	α	β	$\frac{V}{2\pi ab^2}$	$\frac{W(\frac{\sigma_o}{p_o})}{4\pi ab^2\rho g}$	$\frac{t}{b}(\frac{\sigma_o}{p_o})$	F_α	F_β	$\Delta\alpha$	$\Delta\beta$
1	1.5000	1.5000	.53743	.43810	.5000	.29499	.24269	.0885	.0728
2	1.5885	1.5728	.56271	.44979	.5000	.27202	.22630	.0816	.0679
3	1.6701	1.6407	.58448	.46001	.5000	.25255	.21206	.0758	.0636
4	1.7459	1.7043	.60343	.46903	.5000	.26893	.22758	.2228	.1885
5	1.9686	1.8928	.65276	.49306	.5000	.83353	.72064	.2501	.2162
6	2.2187	2.1090	.69862	.52328	.5069	-.28840	-.28654	-.0865	-.0860
7	2.1322	2.0231	.68274	.51165	.5035	-.05475	-.08112	-.0164	-.0243
8	2.1189	1.9915	.67861	.50850	.5025	.01648	-.01889	.0049	-.0057
9	2.1288	1.9802	.67860	.50846	.5024	.01672	-.01885	.0050	-.0057
10	2.1389	1.9689	.67859	.50841	.5024	.01689	-.01889	.0051	-.0057
11	2.1490	1.9575	.67859	.50837	.5023	.01705	-.01896	.0051	-.0057
12	2.1593	1.9461	.67858	.50832	.5023	.01721	-.01899	.0052	-.0057
13	2.1696	1.9347	.67856	.50827	.5022	.01756	-.01886	.0053	-.0057
14	2.1801	1.9233	.67856	.50823	.5022	.01758	-.01905	.0053	-.0057
15	2.1998	1.9107	.67935	.50875	.5023	.00421	-.03187	.0021	-.0159
16	2.2151	1.8891	.67884	.50830	.5021	.01305	-.02400	.0065	-.0120
17	2.2323	1.8690	.67865	.50807	.5020	.01655	-.02106	.0083	-.0105
18	2.2505	1.8494	.67857	.50793	.5019	.01806	-.01998	.0090	-.0100
19	2.2692	1.8299	.67853	.50783	.5017	.01883	-.01960	.0094	-.0098
20	2.2883	1.8105	.67851	.50774	.5016	.01927	-.01957	.0096	-.0098
21	2.3078	1.7911	.67849	.50764	.5015	.01971	-.01951	.0099	-.0098
22	2.3277	1.7716	.67847	.50754	.5014	.02008	-.01952	.0100	-.0098
23	2.3478	1.7521	.67846	.50745	.5013	.02044	-.01955	.0102	-.0098
24	2.3684	1.7326	.67844	.50735	.5012	.02080	-.01959	.0104	-.0098
25	2.3893	1.7130	.67843	.50725	.5011	.02116	-.01963	.0106	-.0098
26	2.4105	1.6933	.67842	.50715	.5010	.02153	-.01966	.0108	-.0098
27	2.4321	1.6737	.67840	.50704	.5008	.02189	-.01968	.0110	-.0098
28	2.4541	1.6540	.67838	.50693	.5007	.02227	-.01970	.0111	-.0099
29	2.4765	1.6343	.67837	.50681	.5005	.02276	-.01955	.0114	-.0098
30	2.4992	1.6146	.67836	.50670	.5004	.02310	-.01967	.0116	-.0098
31	2.5224	1.5948	.67834	.50659	.5003	.02345	-.01974	.0117	-.0099

TABLE II ITERATION RESULTS FOR $\lambda=\lambda_1$, $\mu=\mu_1$, $b/a=1$, $\alpha_o=2.0$, $\beta_o=3.0$

Step	α	β	$\frac{V}{2\pi ab^2}$	$\frac{W(\frac{\sigma_o}{p_o})}{4\pi ab^2 \rho g}$	$\frac{t}{b}(\frac{\sigma_o}{p_o})$	F_α	F_β	$\Delta\alpha$	$\Delta\beta$
1	2.0000	3.0000	.73915	.55381	.5151	-.99584	-.64496	-.2988	-.1935
2	1.7013	2.8065	.69653	.52338	.5075	-.32702	-.21005	-.0981	-.0630
3	1.6031	2.7435	.68018	.51171	.5042	-.02135	-.03251	-.0064	-.0098
4	1.5967	2.7337	.67866	.51062	.5040	.01177	-.01391	.0035	-.0042
5	1.6003	2.7296	.67881	.51071	.5039	.00845	-.01582	.0025	-.0048
6	1.6028	2.7248	.67879	.51069	.5039	.00887	-.01562	.0027	-.0047
7	1.6055	2.7201	.67879	.51068	.5039	.00886	-.01566	.0027	-.0047
8	1.6108	2.7107	.67879	.51066	.5039	.00893	-.01570	.0027	-.0047
9	1.6162	2.7013	.67879	.51064	.5039	.00900	-.01574	.0027	-.0047
10	1.6270	2.6824	.67879	.51060	.5039	.00914	-.01582	.0027	-.0047
11	1.6381	2.6634	.67876	.51055	.5039	.00955	-.01574	.0029	-.0047
12	1.6493	2.6443	.67877	.51051	.5039	.00943	-.01598	.0028	-.0048
13	1.6607	2.6251	.67877	.51047	.5039	.00958	-.01606	.0029	-.0048
14	1.6723	2.6058	.67876	.51042	.5039	.00973	-.01615	.0029	-.0048
15	1.6840	2.5864	.67876	.51038	.5038	.00989	-.01623	.0030	-.0049
16	1.6960	2.5668	.67876	.51033	.5038	.01005	-.01632	.0030	-.0049
17	1.7081	2.5472	.67875	.51028	.5038	.01022	-.01640	.0031	-.0049
18	1.7204	2.5275	.67875	.51023	.5038	.01039	-.01649	.0031	-.0050
19	1.7330	2.5077	.67875	.51018	.5038	.01056	-.01658	.0032	-.0050
20	1.7457	2.4877	.67874	.51012	.5037	.01074	-.01667	.0032	-.0050
21	1.7587	2.4677	.67874	.51007	.5037	.01093	-.01676	.0033	-.0050
22	1.7719	2.4475	.67874	.51001	.5037	.01112	-.01685	.0033	-.0051
23	1.7853	2.4273	.67873	.50996	.5036	.01131	-.01694	.0034	-.0051
24	1.7989	2.4071	.67873	.50990	.5036	.01147	-.01704	.0034	-.0051
25	1.8128	2.3866	.67872	.50984	.5036	.01171	-.01711	.0035	-.0051
26	1.8269	2.3660	.67872	.50978	.5035	.01192	-.01721	.0036	-.0052
27	1.8412	2.3453	.67870	.50972	.5035	.01205	-.01717	.0036	-.0052
28	1.8558	2.3247	.67870	.50966	.5035	.01251	-.01728	.0038	-.0052
29	1.8708	2.3038	.67870	.50958	.5034	.01277	-.01734	.0038	-.0052
30	1.8860	2.2828	.67871	.50952	.5034	.01280	-.01758	.0038	-.0053

TABLE III ITERATION RESULTS FOR $\lambda=\lambda_1$, $\mu=\mu_1$, b/a=2, α_o=1.5, β_o=1.5

Step	α	β	$\frac{V}{2\pi ab^2}$	$\frac{W(\frac{\sigma_o}{p_o})}{4\pi ab^2 \rho g}$	$\frac{t}{b}(\frac{\sigma_o}{p_o})$	F_α	F_β	$\Delta\alpha$	$\Delta\beta$
1	1.5000	1.5000	.53743	.62673	.5000	.29499	.24269	.0885	.0728
2	1.5885	1.5728	.56271	.63813	.5000	.27202	.22630	.0816	.0679
3	1.6701	1.6407	.58448	.64838	.5000	.25255	.21206	.0758	.0636
4	1.7459	1.7043	.60343	.65763	.5000	26.8590	22.7980	.2312	.1963
5	1.9771	1.9006	.65449	.68397	.5000	.72402	.70662	.2172	.2120
6	2.1944	2.1126	.69661	.70714	.5000	-.06577	.03817	-.0197	.0115
7	2.1303	2.1692	.69507	.70564	.5000	-.04389	.05289	-.0132	.0159
8	2.0792	2.2325	.69515	.70509	.5000	-.04058	.05212	-.0122	.0156
9	2.0317	2.2945	.69528	.70462	.5000	-.03797	.05084	-.0114	.0153
10	1.9872	2.3549	.69540	.70419	.5000	-.03561	.04950	-.0107	.0149
11	1.9455	2.4137	.69553	.70380	.5000	-.03342	.04814	-.0100	.0144
12	1.9063	2.4709	.69566	.70343	.5000	-.03135	.04680	-.0094	.0140
13	1.8696	2.5264	.69579	.70310	.5000	-.02940	.04543	-.0088	.0136
14	1.8012	2.6200	.69601	.70259	.5000	-.02680	.04309	-.0161	.0259
15	1.7494	2.7211	.69626	.70210	.5000	-.02346	.04044	-.0141	.0243
16	1.6956	2.8159	.69650	.70169	.5000	-.02089	.03796	-.0125	.0228
17	1.6475	2.9049	.69673	.70135	.5000	-.01870	.03561	-.0112	.0214
18	1.6043	2.9883	.69695	.70107	.5000	-.01680	.03339	-.0101	.0200
19	1.5656	3.0666	.69716	.70084	.5000	-.01515	.03132	-.0091	.0188
20	1.5305	3.1400	.69736	.70065	.5000	-.01370	.02938	-.0082	.0176
21	1.5065	3.1920	.69751	.70053	.5000	-.01273	.02801	-.0076	.0168
22	1.4770	3.2577	.69769	.70040	.5000	-.01159	.02628	-.0070	.0158
23	1.4502	3.3195	.69789	.70030	.5000	-.01091	.02453	-.0065	.0147
24	1.4256	3.3773	.69803	.70020	.5000	-.00966	.02317	-.0058	.0139
25	1.4032	3.4316	.69816	.70013	.5000	-.00884	.02178	-.0053	.0131
26	1.3826	3.4827	.69835	.70007	.5000	-.00817	.02043	-.0049	.0123
27	1.3636	3.5306	.69849	.70002	.5000	-.00747	.01923	-.0045	.0115
28	1.3463	3.5757	.69862	.69999	.5000	-.00688	.01809	-.0041	.0109
29	1.3302	3.6182	.69875	.69997	.5000	-.00641	.01702	-.0038	.0102
30	1.3154	3.6582	.69887	.69996	.5000	-.00590	.01601	-.0035	.0096
31	1.3017	3.6957	.69898	.69995	.5000	-.00544	.01509	-.0033	.0091

TABLE IV ITERATION RESULTS FOR $\lambda=\lambda_1$, $\mu=\mu_1$, b/a=2, α_o=2.0, β_o=3.0

Step	α	β	$\frac{V}{2\pi ab^2}$	$\frac{W(\frac{\sigma_o}{p_o})}{4\pi ab^2 \rho g}$	$\frac{t}{b}(\frac{\sigma_o}{p_o})$	F_α	F_β	$\Delta\alpha$	$\Delta\beta$
1	2.0000	3.0000	.73915	.72761	.5000	-.46258	-.21858	-.1388	-.0656
2	1.8612	2.9344	.72206	.71693	.5000	-.31011	-.12583	-.0930	-.0378
3	1.7682	2.8967	.70998	.70953	.5000	-.18587	-.05414	-.0558	-.0162
4	1.7124	2.8804	.70272	.70513	.5000	-.10019	-.00669	-.0301	-.0020
5	1.6824	2.8784	.69912	.70294	.5000	-.05330	.01847	-.0160	.0055
6	1.6664	2.8840	.69761	.70199	.5000	-.03224	.02937	-.0097	.0088
7	1.6567	2.8928	.69704	.70160	.5000	-.02379	.03344	-.0071	.0100
8	1.6496	2.9028	.69685	.70143	.5000	-.02050	.03476	-.0062	.0104
9	1.6434	2.9132	.69680	.70135	.5000	-.01915	.03506	-.0058	.0105
10	1.6377	2.9238	.69680	.70130	.5000	-.01851	.03498	-.0056	.0105
11	1.6161	2.9653	.69689	.70115	.5000	-.01735	.03398	-.0052	.0102
12	1.5957	3.0056	.69700	.70102	.5000	-.01646	.03292	-.0049	.0099
13	1.5763	3.0446	.69711	.70090	.5000	-.01563	.03188	-.0047	.0096
14	1.5579	3.0825	.69721	.70080	.5000	-.01485	.03088	-.0045	.0093
15	1.5361	3.1281	.69733	.70068	.5000	-.01392	.02969	-.0084	.0178
16	1.5039	3.1977	.69752	.70052	.5000	-.01263	.02786	-.0076	.0167
17	1.4746	3.2630	.69770	.70039	.5000	-.01148	.02615	-.0069	.0157
18	1.4480	3.3243	.69790	.70028	.5000	-.01048	.02455	-.0063	.0147
19	1.4237	3.3818	.69804	.70019	.5000	-.00958	.02306	-.0058	.0138
20	1.4014	3.4359	.69820	.70012	.5000	-.00878	.02167	-.0053	.0130
21	1.3809	3.4867	.69840	.70007	.5000	-.00811	.02033	-.0049	.0122

TABLE V ITERATION RESULTS FOR $\lambda = \lambda_2$, $\mu = \mu_2$, b/a=1

Step	α	β	$\frac{V}{2\pi ab^2}$	$\frac{W(\frac{\sigma_o}{p_o})}{4\pi ab^2 \rho g}$	$\frac{t}{b}(\frac{\sigma_o}{p_o})$	F_α	F_β	$\Delta\alpha$	$\Delta\beta$
1	3.0000	3.0000	.80611	.60146	.5238	-.05923	-.05224	-.1777	-.1567
2	2.8223	2.8433	.78954	.58952	.5219	-.06529	-.05730	-.1959	-.1719
3	2.6265	2.6714	.76878	.57453	.5192	-.07086	-.06187	-.2126	-.1856
4	2.4139	2.4858	.74268	.55566	.5153	-.07227	-.06293	-.2168	-.1888
5	2.1971	2.2970	.71143	.53294	.5097	-.05642	-.04991	-.1693	-.1497
6	2.0278	2.1473	.68281	.51206	.5039	-.00851	-.01119	-.0255	-.0336
7	2.0023	2.1137	.67719	.50794	.5026	.00605	.00053	.0182	.0016
8	2.0205	2.1153	.67928	.50944	.5030	.00046	-.00402	.0014	-.0121
1	2.5000	2.5000	.75000	.56092	.5164	-.07211	-.06410	-.0721	-.0641
2	2.4279	2.4359	.74052	.55404	.5149	-.07125	-.06341	-.0713	-.0634
3	2.3566	2.3725	.73062	.54684	.5132	-.06854	-.06117	-.0685	-.0612
4	2.2881	2.3113	.72056	.53951	.5114	-.06330	-.05683	-.0633	-.0568
5	2.2248	2.2545	.71074	.53233	.5095	-.05507	-.05002	-.0551	-.0500
6	2.1697	2.2045	.70172	.52577	.5077	-.04418	-.04101	-.0442	-.0410
7	2.1256	2.1635	.69414	.52022	.5062	-.03206	-.03100	-.0321	-.0310
8	2.0935	2.1325	.68835	.51599	.5049	-.02082	-.02172	-.0208	-.0217
9	2.0727	2.1108	.68437	.51307	.5041	-.01202	-.01445	-.0120	-.0145
10	2.0607	2.0963	.68188	.51123	.5035	-.00605	-.00952	-.0061	-.0095
11	2.0546	2.0868	.68041	.51015	.5032	-.00239	-.00650	-.0024	-.0065
1	4.0000	2.0000	.80000	.59625	.5212	-.04804	-.07411	-.0480	-.0741
2	3.9520	1.9259	.79348	.59140	.5203	-.04939	-.07776	-.0494	-.0778
3	3.9026	1.8481	.78631	.58605	.5192	-.05070	-.08166	-.0507	-.0817
4	3.8519	1.7665	.77837	.58011	.5178	-.05189	-.08576	-.0519	-.0858
5	3.8000	1.6807	.76958	.57350	.5163	-.05282	-.08990	-.0528	-.0899
6	3.7472	1.5908	.75982	.56610	.5144	-.05324	-.09376	-.0532	-.0938
7	3.6939	1.4971	.74898	.55782	.5122	-.05273	-.09669	-.0527	-.0967
8	3.6412	1.4004	.73704	.54864	.5096	-.05063	-.09744	-.0506	-.0974
9	3.5906	1.3029	.72414	.53862	.5065	-.04591	-.09385	-.0459	-.0939
10	3.5447	1.2091	.71082	.52814	.5030	-.03744	-.08295	-.0374	-.0830
11	3.5072	1.1261	.69826	.51874	.4994	-.00550	-.00138	-.0055	-.0014

TABLE VI ITERATION RESULTS FOR $\lambda=\lambda_3$, $\mu=\mu_3$, b/a=1

Step	α	β	$\frac{V}{2\pi ab^2}$	$\frac{W(\frac{\sigma_o}{p_o})}{4\pi ab^2 \rho g}$	$\frac{t}{b}(\frac{\sigma_o}{p_o})$	F_α	F_β	$\Delta\alpha$	$\Delta\beta$
1	1.5000	1.5000	.53743	.43810	.5000	.29499	.24269	.1838	.1512
2	1.6838	1.6512	.58787	.46162	.5000	.24949	.20988	.2121	.1784
3	1.8959	1.8296	.63745	.48553	.5000	.10529	.09039	.2106	.1808
4	2.1064	2.0104	.67902	.50889	.5026	.03332	.02410	.0667	.0482
5	2.1731	2.0586	.68997	.51692	.5050	.02556	.01793	.0511	.0359
6	2.2242	2.0944	.69793	.52274	.5067	.02035	.01376	.0407	.0275
7	2.3259	2.1606	.71246	.53335	.5100	.01173	.00680	.0235	.0136
8	2.3868	2.1937	.72019	.53898	.5110	.00761	.00342	.0152	.0069
9	2.4274	2.2100	.72473	.54228	.5118	.00532	.00153	.0107	.0031
10	2.4380	2.2131	.72584	.54306	.5120	.00479	.00109	.0096	.0022
1	2.5000	3.0000	.77759	.58111	.5206	-.01458	-.01210	-.0292	-.0242
2	2.4709	2.9758	.77447	.57887	.5202	-.01373	-.01149	-.0275	-.0230
3	2.4434	2.9528	.77145	.57670	.5198	-.01286	-.01088	-.0257	-.0218
4	2.4177	2.9311	.76856	.57463	.5194	-.01199	-.01027	-.0240	-.0205
5	2.3937	2.9105	.76581	.57265	.5190	-.01112	-.00966	-.0222	-.0193
6	2.3715	2.8912	.76320	.57077	.5187	-.01025	-.00906	-.0205	-.0181
7	2.3150	2.8403	.75626	.56578	.5177	-.00778	-.00736	-.0156	-.0147
8	2.2728	2.7992	.75070	.56178	.5168	-.00562	-.00590	-.0112	-.0118
9	2.2428	2.7663	.74642	.55869	.5162	-.00385	-.00471	-.0077	-.0094
10	2.2284	2.7482	.74419	.55708	.5158	-.00288	-.00406	-.0058	-.0081
1	3.0000	2.0000	.75000	.56001	.5151	-.00521	-.00949	-.0104	-.0190
2	2.9896	1.9810	.74796	.55847	.5148	-.00455	-.00885	-.0091	-.0177
3	2.9805	1.9633	.74606	.55707	.5144	-.00393	-.00825	-.0079	-.0165
4	2.9726	1.9468	.74430	.55576	.5141	-.00334	-.00767	-.0067	-.0153
5	2.9603	1.9172	.74123	.55346	.5135	-.00229	-.00663	-.0046	-.0133
6	2.9521	1.8916	.73869	.55154	.5130	-.00140	-.00575	-.0028	-.0115
7	2.9473	1.8694	.73660	.54997	.5126	-.00066	-.00500	-.0013	-.0100
8	2.9460	1.8594	.73570	.54930	.5124	-.00034	-.00468	-.0007	-.0094

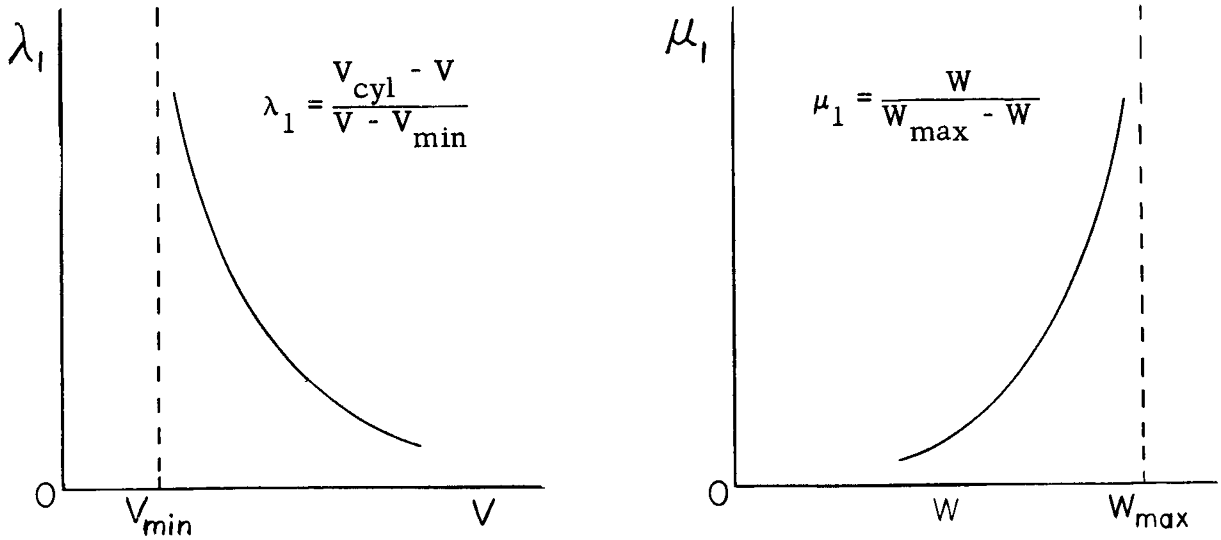

Fig. 1 Weighting Functions λ_1 and μ_1

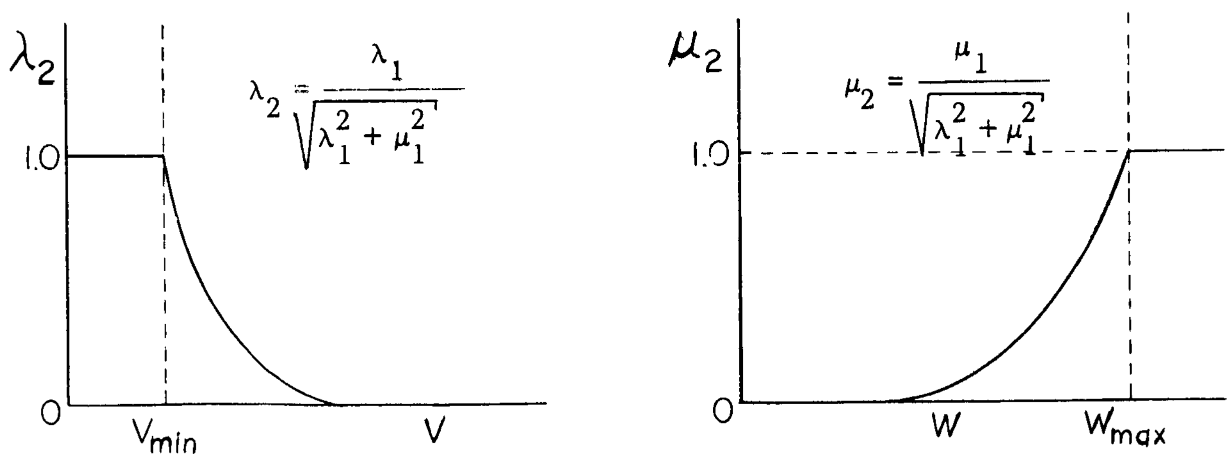

Fig. 2 Weighting Functions λ_2 and μ_2

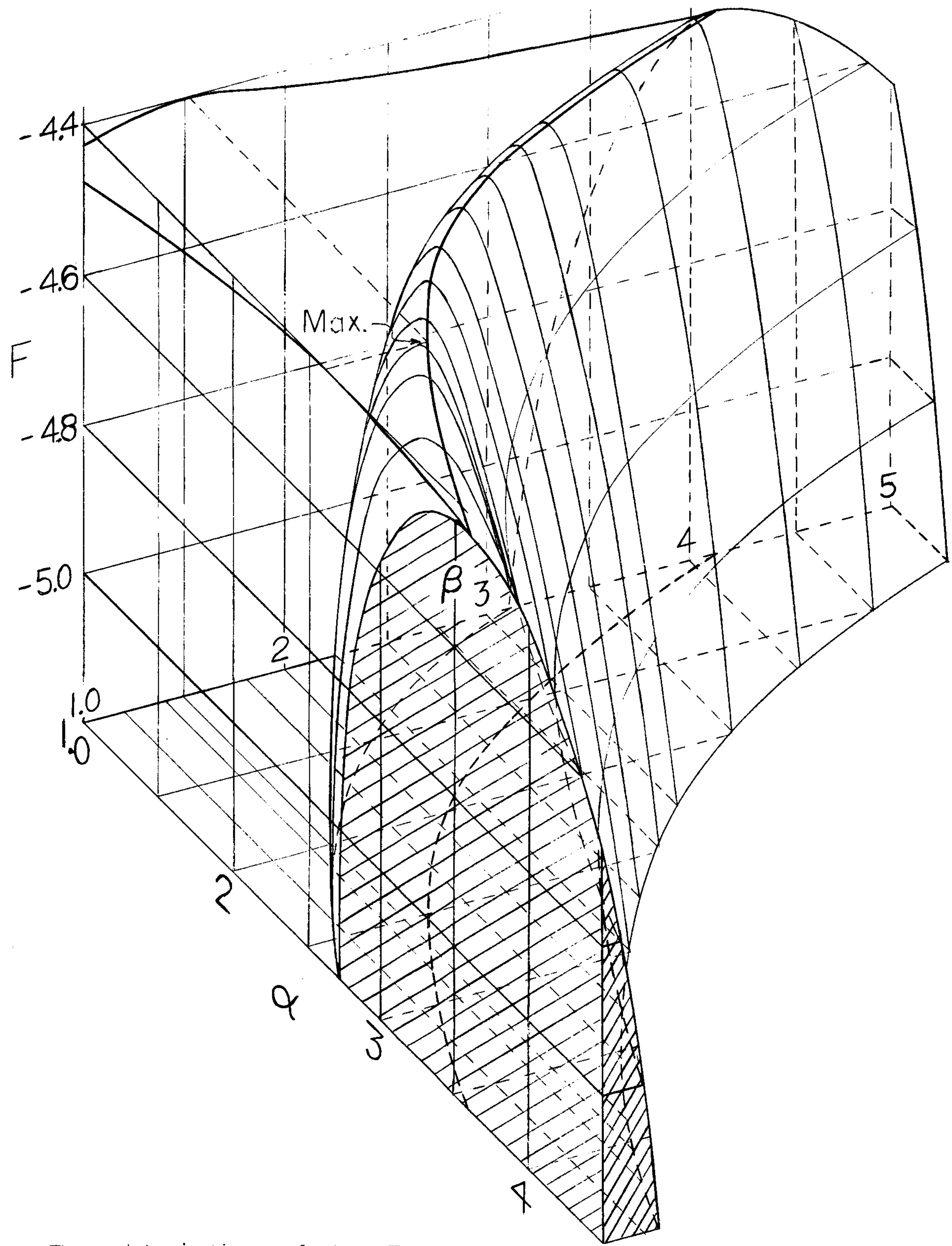

Fig. 3 Variation of the Function F with α and β

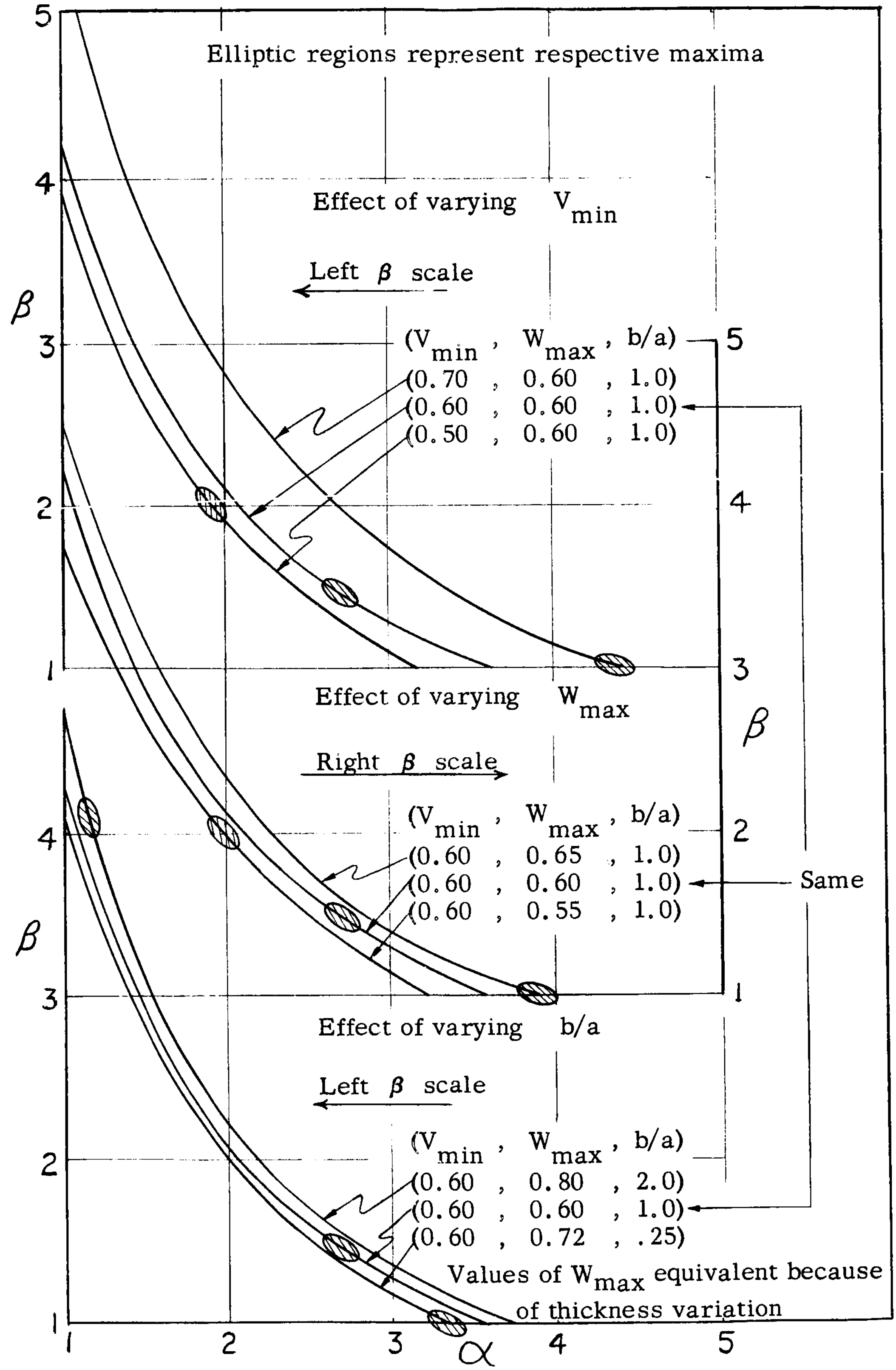

Fig. 4 Variation in Projections of Ridges on α-β Plane

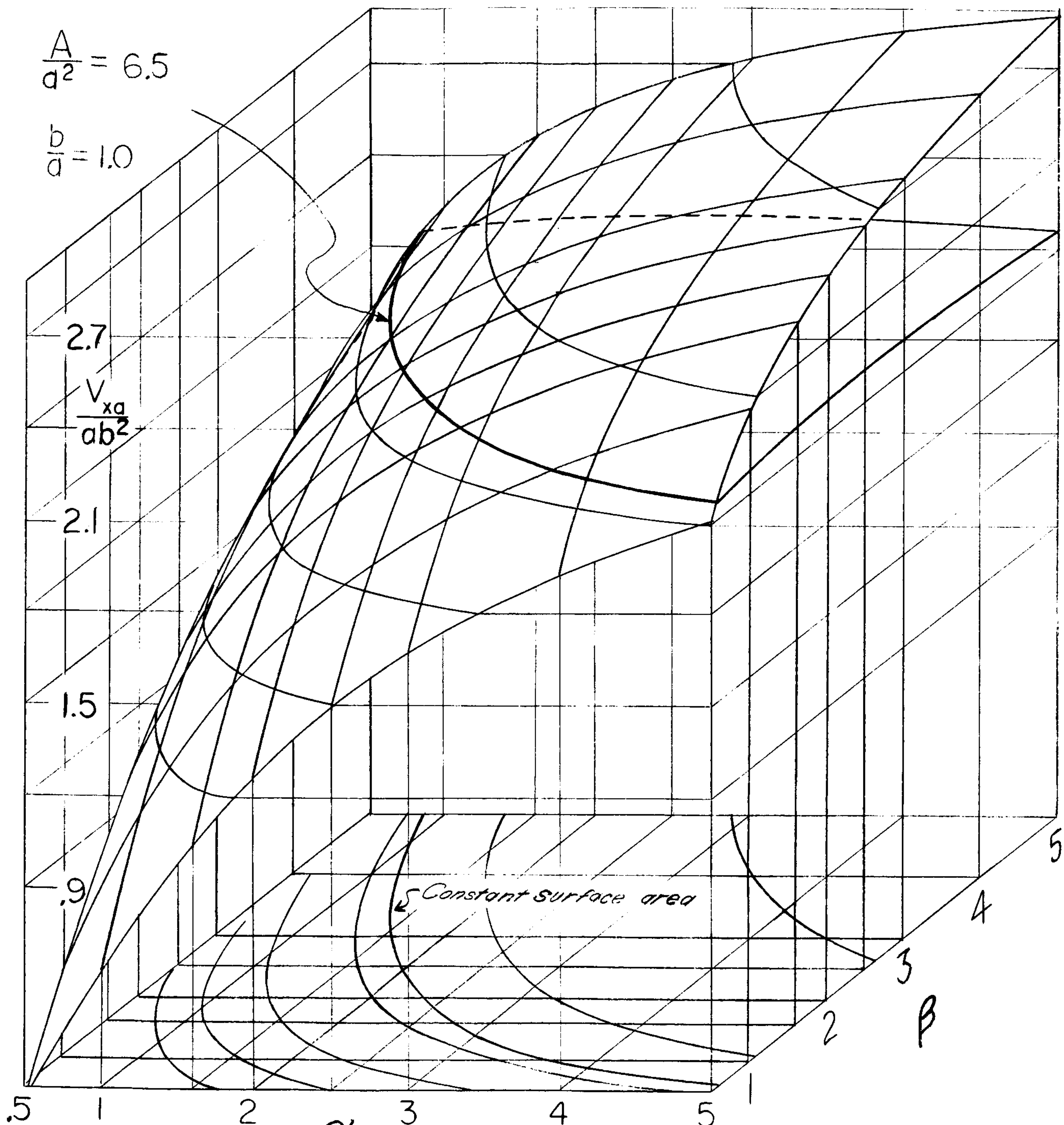

Fig.5 Variation in V_{xa}/ab^2 with α and β

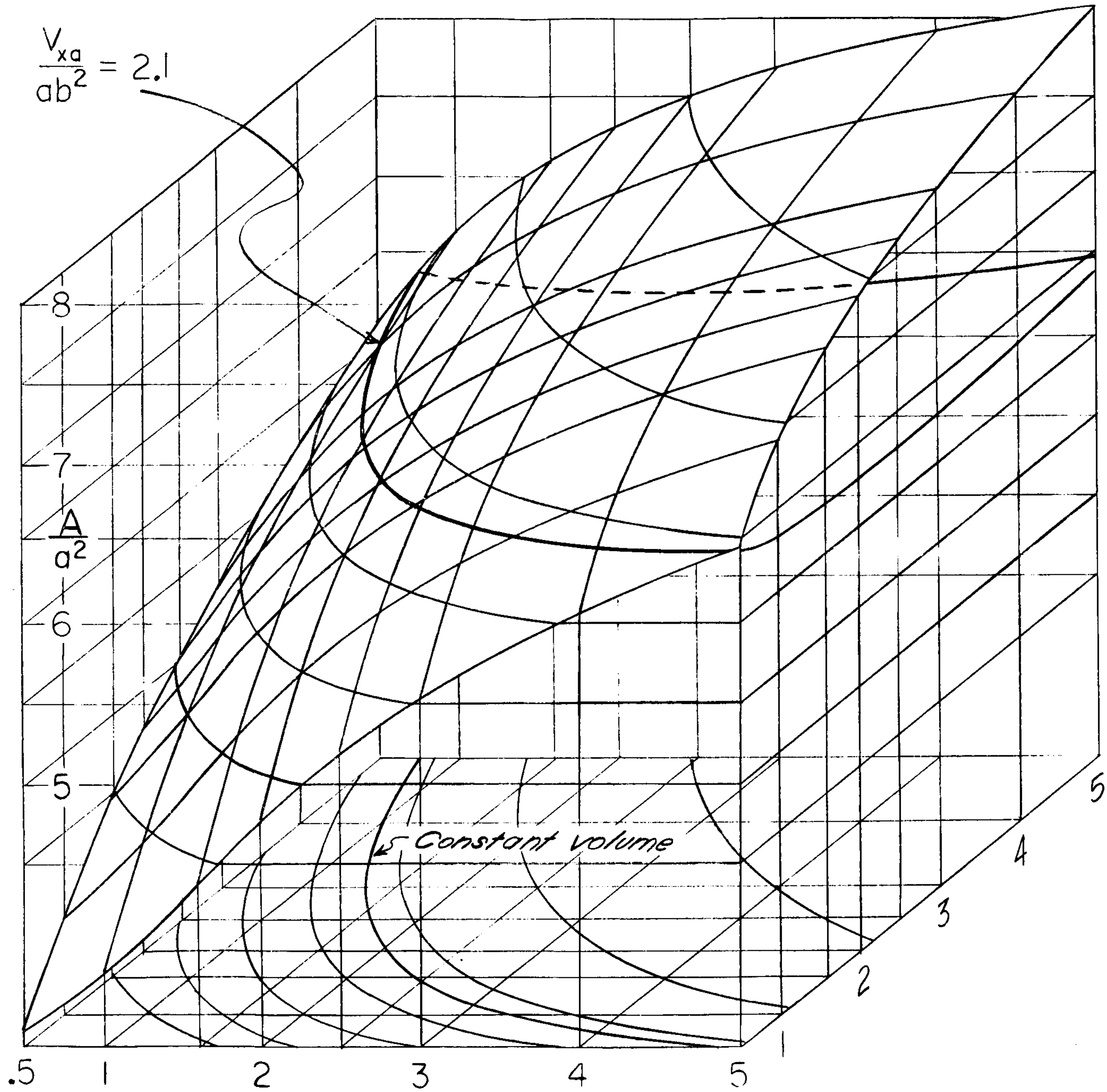

Fig.6 Variation in A/a^2 with α and β for $b/a = 1.0$

Recent T. & A. M. Reports

No.	Title	Date
263	"Evaluation of Bonding Characteristics of Deformed Wire," by Russell S. Jensen and Clyde E. Kesler.	May, 1964
264	"Geometrical and Inertial Properties of a Class of Thin Shells of a General Type," by Will J. Worley and Han-chung Wang.	June, 1964
265	"The Fatigue Toughness of Metals: A Data Compilation," by Gary R. Halford.	June, 1964
266	"On a Theory for Axisymmetric Elastic Shells of Moderate Thickness," by R. J. Nikolai and A. P. Boresi.	July, 1964
267	"The Stress Distribution in a Notched Semi-infinite Plate," by D. Shadman.	August, 1964
268	"Yield Behavior of Niobium Single Crystals," by D. C. Huffaker.	September, 1964
269	"Euler Buckling of a Ring-Reinforced Cylindrical Shell Subjected to External Pressure," by H. L. Langhaar, A. P. Boresi and C. C. Fretwell.	September, 1964
270	"Photoelastic Study of the Stresses Near Openings in Pressure Vessels," by N. C. Lind and C. E. Taylor.	October, 1964
271	"Theoretical and Experimental Investigation of the Tensile Moduli of Parallel Filament Composites," by John W. Melvin.	November, 1964
272	"Third Conference on Fundamental Research in Plain Concrete," by Clyde E. Kesler.	November, 1964
273	"The Effect of Temperature on Cycle Dependent Deformation," by Brian R. Gain.	December, 1964
274	"An Investigation Into the Effect of Environmental Treatments on the Strength of E Glass Fibers," by N. M. Cameron.	January, 1965
275	"Crack Extension in Fiberglass Reinforced Plastics and a Critical Examination of the General Fracture Criterion," by E. M. Wu and R. C. Reuter, Jr.	January, 1965
276	"Applications of Lasers to Photoelasticity," by C. E. Taylor, C. E. Bowman, W. P. North, and W. F. Swinson.	February, 1965

PUBLIC SERVICE

RESEARCH TAM TEACHING

UNIVERSITY OF ILLINOIS