# Optimization of parabolic steel box arches.

**Hani Mahmoud Mohamed Hasan Mohdaly**

Civil Engineering

January 1997

Abstract

A generalized optimization procedure has ben formulated and coded into the computer to find the least-weight design of a parabolic box-shaped steel arch of variable corss section subjected to a set of constraints, imposed by geometrical, strength and practical requirements. The strength reqquirements of AISC specifications have been followed. The web is considered to have a smooth profile. The flange areas are varied on the basis of a preassigned number of cut-off points. The iterative search procedure which incorporates dynamic programming and curve smoothening subroutines yield an optimum design within the prescribed constraints.

The applicability of the developed optimization program has been demonstrated with several illustrative examples, which also provide some interesting observations with regard to optimum design. A parametric study realted to rise to span ratio, span length, load intensities and web plate thickness is carried out through several examples. The sensitivity of a parabolic arch to the two important design variables, the span to rise ratio and the web thickness, is investigated.

# Optimization of Parabolic Steel Box Arches

by

Hani Mahmoud Mohamed Hasan Mohdaly

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

**MASTER OF SCIENCE**

In

**CIVIL ENGINEERING**

January, 1997

# OPTIMIZATION OF PARABOLIC STEEL BOX ARCHES

BY

**HANI MAHMOUD MOHAMED HASAN MOHDALY**

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

**KING FAHD UNIVERSITY OF PETROLEUM & MINERALS**

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

**MASTER OF SCIENCE**

**In**

**CIVIL ENGINEERING**

JANUARY, 1997

UMI Number: 1385299



UMI
300 North Zeeb Road
Ann Arbor, MI 48103

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS
DHAHRAN 31261, SAUDI ARABIA

COLLEGE OF GRADUATE STUDIES

This thesis, written by **Hani Mahmoud Mohamed Hasan Mohdaly** under the direction of his thesis Advisor and approved by his Thesis Committee, has been presented to and accepted by the Dean of the College of Graduate Studies, in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE.

Thesis Committee

Dr. Abul Kalam Azad
Thesis Adviser

Dr. Khaled Y. Medallah
Co-Advisor

Dr. Mohammad H. Baluch
Member

Dr. Saeid A. Alghamdi
Member

Dr. Sahl Abduljauwad
Department Chairman

Dr. Abdallah M. Al-Shehri
Dean, College of Graduate Studies

19-2-97
Date

This thesis is dedicated to my parents.

## ACKNOWLEDGMENT

Praise and thanks be to Almighty Allah for his limitless help and guidance; and peace and prayers be upon his Prophet.

Acknowledgment is due to *King Fahd University of Petroleum and Minerals* for having given me the opportunity to achieve this work through its tremendous facilities.

My expression of thanks to my thesis advisor, *Prof. Abul Kalam Azad*, does not arise from convention. Rather, from my heart-felt gratitude for his constant counsel, matchless patience and extraordinary attentiveness. I am deeply indebted to my thesis co-advisor *Dr Khaled Medallah,* for his guidance, generous support and encouragement at all time. I would also like to express my gratefulness and appreciation to *Dr. Saeid A. Alghamdi* for the great effort exerted by him in reviewing the thesis and for his valuable suggestions and helpful remarks. I would like to thank the committee member, *Prof. M. Baluch,* for his kind co-operation and for all the courses he taught me during my study in the university.

I would further express my thanks and appreciation to my company *Rabiah & Nassar Precast Factory* and to *Mr. Ayman Zaid, Mr. Mahmoud Abulwafa* and *Mr. Abdulwahab Al-Nassar* for their understanding, encouragement and support.

I express my special thanks to my dear friend *Yassir Awad* for his care, help in final thesis editing and his support. Special thanks are to my friend *Ibrahium AbdulRaof* for

his assistance in drawing thesis charts, concern at all times and encourage. I also extend my special thanks to *Salauddin Mahmoud*, *Asad Alrahman*, *Wael Elleithy*, all other friends, teachers and colleagues for their help, moral support and memorable days we shared together.

Last but not the least, I thank my parents, brothers & sister for their love, sacrifices, prayers and understanding.

## TABLE OF CONTENTS

## LIST OF FIGURES

# ABSTRACT

FULL NAME OF STUDENT : **Hani Mahmoud Mohmed Hasen Mohdaly**

TITLE OF STUDY : **Optimization of Parabolic Steel Box Arches**

MAJOR FIELD : **Civil Engineering ( Structures)**

DATE OF DEGREE : **January 1997**

A generalized optimization procedure has been formulated and coded into the computer to find the least-weight design of a parabolic box-shaped steel arch of variable cross section subjected to a set of constraints, imposed by geometrical, strength and practical requirements. The strength requirements of AISC specifications have been followed. The web is considered to have a smooth profile. The flange areas are varied on the basis of a preassigned number of cut-off points. The iterative search procedure which incorporates dynamic programming and curve smoothening subroutines yield an optimum design within the prescribed constraints.

The applicability of the developed optimization program has been demonstrated with several illustrative examples, which also provide some interesting observations with regard to optimum design. A parametric study related to rise to span ratio , span length, load intensities and web plate thickness is carried out through several examples. The sensitivity of a parabolic arch to the two important design variables, the span to rise ratio and the web thickness, is investigated.


**MASTER OF SCIENCE DEGREE**


**KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS**
**Dhahran, Saudi Arabia**
**January 1997**

## خلاصــة الرسالــة

اسم الطالب الكامل : **هاني محمود محمد حسن مهدلي**

عنوان الدراسة : **إيجاد التصميم الأمثل للأقواس الحديدية ذات الأشكال الإقطاعية المكافئة والمقاطع العرضية الصندوقية**

التخصص : **هندسة مدنية**

تاريخ الشهادة : **يناير ١٩٩٧**

تناولت الرسالة تطوير طريقة شاملة للتصميم الأمثل للأقواس الحديدية ذات الأشكال الإقطاعية المكافئة والمقاطع العرضية الصندوقية غير المنتظمة والمعرضة لمجموعة من القيود المتعلقة بالشكل والأبعاد الهندسية وقوة الإجهاد والاعتبارات العملية. إن وزن الحديد المستخدم لصناعة القوس هي الدالة التي يراد الوصول إلي حدها الأدنى. إن التصميم الأمثل يتفق مع شروط مواصفات البناء AISC .إن عمق المقطع متغير تغير منتظم ناعم. كما يستخدم عدد محدد سابقا من الصفائح متغيرة السماكة لعرض المقطع. وقد تم استخدام النهج البحثي التكراري المدمج مع برنامج فرعي للبرمجة الديناميكية وآخر لتنعيم المنحني للوصول إلى التصميم الأمثل .ولقد تم ترجمة النهج إلى برنامج حاسب آلي

وقد تم توضيح إمكانات البرنامج من خلال دراسة عدة حالات أمدتنا بملاحظات وإرشادات جيدة عن التصميم الأمثل للأقواس. كما تم عمل دراسة ثوابت تتعلق بإيجاد نسبة الارتفاع إلى طول باع القوس المثلي، وتأثير طول باع الجسر ومقدار شدة التحميل بالإضافة إلى سماكة الصفيحة المستخدمة لصناعة العمق من خلال إجراء تطبيقات عدة علي أمثلة مختلفة. وقدمت ملاحظات عن حساسية وزن القوس للتغير في نسبة الارتفاع إلى طول باع القوس المثلي وسماكة الصفيحة المستخدمة لصناعة العمق.

**درجة الماجستير في العلـوم**

**جامعة الملك فهد للبترول والمعادن**

**الظهران، المملكة العربية السعودية**

**يناير ١٩٩٧**

# CHAPTER I

# INTRODUCTION

## 1.1 GENERAL

Increasing construction cost and limitation in available resources together with our improved understanding of material behavior and the significant improvement in computing technology have given impetus to seek optimum design either on the basis of minimum cost or minimum weight of the material, while satisfying the criteria for safety, performance and function of the structure.

Arch type structures, which are often used in bridges, dams and vaulted roofs, are frequently encountered in civil engineering works. They are known for their aesthetical lines and the larger distance they can span. Examples of such structures are, the Rainbow arch near the Niagara falls (USA) Fig.1.1, the Moldava bridge (CSSR) Fig.1.2, the Malaren bridge (Stockholm) Fig.1.3 and the Matoya bridge (Japan) Fig.1.4.

Arches derive their strength from their midsurface shapes, as well as their cross sectional properties. It is well known that the shape has a great influence on the economy. Within an upper limit and lower limit of height, an arch may take any shape.

Figure 1.1 : Rainbow arch near the Niagara falls, USA.

Figure 1.2 : Arch bridge near the Moldava reservoir, CSSR.

Figure 1.3 : Western bridge across the Malaren lake in Stockholm.

Figure 1.4 : Motoya bridge, Japan.

Parabolic arches are frequently used in practice due to their aesthetic appeal and better geometrical configuration. Moreover, parabolic shape is the optimal or near optimal shape for a wide range of loads.

On cross-sectional properties, the same cross section of the arch may be used all over, or it can be varied to achieve saving in material cost. However, too many variations of cross sections are not encouraged to avoid increase in construction cost. In some cases the shape is continuously changed in a way that makes it practical and easy to construct. It is emphasized that optimal variation of arch cross-sections along its span has not yet been investigated. The parabolic variation of cross section depth, proposed in this study, saves material, yet it is easy to construct.

Optimal design of steel arches in terms of minimum weight of material is investigated in this study. An iterative procedure is usually employed to choose the best geometry and cross-section of the arch. This is a tedious and a time consuming process. Usually, engineers will stop the iteration if a feasible solution is obtained. The arch obtained by this way is not necessarily optimum or even economical. The computer coded programs of automated search can be used to seek optimal solution. Recent developments in numerical methods, and the use of computers have made it possible to obtain a near optimum practical design for almost any engineering problem.

## 1.2 SCOPE AND OBJECTIVES

This study focuses on the optimal design of parabolic steel box arches subjected to prescribed loading. The proposed work is significant because it yields a near optimal design that satisfies all relevant provisions of the American Institute of Steel

Construction (AISC) specifications (1), practical design considerations and geometrical restrictions if any exist.

The scope of this study is to find a least-weight design of a steel arch of a parabolic shape having a box-type cross-section conforming to both structural and practical requirements. The box section is selected for its stiffness and lateral buckling resistance. It is assumed to be non-uniform in depth, having different flange areas with constant width. The scope of the work is limited to the parabolic arch shape only. The arch is assumed to have adequate lateral supports, so that premature lateral buckling is prevented.

In seeking an optimum arch design, an optimization procedure has been formulated and then coded into the computer. The total steel weight of the arch is taken as the objective function, and the constraints are related to the arch rise, allowable stresses, section properties and the combined action of axial compression and bending moments. The interaction equation for AISC specification (1) is used to determine the required section. The section is proportioned so that its optimum properties are obtained by satisfying AISC requirement of width-thickness (d/t) ratio, shear and moment requirements and the practical steel plate thickness. Mathematical methods are used to obtain a smooth profile of the web and a selected number of flange areas of the arch.

The primary objectives of this study are as follows :

1. Develop an analytical approach for optimization of a parabolic arch of non-uniform box-type cross-section, with variable depth and flange areas for a prescribed loading and support conditions. The design would be based on the

strength requirements of the AISC specifications. And the arch geometry would show a smooth profile with varying flange areas.

2. Develop a generalized computer program to find an optimum feasible design of arch, based on the proposed methodology.

3. Conduct a parametric study to observe the sensitivity of the optimum design to changes in rise, span and load.

4. Suggest some guidelines for design of arches based on findings of this study.

# CHAPTER II

# LITERATURE REVIEW

Optimization of arches, has received general attention, resulting in a number of publications. In this literature review, references on the subject of structural optimization of arches are only reviewed to focus the general state of the art.

Optimum arch design have been obtained in various ways using various optimization criteria such as zero moment, unit value of the square of the arch slope and minimum weight. Different approaches to optimum arch shape that varies from classical varitional procedure to mathematical programming are used. Fundamentals of these approaches are well covered in various books like for example [2-7].

A number of the earlier work dealing with the optimal design of arches has been reported in literature [8-11]. In most of these studies, the general arch shape is assumed a priori. Farshad (11) investigated the optimal design of arches both from the point of view of arch shape and the cross-sectional area. In all these studies, a classical varitional procedures were applied to determine the optimal arch and the resulting equations were solved exactly. Because of the analytical difficulties inherent in these studies, optimal designs are obtained for simple cases.

In 1979, Prager and Rozvany (12) derived a slightly modified version of a theorem previously developed by Michell in 1904 (13). Michell's theory formulated the basis for optimization of structural layout (14), but it was limited only to least weight trusses and least weight grillage until Prager pointed out that a very dense grillage may be regarded as an anisotropic plate. Then, he considered curved structures as a dense system of intersecting arches. Prager structure consists of intersecting arches or cables for which the shape of the middle surface and the member layout are to be optimized. Prager and Rozvany derived an optimality criteria for an archgrid, on the assumption that optimal solution for a single load condition consists of arches which are subject only to compression and no bending moments so that all cross-sections are stressed uniformly to a prescribed stress. Thus, Prager structure is a special class of Michell frame for which : (a) either the compressive or the tensile permissible stresses tend to zero, (b) the position of the loads is unspecified and to be optimized .

In a similar investigation to Ref. (12), but for "one-way system" of parallel arches, Rozvany and others (15) derived an optimality condition for fully stressed flexureless arches : a necessary condition for the minimum weight is that the mean square value of the arch slope is unity . Based on this, they obtained a closed form analytical solution for flexureless arches of uniform cross-section subjected to design-dependent loads such as self weight and weight of roof sheeting in addition to symmetrical or non-symmetrical, uniformly distributed or concentric external loads. He also carried a cost sensitivity analysis.

Shape optimization of arches under bending and axial compression briefly treated by Rozvany, Wang and Dow (16) in 1980. The most important conclusion of this paper is that, while for a single system of point loads, the optimal arch consists of straight

segments, curved geometry is the optimal solution if several alternative system of point loads are considered .

Latter, Rozvany and Wange (17) provided a full derivation of optimality criteria from both Prager layout theory, and from Michell theory. Moreover, they provided a systematic method for constructing plane Prager structure for any system of vertical load. Also they extended the method to be applicable to non parallel load system, with or without self weight, and pinned jointed supports at different levels (18).

Lipson and Haque (19) presented a numerical study of optimal design of arches of uniform depth and width but variable plate thickness, using an automated design routine. The optimization process is conducted in two levels: Y-coordinates are found first, then required section is calculated. Results show that for a wide range of uniformly distributed loads, the optimal arch shapes closely resemble the parabolic curve. As the loads deviate further and further from the uniformly distributed load cases, they tend to produce higher moments in the parabolic arch. The optimal arches become shallower in general while the arch heights at 1/4 and 3/4 span lengths tend to increase above the parabolic curve.

Wassermann (20) used eight-node isoparametric solid finite elements to model a double-curved arch dam with the positions of some key points and tangent vectors of surfaces as design parameters, and solved the optimization problem by a sequential linear programming method (2).

Ermopoulos and Ioannidis (21) presented an analytical investigation and sensitivity analysis of the optimum rise of steel arch bridges. A computer program that uses an

iteration procedure is designed to examine two cases: a constant moment of inertia case and a variable moment of inertia case using a previously defined geometric function. The most important conclusion of this paper is as follows, in all pin-ended arch bridges the value of the optimum rise to span ratio is almost constant and the total weight is not sensitive to changes in this ratio around the optimum value

Ang, Teo and Wang (22) investigated the optimal shape of plastically designed arches under a uniformly distributed load. Using the specific cost function derived by Rozvany (16), they presented a numerical procedure to obtain the optimum arch shape under bending and axial load. Because of the difficulties inherent in their procedure, the procedure is limited only to arches under uniformly distributed loads. Latter (6), Teo and Wang studied briefly the effect of equal end moments and height constraint on the optimal arch total weight.

Yao and Choi (23), developed a general three-dimensional surface design optimization model. In their work, the double-curved arch dam solved by Wassermann is treated using a higher-order finite element approximation, to obtain continuum shape design sensitivity expressions. The total volume of the arch dam is taken as the cost of the design problem, and constraints are imposed on the principal stresses on the surface of the arch dam, since the highest stresses are observed on the surfaces.

Bong and others (24) developed in 1992 a rational and practical mathematical model that uses a series of methods of mathematical programming to determine the optimum shapes of arch dams made of plain concrete and subjected to static and dynamic loads. The dam shape consists of simple smooth vertical and horizontal curves subjected to geological, topographical conditions and even to requirements of traffic. Other

constraints ,stress and stability constraints , are to satisfy design code requirement and some empirical conditions proposed by design engineers. Consequently, their model is reasonable and acceptable to design engineers. About 30 dams have been optimized by the writers, resulting in a considerable reduction in investment of resources and a higher speed of design.

Review of the literature indicates that arches are optimized based on closed form solutions and numerical methods. Although closed form solutions provide interesting insight into the optimum design of arches, they are applied without meaningful constraints on the geometric form. As a result, they yield impractical solutions. Only, a few numerical works are conducted for two dimensional arches. Neither of the closed form solutions nor numerical studies provided an investigation for the variation of optimal arch cross-section along its axis.

The present study addresses the problem designwise, by adhering to the practical design shape and design requirements. The use of proper geometrical and practical engineering constraints, the simplicity of the selected parabolic shape and the smooth variation of the depth resulted in a practical arch design.

# CHAPTER III

# GENERAL FORMULATION

## 3.1 GENERAL

Structural optimization seeks the selection of design variables to achieve its goal of optimality defined by the objective function for specified loading or environmental conditions within the limits (constraints) placed on the structural behavior, geometry, or other factors. The three basic features - **design variables, objective function and constraints** - contrive to form the design problem. The optimal design problem can be formulated mathematically as one of choosing the vector of design variables {Z} that minimize the objective function W(Z), subject to the constraints

$$g_j(Z) \geq 0 \qquad j = 1, \ldots, m \qquad (\text{ Inequality constraints }) \qquad (3.1)$$

$$g_j(Z) \geq 0 \qquad j = m+1, \ldots, k \qquad (\text{ Equality constraints }) \qquad (3.2)$$

with explicit bounds on design variables

$$Z_i^l \leq Z_i \leq Z_i^u \qquad (3.3)$$

where,

| | | |
|---|---|---|
| $Z_i^l$ | = | the lowest allowed value for the $i^{th}$ design variable, |
| $Z_i^u$ | = | the highest allowed value for the $i^{th}$ design variable, |
| k | = | the total number of constraints, |
| m | = | the total number of inequality constraints. |

Design may be defined as matching member response to required capacity at each limit state. Selection of the optimal solution from amongst any number of feasible design alternatives requires the development of a suitable merit function which reflects the desired criterion. An optimal design is then defined as a feasible solution which best satisfies the required criteria.

## 3.2 PROBLEM STATEMENT

The pertinent variables of the design problem are shown in Figure 3.1, which represent an arch of a parabolic profile. The variables L, h denote, respectively, the arch span, which is preassigned and the arch rise, whose optimum value is to be defined. The typical cross-section of the arch consists of a symmetrical box section whose dimensions are described by the variables $A_f$, d, $t_w$, as shown in Figure 3.1.

The arch cross section is assumed to be non-uniform in depth, having different flange areas and with constant width. It is made of unstiffened web, non-composite steel sections. Considering available plate thickness, a uniform web plate thickness is chosen, while the depth is allowed to vary in a prescribed geometry. Unlike the web, the flange area has to be varied in steps, for which the number of cut of points has to be preassigned. Although any number of cut of points is possible, from practical design

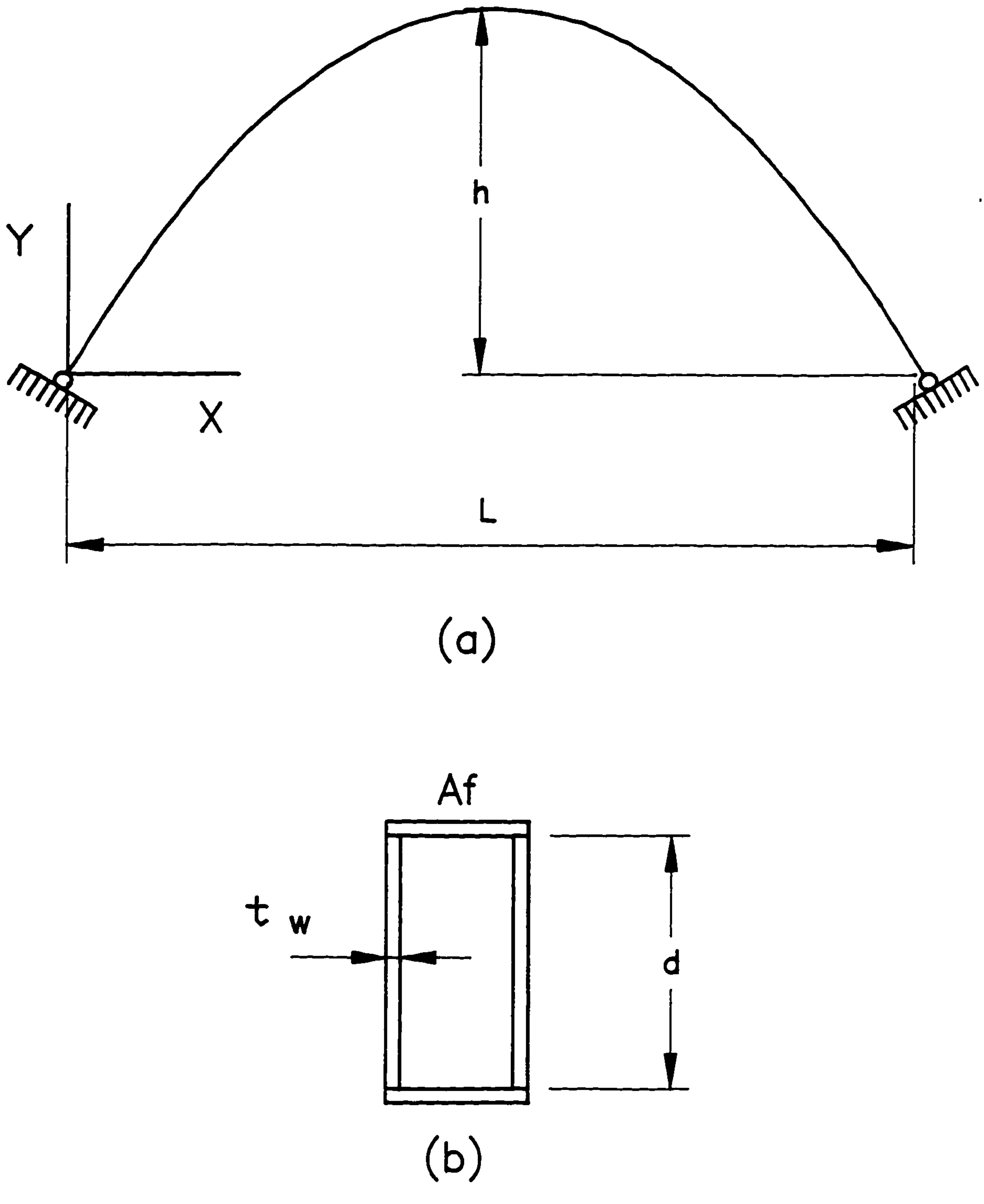



Figure 3.1 : Definition Sketch (a) Parabolic Arch Profile (b) Arch Cross-section

point of view, the number has to be limited only to a few to reduce the cost of welding.

The problem then is to find an arch shape and its cross-sectional dimensions subjected to both geometry constraints and stress constraints, so that the structure is able to carry safely the external loads and at the same time attains the minimum weight .

## 3.3 DESIGN VARIABLES

A structural system can be described by a set of parameters, some of which are viewed as variables during the optimization process. Those quantities defining a structural system that are fixed during the automated design are called preassigned parameters and they are not varied by the optimization algorithm. In the problem under investigation, the preassigned parameters represent the mechanical or physical properties of the material, the number of flange cut offs, the type of support and the total span of the arch. Those quantities that are not preassigned are called design variables. The preassigned parameters together with the design variables, will completely describe a design.

A proper optimization process requires appropriate selection of design variables. In this research the design variables considered are

(a) the arch shape or geometry and

(b) member sizes or cross-sectional dimensions.

### 3.3.1 Arch Geometry

The profile of the arch is assumed to be parabolic (Fig. 3.1). As the total span of the

arch is a preassigned value, the only geometric design variable is the arch rise, h. The equation of the arch can be defined as

$$Y = \frac{4h}{L^2}(X)(L - X) \tag{3.4}$$

where :

| | | |
|---|---|---|
| X | = | horizontal coordinates of the arch, |
| Y | = | vertical coordinates of the arch. |

### 3.3.2 Cross-sectional Dimensions

The typical section of the arch is shown in Figure 3.1. The typical cross-section of the arch is taken as a symmetrical box-section. The three design variables are the web plate thickness, the web depth and the flange area are as indicated in Figure 3.1.

**(a) Web Plate Thickness:** A uniform web plate thickness, $t_w$, is used through the whole arch. Among all available plate thickness, the optimum thickness of the web plate has to be chosen which would yield least weight design .

**(b) Web Depth:** The depth, d, varies in a geometry, prescribed as a parabolic form, to provide a smooth, practical profile. The optimum profile of the web has to be determined which would yield least weight design .

**(c) Flange Area:** As the flange area is a discrete variable, it has to be varied in steps, for which the number of cut of points has to be preassigned. In practical design , a few cut of points are often preferred, as too many cut off points are not cost-effective.

## 3.4 OBJECTIVE FUNCTION

Generally, a design objective function is the function whose least, or greatest, value is sought. It constitutes a basis for the selection of one of several alternative acceptable designs. In this work, total weight of the arch is taken as the objective function. It's required to find the minimum weight of the arch that satisfies design constraints.

For mathematical formulation, the arch is discretized into n-number of nodal points. as shown in Figure 3.2. For convenience, a segment is considered as a straight line and the objective function becomes

$$W = \sum_{i=1}^{n} \rho A_i L_i \qquad (3.5)$$

where $A_i$ is the cross-sectional area at node i. $A_i$ is given as

$$A_i = 2(d_i t_w + A_{fi}) \qquad (3.6)$$

where,

| | | |
|---|---|---|
| $A_{fi}$ | = | flange area at node i, |
| $d_i$ | = | depth at node i, |
| $t_w$ | = | web plate thickness. |
| $L_i$ | = | length of element i between nodes (i+1) and i, |
| n | = | number of nodes. |
| $\rho$ | = | unit weight of steel. |

Thus W is a function of the four design variables h, $t_w$, $d_i$ and $A_{fi}$. That is

$$W = f(h, t_w, d_i, A_{fi}) \qquad (3.7)$$

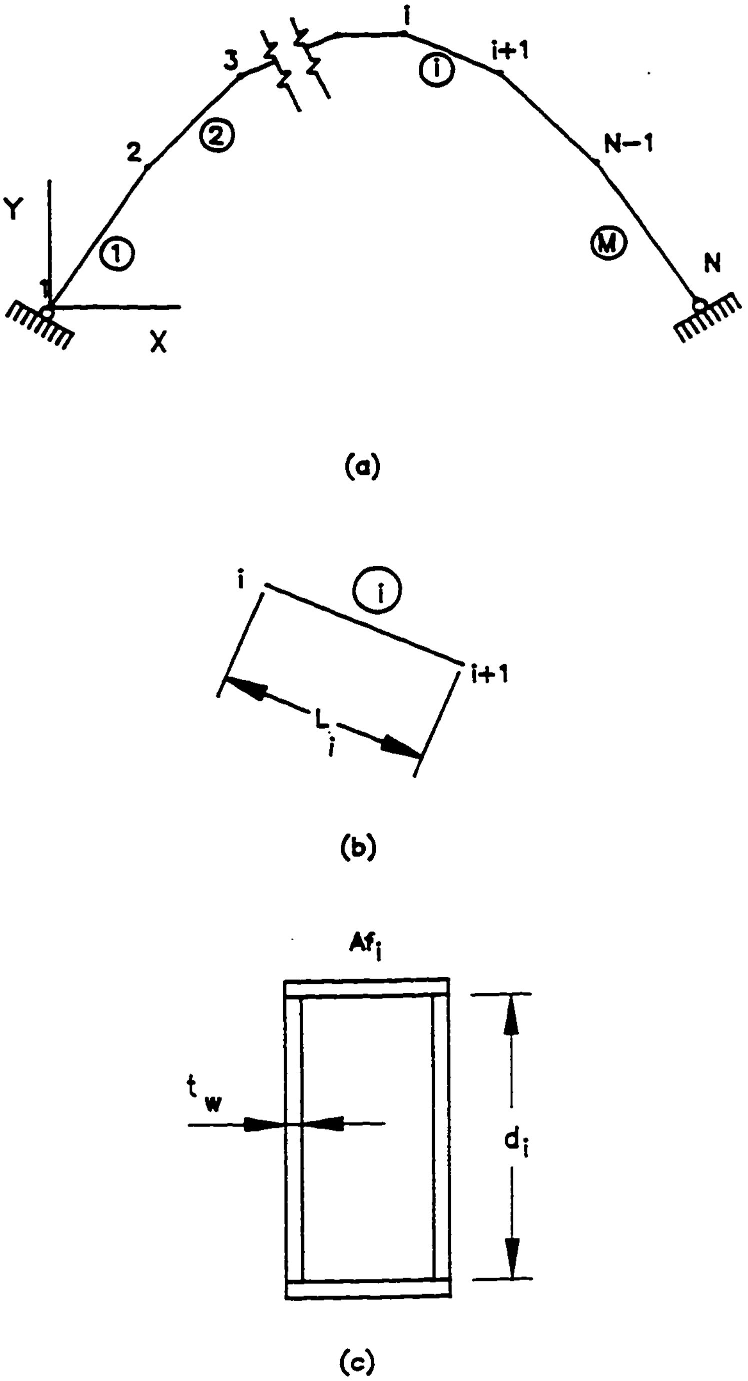



Figure 3.2 : Arch Discretization into n-number of points. (b) Typical Element (c) Typical Element Cross-section

## 3.5 CONSTRAINTS

Any set of permissible values for the design variables represents a design. Clearly, some designs are useful solutions to the optimization problem, but others may be inadequate in terms of function, strength, or other considerations. If a design meets all requirements placed on it, it will be called a feasible design. The restrictions that must be satisfied in order to produce a feasible design are called constraints.

In this study, the constraints are of types: geometry constraints and stress constraints. Deflection constraints have not been considered, although they can be included. It is assumed that as deflections of arch structures are relatively small, deflection conditions are normally satisfied by feasible designs that satisfy stress and geometry constraints.

### 3.5.1 Geometry Constraints

Meaningful restrictions in the geometry are imposed to satisfy the bounds of a practical or an acceptable design. It includes two categories.

**(a) Constraints on Rise:**

$$g_1 = h - h^u \leq 0 \tag{3.8}$$

$$g_2 = h^l - h \leq 0 \tag{3.9}$$

Where $h^u$ represent an upper bound on the height, and $h^l$ represent the lower bound. These restrictions are imposed from practical considerations which generally include requirements of clearance, aesthetic and construction.

**(b) Constraints on Cross-sectional Dimensions:**

$$g_3 = d^l - d \leq 0 \tag{3.10}$$

$$g_4 = A_f^l - A_f \leq 0 \tag{3.11}$$

$$g_5 = t_w - t_w^u \leq 0 \tag{3.12}$$

$$g_6 = t_w^l - t_w \leq 0 \tag{3.13}$$

Where $d^l$, $A_f^l$ and $t_w^l$ represent the lower bound on the depth, the area of the flange plate and the thickness of the web plate and $t_w^u$ represents the upper bound on the thickness of the web plate. These constraints on the minimum and maximum dimensions are generally ascertained from practical considerations.

**(d) Local Buckling Constraint on d/t Ratio :** Unstiffened web section is used in this study. The aspect ratio limit is as follows :

$$g_r = \frac{d}{t_w} - 100 \leq 0 \tag{3.14}$$

### 3.5.2 Stress Constraints

The allowable stress design method under service load is used in this study. Although the allowable stresses used in this work are based on AISC -Allowable Stress Design code (1), any other specification can be easily adopted. As a section is subjected to the combined action of compressive normal force N, bending moment M and shear force V, the combined normal stress from N and M and the shear stress from V must be calculated at a section and checked with the allowable stresses prescribed by the code.

**(a) Combined Normal Stress Constraint :** AISC interaction equation, to satisfy the full stress design criterion for a given arch, is

$$\frac{f_a}{F_a} + \frac{f_b}{F_b} = 1.0 \tag{3.15}$$

where,

$f_a$ = computed axial stress $= \dfrac{N_i}{A_i}$,

$f_b$ = computed compressive bending stress $= \dfrac{M_i}{S_i}$,

$F_a$ = axial compressive stress that would be permitted if axial force alone existed,

$F_b$ = compressive bending stress that would be permitted if bending moment alone exist,

$N_i$ = axial force at node i,

$M_i$ = bending moment at node i,

$S_i$ = elastic section modulus about the major axis at node i.

The section is proportioned on the basis of a non compact section. Thus

$$\frac{F_b}{0.60F_y} \leq 1 \tag{3.16}$$

in which $F_y$ = yield stress of steel used.

It is assumed that lateral buckling is prevented by providing lateral bracing and $F_a$ is calculated from rational methods.

**(b) Shear Stress Constraint :** The AISC-ASD specifications define

$$C_v = \frac{240300}{(d/t_w)^2 F_y} \qquad \text{if } C_v \le 0.8 \tag{3.17}$$

$$C_v = \frac{190}{(d/t_w)}\sqrt{\frac{5.34}{F_y}} \qquad \text{if } C_v > 0.8 \tag{3.18}$$

the shear stress constraint is

$$g_v = f_v - \frac{C_v}{2.89} F_y \le 0 \qquad \text{if } \frac{C_v}{2.89} < 0.4 \tag{3.19}$$

$$g_v = f_v - 0.4\, F_y \le 0 \qquad \text{if } \frac{C_v}{2.89} \ge 0.4 \tag{3.20}$$

where,

$F_v$ = computed shear stress $= \dfrac{V_i}{d_i t_w}$,

$V_i$ = shear force at node i.

## 3.6 ANALYSIS

The arch is a structural system in which the primary internal force is axial compression. Most arches do develop some bending in addition to the compression. However, the bending moments that a given set of loads produces in an arch are much smaller than those produced in a beam of the same length. As a sequence, arches are able to span much larger distances than beams.

Unlike a beam, which develops only vertical reactions at its support, the arch gives rise to both horizontal thrusts and vertical forces at its supports. In fact, it is the horizontal

reactions that are responsible for the relatively small bending moments in an arch. Thus, the moment at midspan for the beam in Fig. 3.3 is equal 500 KN.m. By comparison, the moment at the middle of the arch is only 158 KN.m.

Structural analysis is an essential part in any optimal design formulation and solution. The basic objective of frame analysis is to determine the bending moments, shear, and thrusts, at each section. The design of the arch is based on these analysis results.

Furthermore, in this work, the analysis must be repeated many times during optimization process. This require the use of a simple representative model.

In this study, finite element method is used to conduct a linear elastic analysis of the arch. For the purpose of analysis, the structure is discretized as an assembly of straight beam elements joining the nodes. Internal forces are then calculated at each element.

## 3.7 PROPORTIONING OF OPTIMUM BOX-SECTION

At a typical cross-section, the total area is simply given as

$$A = 2(dt_w + A_f) \tag{3.21}$$

Since the thickness of the flange is small compared with d, the elastic section modulus about the major axis, S, can be written approximately as

$$S = \frac{t_w d^2}{3} + A_f\, d \tag{3.22}$$

Substitution of the value of $A_f$ from Eq. (3.21) in Eq. (3.22), the elastic section

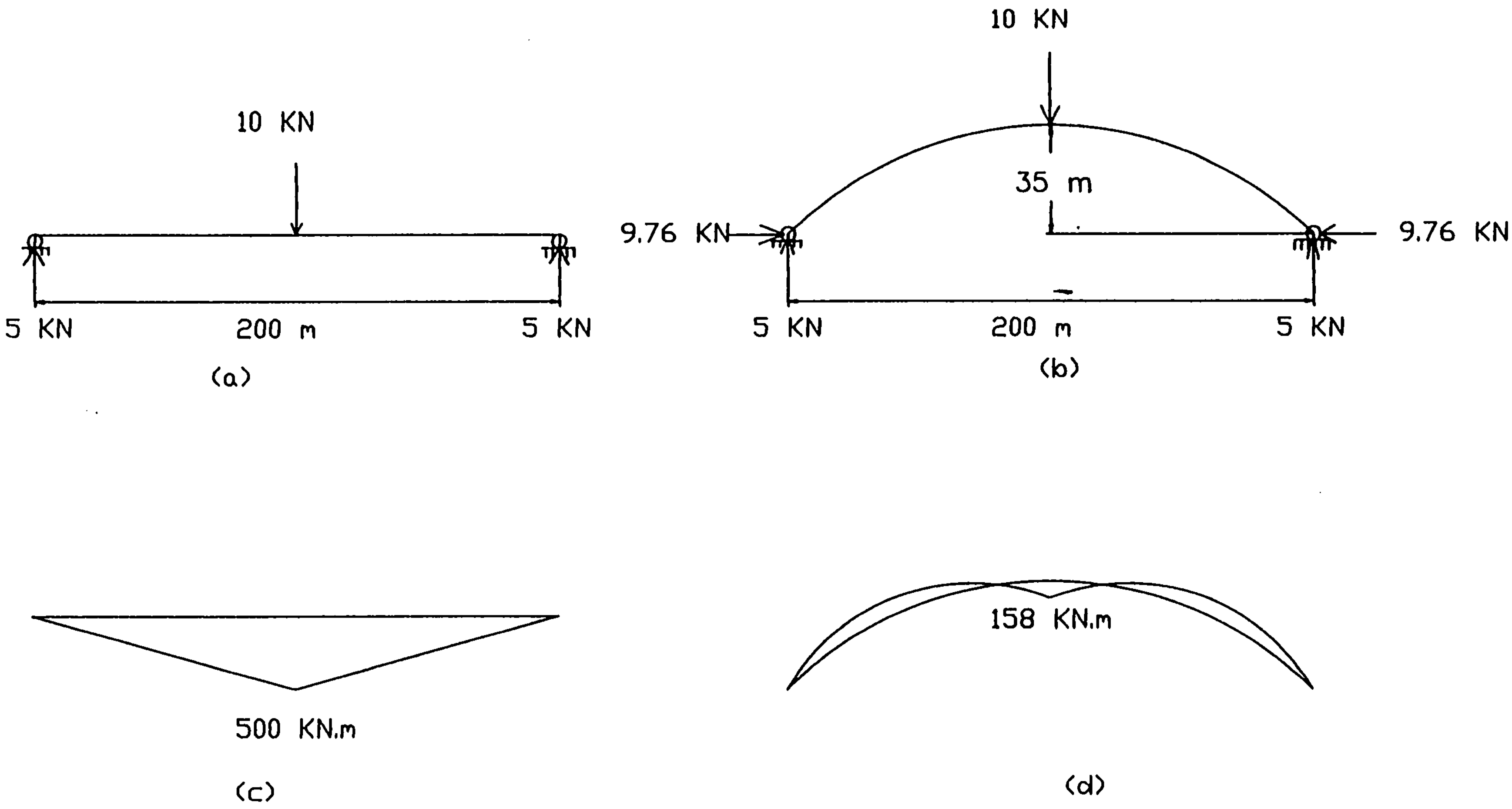



Figure 3.3: Comparing an arch system with a beam of equivalent span and subjected to the same load

(a) a beam example (b) an equivalent span arch (c) the beam moment diagram (d) the arch moment diagram.

modulus may be rewritten as

$$S = \frac{Ad}{2} - \frac{2}{3} t_w d^2 \tag{3.23}$$

For a given area, the section modules is maximized by setting

$$\frac{dS}{dd} = 0 \tag{3.24}$$

which leads to optimum depth $d_{opt}$ for a given area A as

$$d_{opt} = \frac{3A}{8t_w} \tag{3.25}$$

the area in terms of optimum d is

$$A = \frac{8}{3} t_w d \tag{3.26}$$

and the corresponding

$$A_f = \frac{1}{8} A \tag{3.27}$$

Thus the optimum section modulus is given as

$$S = \left(\frac{d}{4}\right) A \tag{3.28}$$

## 3.8 CALCULATION OF REQUIRED OPTIMUM SECTION

If M is the design moment and P is the design axial force then the AISC interaction equation (3.15) can be written as

$$\frac{P}{A\,F_a} + \frac{M}{S\,F_b} = 1.0 \qquad (3.29)$$

Substituting Eq. (3.28) in Eq. (3.29) and multiplying both sides by (Ad) yields

$$Ad = \frac{Pd}{F_a} + \frac{4M}{F_b} \qquad (3.30)$$

substituting the value of A from equation (3.25) in Eq. (3.30.) yields

$$(\frac{8t_w}{3})d_{opt}^{\,2} - (\frac{P}{F_a})d_{opt} - (\frac{4M}{F_b}) = 0 \qquad (3.31)$$

Solving

$$d_{opt} = \frac{-b + \sqrt{b^2 - 4a\,c}}{2a} \qquad (3.32)$$

where,

$$a = \frac{8t_w}{3}$$

$$b = -\frac{P}{F_a}$$

$$c = -\frac{4M}{F_b}$$

## 3.9 PROBLEM FORMULATION

The aforementioned problem can be formulated as follows : find a design vector

$$W = (w_i) = (\,h\;, t_w\,, d_i\,, A_{fi}\,) \qquad i=1, 2, \ldots n \qquad (3.33)$$

subject to the geometrical constraints of Eqns. 3.8 - 3.14 and the stress constraints of Eqns. 3.15 - 3.20 , so that the objective function

$$W=\sum_{i=1}^{n} \rho A_i L_i$$

where,

$$Ai = 2(d_i t_w + A_{fi})$$

attains minimum value within a specified design space.

# CHAPTER IV

# METHOD OF SOLUTION & COMPUTER PROGRAM

## 4.1 GENERAL

In this chapter, the method of solution for the problem formulated in Chapter-3 is presented, and a computer code for optimization of the problem under investigation in this research called **POPSAR** (**P**rogram for **O**ptimization of **P**arabolic **S**teel **AR**ches) is introduced. It can handle parabolic box-shaped steel arches subjected to prescribed loading, and can be extended to solve general shape arches .

The problem begins with an initial feasible point in the design space, using upper limit of height and maximum thickness of web plate. The resulting formulation of the problem and the method of solution under investigation are employed to obtain the optimum solution for the height and the thickness of web plate, and to calculate the corresponding merit function. The merit function will be checked for convergence. If convergence is satisfied within the specified tolerances, then the optimum design is reached, otherwise another design cycle will be necessary and so on till convergence is being satisfied. The method of solution is described in the following sections.

## 4.2 STRUCTURAL ANALYSIS

An essential part in any optimum structural design system is to prepare the analysis module which predicts and demonstrates the behavior and response of the structure under the functional requirements. This module analyzes the structure for the initial value of height and thickness of web plate, and generates all the necessary information that will be used in cost function and constraints.

For this purpose, a general frame analysis routine called **ANASYS** is used for linear elastic analysis of the arch. The program was first varied by comparing results with STAAD III package. Sample results from this program are shown in Appendix A to show that the program is fairly accurate.

For the initial value of height and thickness of web plate , the arch is idealized as (n) straight elements, linear analysis is conducted and the member end forces for each element are calculated. The worst values of axial load, bending moment and shear force, $\{ P_i, M_i, V_i \}$ , will be placed in a matrix named Force.

Now, the required data for optimum section calculations is obtained, and the program is ready to start the next step: determination of optimum depth.

## 4.3 DETERMINATION OF DEPTH

Formulas derived in Chapter 3 are now used to find optimal arch cross-sectional dimensions ( $d_i$ , $A_{fi}$ ) , subjected to geometrical and stress constraints, so that the arch attains its minimum weight for a specified height. **SODESH**, a subroutine for

Optimum Design for a Specified Height and thickness of web plate, is employed as an optimizer to determine an optimal solution for the design problem of a specified merit function and constraints.

Optimum depth is determined, for each node, by using Eq. 3.32, derived in chapter 3, with the element forces and initial design data. Cross-sections are proportioned in accordance to Eqns. 3.25, 3.27 and 3.21. Optimum flange area, optimum section area, and optimum section modulus are calculated.

Next, geometrical constraints of Eqns. 3.8 - 3.14 and stress constraints of Eqns. 3.15 - 3.20 are checked. If required, the optimum depth will be modified to incorporate any of these constraints, and hence the flange area.

The outcome of these steps is in the form of discrete data points. The theoretical optimum arch for a given height and thickness of web plate may be obtained by connecting these points together, but this leads to an irregular depth profile which is an impractical design.

## 4.4 SMOOTHENING OF WEB PROFILE :

Instead of requiring the depth profile to pass by all calculated points, the chosen profile should fit the data as closely as possible. In other words, it's required to determine a function y(x) for the depth profile that fits the data points by a smooth curve, or profile for (d/2) above and below the arch mid-surface. Figure 4.1 shows an example of smooth curve profile in comparison with the calculated one ; the solid lines and the dashed lines represent a polynomial curve fitting and the calculated (d/2) respectively.

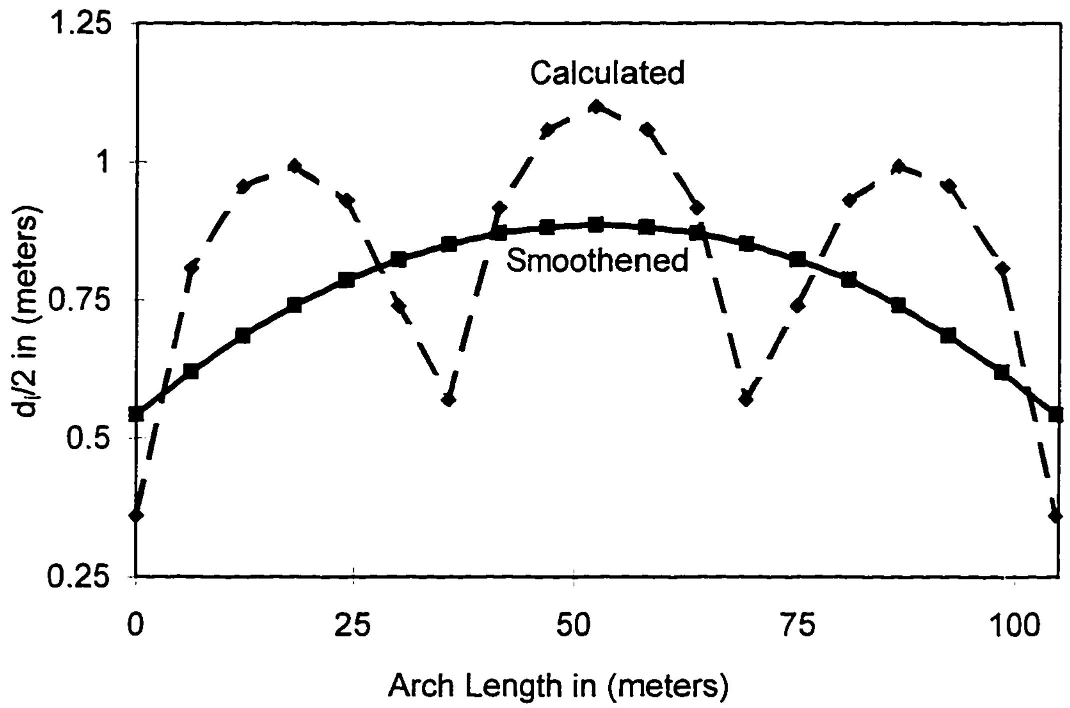



Figure 4.1 : Plot of Smooth and Calculated Depth Profile versus Arch Length.

**4.4.1 The Polynomial Fitting :**

In polynomial fitting, the paired points $(x_i,f(x_i))$ resulted from the previous step are approximated by the function $(x_i,y_i)$ where $x_i$ = arch length at node i, $f(x_i)$ = calculated $d_i/2$ at node i and $y_i$ smoothened $d_i/2$ as per the following function :

$$y_i = a + bx_i + cx_i^2 \tag{4.1}$$

It is required to find the best values for a, b, and c, so that the approximating polynomial gives the best possible trend of the data points with least error.

The least square principle states that for the best fit, the sum of the squares of the deviations, SS, of the data points from the corresponding model points should be a minimum. The sum of the squares of the deviations can be expressed as

$$SS = \sum_{i=1}^{n}[y_i - f(x_i)]^2 \tag{4.2}$$

$$SS = \sum_{i=1}^{n}\left[\left(a + bx_i + cx_i^2\right) - f(x_i)\right]^2 \tag{4.3}$$

Differentiating SS with respect to a, b and c yields

$$\begin{bmatrix} n & \sum x_i & \sum x_i^2 \\ \sum x_i & \sum x_i^2 & \sum x_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \end{bmatrix}\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \sum f(x_i) \\ \sum x_i f(x_i) \\ \sum x_i^2 f(x_i) \end{bmatrix} \tag{4.4}$$

Solving, the values of a, b and c that give the best approximation of Eq. 4.1 are obtained. Thus, the smooth depth can be formulated as

$$(d/2)_{smoothen} = a + bS + cS^2 \tag{4.5}$$

Figure 4.2 shows an example of web profile smoothened using Eq. 4.5. It is observed from the figure that the polynomial fitting provides a smooth and feasible profile to construct for (d/2). Polynomial fitting is further illustrated through a detailed example in Appendix B.

**4.4.2 Modifying the Polynomial Fitting :**

After smoothening the web profile, the flange areas are recalculated, corresponding to the new value of $d_i$ and the geometry and stress constraints are rechecked. As shown in Appendix B, the smoothening of the depth profile results in a slight increase in the total weight of the arch, due to the following two reasons : (i) the addition of the flange areas at some points as per the constraint $g_4$, Eq. 3.11, although they are not required to satisfy AISC interaction equation and (ii) the departure from the optimum depth of box-section derived in Chapter 3.

As shown later in Chapter 5, the weight function is insensitive to the proportioning of the section in the vicinity of the global optimum. Thus, the increase in the weight function is mainly due to the addition of flange areas not theoretically needed at some points. The slight lowering of the smooth depth profile helps to minimize this addition of flange areas and hence the merit function.

After modifying the smooth depth profile, the flange areas are again recalculated, and the geometry and stress constraints are rechecked.

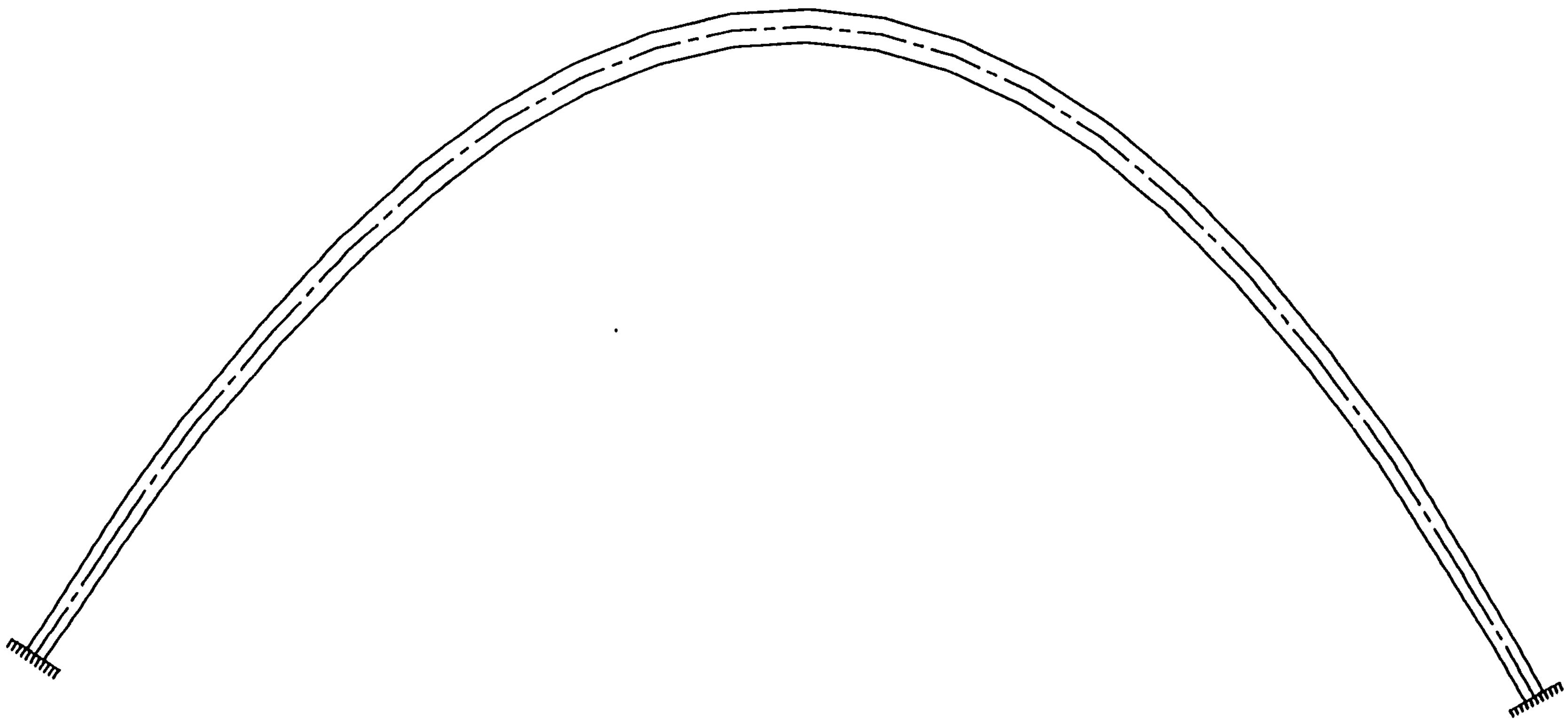


Figure 4.2 : An Example of Polynomial fitting of Web Profile for a Parabolic Arch.

## 4.5 CURTAILMENT OF FLANGE PLATES USING DYNAMIC PROGRAM :

As shown in Fig. 4.3, the calculated flange areas in the previous step are discrete values at elements. That is for each element, the flange plate is to be cut off and welded to another one with a different area, which is impractical. For a practical design, the number of cut off points has to be limited to only a few. Thus, it is required to divide the total arch length, S, into m segments or stages, each with independent flange area. An example of flange curtailment in which the arch length is divided into three segments $S_1$, $S_2$ and $S_3$ each with uniform flange area is shown in Fig. 4.4.

As the curtailment of flange plates is the last step in the optimization process. The flange area in any segment shall fulfills the design requirements at this stage. Thus, the maximum flange area governs at each segment. For a preassigned number of flange cut offs, this losses of flange areas can be minimized by selecting the optimum subdivision of arch length. Thus, the objective is to determine the optimum subdivision of S that will minimize the total flanges weight.

$$W_f = \lambda \sum_{k=1}^{m} A_{fk} \cdot S_k \tag{4.6}$$

where,

| | | |
|---|---|---|
| $W_f$ | = | the total flange weight, |
| $\lambda$ | = | unit weight of steel, |
| $S_k$ | = | the total length for stage k, |
| $A_{fk}$ | = | the flange area at stage k, |
| m | = | the preassigned number of flange plates. |

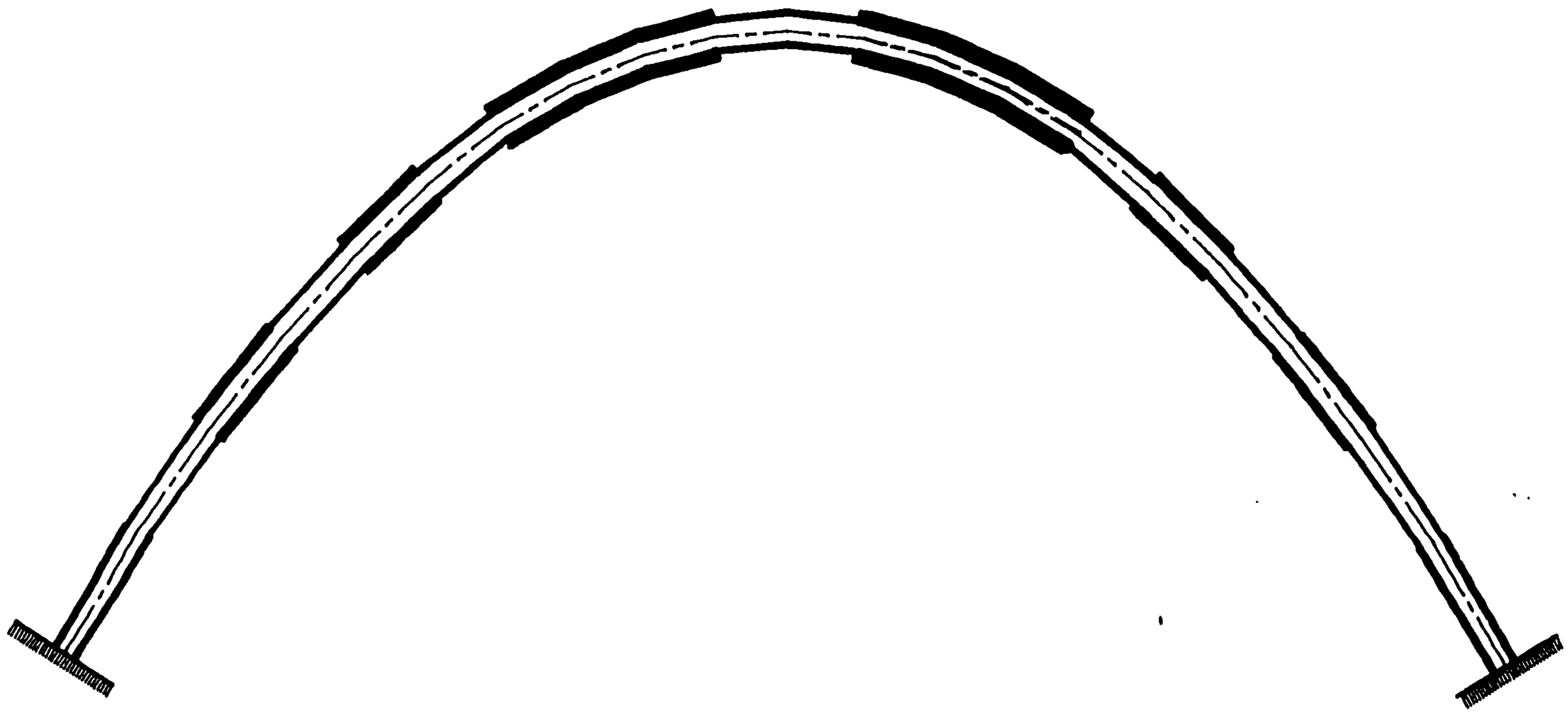


Figure 4.3 : An Example of the Flange Plates As Calculated.

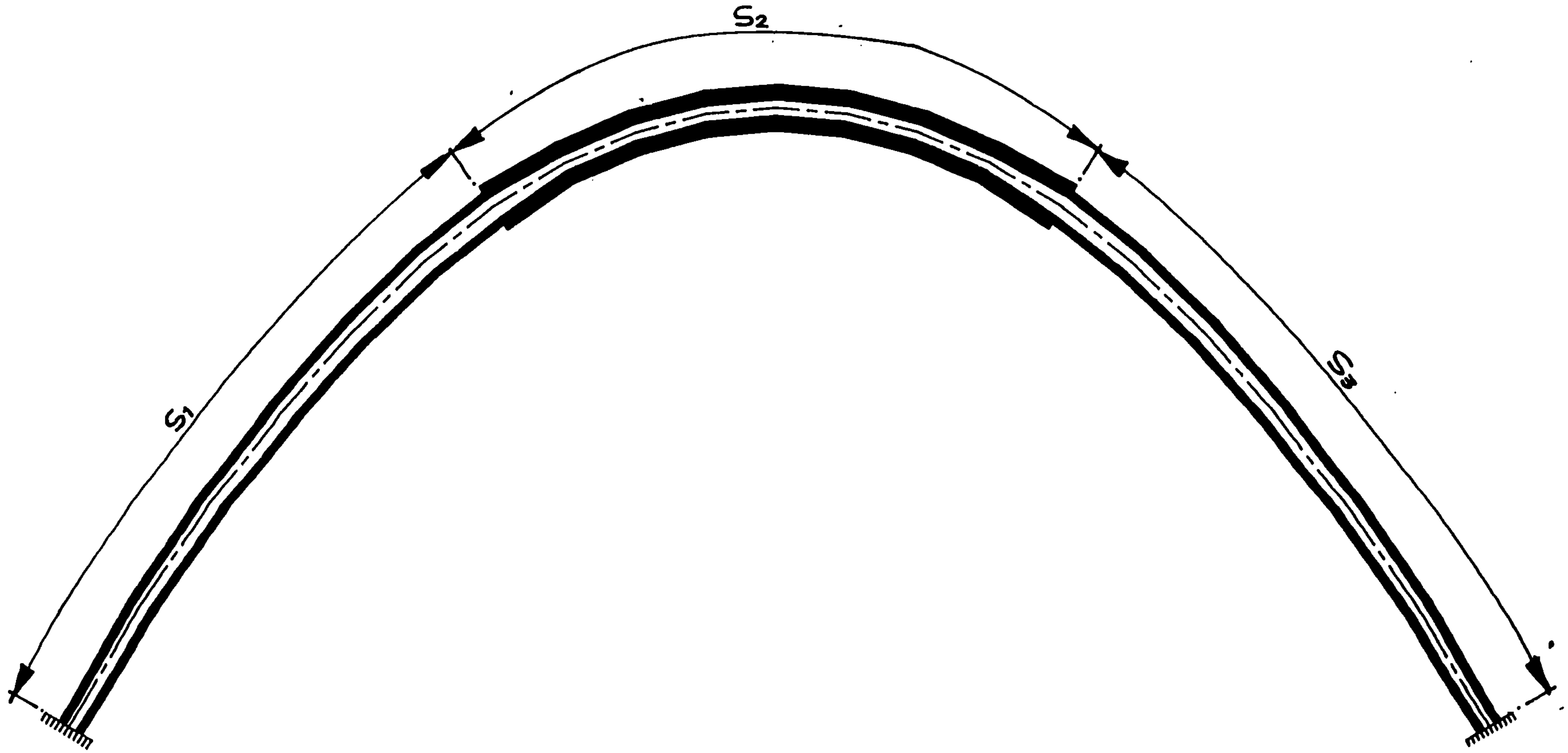


Figure 4.4 : An Example showing Curtailment of Flange Plates Using Dynamic Programming.

A Dynamic Programming procedure is developed to solve this problem in stages An example explaining Dynamic Programming is provided in Appendix- C.

Defining flange area and the smooth depth profile, a step for the practical optimum design of arch for a given rise and thickness of web plate is completed. The next step is to iterate for optimum rise. Finally, the program iterates for optimum thickness of web plate. Obtaining the optimum rise and the optimum thickness of web plate, the optimization process is completed.

### 4.6 FLOW CHART

The method of solution described above is coded into a computer program. The following is the step by step breakdown of the program:

1. The input data includes the span of arch, limits of the rise , loads, material properties, allowable yield and compression strength, limits of web plate thickness, preassigned number of flange cut offs, and the number of elements.

2. An initial geometry is assumed using the specified maximum rise, the maximum thickness of web plate and a hypothetical uniform cross section for the entire arch.

3. For the assumed geometry, the structure is analyzed for the given loading, and the maximum values of bending moment, axial compressive force and shear force are calculated at nodal points.

4. For the computed forces, optimum cross sections are obtained at each node , using

Eqns. 3..33, 3.21, 3.26 and 3.27 and constraints of Eqns 3.8-3.20 are checked.

5. The web profile is smoothed using *polynomial curve fitting* .

6. Flange areas, and total weight are then calculated for the smoothed depth profile

7. Smoothed depth profile is modified to minimize the increase in the arch weight due to curve fitting.

8. For the recalculated flanges, *dynamic programming* is used to determine the optimum curtailment of flange plates corresponding to the chosen cut off number. Now the design is completed.

9. The structure is then reanalyzed and steps (3-8) are repeated to seek convergence in the design. The final design represents the optimum design corresponding to the chosen rise of the arch and thickness of the web plate.

10. The optimum rise is searched iteratively by sequentially reducing the value of h, and repeating the above steps until the least weight of the arch, corresponding to the chosen thickness of the web plate, is obtained.

11. The optimum thickness of the web plate is searched iteratively by sequentially reducing the value of $t_w$, and repeating the above steps until the least weight of the arch is obtained.

12. Results, optimum rise, smoothed depth profile and flange areas cut offs for the

optimum rise are written in the output file.

The flow chart of the computer program **POPSAR** is depicted in Fig. 4.5 and Fig.4.6. The program consists of a main subroutine called **SOPSAR** that iteratively searches the optimum arch rise and web plate thickness. **SOPSAR** uses two subroutines : (i) **Geom**, a subroutine that generates all the necessary geometric data and (ii) **SODESH,** a root subroutine that determines the local optimum design corresponding to a specified rise and a chosen web plate thickness.

**SODESH** employs four subroutines : (i) **ANASYS**, a general analysis subroutine, (ii) **OPTIMM**, a subroutine that computes the optimum cross-section ,(iii) **SMOOTH**, a subroutine that smoothen the web profile and (iv) **DYNAMIC**, a subroutine that determine the flanges cut offs.

Program listing, typical input file, and typical main output file are listed in Appendixes D, E and F .

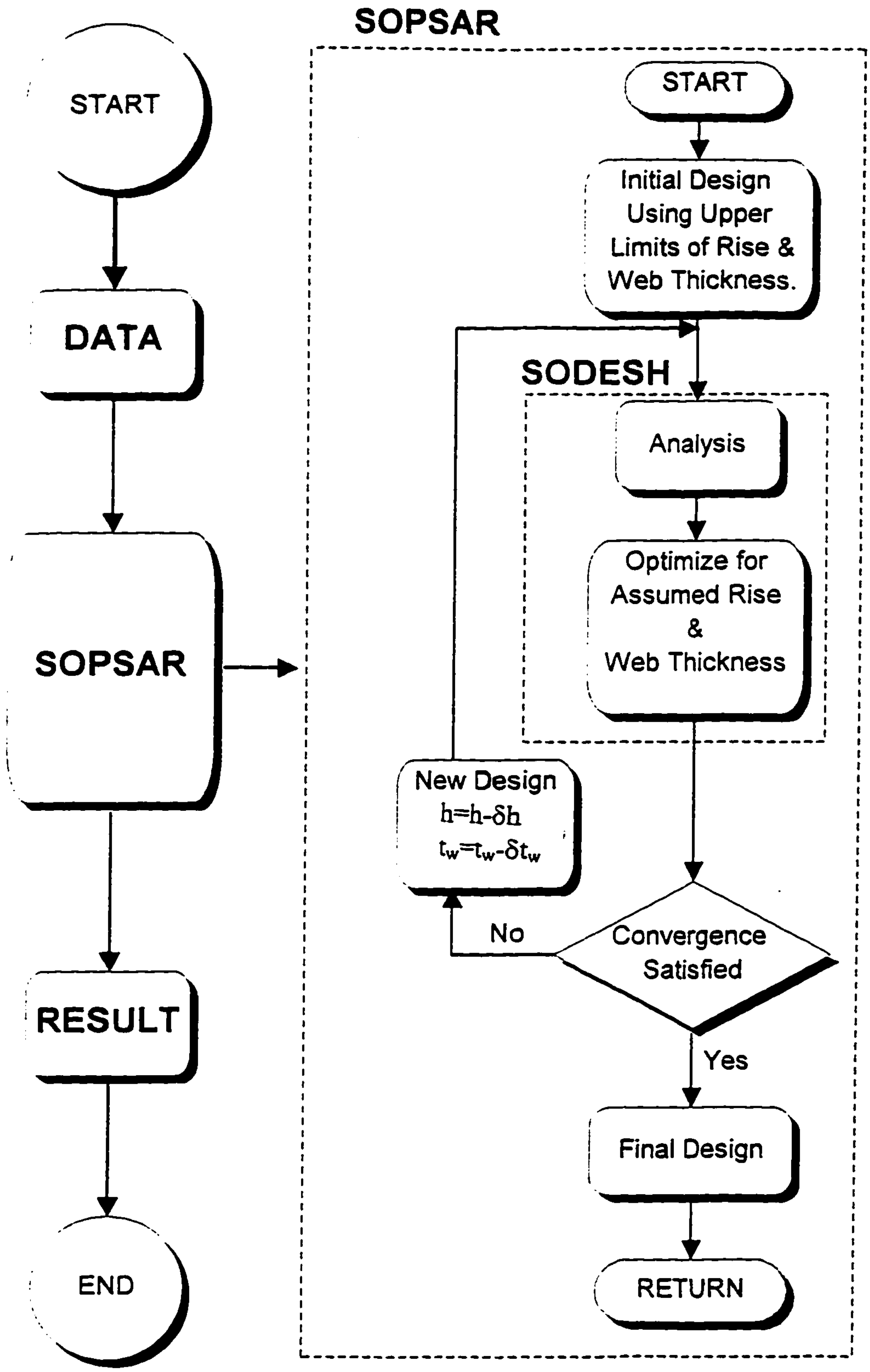



Figure 4.5 : Flow Chart of **POPSAR**

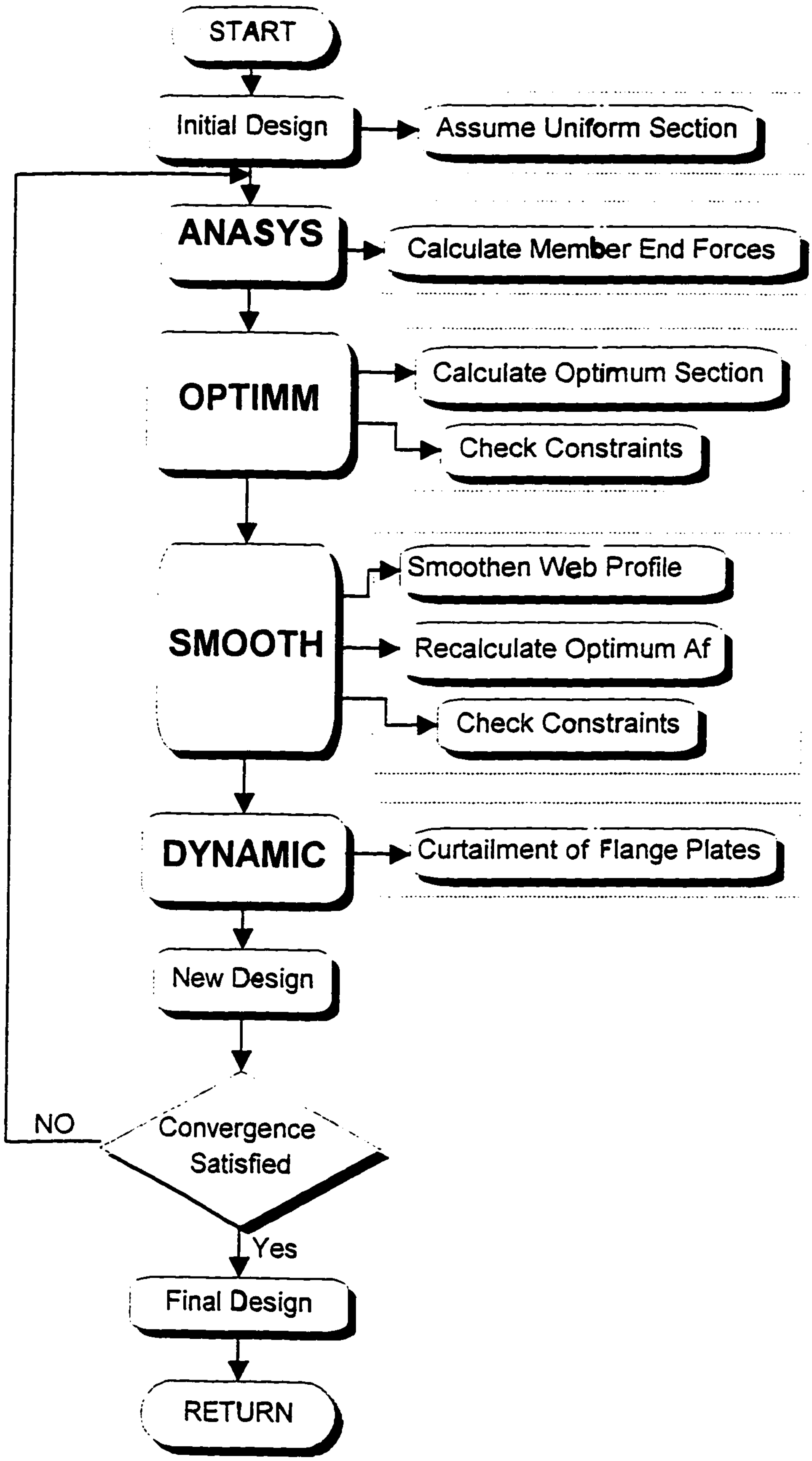



Figure 4.6 : Flow Chart of **SODESH**

# CHAPTER V

# APPLICATIONS, RESULTS AND DISCUSSIONS

## 5.1 GENERAL

The capabilities of POPSAR as developed in this study are illustrated through different examples. The cases selected provide information about the trend of the cost function, the optimum arch rise and cross-sections and the influence of important design variables on the optimal solution. The geometry of these example arches and the loading are chosen at random.

Four cases are considered with one case having a uniformly distributed load of full length which is used to ascertain the reliability and performance of the present computational scheme in optimizing the design of an arch for which the optimal solution is obtained analytically. Two cases having uniformly distributed loads of partial lengths are used to study the effect of the design variables on the applied design. The last case is an arch subjected to unsymmetrical loading which is used to show the applicability of the computer program to readily solve various design problems.

## 5.2 EXAMPLES

### CASE 1 : a Hinged Arch Subjected to a Uniformly Distributed Load of Full Length:

The following data is applicable to a hinged arch, subjected to a uniformly distributed load of full length as shown in Fig. 5.1, whose

span, L = 100 meter.

The rise will be conforming to the limits : $10 \leq h \leq 100$ .

Uniform load intensity, w = 100 KN/m .

Preassigned number of flange plates, m = 3 .

Limits for web plate thickness : $15 \leq t_w \leq 25$ mm .

Material Properties :

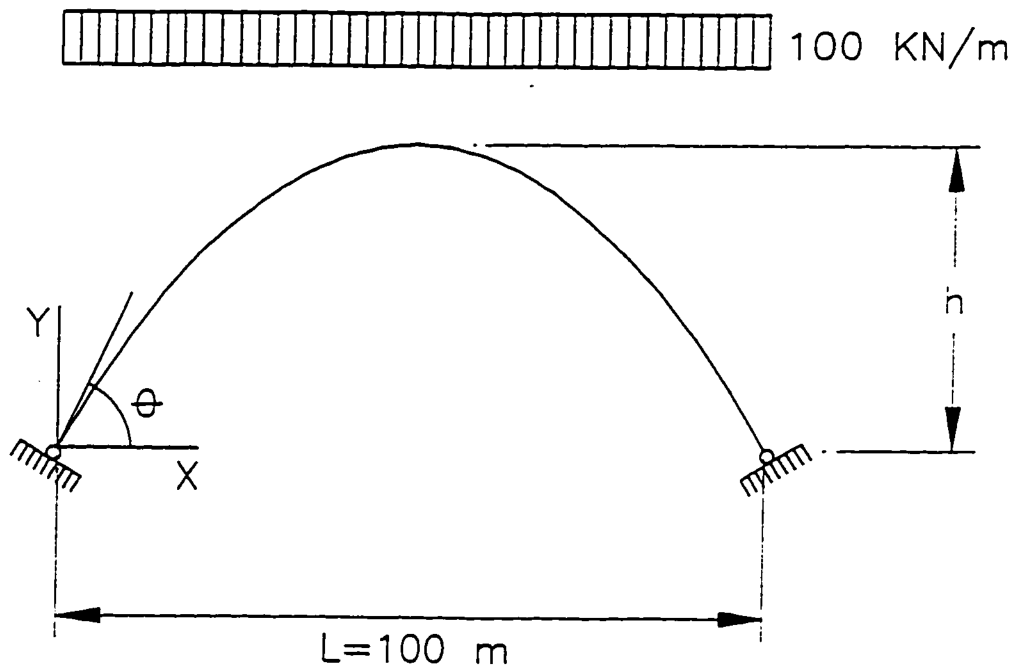
Modulus of elasticity of steel , E = 200 GPa .

Steel yield strength , $F_y$ = 248 Mpa .

Allowable compression strength, $F_a$ = 100 Mpa .

The optimal arch shapes are obtained using the present automated design routine developed earlier in this study . In this case, the arch is represented by 20 straight members. The program starts with an initial design using the upper limit of rise, the maximum thickness of web plate and a uniform cross section.

The optimum arch rise is searched iteratively by sequentially reducing the value of h. The variation of the merit function which is the weight of the parabolic arch, with the ratio h/L is plotted in Figure 5.2, by nondimensionalizing the merit function as




5.1 : Case 1, a Hinged Arch Subjected to a Uniformly Distributed Load of Full Length.

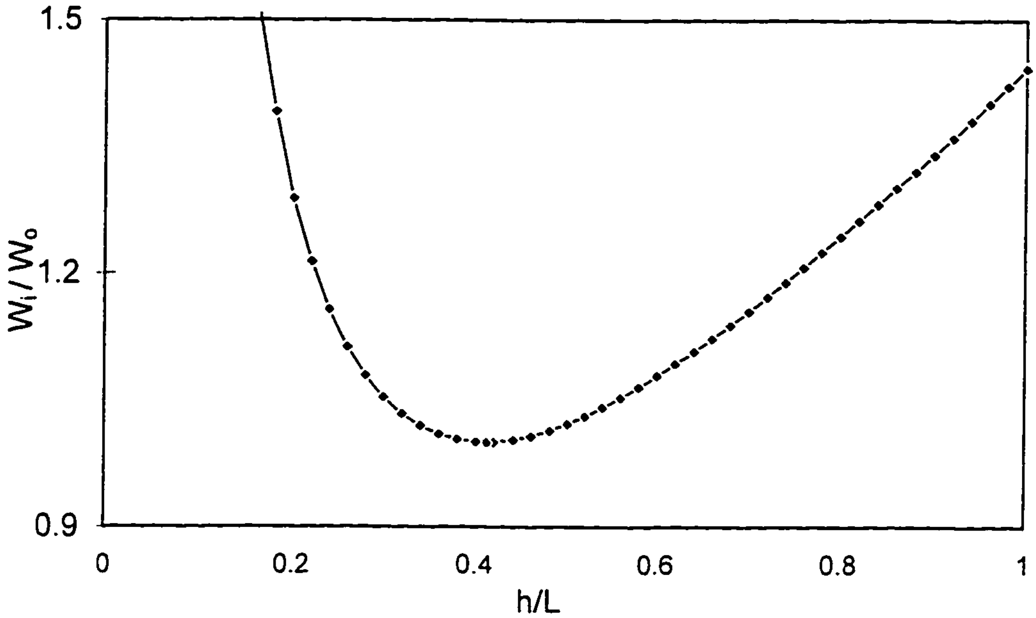



Figure 5.2 : Plot of Total weight Versus h/L ratio for Case 1.

$W_i/W_o$ , where $W_i$ is the weight of the arch at $(h/L)_i$ at the ith step of iteration and $W_o$ is the optimum weight at the optimum h/L ratio. The plot shows that the minimum weight of the parabolic arch occurs at a ratio h/L = 0.41, below and above which the weight increases.

This can be explained as follows. The arch in this case is subjected to pure compression only. Thus the weight function is a function of two variables the total arch length and the cross-sectional areas which is a function of axial compressive stresses. As the h/L ratio increases, the total arch length increases and the axial stresses decreases. On the other side, as the h/L ratio decreases, the total arch length decreases and the axial stresses increases. At a ratio h/L = 0.41, the optimum combination of arch length and cross-sectional area exists.

Comparing the resulting ratio of h/L with the previous analytical work of Rozvany and Prager (12) who derived an optimum h/L ratio = 0.433 for parabolic arches, it is evidently clear that the numerical method developed in this research demonstrated reliable result conforming to the almost exact solution.

The plot also shows that any parabolic arch shape having h/L ratio ranges between 0.34 and 0.50 can at most be 2% heavier than the optimum arch. Thus, for an economical design of an arch, subjected to a uniformly distributed loads of full length, a ratio of h/L between 0.34 and 0.48 can be chosen.

The required depth profile from Eq. 3-32 is shown in Figure 5.3 by nondimensionalizing the depth as $d_i/d_c$ , where $d_i$ is the depth at node i and $d_c$ is the depth at crown. It is obvious from the figure that the resulting depth has a smooth

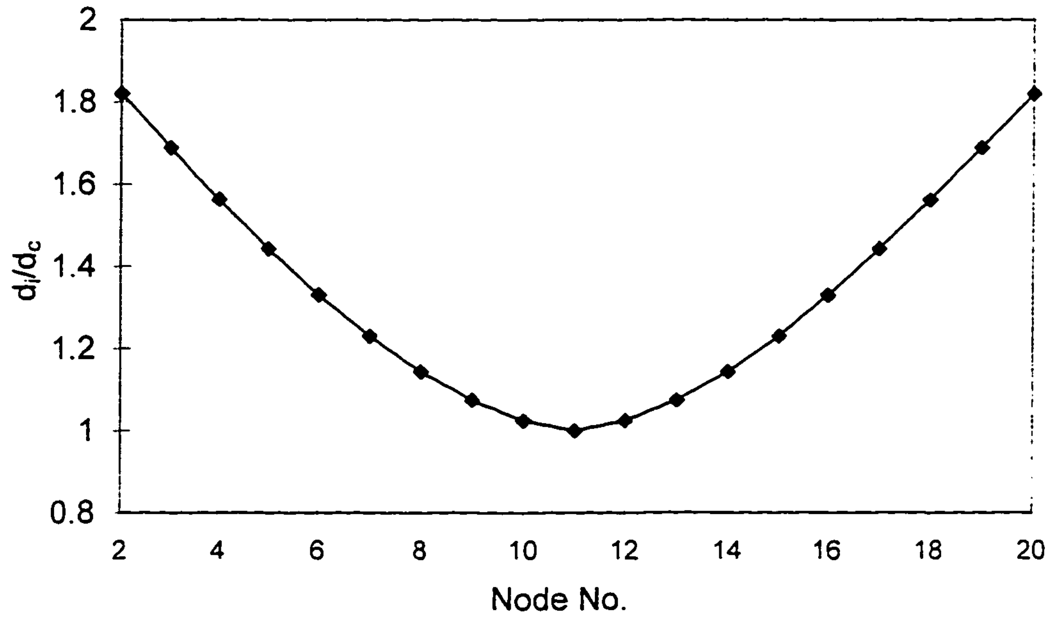



Figure 5.3 : Plot of Required Depth Profile ( Case 1, h=41 m , $t_w$ = 15 mm )

regular profile. It resembles a function which can be derived as follows. As a parabolic arch under uniformly distributed loads is subjected to pure compression only, Eq. 3.29 reduces to

$$A = \frac{P}{F_a} \tag{5.1}$$

and as the thrust is expressed as

$$P = \frac{H_F}{\cos\theta} \tag{5.2}$$

The area and hence the depth, can be represented as

$$A_i = \frac{A_c}{\cos\theta} \tag{5.3}$$

$$d_i = \frac{d_c}{\cos\theta} \tag{5.4}$$

where,

| | | |
|---|---|---|
| $H_F$ | = | horizontal reaction at the arch support, |
| $\theta$ | = | angle between the tangent of the arch and the horizontal axis, |
| $A_c$ | = | area at the crown of the arch, |
| $d_c$ | = | depth at the crown of the arch. |

As the parabolic variation of depth is the chosen profile in this study, the parabolic smoothed depth profile is plotted in Figure 5.4 by nondimensionalizing the depth as before. And Figure 5.5 shows the optimum flange plates cut offs. Both the smoothening of the web profile and the flange cut offs would result in 4% weight increase compared with the cosine function profile of web and flange.

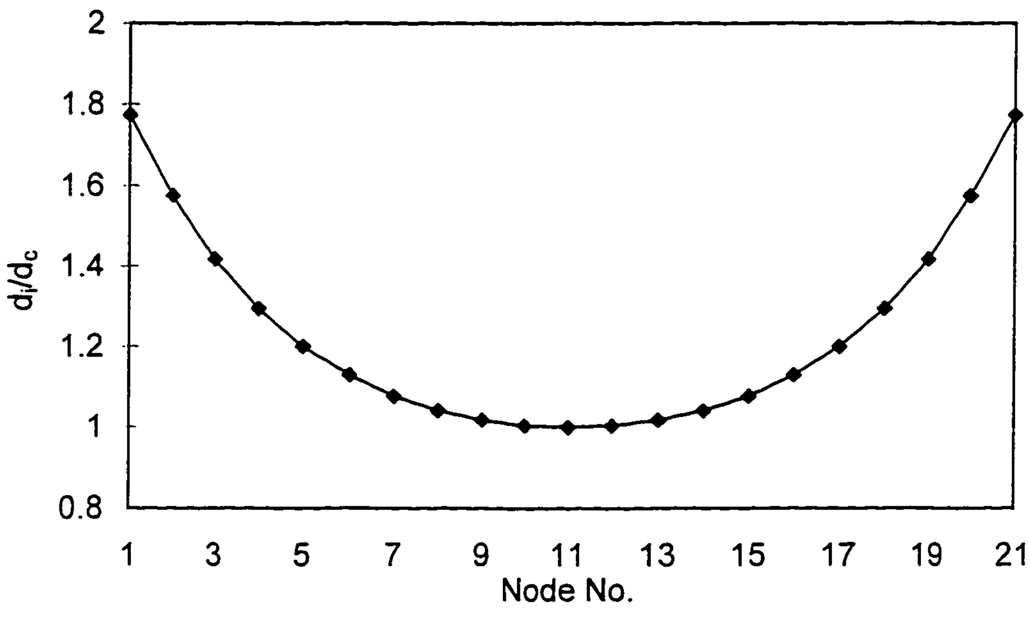



Figure 5.4 : Plot of Smoothed Depth Profile ( Case 1, h=41 m , $t_w$ = 15 mm )

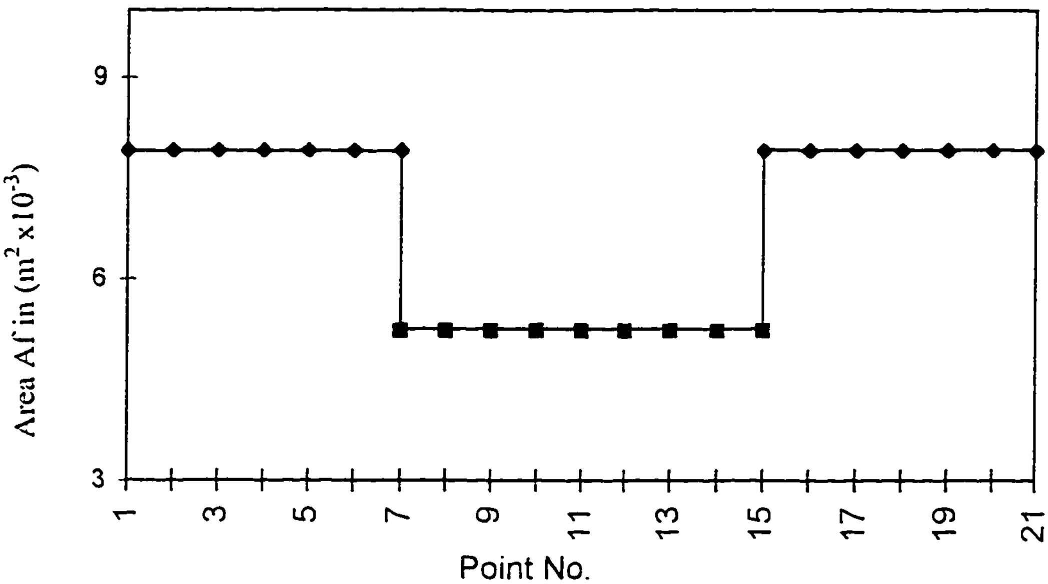



Figure 5.5 : Plot of Optimum flange Cut offs ( Case 1, h=41 m , $t_w$ = 15 mm )

Figure 5.6 indicates that hinged arches subjected to a uniformly distributed load of full length is insensitive to thickness of web plates. This can be explained by Eq. 5.1, which indicates that the required area is a function of applied compressive force and the total cross-sectional areas. As long as the total cross-sectional areas are provided, any thickness of web plate may be used.

In order to verify the independence of the optimum h/L ratio for uniformly loaded arches on their span, several designs were performed considering different spans from 50 m to 200 m. The optimum h/L ratios for spans 50, 100, 150 and 200 are 0.4081, 0.4075, 0.4092 and 0.4054 respectively, which establishes the fact that the h/L ratio can be approximated as 0.41 for all spans. Figure 5.7 shows the variation of the total weight of the arch with respect to all examined spans and h/L ratios. It is observed from the figure that the optimum h/L ratio is almost constant for different spans, at around 0.41. The plot also shows an interesting observation. The weight of arch is less sensitive to the change in rise in the vicinity of the optimum only for short span arches. As the span increases, the curve has a better defined minimum weight and hence optimum h/L.

Parabolic arches, subjected to uniformly distributed loads of full lengths, were also studied for different load intensities, to examine the independence of the optimum h/L ratio on the load intensity. Figure 5.8 shows the variation of the total weight with respect to all examined load intensities and h/L ratios. It is evident from the figure that the optimum h/L ratio is invariant with the load intensity.

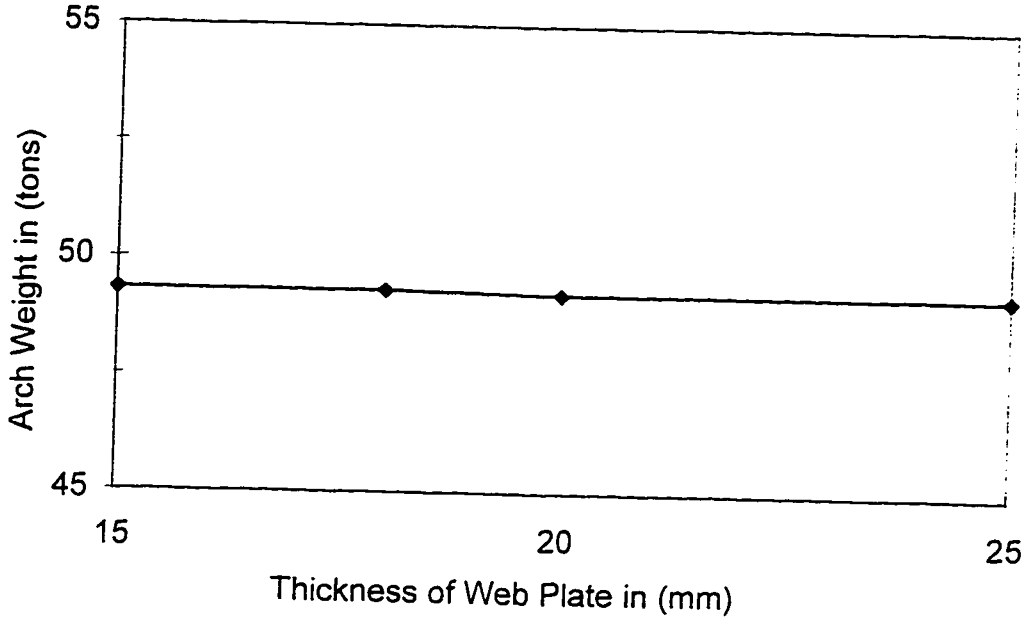



Figure 5.6 : Plot of Optimum Arch Weight Versus Plate Thickness, for Case 1

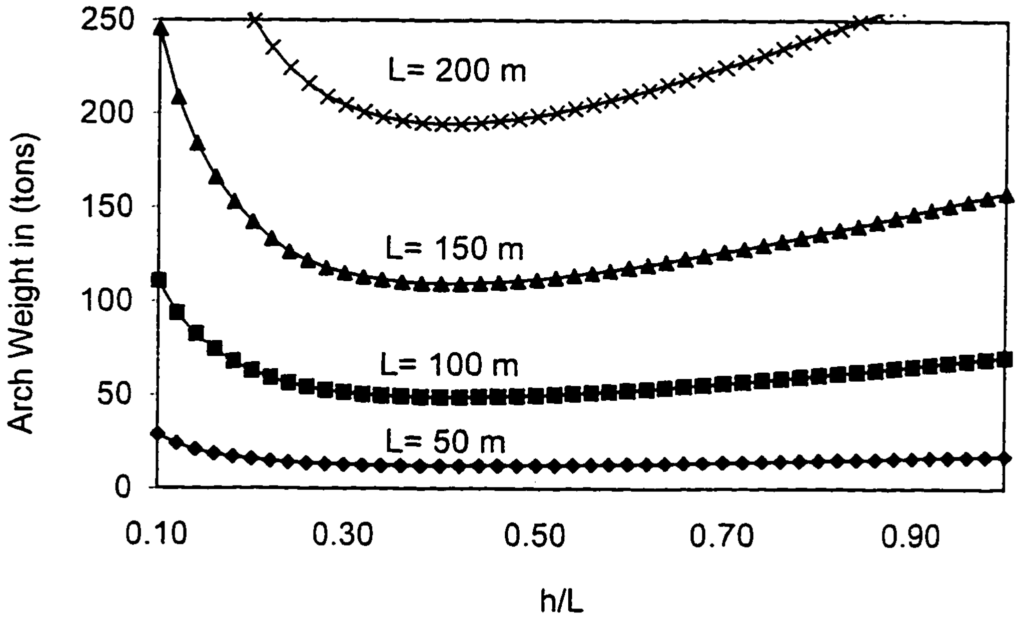
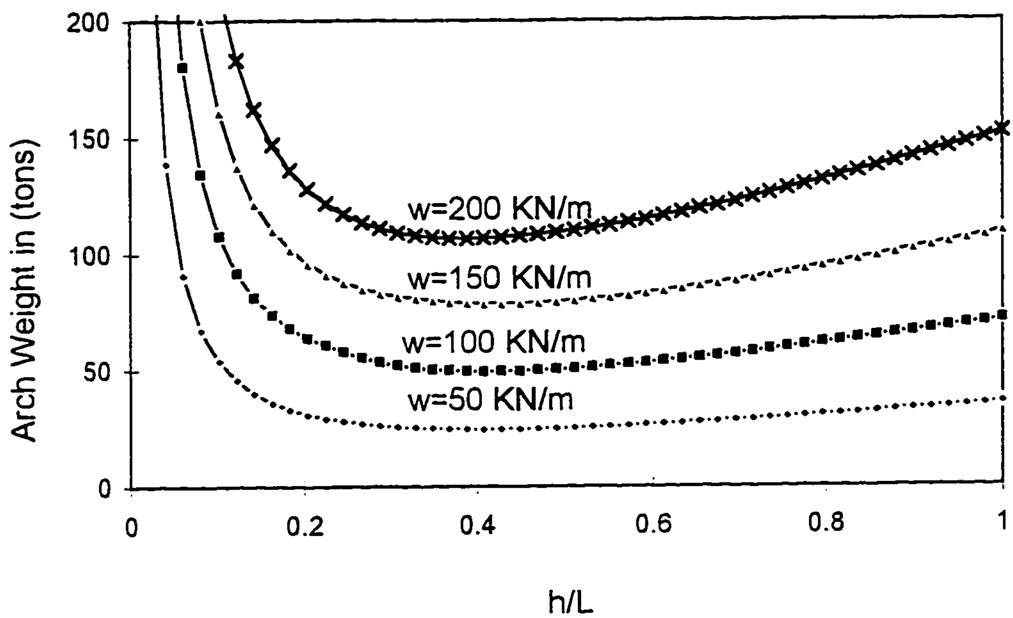



Figure 5.7 : Plot of Arch Weight for Different Spans versus h/L ratio, for Case 1.




5.8 : Plot of Arch Weight for Different Loads Intensities versus h/L ratio, for Case 1.

**CASE 2:a Hinged Arch Subjected to a Uniformly Distributed Load of Partial Length**

This load case produces axial compression, bending moment and shear force in the arch. The following data is applicable to a hinged arch, subjected to a uniformly distributed load at middle third as shown in Fig. 5.9, whose

span L = 100 meter.

The rise will be conforming to the limits : $5 \leq h \leq 100$ .

Uniform load intensity, w = 100 KN/m .

Preassigned number of flange plates, m = 5 .

Range of web plate thickness : $8 \leq t_w \leq 30$ mm .

Material Properties :

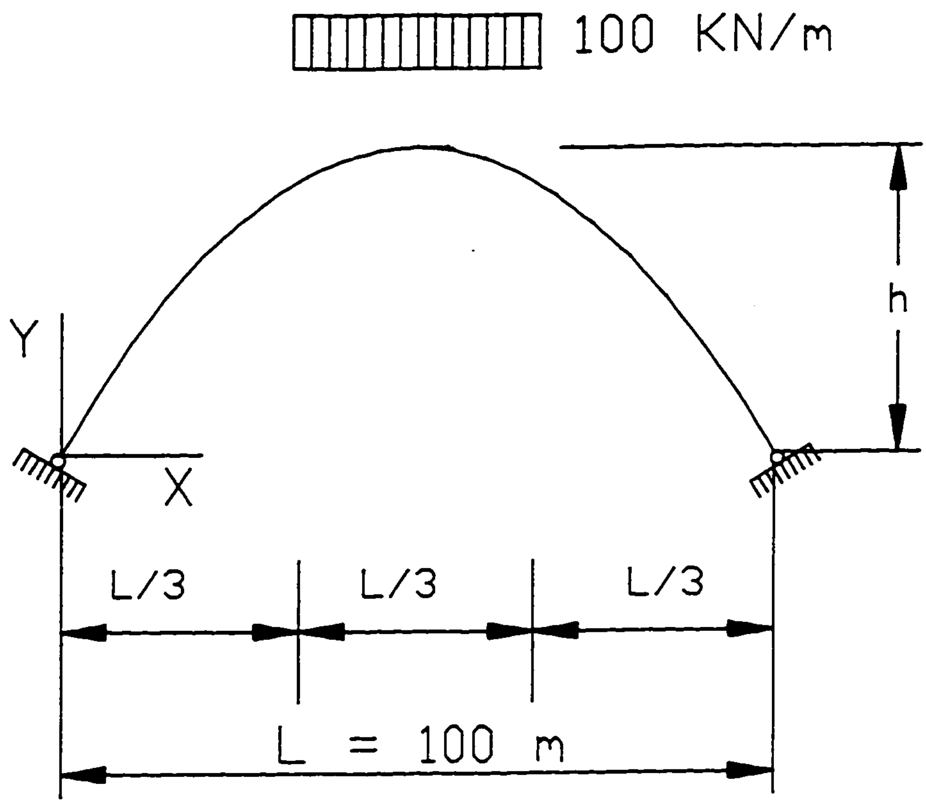
Modulus of elasticity of steel, E = 200 GPa .

Steel Yield Strength , $F_y$ = 248 Mpa .

Allowable compression strength, $F_a$ = 100 MPa .

The structure is discritized into 18 elements . The program starts with an initial design using an upper bound of rise, and a uniform cross section. Then, the arch is analysed. Fig. 5.10 shows a typical moment diagram for this case.

The optimum arch rise is then searched iteratively by sequentially reducing the value of h from the initial value of 100 m. The plot of the nondimensional merit function $W_i/W_o$ versus h/L ratios is shown in Figure 5.11. The plot shows that the minimum weight of the parabolic arch occurs at a ratio h/L = 0.207. Comparing with Case 1, this is almost half the ratio found in Case 1.




5.9 : Case 2, a Hinged Arch Subjected to a Uniformly Distributed Load of Partial Length.

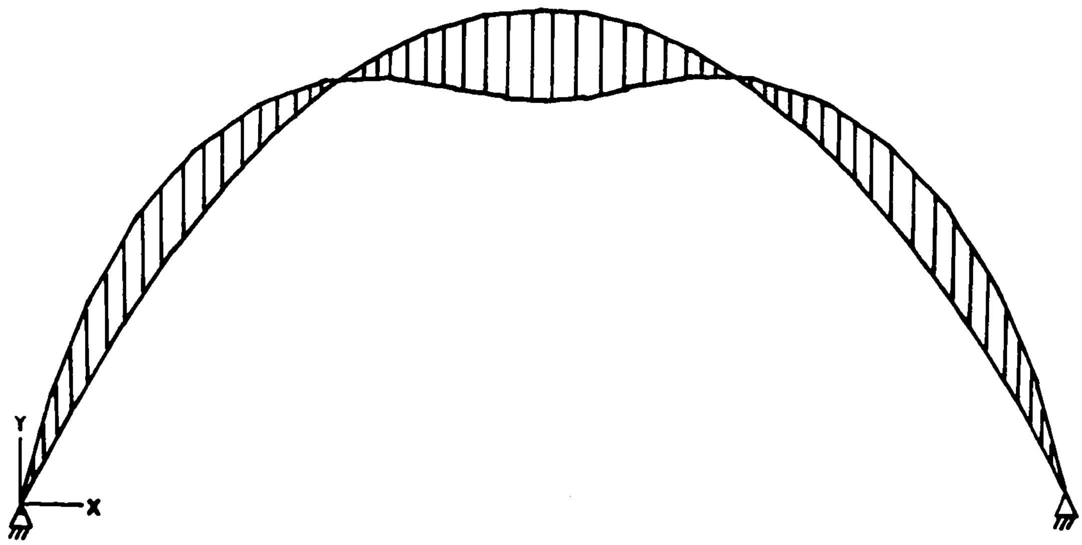



Figure 5.10 : Case 2, Moment Diagram.

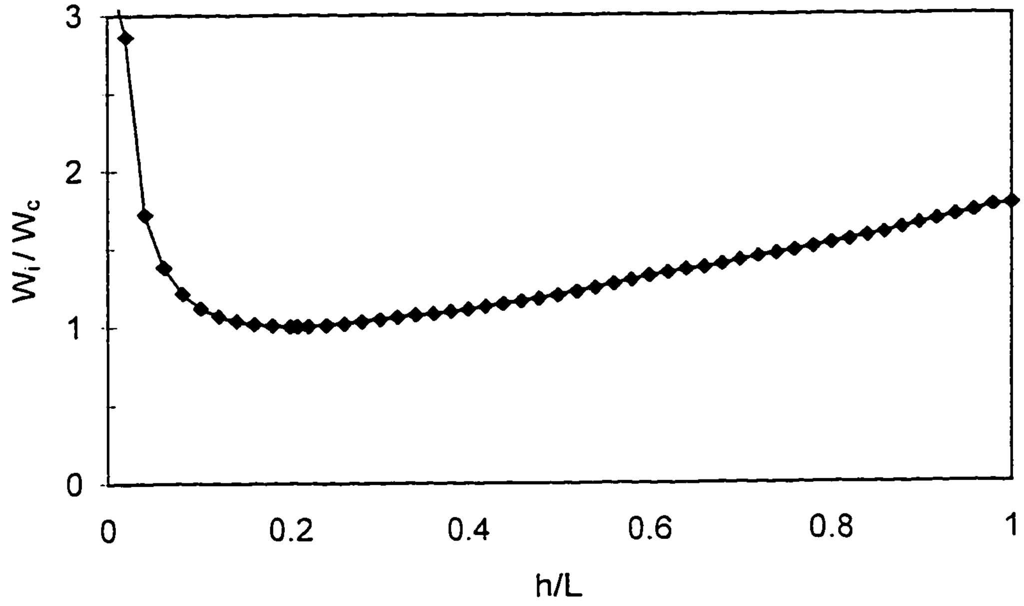



Figure 5.11 : Plot of Total weight Versus h/L ratio for Case 2.

This can be explained as follows. The arch in this case is subjected to both compression and bending moment. As the h/L ratio increases, the total arch length increases and the axial stresses decreases but the moment diagram does not show a tendency to decrease nor increase. Thus at lower h/L ratio than case 1, the optimum weight function is reached.

Of some practical importance is the fact that the weight of the parabolic arch for this case is rather insensitive to changes in the arch rise near the optimum solution, in the h/L range from 0.175 to 0.225 . Furthermore, as the ratio h/L changes between 0.16 and 0.27, the weight of the parabolic arch varies by about 2% only from the optimal weight.

The optimum thickness of web plate is also searched iteratively by sequentially reducing the value of $t_w$ from the initial value of 30 mm. The effect of web thickness on the total arch weight is shown in figure 5.12 for various h/L ratios. It is observed from the figure that the minimum weight occurs at a web thickness equals to 18 mm. Also, the optimum h/L ratio slightly varies with the variation of web thickness. The variation is in the insensitive area defined above, from 0.175 to 0.225.

To show more clearly the influence of web thickness on the height, Figure 5.13 is drawn to summarize data in figure 5.12. In Figure 5.13, $W_t$ is the minimum weight corresponding to a plate thickness t and $W_0$ is the minimum weight corresponding to the optimum web thickness of 18 mm. The figure shows that as the thickness increases from 8 mm to 15 mm, the weight decreases rapidly. Then, from 15 mm to 18 mm the change is virtually negligible. Thereafter weight increases again . This proves that the optimum arch weight is sensitive to the web thickness.

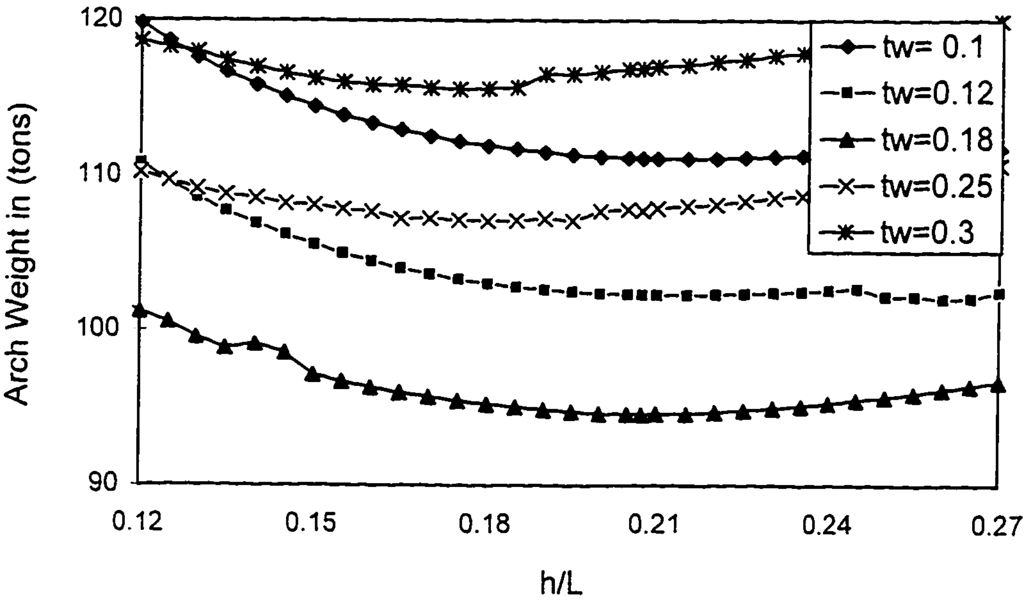



Figure 5.12 : Plot of Arch Weight for Different Web Thickness versus h/L, for Case 2 .

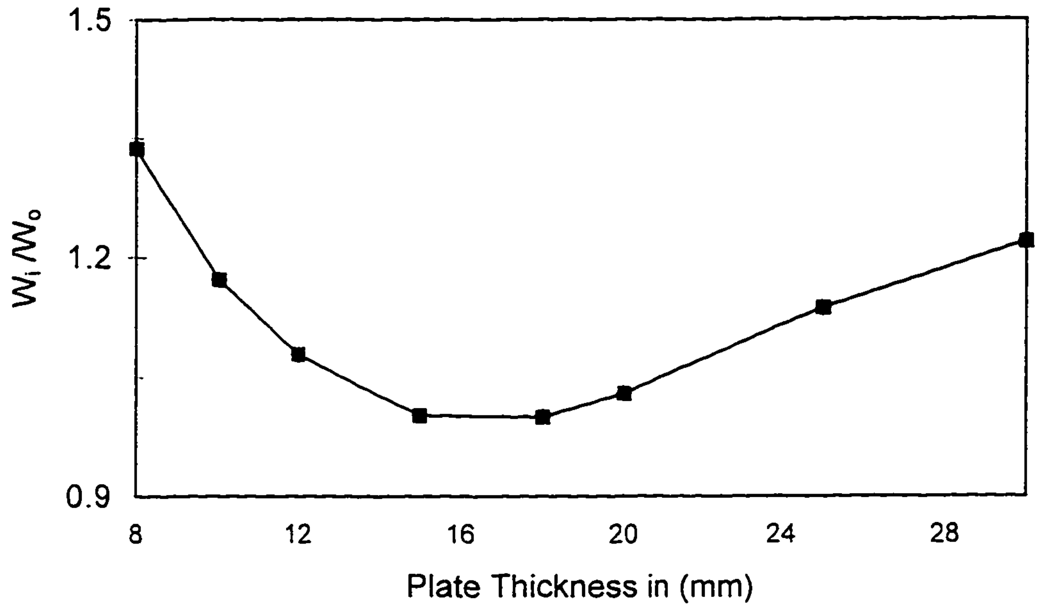



Figure 5.13 : Plot of Weight versus Web Thickness, for Case 2 .

Plots of optimum section properties of the arch are shown in Figures 5.14 and 5.15. Figure 5.14 shows smoothed depth profile by nondimensionalizing the depth as before. Figure 5.15 shows variations of flange areas which is obtained through the use of dynamic programming.

Case 2, was also studied for different spans to see the influence of span on optimum h/L. Figure 5.16 shows the variation of the total weight of the arch with respect to all examined spans and h/L ratios. As noted from the figure the optimum h/L ratio is almost constant for different spans. It is also clear that, as the span increases, the curve has a better defined minimum and therefore an easily determine optimum h/L.

Case 2, was also investigated for different load intensities. Figure 5.17 shows the variation of the total weight with respect to all examined load intensities and h/L ratios. It is observed from the figure that the optimum h/L ratio is invariant with the load intensity.

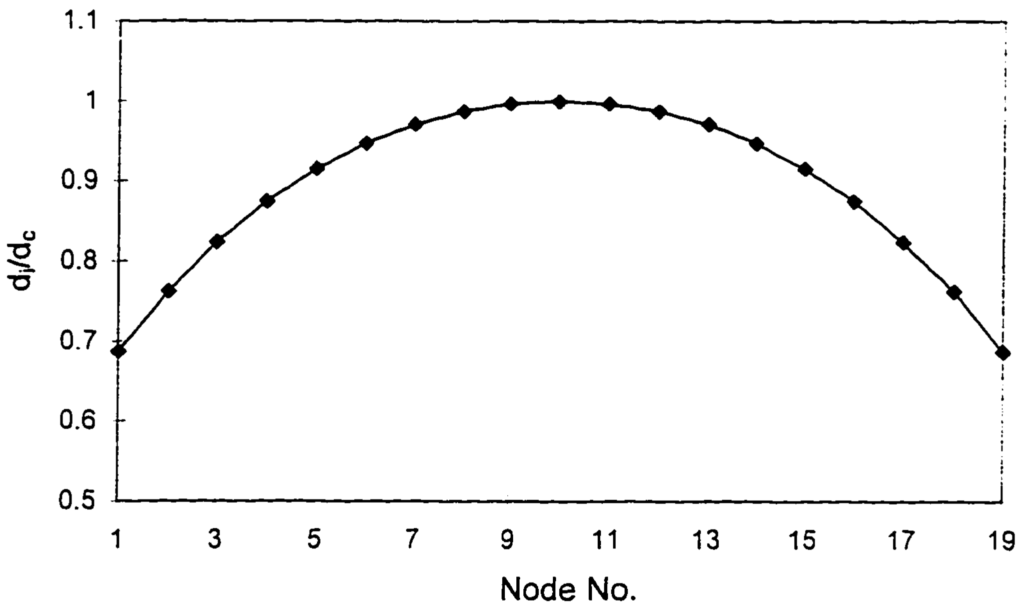



Figure 5.14 : Plot of Optimum Depth Profile ( Case 2, h=20.7 m , $t_w$ = 18 mm )

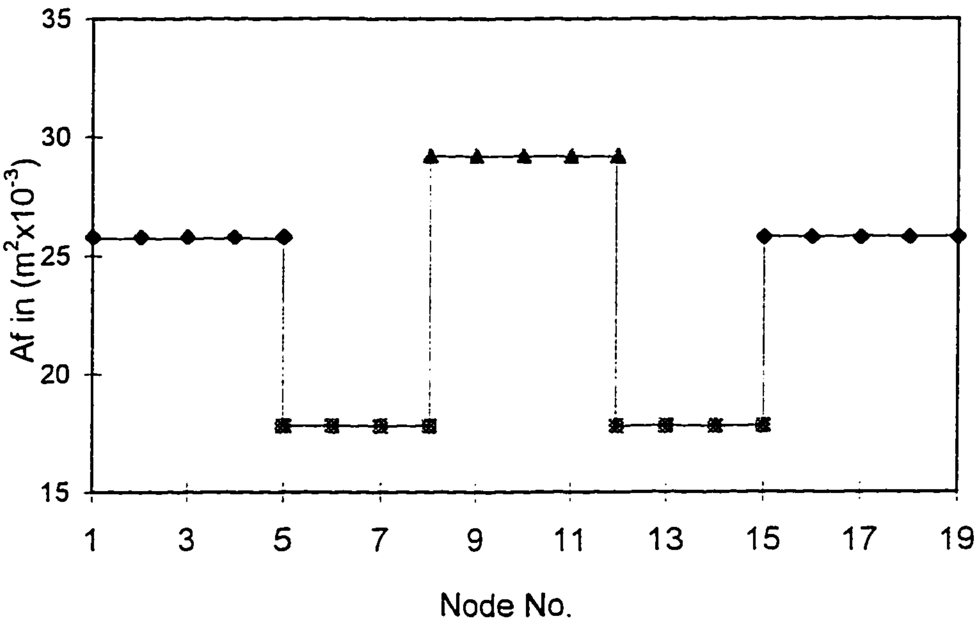



Figure 5.15 : Plot of Optimum flange Cut offs ( Case 2, h=20.7 m , $t_w$ = 18 mm )

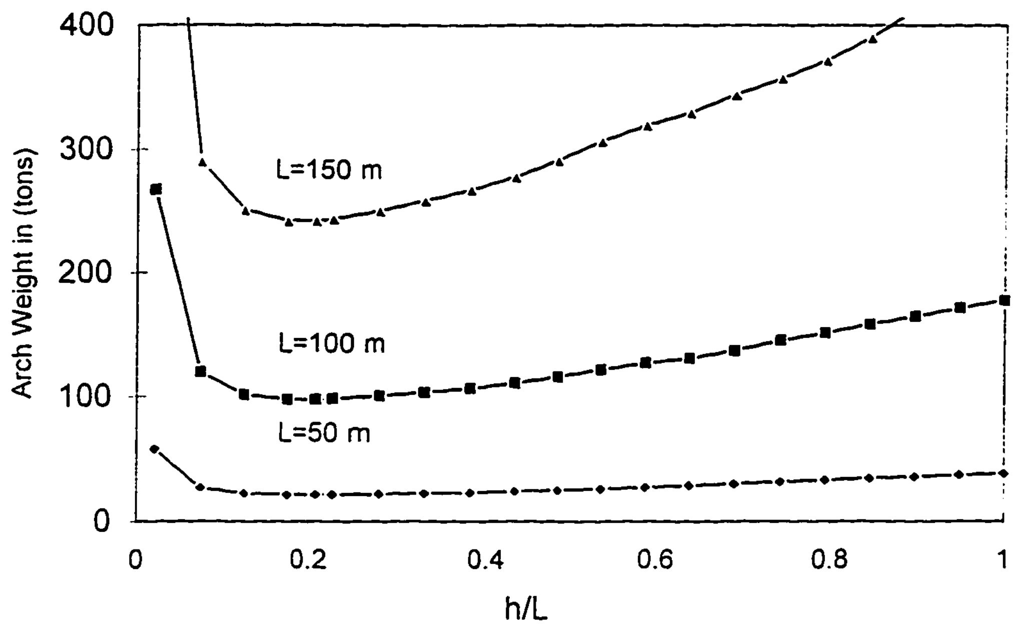



Figure 5.16 : Plot of Arch Weight for Different Spans versus h/L ratio, for Case 2.

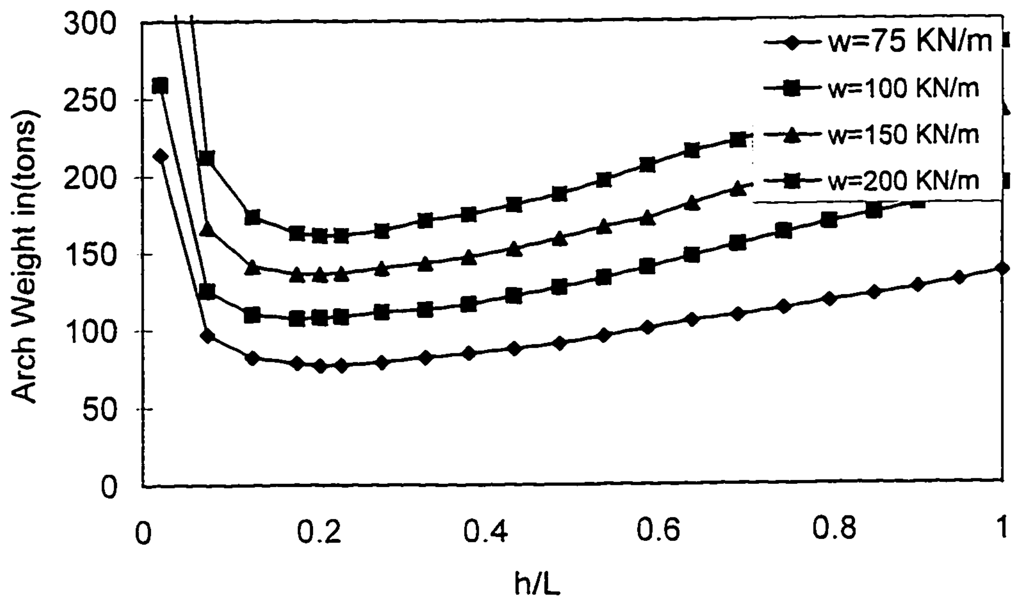



Figure 5.17 : Plot of Arch Weight for Different Loads Intensities versus h/L ratio, for Case 2.

**CASE 3: a Fixed Arch Subjected to a Uniformly Distributed Load of Partial Length**

The following data is applicable to a fixed arch, subjected to a uniformly distributed loads at middle third as shown in Fig. 5.18, whose

span, L = 100 meter .

The rise will be conforming to the limits : $5 \leq h \leq 100$ .

Uniform load intensity, w = 100 KN/m .

Preassigned number of flange plates, m = 5 .

Range of web plate thickness : $10 \leq t_w \leq 30$ mm .

Material Properties :

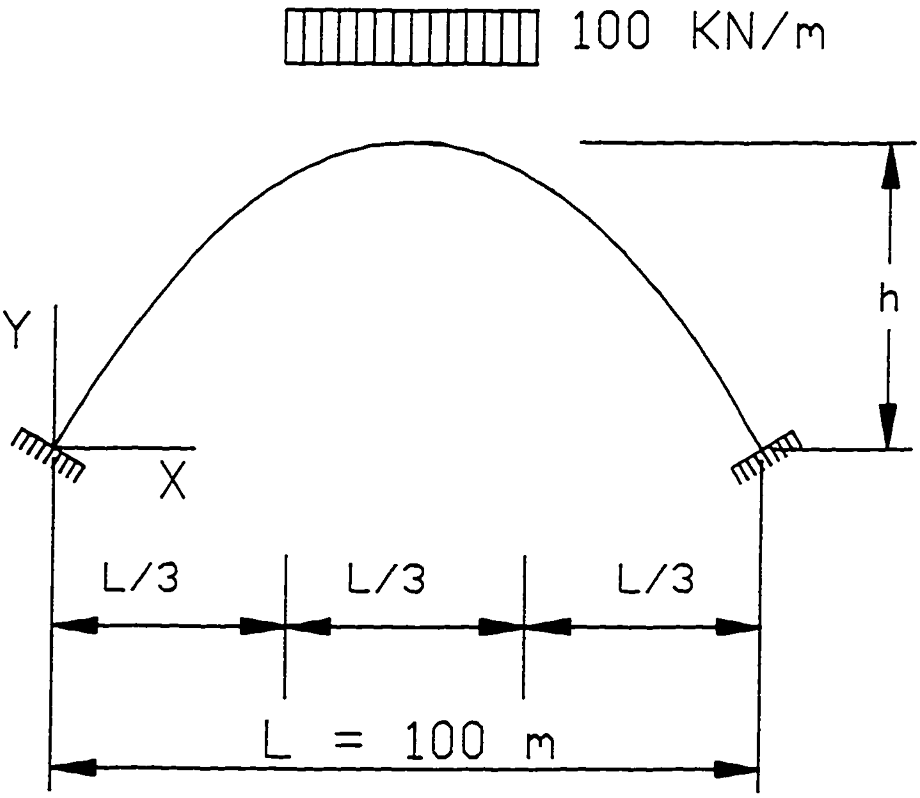
Modulus of elasticity of steel, E = 200 GPa .

Steel Yield Strength , $F_y$ = 248 MPa .

Allowable compression strength, $F_a$ = 100 MPa .

The structure is discretized into 18 elements. Similar to case II, the optimum rise, and the optimum cross section properties are calculated. The plot of the optimum parabolic non dimensional depth profile of the arch is shown in Figure 5.19 . unlike the pin-ended arch, the plot is observed to have a flatter web profile. A plot of a typical moment diagram for case 3 is shown in Fig. 5.20. It is observed from the Figure that both maximum positive and negative moments exists and they occur at different locations. Consequently, the depth profile is so irregular that the smoothening of it results in an almost flat profile. Figure 5.21 shows the optimum cut offs of flange areas.

As the variation of the optimum depth is less than 10% (Figure 5.19), one can use the




5.18 : Case 3, a Fixed Arch Subjected to a Uniformly Distributed Load of Partial Length.

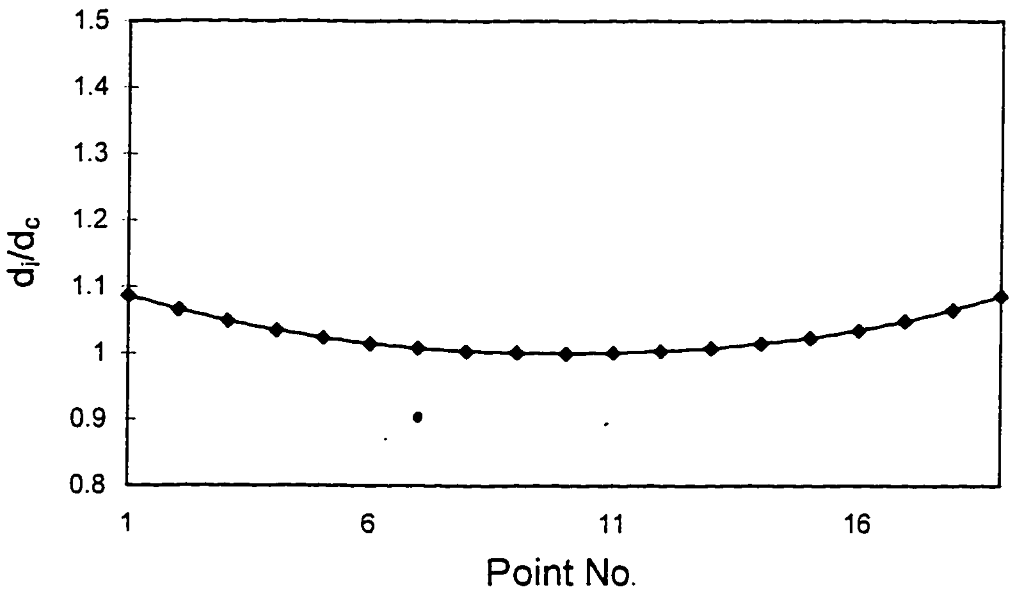



Figure 5.19 : Plot of Optimum Depth Profile ( Case 3, h=21.5 m , $t_w$ = 18 mm )

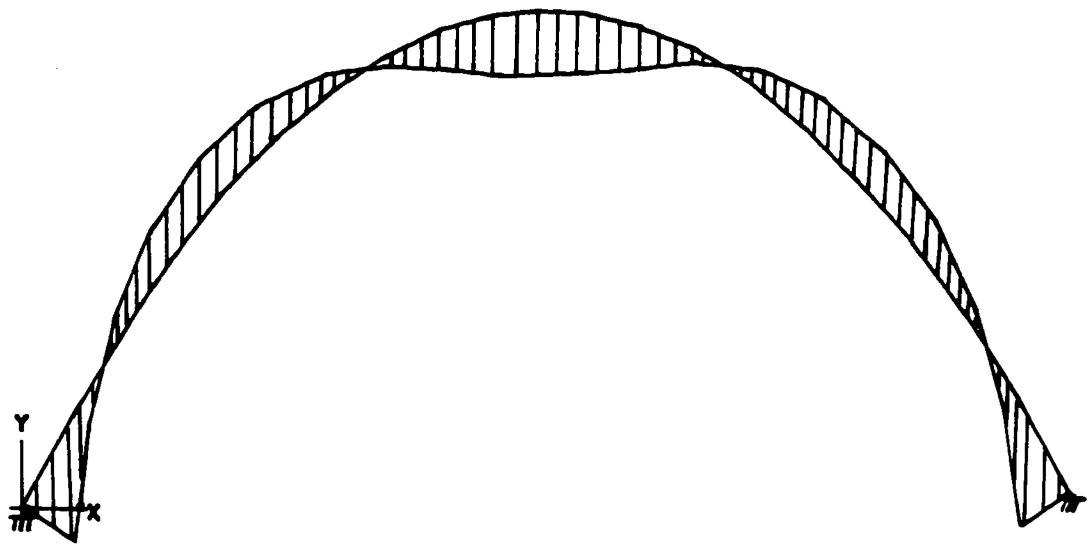



Figure 5.20 : Case 3, Moment Diagram.

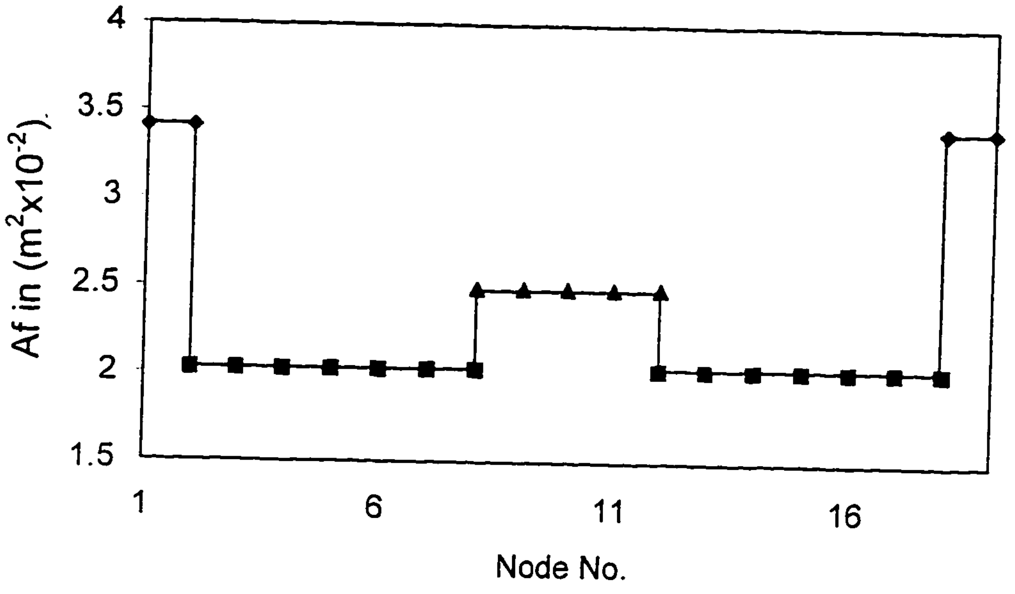



Figure 5.21 : Plot of Optimum flange Cut offs ( Case 3, h=21.5 m , $t_w$ = 18 mm )

straight line curve fitting (uniform depth) for this case. Figure 5.22 shows the calculated depth, smooth depth using both the line fitting and the polynomial fitting. Figure 5.23 shows the total weight of the arch using: (i) polynomial fitting of web and flange cut offs, (ii) uniform depth and the same number of flange cut offs and (iii) uniform depth and uniform flange. It is observed from the figure that the total weight of the arch resulted using (i) and (ii) is almost equivalent for h/L ratio greater than 0.20 , below which the polynomial fitting is more economical. It is also observed that both proposals are economical than (iii). The flanges cut offs, in (i) and (ii) results in a 20% saving in the total weight of the arch at optimum h/L ratio. Thus for Case 3, the use of uniform depth and a few number of flange cut offs is a good alternative.

The plot of the nondimensional merit function $W_i/W_o$ , versus h/L ratios is shown in Figure 5.24. The plot shows that the minimum weight of the parabolic arch occurs for this case at a ratio h/L = 0.215 . The plot also shows that any fixed-ended parabolic arch having h/L ratio ranges between 0.16 and 0.28 can at most be 2% heavier than the optimum arch. Thus, for an economical design of a fixed-ended arch, subjected to a uniformly distributed loads of partial length, a ratio of h/L between 0.16 and 0.28 can be chosen.

The optimal shape of the arch for Case 3 is investigated for various spans. Figure 5.25 shows the variation of the total weight of the arch with respect to all examined spans and h/L ratios. As noted from the Figure, the optimum h/L ratio varies with spans between 0.215 and 0.17. Thus it can be concluded that for a fixed-ended arch , the optimum h/L ratio, unlike hinged arches, depends on the span.

Case 3, was also investigated for different load intensities. Figure 5.26 shows the

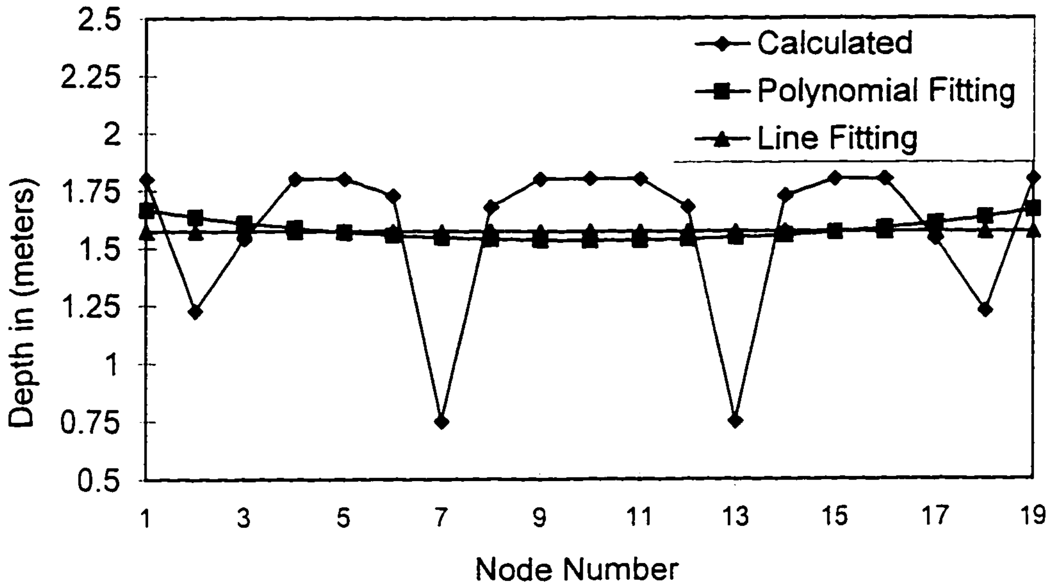



Figure 5.22 : Plot of Calculated Depth Profile, Smoothened depth profile using the polynomial fitting and the uniform depth (Case 3, h=21.5 m , $t_w$ = 18 mm)

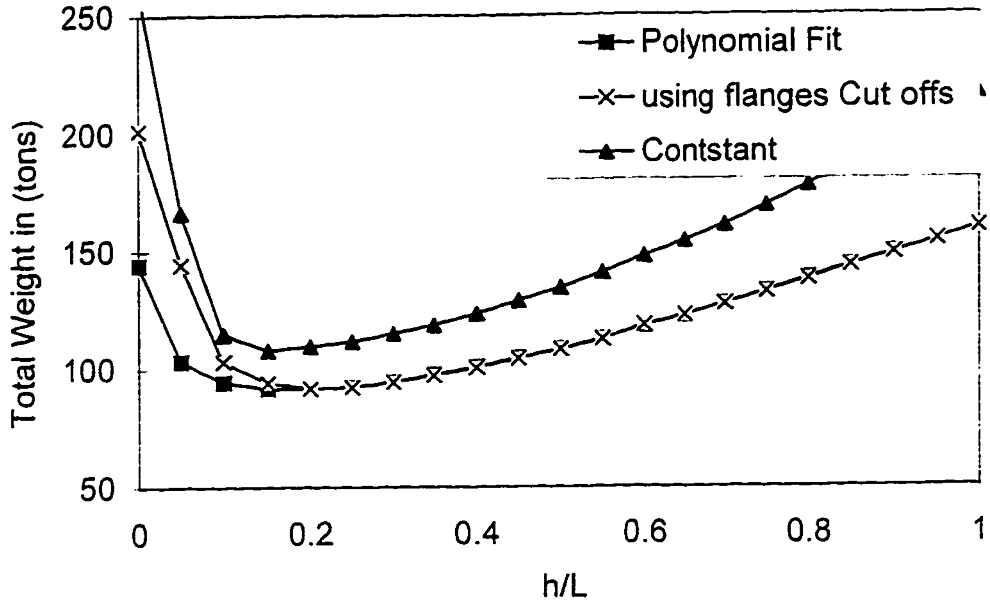



Figure 5.23 : Plot of the total weight of the arch using: (i) polynomial fitting of web and flange cut offs, (ii) uniform depth and the same number of flange cut offs and (iii) uniform depth and uniform flange ( Case 3, h=21.5 m , $t_w$ =18 mm ).

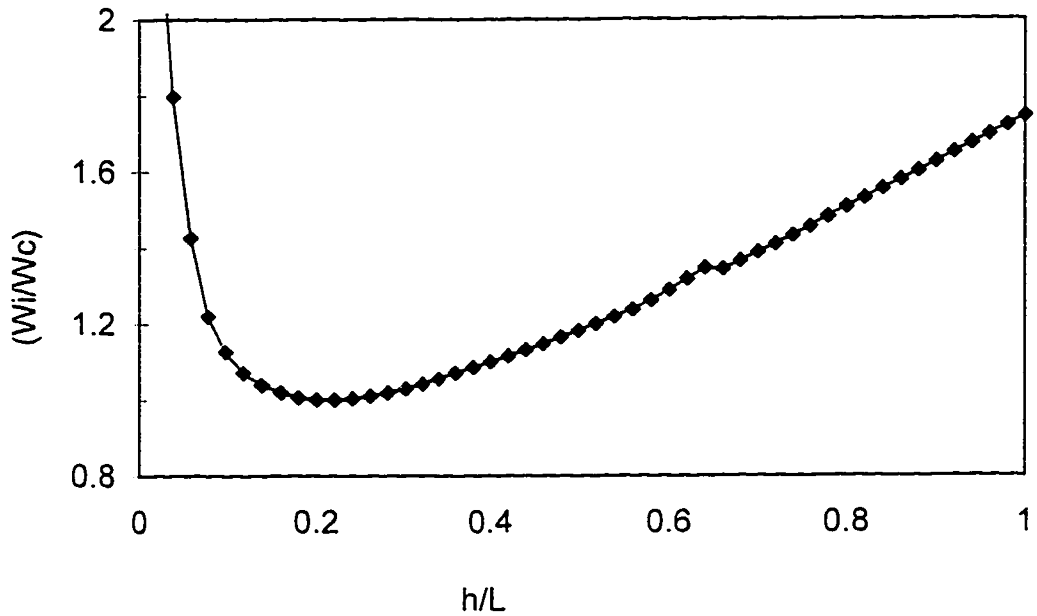



Figure 5.24: Plot of Total weight Versus (h/L) ratio for Case 3.

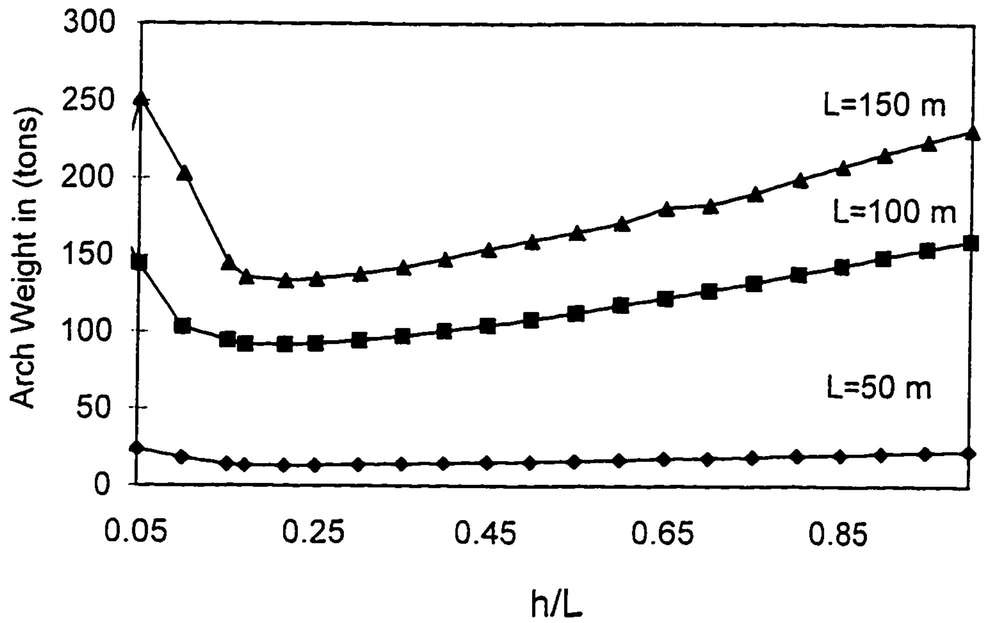



Figure 5.25 : Plot of Arch Weight for Different Spans versus h/L ratio, for Case 3.

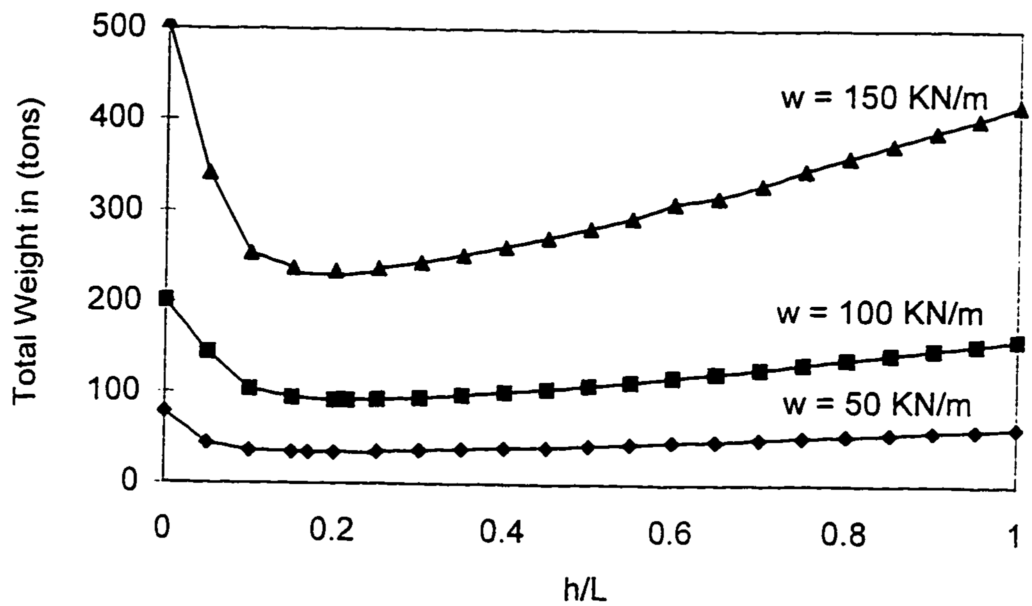



Figure 5.26 : Plot of Arch Weight for Different Loads Intensities versus h/L ratio, for Case 3.

variation of the total weight with respect to all examined load intensities and h/L ratios. Unlike hinged arches, the optimum h/L ratio varies with the load intensity from 0.215 to 0.18 . Thus, for a fixed-ended arch , the optimum h/L ratio, also depends on the load intensity.

A comparison of optimum arch weight at optimum rises for different plate thickness is plotted in figure 5.27 , by nondimensionalizing the weight function as before . The plot shows similar conclusion as in Case 2.

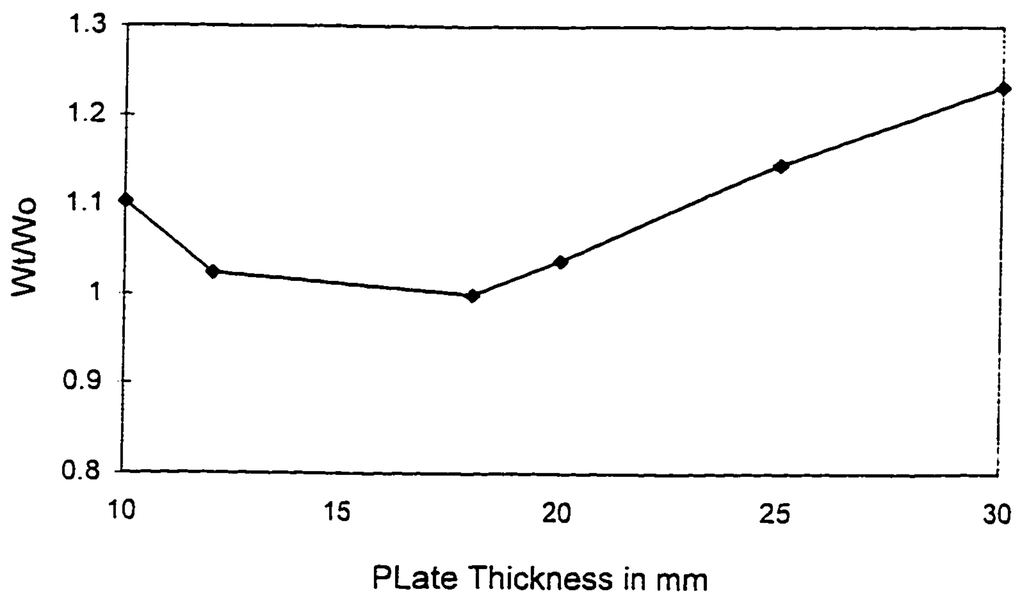



Figure 5.27 : Plot of Weight versus Web Thickness, for Case 3 .

**CASE 4 : Hinged Arches Subjected to a nonsymmetrical Concentrated Loads**

The following data is applicable to a hinged arch, subjected to a nonsymmetrical concentrated Load at L/4 as shown in Fig. 5.28, whose span, L = 100 meter :

The rise will be conforming to the limits : $5 \leq H \leq 45$ .

The concentrated load, P = 1000 KN .

Preassigned number of flange plates, m =5.

Limits for web plate thickness : $10 \leq t_w \leq 20$ mm.

Material Properties :

Modules of elasticity of Steel , E = 200 GPa .

Steel Yield Strength , $F_y$ = 248 MPa.

Allowable compression strength, $F_a$ = 100 MPa.

The structure is discretized into 20 elements. Similar to cases I, II and III, the optimum rise, and the optimum cross section properties are calculated. Figure 5.29 shows the optimum parabolic depth profile for both the selected case and for an alternating load case in which the concentrated load is alternating between L/4 and 3L/4. The solid line represents case 4, while the dashed line represents the case of an alternating concentrated load. Figure 5.30 shows the optimum flange cut offs.

The plot of the nondimensional merit function $W_i/W_o$ , versus h/L ratios is shown in Figure 5.31. The plot shows that the minimum weight of the parabolic arch occurs at a ratio h/L = 0.10 . The plot indicates that for the alternating load case, the total arch weight is more due to the addition of unnecessary areas to make the arch symmetrical despite the fact that loading is nonsymmetrical (Figure 5.29).

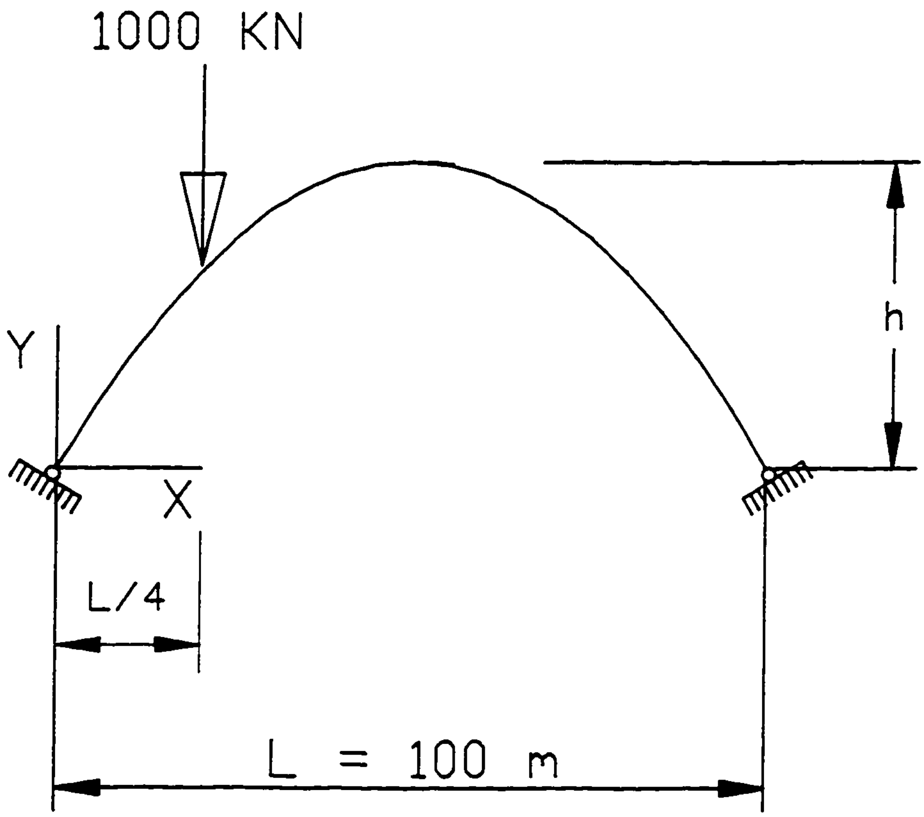



Figure 5.28 : Case 4, a Hinged Arch Subjected to a non symetric Concentric Load at L/4.

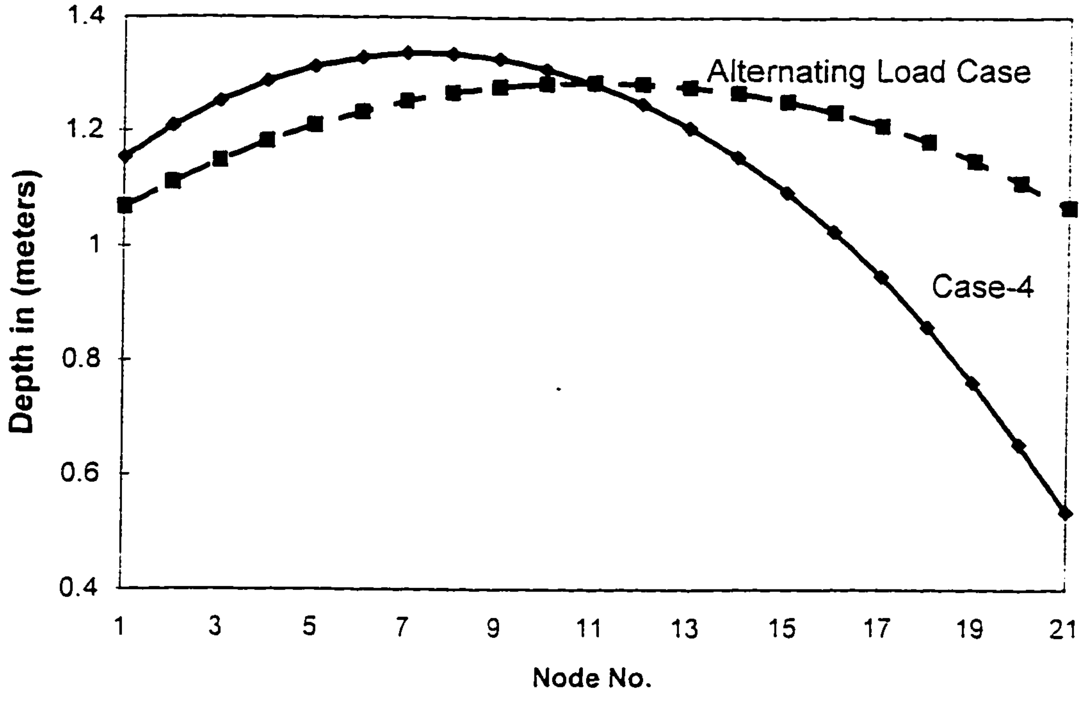



Figure 5.29 : Plot of Optimum Depth Profile ( Case 4, h=10 m , $t_w$ = 16 mm )

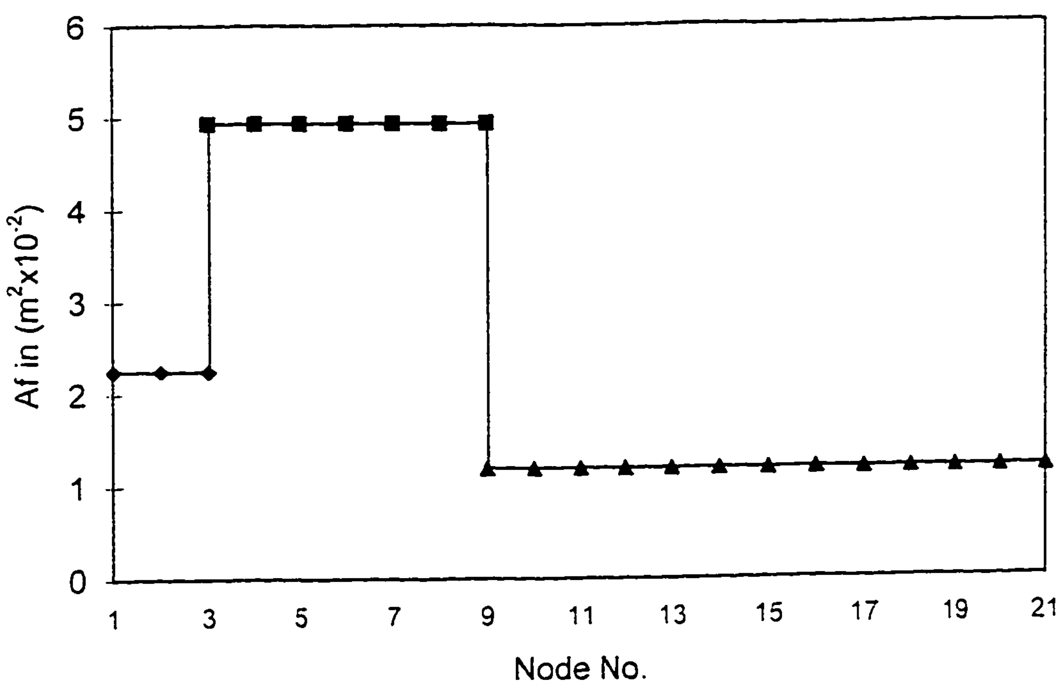



Figure 5.30 : Plot of Optimum flange Cut offs ( Case 4, h=10 m , $t_w$ = 16 mm )

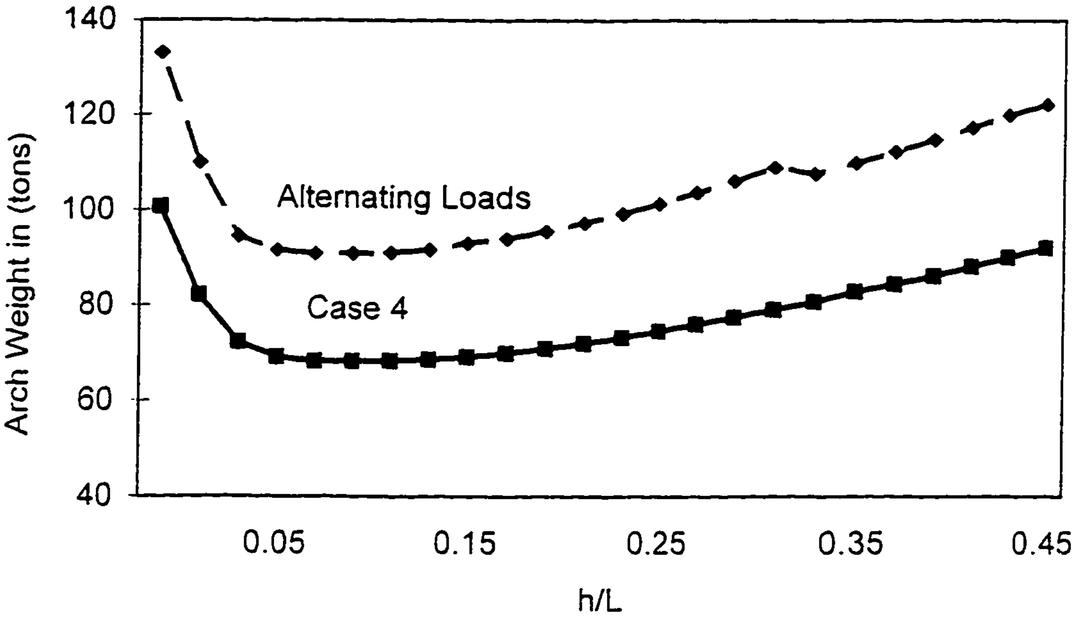



Figure 5.31 : Plot of Total Arch Weight Versus h/L Ratios for case 4.

## 5.3 REMARKS ON SENSITIVITY ANALYSIS

A study of the sensitivity of the total weight of arches in the proceeding examples with regard to changes in the design variables indicats that the total arch weight is sensitive to two design variables: h/L ratio and web plate thickness.

Figures 5.2, 5.11, 5.24 and 5.31 shows the variation of the weight of the parabolic arch with the ratio h/L for Cases 1, 2, 3 and 4 respectively. A study of the sensitivity of these curves reveals two noteworthy observations. One is related to the existence of the optimum h/L ratio and the other one related to the insensitiveness of the least weight solution in the vicinity of the optimum (h/L). First, the optimum h/L ratio for a parabolic arch subjected to a prescribed loading occurs at a well defined point in the curve below and above which the total weight increases. Second, the weight of the parabolic arch is insensitive to changes in the vicinity of optimum h/L. Thus, within a range of h/L ratios, 10% to 15% in the proceeding examples, an economical design may be obtained. Moreover, it is observed from Figures 5.7, 5.15 and 5.23 that the weight of arch is less sensitive to changes in the rise in the vicinity of the optimum only for short span arches. As the span increases, the curve has a better defined minimum weight and hence optimum h/L.

The choice of web thickness has a great influence on the weight of the parabolic arch subjected to both axial force and bending moment. For example, for Case 2, from Figure 5.12, the use of 8 mm thick web plate instead of the optimum one which is 18 mm thick, results in 33% increase in the weight. Thus, the arch is very sensitive to web thickness. But, unlike the h/L design variable, the optimal value is not well defined. This is because of the following 2-constraints: d/t constraint (Eq. 3.14) and constraint

on Af (Eq. 3.11). During step No. 4, Section 4.6, the use of thin steel plate together with the d/t constraint result in limited depth, smaller $A_w/A_t$ ratio and a smaller section modulus. The optimum web thickness will minimize both effects for most elements of the arch. This may be easily determined for one element, but doing this process for a set of variant elements results in a random global optimum thickness. Thus several acceptable web thickness must be tried to find the optimum web thickness.

# CHAPTER VI

# CONCLUSIONS AND RECOMMENDATIONS

## 6.1 CONCLUSIONS

This study deals with the development of an optimization procedure to find a least-weight design of a parabolic box-shaped steel arch of variable cross section subjected to a set of constraints, imposed by geometrical, strength and practical requirements. The strength requirements of AISC specifications have been followed and the arch is assumed to be laterally supported. The web profile is considered to have a smooth profile. The flange areas are varied on the basis of a preassigned maximum number of cut-off points. The applicability of the developed optimization program has been demonstrated with several illustrative examples, which also provide some interesting observations with regard to optimum design.

Based on this study, the following conclusions are drawn :

1. A generalized computer program POPSAR has been developed to readily determine the optimum design of a non-uniform parabolic arch with a smooth web profile and variable flange areas. The iterative search

procedure which incorporates dynamic programming and curve smoothening subroutines yields an optimum design within the prescribed constraints.

2. The most important design variable for a parabolic arch is the h/L ratio. For a parabolic arch subjected to a prescribed loading an optimum h/L ratio exists, below and above which the weight increases. However, it has been observed that the optimum design is insensitive to h/L in the vicinity of the optimum for all parabolic arches.

3. For a parabolic arch subjected to uniformly distributed loads of full length, which represent the case of pure compression in the arch, the optimal arch rise is 0.41 times the span length, i.e. h/L=0.41. However, an economical design of a parabolic arch subjected to uniformly distributed loads of full length can be obtained with a h/L ratio between 0.34 and 0.50, as the arch weight is relatively insensitive to h/L within this range.

4. For a parabolic arch subjected to a uniformly distributed load of full length, the optimal h/L ratio is invariant with span length , load intensity and web plate thickness .

5. As loads deviate from the case of uniformly distributed loads of full length, they produce moments in a parabolic arch, and consequently, the optimal arch shape become shallower than h/L = 0.41. In other words, for any

parabolic arch (fixed or pinned) with partial load lengths or concentrated loads, the optimum h/L < 0.41

6. For a pinned pinned parabolic arch subjected to a uniformly distributed load w over a partial length of $\alpha L$ ($\alpha<1$), the optimum h/L (< 0.41 ) is invariant to the load intensity w and the span L. However, for a fixed-fixed parabolic arch the same is not true, as the optimum h/L is dependent upon w and L.

7. Apart from the most significant design variable h/L, the other important variable is the web thickness, as it influences the weight to a high degree. For an arch subjected to both axial force and bending moment, an optimum web thickness must be determined by iterative procedure as prescribed.

## 6.2 RECOMMENDATIONS :

The following recommendations are made with regard to future work in this area

1. Future study has to be directed toward the determination of the allowable axial compressive stress.

2. Future research is recommended to be conducted on stiffened web arches.

# APPENDIX A

## Numerical Examples for Arch Analysis

In this appendix, two examples are analyzed to show that the frame analysis routine used in this study gives accurate results. The geometry of these examples and the loadings are chosen at random. The analysis routine was varied by comparing results with STAAD III package.

As a first example, member end forces are calculated for a hinged arch, subjected to a uniformly distributed load of full length as shown in Fig. A.1, whose span, L = 40 meter, rise, h = 17.32 meter and uniform load intensity, w = 100 KN/m . Twenty finite elements were chosen to approximate the arch and a uniform cross sections are assumed. The arch was analyzed twice using ANASYS routine and STAAD III. As the arch geometry and the loads are symmetric, element end forces and moments in the elements resulting from the load applied to the structure are shown in Table A.1 for the left half of the arch. These forces are in local element coordinate system. Figure A.2 shows the member end actions with their directions.

The arch in this case is subjected to pure compression only. It is evident from Table A.1 that axial forces calculated using both packages are identical. This is because, both packages uses finite element method and straight element approximations.

As a second example member end forces are calculated for a hinged arch, subjected to a uniformly distributed load at middle 30 % as shown in Fig. A.3, whose

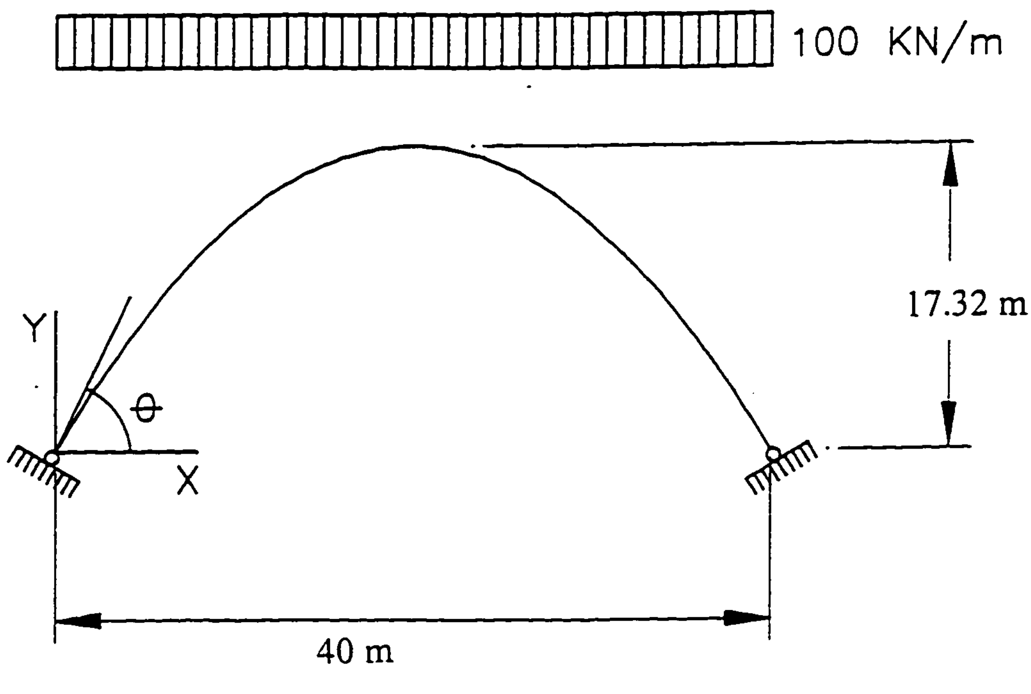



Figure A.1 : Example 1, a Hinged Arch subjected to a Uniformly Distributed Load of Full Length.

TABLE A.1 : Element End Forces for Example 1.

| Element No. | Joint. | Joint Forces | |
|---|---|---|---|
| | | Axial Forces (KN) | |
| | | ANASYS | STAAD III |
| 1 | 1 | 2223 | 2223 |
| | 2 | -2223 | -2223 |
| 2 | 1 | 2055 | 2055 |
| | 2 | -2055 | -2055 |
| 3 | 1 | 1893 | 1893 |
| | 2 | -1893 | -1893 |
| 4 | 1 | 1739 | 1739 |
| | 2 | -1739 | -1739 |
| 5 | 1 | 1595 | 1595 |
| | 2 | -1595 | -1595 |
| 6 | 1 | 1464 | 1464 |
| | 2 | -1464 | -1464 |
| 7 | 1 | 1350 | 1350 |
| | 2 | -1350 | -1350 |
| 8 | 1 | 1258 | 1258 |
| | 2 | -1258 | -1258 |
| 9 | 1 | 1193 | 1193 |
| | 2 | -1193 | -1193 |
| 10 | 1 | 1159 | 1159 |
| | 2 | -1159 | -1159 |

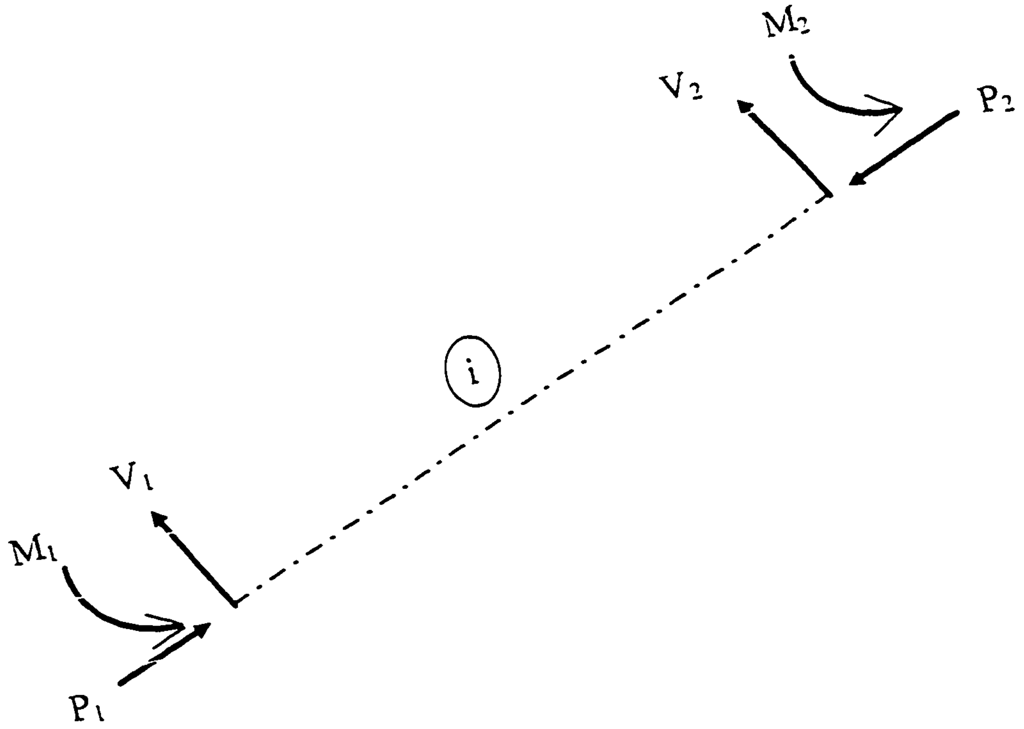



Figure A.2 : Local Coordinte system.

span, $L = 40$ meter, rise, $h = 17.32$ meter and uniform load intensity, $w = 100$ KN/m. The arch is discretized into 20 elements. Using both packages, the arch is analyzed and member end forces and moments are tabulated in Table A.2. It is evident from the Table that both the analysis routine results and STAAD III results are almost identical.

Two sum up, comparing the analysis routine used in this study with the well known analysis package STAAD III, proves that the analysis package is sufficiently accurate.

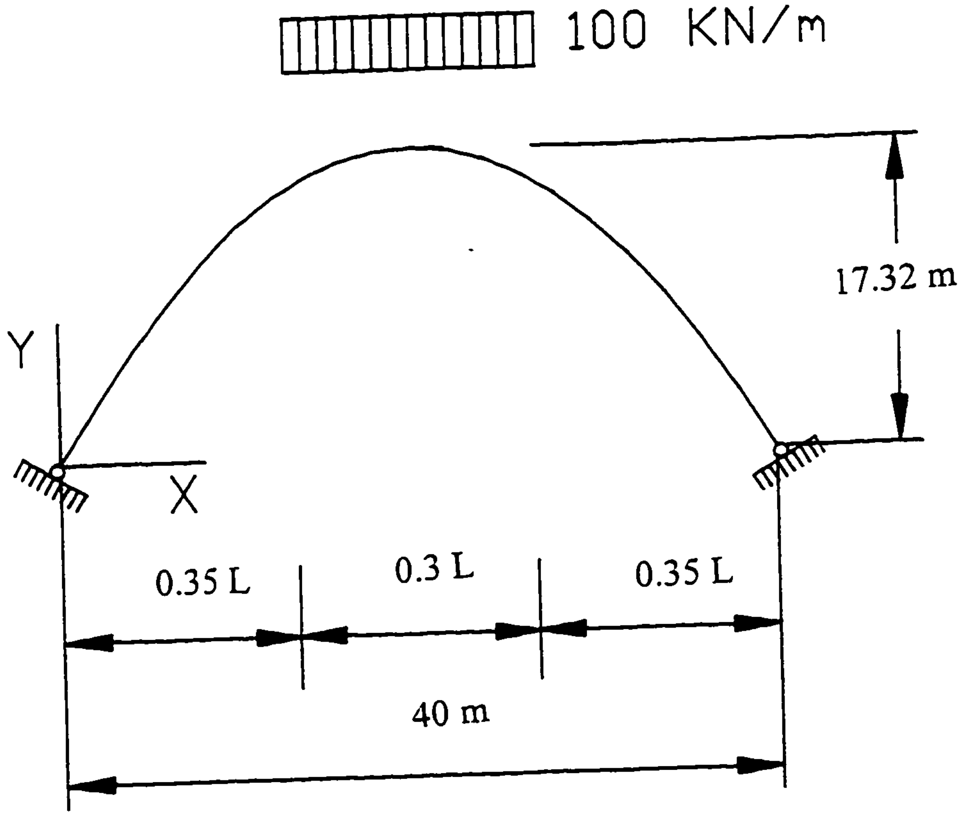



Figure A.3 : Example 2, a Hinged Arch subjected to a Uniformly Distributed Load at Middle 30%.

TABLE A.2 : Element End Forces for Example 2.

| Elem. No. | Jt. | Joint Forces | | | | | |
|---|---|---|---|---|---|---|---|
| | | Axial Forces (KN) | | Shear (KN) | | Moment (KN.m) | |
| | | ANASYS | STAAD III | ANASYS | STAAD III | ANASYS | STAAD III |
| 1 | 1 | 778.7 | 778.7 | -125.9 | -126.0 | 0.0 | 0.0 |
| | 2 | -778.7 | -778.7 | 125.9 | 126.0 | -485.0 | -485.0 |
| 2 | 1 | 784.0 | 784.0 | -86.4 | -86.4 | 485.0 | 485.0 |
| | 2 | -784.0 | -784.0 | 86.4 | 86.4 | -792.6 | -792.4 |
| 3 | 1 | 787.8 | 787.8 | -39.7 | -39.7 | 792.6 | 792.4 |
| | 2 | -787.8 | -787.8 | 39.7 | 39.7 | -922.9 | -922.6 |
| 4 | 1 | 788.6 | 788.6 | 15.6 | 15.6 | 922.8 | 922.6 |
| | 2 | -788.6 | -788.6 | -15.6 | -15.6 | -875.7 | -875.7 |
| 5 | 1 | 784.6 | 784.6 | 81.3 | 81.3 | 875.7 | 875.7 |
| | 2 | -784.6 | -784.6 | -81.3 | -81.3 | -651.3 | -651.1 |
| 6 | 1 | 772.7 | 772.7 | 158.5 | 158.4 | 651.3 | 651.1 |
| | 2 | -772.7 | -772.7 | -158.5 | -158.4 | -249.4 | -249.3 |
| 7 | 1 | 748.9 | 748.9 | 247.7 | 247.5 | 249.4 | 249.3 |
| | 2 | -748.9 | -748.9 | -247.7 | -247.5 | 329.8 | 329.6 |
| 8 | 1 | 668.5 | 668.5 | 255.4 | 255.4 | -329.8 | -329.6 |
| | 2 | -668.5 | -668.5 | -255.4 | -255.4 | 886.4 | 886.2 |
| 9 | 1 | 571.0 | 571.0 | 161.6 | 161.8 | -886.4 | -886.2 |
| | 2 | -571.0 | -571.0 | -161.6 | -161.8 | 1220.0 | 1220.5 |
| 10 | 1 | 518.7 | 518.8 | 55.5 | 55.2 | -1220.0 | -1220.5 |
| | 2 | -518.7 | -518.8 | -55.5 | -55.2 | 1332.0 | 1331.4 |

# APPENDIX B

## Polynomial Curve Fitting

The polynomial curve fitting, investigated in section 4.4, is illustrated through an example. The geometry of this example and the loading are chosen at random. The following data is for a hinged arch subjected to a partial distributed load. The arch is represented by 40 straight members. The optimum design for this arch is at an optimum h/L ratio = 0.20 and using a 15-mm thick web plate.

The optimum depth determined using formulas derived in chapter 3 are shown in Table B.1 . As the arch is symmetrical, only points 1 to 21, are tabulated. It is observed from the table that the (d/t) constraint controls the depth at points 4 to 10 and 17 to 21. Thus the optimum proportioning of the optimum box section ( Eq. 3.25 ) is not reached. Instead of optimum $A_w/A_t$ equals to 0.75, only a global 0.68 ratio is obtained. Table B.2 shows the calculated element properties at optimum design. While the depth is allowed to vary within an element, a uniform flange area is to be used. Thus the maximum of $Af_i$ and $Af_{i+1}$ is used for an element i. Tables B.1 and B.2 are summarized as follows :

| | | | |
|---|---|---|---|
| Total Flange Weight | = | 67.5 | tons |
| Total Web Weight | = | 144 | tons |
| Total Arch Weight | = | 212 | tons |

TABLE B.1 : Calculated Optimum Depth.

| Point No. | Required Depth (cm) | Required Af ($cm^2$) | Notes |
|---|---|---|---|
| 1 | 81 | 40.5 | |
| 2 | 119 | 59.5 | |
| 3 | 140 | 70.1 | |
| 4 | 150 | 83.5 | d/t controls |
| 5 | 150 | 102 | d/t controls |
| 6 | 150 | 115 | d/t controls |
| 7 | 150 | 120 | d/t controls |
| 8 | 150 | 119 | d/t controls |
| 9 | 150 | 111 | d/t controls |
| 10 | 150 | 96.5 | d/t controls |
| 11 | 150 | 74.8 | d/t controls |
| 12 | 133 | 66.6 | |
| 13 | 108 | 54.0 | |
| 14 | 99 | 49.5 | |
| 15 | 130 | 65.0 | |
| 16 | 149 | 74.4 | |
| 17 | 150 | 99.6 | d/t controls |
| 18 | 150 | 119 | d/t controls |
| 19 | 150 | 133 | d/t controls |
| 20 | 150 | 141 | d/t controls |
| 21 | 150 | 143 | d/t controls |

TABLE B.2 : Calculated Element Properties at Optimum Design.

| Element No. | $d_1$ (cm) | $d_2$ (cm) | Af (cm2) |
|---|---|---|---|
| 1 | 81 | 119 | 59.5 |
| 2 | 119 | 140 | 70.1 |
| 3 | 140 | 150 | 83.5 |
| 4 | 150 | 150 | 103 |
| 5 | 150 | 150 | 115 |
| 6 | 150 | 150 | 120 |
| 7 | 150 | 150 | 120 |
| 8 | 150 | 150 | 119 |
| 9 | 150 | 150 | 111 |
| 10 | 150 | 150 | 96.6 |
| 11 | 150 | 133 | 74.8 |
| 12 | 133 | 108 | 66.7 |
| 13 | 108 | 99.0 | 54.0 |
| 14 | 99 | 130 | 65.0 |
| 15 | 130 | 149 | 74.4 |
| 16 | 149 | 150 | 100 |
| 17 | 150 | 150 | 119 |
| 18 | 150 | 150 | 133 |
| 19 | 150 | 150 | 141 |
| 20 | 150 | 150 | 143 |

The calculated depth profile is shown in Figure B.1. It is observed from the figure that the calculated depth is irregular. The figure shows also the smoothened depth profile. It is a parabola whose equation is as follows :

$$(d/2)_{smooth} = 0.6206 + 0.009563\ S - 0.0002177\ S^2 \qquad \text{(A.1)}$$

Table B.3 shows smoothened depth profile and corresponding flange areas. As the table shows, flanges are not required theoretically for the AISC interaction equation at points 1 and 14. However, as per the constrain $g_4$, a minimum flange area of ($10\%\ t_w\ d$) is used. Table A.4 shows element properties for the smooth depths. The result of tables B.3 and B.4 are summarized as follows :

| | | | |
|---|---|---|---|
| Total Flange Weight | = | 75.1 | tons |
| Total Web Weight | = | 143 | tons |
| Total Arch Weight | = | 218 | tons |

Comparing these results with the previous :

$$\%\ \text{Increase in Weight due to Curve Smoothening} = \frac{218 - 212}{212}\ 100$$

$$= 2.8\ \%$$

It shall be noted that this percentage is not constant. It varies with different design examples. However both the d/t and the minimum Af constraints play an important role in calculating the optimum depth and smoothening the depth profile, in various examples.

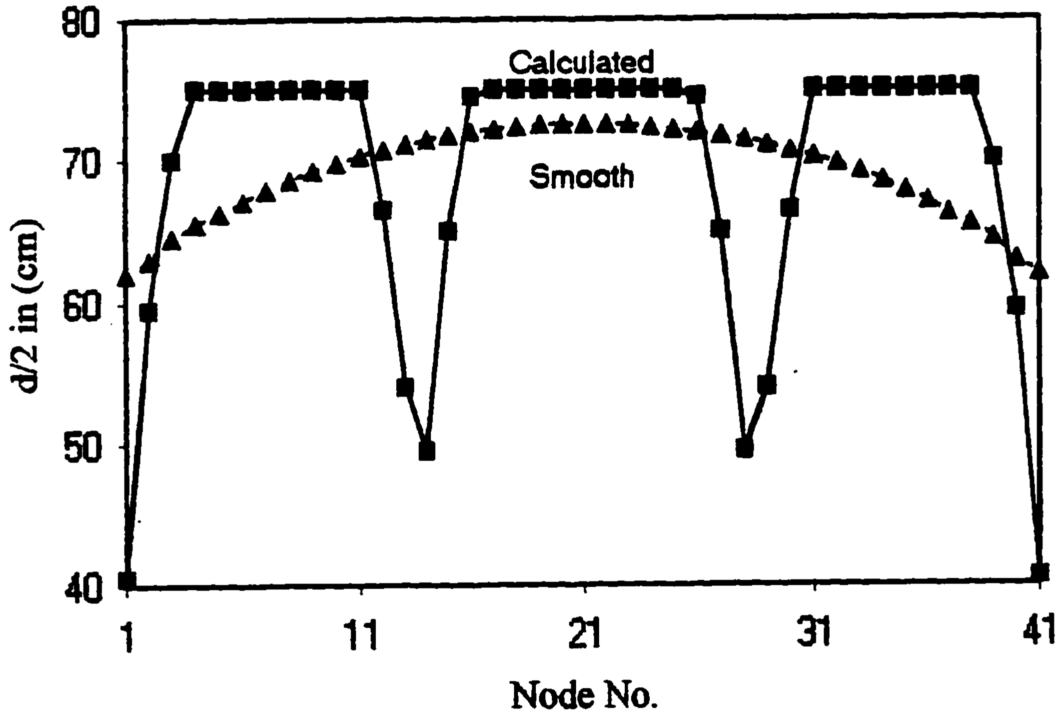



Figure B.1 : Plot of Calculated and Smoothed Depth Profile versus Node No. .

TABLE B.3 : Smooth Depth Profile and Corresponding Flange Areas.

| Point No. | Smooth Depth (cm) | Required Af ($cm^2$) | Engineering Af ($cm^2$) |
|---|---|---|---|
| 1 | 124 | not required | 18.6 |
| 2 | 126 | 48.6 | 48.6 |
| 3 | 129 | 88.0 | 88.0 |
| 4 | 131 | 115 | 115 |
| 5 | 132 | 133 | 133 |
| 6 | 134 | 143 | 143 |
| 7 | 136 | 146 | 146 |
| 8 | 137 | 142 | 142 |
| 9 | 138 | 131 | 131 |
| 10 | 140 | 114 | 114 |
| 11 | 141 | 88.7 | 88.7 |
| 12 | 141 | 54.7 | 54.7 |
| 13 | 142 | 6.52 | 21.3 |
| 14 | 143 | not required | 21.4 |
| 15 | 144 | 45.5 | 45.5 |
| 16 | 144 | 81.8 | 81.8 |
| 17 | 144 | 109 | 109 |
| 18 | 145 | 128 | 128 |
| 19 | 145 | 142 | 142 |
| 20 | 145 | 150 | 150 |
| 21 | 145 | 152 | 152 |

TABLE B.4 : Element Properties for Smooth Depth Profile.

| Element No. | $d_1$ (cm) | $d_2$ (cm) | Af (cm2) |
|---|---|---|---|
| 1 | 124 | 126 | 48.6 |
| 2 | 126 | 129 | 88.0 |
| 3 | 129 | 131 | 115 |
| 4 | 131 | 132 | 133 |
| 5 | 132 | 134 | 143 |
| 6 | 134 | 136 | 146 |
| 7 | 136 | 137 | 146 |
| 8 | 137 | 138 | 142 |
| 9 | 138 | 140 | 131 |
| 10 | 140 | 141 | 119 |
| 11 | 141 | 141 | 88.7 |
| 12 | 141 | 142 | 54.7 |
| 13 | 142 | 143 | 21.4 |
| 14 | 143 | 144 | 45.5 |
| 15 | 144 | 144 | 81.8 |
| 16 | 144 | 144 | 109 |
| 17 | 144 | 145 | 128 |
| 18 | 145 | 145 | 142 |
| 19 | 145 | 145 | 150 |
| 20 | 145 | 145 | 152 |

# APPENDIX C

## Curtailment of Flange Plates Using Dynamic Program :

The Dynamic programming, investigated in section 4.5, is illustrated through an example. The geometry of this example and the loadings are chosen at random. The following data is for a hinged arch subjected to a partial distributed load. The arch is represented by 20 straight members. The optimum design for this arch is at h/L equals to 0.20 and a 12-mm web thickness. The preassigned number of cut off points is 3. The calculated flange areas are shown in Table C.1 . The construction of such an arch requires the use of 20 flange plates welded together, which is impractical. Thus it is required to use dynamic programming to determine the optimum subdivision of S , into three subdivisions $S_1$ , $S_2$ and $S_3$, that minimize the total flange weight ( Eq. 4.6)

The DP model solve the problem in **stages**. For this problem, three stages existed $S_1$ , $S_2$ and $S_3$ These stages are interdependent because all three stages must compete for total arch length. The basic idea of DP is practically to eliminate the effect of interdependence between stages by associating a *state* definition with each stage. A **state** is normally defined to reflect the status of the constraints that bind all stages together. Thus in our example we define the state for stages 1,2 and 3 as follows :

State $k_1$ : the total length assigned to stage 1

State $k_2$ : the total length assigned to stages 1 and 2

State $k_3$ : the total length assigned to stages 1, 2 and 3.

TABLE C.1 : Calculated Flange Areas

| Element No. | Element Length<br>m | Cumulative Length<br>m | Flange Area<br>$m^2$ |
|---|---|---|---|
| 1 | 3.85089135 | 3.85089135 | 4.68709320E-03 |
| 2 | 3.55942297 | 7.41031456 | 6.52101263E-03 |
| 3 | 3.27865863 | 10.68897340 | 6.53968193E-03 |
| 4 | 3.01159477 | 13.70056820 | 6.53968193E-03 |
| 5 | 2.76220703 | 16.46277620 | 5.27188648E-03 |
| 6 | 2.53571630 | 18.99849320 | 2.84088985E-03 |
| 7 | 2.33878493 | 21.33727840 | 7.99999980E-05 |
| 8 | 2.17943907 | 23.51671790 | 3.72228911E-03 |
| 9 | 2.06639409 | 25.58311270 | 5.88338356E-03 |
| 10 | 2.00748563 | 27.59059910 | 6.53923629E-03 |
| 11 | 2.00748563 | 29.59808540 | 6.53923629E-03 |
| 12 | 2.06639409 | 31.66448020 | 5.88322617E-03 |
| 13 | 2.17943907 | 33.84391780 | 3.72214476E-03 |
| 14 | 2.33878493 | 36.18270110 | 7.99999980E-05 |
| 15 | 2.53571630 | 38.71841810 | 2.84102443E-03 |
| 16 | 2.76220703 | 41.48062520 | 5.27196703E-03 |
| 17 | 3.01159477 | 44.49221800 | 6.53972430E-03 |
| 18 | 3.27865863 | 47.77087780 | 6.53972430E-03 |
| 19 | 3.55942297 | 51.33029940 | 6.52101077E-03 |
| 20 | 3.85089135 | 55.18119050 | 4.68697352E-03 |

The value of $k_1$ and $k_2$ are not known exactly, but must lie between 0 and 55.18 (meters). On the other hand $k_3$ which is the total length in this case must be 55.18. Yet, as the flange area is defined over 20 elements. The total length must be a discrete value. Thus an **alternative** or a multiplier ik that ranges between o and 20 is to be used.

The solution start with stage 1, given a specified ik value (ik=0,1,2......20) what would be the best alternative (proposal for 1) that fulfill Eq. 4.7. Table C.2 answer this. For example row number 4 stands for ik=3, that is first three elements forms stage 1. From Eq. 4.7, the maximum flange area for elements 1,2,3 are used. That is Af equal 4.667e-3 $m^2$. The total flange volume for this proposal is 1.8e-2 $m^2$. Similarly, all alternatives from 0 to 20 are tabulated where 20 stands for uniform Af in this table.

Now consider stage 2 calculations. As State $k_2$ defines the total length assigned to stages 1 and 2, the best alternative for stage no 2 can be defined as :

$$\begin{pmatrix} \text{Best proposal} \\ \text{for stages 1\&2} \\ \text{given state k2} \end{pmatrix} = \text{max. all feasible} \left[ \begin{pmatrix} \text{proposal} \\ \text{for stage 2} \end{pmatrix} + \begin{pmatrix} \text{proposal} \\ \text{for stage 1} \end{pmatrix} \right[ .$$

Table C.3 shows stage 2. Similarly, stage 3, shown in table C.4, is defined by :

$$\begin{pmatrix} \text{Best proposal} \\ \text{for stages 1,2\&3} \\ \text{given state k3} \end{pmatrix} = \text{max. all feasible} \left[ \begin{pmatrix} \text{proposal} \\ \text{for stage 3} \end{pmatrix} + \begin{pmatrix} \text{proposal} \\ \text{for stage 2} \end{pmatrix} \right[ .$$

TABLE C.2 : Stage 1

| i | ik(i,1) | AF(i,1) $m^2$ | VOLF(i,1) $m^3$ |
|---|---|---|---|
| 0 | 0 | 0.00000000E-01 | 0.00000000E-01 |
| 1 | 1 | 4.68709320E-03 | 1.80494860E-02 |
| 2 | 2 | 6.52101263E-03 | 4.83227558E-02 |
| 3 | 3 | 6.53968193E-03 | 6.99024871E-02 |
| 4 | 4 | 6.53968193E-03 | 8.95973593E-02 |
| 5 | 5 | 6.53968193E-03 | 0.10766132 |
| 6 | 6 | 6.53968193E-03 | 0.12424410 |
| 7 | 7 | 6.53968193E-03 | 0.13953902 |
| 8 | 8 | 6.53968193E-03 | 0.15379186 |
| 9 | 9 | 6.53968193E-03 | 0.16730542 |
| 10 | 10 | 6.53968193E-03 | 0.18043374 |
| 11 | 11 | 6.53968193E-03 | 0.19356206 |
| 12 | 12 | 6.53968193E-03 | 0.20707563 |
| 13 | 13 | 6.53968193E-03 | 0.22132845 |
| 14 | 14 | 6.53968193E-03 | 0.23662336 |
| 15 | 15 | 6.53968193E-03 | 0.25320613 |
| 16 | 16 | 6.53968193E-03 | 0.27127010 |
| 17 | 17 | 6.53972430E-03 | 0.29096684 |
| 18 | 18 | 6.53972430E-03 | 0.31240836 |
| 19 | 19 | 6.53972430E-03 | 0.33568600 |
| 20 | 20 | 6.53972430E-03 | 0.36086977 |

TABLE C.3 : Stage 2

| i | ik(i,1) | AF(i,1) $m^2$ | VOLF(i,1) $m^3$ |
|---|---|---|---|
| 0 | 0 | 0.00000000E-01 | 0.00000000E-01 |
| 1 | 1 | 4.68709320E-03 | 1.80494860E-02 |
| 2 | 1 | 6.52101263E-03 | 4.12605293E-02 |
| 3 | 2 | 6.53968193E-03 | 6.27683699E-02 |
| 4 | 3 | 6.53968193E-03 | 8.24632421E-02 |
| 5 | 4 | 6.53968193E-03 | 0.10052720 |
| 6 | 1 | 2.84088985E-03 | 0.11486501 |
| 7 | 2 | 2.84088985E-03 | 0.12150925 |
| 8 | 3 | 3.72228911E-03 | 0.13391814 |
| 9 | 5 | 5.88338356E-03 | 0.15950693 |
| 10 | 9 | 6.53968193E-03 | 0.17329963 |
| 11 | 10 | 6.53968193E-03 | 0.18642795 |
| 12 | 11 | 6.53968193E-03 | 0.19994152 |
| 13 | 12 | 6.53968193E-03 | 0.21419434 |
| 14 | 1 | 7.99999980E-05 | 0.22151555 |
| 15 | 3 | 3.72214476E-03 | 0.23333140 |
| 16 | 4 | 5.27196703E-03 | 0.25882602 |
| 17 | 16 | 6.53972430E-03 | 0.28383255 |
| 18 | 17 | 6.53972430E-03 | 0.30527410 |
| 19 | 18 | 6.53972430E-03 | 0.32855174 |
| 20 | 1 | 4.68697352E-03 | 0.35373503 |

TABLE C.4 : Stage 3

| i | ik(i,1) | AF(i,1) $m^2$ | VOLF(i,1) $m^3$ |
|---|---|---|---|
| 0 | 0 | 0.00000000E-01 | 0.00000000E-01 |
| 1 | 1 | 4.68709320E-03 | 1.80494860E-02 |
| 2 | 1 | 6.52101263E-03 | 4.12605293E-02 |
| 3 | 1 | 6.53968193E-03 | 6.27019182E-02 |
| 4 | 2 | 6.53968193E-03 | 8.23967904E-02 |
| 5 | 1 | 5.27188648E-03 | 9.70252901E-02 |
| 6 | 1 | 2.84088985E-03 | 0.10773090 |
| 7 | 2 | 2.84088985E-03 | 0.11437513 |
| 8 | 3 | 3.72228911E-03 | 0.12678401 |
| 9 | 1 | 5.88338356E-03 | 0.14607553 |
| 10 | 2 | 6.53923629E-03 | 0.16055821 |
| 11 | 3 | 6.53923629E-03 | 0.17368564 |
| 12 | 4 | 6.53923629E-03 | 0.18719828 |
| 13 | 5 | 6.53923629E-03 | 0.20145014 |
| 14 | 1 | 7.99999980E-05 | 0.21438144 |
| 15 | 3 | 3.72214476E-03 | 0.22619729 |
| 16 | 1 | 5.27196703E-03 | 0.24789366 |
| 17 | 2 | 6.53972430E-03 | 0.27109045 |
| 18 | 3 | 6.53972430E-03 | 0.29253200 |
| 19 | 4 | 6.53972430E-03 | 0.31580964 |
| 20 | 5 | 6.53972430E-03 | 0.34099340 |

The final curtailment of flange plates is shown in Table C.5. Figures 5.5, 5.14, 5.19 and 5.28 are all examples of dynamic programming.

TABLE C.5 : Resulted Dynamic Flange Areas

| Element No. | Element Length m | Cumulative Length m | Flange Area $m^3$ |
|---|---|---|---|
| 1 | 3.85089135 | 4.68709320E-03 | 7.99999980E-05 |
| 2 | 7.41031456 | 6.52101263E-03 | 4.68709320E-03 |
| 3 | 10.68897340 | 6.53968193E-03 | 6.52101263E-03 |
| 4 | 13.70056820 | 6.53968193E-03 | 6.53968193E-03 |
| 5 | 16.46277620 | 5.27188648E-03 | 5.27188648E-03 |
| 6 | 18.99849320 | 2.84088985E-03 | 2.84088985E-03 |
| 7 | 21.33727840 | 7.99999980E-05 | 7.99999980E-05 |
| 8 | 23.51671790 | 3.72228911E-03 | 7.99999980E-05 |
| 9 | 25.58311270 | 5.88338356E-03 | 3.72228911E-03 |
| 10 | 27.59059910 | 6.53923629E-03 | 5.88338356E-03 |
| 11 | 29.59808540 | 6.53923629E-03 | 6.53923629E-03 |
| 12 | 31.66448020 | 5.88322617E-03 | 5.88322617E-03 |
| 13 | 33.84391780 | 3.72214476E-03 | 3.72214476E-03 |
| 14 | 36.18270110 | 7.99999980E-05 | 7.99999980E-05 |
| 15 | 38.71841810 | 2.84102443E-03 | 7.99999980E-05 |
| 16 | 41.48062520 | 5.27196703E-03 | 2.84102443E-03 |
| 17 | 44.49221800 | 6.53972430E-03 | 5.27196703E-03 |
| 18 | 47.77087780 | 6.53972430E-03 | 6.53972430E-03 |
| 19 | 51.33029940 | 6.52101077E-03 | 6.52101077E-03 |
| 20 | 55.18119050 | 4.68697352E-03 | 4.68697352E-03 |

# APPENDIX D

# PROGRAM LISTING

```
C                              IN THE NAME OF ALLAH
C                              ~~~~~~~~~~~~~~~~~~~~
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      COMMON/FEM3/XDCAL(25,51),XAF(25,51),RVOL(51,2),IRISE,SRISE

      CALL DATA
      CALL SOPSAR
      CALL RESULT
      STOP
      END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```
C-------------------------------------------------------------------
C                              SUBROUTINE DATA
C                              ~~~~~~~~~~~~~~~
C      1) TITLE
C
C      2) CONTROL PARAMETERS
C               SPAN   : TOTAL ARCH SPAN
C               RISE   : ARCH RISE
C               NELEM  : TOTAL NUMBER OF ELEMENTS
C               NSTAGE : NUMBER OF STAGES
C
C      3) UNIFORM LIVE LOAD
C               ULL(I) : UNIFORM LIVE LOAD MATRIX
C
C      4) CONCENTRIC  LOAD
C         RLOAD(I,IDOF) : APPLIED NODAL CONCENTRIC LOAD MATRIX
C
C      5) PRESCRIBED NODAL DISPLACEMENT
C           IFPRE(3*I) : 1 FOR RESTRAINED ACTION, 0 FOR FREE MOVEMENT
C           FIXED(3*I) : 0.0, OR PRESCRIBED SUPPORT SETTLEMENT VALUE
C
C      6) MATERIAL CONSTANTS
C               YOUNG : YOUNG'S MODULUS
C                FY   : YEILD STRENGT
C               WEBT  : WEB THICKNESS
C                UW   : UNIT WEIGHT
C
C      7) DEFAULT SECTION PROPERTIES {PROPS}  :
C                    PROPS(I,1)= FLANGE AREA
C                    PROPS(I,2)= DEPTH OF SECTION
C                    PROPS(I,3)= CROSS SECTION AREA
C                    PROPS(I,4)= MOMENT OF INERTIA
C                    PROPS(I,5)= SECTION MODULUS
```

```
C-----------------------------------------------------------------------
      SUBROUTINE DATA
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      COMMON/FEM3/XDCAL(25,51),XAF(25,51),RVOL(51,2),IRISE,SRISE
      DIMENSION ICODE(3),PRESC(3),TITLE(20)

C Input & output file
C-------------------
      open(10,file='POPSAR.in',status='old')
      open(11,file='POPSAR.out',status='old')
      open(9,file='ANALYSIS.out',status='old')
      open(8,file='optimum.out',status='old')
      open(7,file='SMOOTH.out',status='old')
      open(6,file='dynamic.out',status='old')

C READ & WRITE DATA
C-----------------
      READ(10,91)TITLE
      WRITE(11,91)TITLE
      WRITE(9,91)TITLE
      WRITE(8,91)TITLE
      WRITE(7,91)TITLE
      WRITE(6,91)TITLE

      read(10,91)TITLE
      WRITE(11,91)TITLE
      WRITE(9,91)TITLE
      WRITE(8,91)TITLE
      WRITE(7,91)TITLE
      WRITE(6,91)TITLE
   91 FORMAT(20A4)

C READ & WRITE CONTROL PARAMETER
C------------------------------
      READ(10,91)TITLE
      READ(10,*)SPAN,NELEM,NSTAGE
      WRITE(9,91)TITLE
      WRITE(9,*)SPAN,NELEM,NSTAGE
      IF(NELEM.GE.24) STOP
      NPOIN= NELEM + 1
      NSVAB= 3 * NPOIN
      READ(10,91)TITLE
      READ(10,*)RISE,RISEM,NRISES,SRISE
      WRITE(9,91)TITLE
      WRITE(9,*)RISE,RISEM,NRISES,SRIZE

C READ & WRITE UNIFORM LIVE LOAD
```

```
C------------------------------
      DO 10 I=1,NPOIN-1
 10   ULL(I)=0.0
      READ(10,91)TITLE
      READ(10,*)UL,ISPON,IEPON
      WRITE(9,*)'ELEMENT NO.        UNIFORM LIVE LOAD'
      IEPO=IEPON-1
      DO 20 I=ISPON,IEPO
      ULL(I)=UL
      WRITE(9,*)I,ULL(I)
  20  CONTINUE

C READ & WRITE NODAL APPLIED LOADS
C------------------------------
      READ(10,91)TITLE
      WRITE(9,99)
   99 FORMAT(1H0,5X,4HNODE,7X,5HLOADS)
      DO 60 IPOIN=1,NPOIN
      DO 60 IDOFN=1,3
   60 RLOAD(IPOIN,IDOFN)=0.0
   70 READ(10,*)IPOIN,(RLOAD(IPOIN,IDOFN),IDOFN=1,3)
      WRITE(9,93)IPOIN,(RLOAD(IPOIN,IDOFN),IDOFN=1,3)
      IF(IPOIN.LT.NPOIN) GO TO 70
 93   FORMAT(I10,3F15.5)

C READ & WRITE PRESCRIBED NODAL DISPLACEMENT
C------------------------------------------
      DO 40 ISVAB=1,NSVAB
      IFPRE(ISVAB)=0
   40 FIXED(ISVAB)=0.0
      READ(10,91)TITLE
      WRITE(9,98)
   98 FORMAT(1H0,1X,28HRESTRAINED NODES,FIXITY CODE,
     .22H AND PRESCRIBED VALUES)
      DO 50 IBOUN=1,2
      READ(10,*)NODFX,(ICODE(IDOFN),PRESC(IDOFN),IDOFN=1,3)
      WRITE(9,94) NODFX,(ICODE(IDOFN),PRESC(IDOFN),IDOFN=1,3)
   94 FORMAT(I3,3(I7,F13.3))
      DO 50 IDOFN=1,3
      INDEX=(NODFX-1)*3+IDOFN
      IFPRE(INDEX)=ICODE(IDOFN)
   50 FIXED(INDEX)=PRESC(IDOFN)

C READ & WRITE MATERIAL CONSTANTS
C------------------------------
      READ(10,91)TITLE
      READ(10,*)YOUNG,FY,FA,WEBT,UW
      WRITE(9,91)TITLE
      WRITE(9,*)YOUNG,FY,FA,WEBT,UW
      WRITE(9,*)
      WRITE(9,*)'end of output data'
      RETURN
      END
```

```
NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

C-----------------------------------------------------------------------
C                          SUBROUTINE SOPSAR
C                          ~~~~~~~~~~~~~~~~~
C
C FUNCTION :
C     * TO  OPTIMIZE THE ARCH
C-----------------------------------------------------------------------
      SUBROUTINE SOPSAR
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      COMMON/FEM3/XDCAL(25,51),XAF(25,51),RVOL(51,2),IRISE,SRISE

      IRISE=1
      CALL GEOM
      CALL SODESH
      IF (RISE.NE.RISEM) THEN
            XDIFF=(RISE-RISEM)/(NRISES-1)
            DO 20 IRISE=2,NRISES
                  RISE=RISE-XDIFF
                  CALL GEOM
                  CALL SODESH
20            CONTINUE
       ENDIF

      NRISES = NRISES + 1
      IRISE  = NRISES
      RISE   = SRISE
      CALL GEOM
      CALL SODESH

      RETURN
      END

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

C-----------------------------------------------------------------------
C                          SUBROUTINE RESULT
C                          ~~~~~~~~~~~~~~~~~
C
C-----------------------------------------------------------------------
      SUBROUTINE RESULT
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
```

```
     .              XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .              ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .              CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      COMMON/FEM3/XDCAL(25,51),XAF(25,51),RVOL(51,2),IRISE,SRISE

      WRITE(11,*)
      WRITE(11,*)' ---------------------------------------------------'
      WRITE(11,*)'   RISE/SPAN          WEIGHT           VOL/ OPT.RISE   '
      WRITE(11,*)'   -------------------------------------------------'
      DO 237 I=1,NRISES
         WT    = 78.5 * RVOL(I,2)
         SRRAT = RVOL(I,1)/SPAN
         OPRAT = RVOL(I,2)/RVOL(IRISE,2)
         WRITE(11,*)SRRAT,WT,OPRAT
237   CONTINUE

      WRITE(11,*)
      WRITE(11,*)' ---------------------------------------------------'
      WRITE(11,*)'   SPAN     RISE      RISE/SPAN        OPTIMUM VOLUME'
      WRITE(11,*)'   -------------------------------------------------'
      DO 1237 I=1,NRISES
         WRITE(11,*)SPAN,RVOL(I,1),RVOL(I,1)/SPAN,RVOL(I,2)
1237   CONTINUE
      JRISE =NRISES
      DO 257 J=JRISE,NRISES
      WRITE(11,*)
      WRITE(11,*)'   RISE/SPAN    =    ',RVOL(J,1)/SPAN
      WRITE(11,*)
      WRITE(11,*)'----------------------------------------------'
      WRITE(11,*)'   NODE NO              DEPTH              DEPTH/Dc'
      WRITE(11,*)'----------------------------------------------'
      DO 257 I=1,NPOIN
         WRITE(11,*)I,XDCAL(I,J),XDCAL(I,J)/XDCAL(((NPOIN+1)/2),J)
257   CONTINUE

      DO 267 J=JRISE,NRISES
      WRITE(11,*)
      WRITE(11,*)'   RISE/SPAN    =    ',RVOL(J,1)/SPAN
      WRITE(11,*)
      WRITE(11,*)'-----------------------------------------------'
      WRITE(11,*)'   ELEMENT NO                     AF           AF/AFc '
      WRITE(11,*)'-----------------------------------------------'
      DO 267 I=1,NELEM
         WRITE(11,*)I,XAF(I,J),XAF(I,J)/XAF((NELEM/2),J)
267   CONTINUE
      RETURN
      END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```

C-----------------------------------------------------------------
```

```
C                           SUBROUTINE GEOM
C                           ~~~~~~~~~~~~~~~
C FUNCTION :
C            1) TO CALC X,Y  COORDINATES : {COORD}
C            2) TO WRITE
C                        PROPS(I,6)= X2 - X1
C                        PROPS(I,7)= Y2 - Y1
C                        PROPS(I,8)= ELEMENT LENGTH
C                        PROPS(I,9)= Cumulative ELEMENT LENGTH
C-----------------------------------------------------------------------------
         SUBROUTINE GEOM
         COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .              RISEM,FA,NRISES
         COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .              FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .              XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .              ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .              CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
         COMMON/FEM3/XDCAL(25,51),XAF(25,51),RVOL(51,2),IRISE,SRISE

         WRITE(9,191)
         WRITE(9,192)
         DO 110  IPOIN=1,NPOIN
         X=(IPOIN-1)*SPAN/NELEM
         COORD(IPOIN,1)=X
         COORD(IPOIN,2)=4*X*(SPAN-X)*(RISE)/(SPAN**2)
         WRITE(9,*)IPOIN,(COORD(IPOIN,J),J=1,2)
 110     CONTINUE

         WRITE(9,291)
         WRITE(9,292)
            cleng(1)=0.00000001
         DO 120 I=1,Nelem
             PROPS(I,6)= COORD(I+1,1)-COORD(I,1)
             PROPS(I,7)= COORD(I+1,2)-COORD(I,2)
             PROPS(I,8)= SQRT(PROPS(I,6)**2+PROPS(I,7)**2)
             if (i.EQ.1) then
             props(i,9)=props(i,8)
             ELSE
             props(i,9)=props(i,8) + props(i - 1,9)
             endif
             cleng(i+1)=cleng(i)+props(i,8)
             WRITE(9,*)I,PROPS(I,8),PROPS(I,9)
 120     CONTINUE
 191     FORMAT(//,'          NODE #        X- Coord.          Y- Coord.')
 192     FORMAT('          ------        ---------          ---------')
 291     FORMAT(//,'          ELEM #        LENGTH           CUML. LENGTH')
 292     FORMAT('          ------        ------         ------------')
         RETURN
         END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```
C----------------------------------------------------------------------
C                           SUBROUTINE SODESH
C                           ~~~~~~~~~~~~~~~~~
C FUNCTION :
C    * TO  OPTIMIZE THE ARCH FOR A SPECIFIED HEIGHT
C----------------------------------------------------------------------
      SUBROUTINE SODESH
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .               RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .               FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .               XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .               ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .               CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      COMMON/FEM3/XDCAL(25,51),XAF(25,51),RVOL(51,2),IRISE,SRISE


C INITIAL DESIGN
C--------------
      ITERS=0
      IZ=0
      DO 80 I=1,NELEM
              PROPS(I,2)= SPAN/40
              PROPS(I,1)= WEBT*PROPS(I,2)/3
              PROPS(I,3)= 8*WEBT*PROPS(I,2)/3
              PROPS(I,4)= (WEBT*PROPS(I,2)**3)/3
              PROPS(I,5)= (2*WEBT*PROPS(I,2)**2)/3
80    CONTINUE
      WRITE(9,*)'DEFAULT SECTION PROPS : AF , DEPTH , A , I , S'
      WRITE(9,97)(PROPS(1,J),J=1,5)
97    FORMAT(/,5E12.4)
      CALL anasys
      CALL OPTIMM
      CALL smooth
      CALL dynmic


C ITERATE
C--------
555   IZ=41
      DO 503 I=1,NELEM
         XDCAL(I,IRISE) =DCAL(I)
         XAF(I,IRISE)   =AF(I)
         PROPS(I,1)     =AF(I)
         PROPS(I,2)     =(DCAL(I)+DCAL(I+1))/2
         PROPS(I,3)     =2* AF(I)+2*PROPS(I,2)*WEBT
         PROPS(I,5)     =AF(I)*PROPS(I,2)+WEBT*(PROPS(I,2)**2)/3
         PROPS(I,4)     =PROPS(I,2)*PROPS(I,5)/2
503   CONTINUE
      XDCAL(NPOIN,IRISE) = DCAL(NPOIN)

         CALL anasys
```

```
            CALL OPTIMM
            CALL smooth
            CALL dynmic

        DO 504 I=1,NELEM
            X = ABS(AF(I)-XAF(I,IRISE))
            Y =0.5*(AF(I)+XAF(I,IRISE))
            Z = (X/Y) *100
            IF(Z.GT.1.000)THEN
                 WRITE(6,*)'FLANGE AREA IS CHANGED AT ELEM #',I
                 IZ=IZ-1
            ENDIF
504     CONTINUE


        DO 505 I=1,NPOIN
            X = abs(XDCAL(I,IRISE)-DCAL(I))
            Y =0.5*(XDCAL(I,IRISE)+DCAL(I))
            Z = (X/Y)*100
            IF(Z.GT.1.000)THEN
                 WRITE(7,*)'WEB DEPTH IS CHANGED AT POINT #',I
                 IZ=IZ-1
            ENDIF
505     CONTINUE

        IF((ITERS.LT.25).AND.(IZ.LT.41)) THEN
             iters=ITERS+1
             do 501 i=9,6,-1
                write(i,*)'*************************************************'
                write(i,*)'ANALYSIS & DESIGN ARE RECONDUCTED FOR',ITERS
501          CONTINUE
             GOTO 555
        ELSE
             WRITE(7,*)'41 CHECKS PEFORMED,OUT OF WHICH  ',IZ,'  ARE OK'
             WRITE(6,*)'41 CHECKS PEFORMED,OUT OF WHICH  ',IZ,'  ARE OK'
        ENDIF

        VOL =0.00
        DO 605 I=1,NELEM
           AREA = 2*XAF(I,IRISE) + WEBT*(XDCAL(I,IRISE)+XDCAL(I+1,IRISE))
           VOL  = VOL + AREA * PROPS(I,8)
605     CONTINUE
        RVOL(IRISE,1)=RISE
        RVOL(IRISE,2)=VOL
        RETURN
        END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```
c-----------------------------------------------------------------
C                          SUBROUTINE ANASYS
C                          ~~~~~~~~~~~~~~~~~
```

```
C FUNCTION :
C     * TO  CALC & WRITE  REQUIRED SECTION
C-----------------------------------------------------------------------
      SUBROUTINE ANASYS
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)

      CALL STIFF
      CALL LOADS
      CALL ASSEMB
      CALL GREDUC
      CALL BAKSUB
      CALL ENDFOR
      RETURN
      END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```
c-----------------------------------------------------------------------
C                          SUBROUTINE OPTIM
C                          ~~~~~~~~~~~~~~~~
C
C FUNCTION :
C     * TO  CALC & WRITE  REQUIRED SECTION
C-----------------------------------------------------------------------
      SUBROUTINE OPTIMM
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)

      real area(25)

C      Intializing
C      -----------
          c1=0.00
          c2=0.00
          c3=0.00
          bb=0.00
          cc=0.00
           d=0.00
     avgarea=0.00
      volume=0.00
        volw=0.00
        volf=0.00
```

```
              rat=0.00
          DO 111 IPOIN=1, NPOIN
             dcal(IPOIN)  = 0.00
               af(IPOIN)  = 0.00
             AREA(IPOIN)  = 0.00
 111      CONTINUE

          FB = 0.60*FY
          FV = 0.40*FY

C          Calculating Optimum Area
C          ------------------------
 1238      DO 555 IPOIN=1,NPOIN
             c1     = force(IPOIN,1)/(2*FA)
             C2     = FORCE(IPOIN,3) * 32 * WEBT / (3*FB)
             AREA(IPOIN) = C1 + SQRT ( C1**2 + C2)
             DCAL(IPOIN) = 3*AREA(IPOIN)/(8*WEBT)
             AF(ipoin)=AREA(IPOIN)/8
 555       CONTINUE

C          CHECKING FOR SHEAR & BUCKLING
C          -----------------------------
           DO 666 IPOIN=1,NPOIN
             SHEAR = 2*DCAL(iPOIN)*WEBT*FV
             IF (SHEAR.LT.FORCE(IPOIN,2))THEN
                DCAL(IPOIN) = force(IPOIN,2)/(2*WEBT*FV)
                WRITE(8,*) 'SHEAR CONTROLLS DEPTH AT NODE #',IPOIN
             ENDIF
             C3 = DCAL(IPOIN)/WEBT
             IF (C3.GT.100) THEN
                WRITE(8,*) 'LOCAL BUCKLING CONTROLLS DEPTH AT NODE #',IPOIN
                DCAL(IPOIN)= 100 * webt
             ENDIF

             IF ((SHEAR.LT.FORCE(IPOIN,2)).or.(c3.gt.100)) THEN
                d =dcal(ipoin)
                BB= 4*webt*d/3-force(ipoin,1)/(2*Fa)-force(ipoin,3)/(d*fb)
                CC=((webt*d)**2)/3-WEBT*d*fORCE(ipoin,1)/(6*Fa)
     .              -WEBT*force(ipoin,3)/Fb
                AF(ipoin)= (-BB+SQRT( BB*BB -4*CC ))/2
                area(ipoin)=2*webt*d + 2*AF(ipoin)
             endif
 666         CONTINUE

          WRITE(8,*) '      ------     ----------     -------     ----------'
          WRITE(8,*) '       PT NO:    REQ. DEPTH     REQ. Af     TOTAL AREA'
          DO 777 IPOIN=1, NPOIN
            IF (AF(IPOIN).GE.0.000) THEN
                WRITE(8,*) IPOIN,DCAL(IPOIN),AF(ipoin),AREA(IPOIN)
            ELSE
                AF(IPOIN)=0.00000
                WRITE(8,*) IPOIN,DCAL(IPOIN),'NOT REQ.',AREA(IPOIN)
            ENDIF
```

```
 777    CONTINUE

        VOLUME = 0.0
        DO 888 IELEM=1, NELEM
            AVGAREA =( AREA(IELEM) + AREA(IELEM+1) ) / 2
            VOLUME  = VOLUME + AVGAREA*PROPS(IELEM,8)
 888    CONTINUE
        WRITE(8,*) '--------------------------------------------------------'

        WRITE(8,*) '  TOTAL VOLUME     =     ',VOLUME,'m^3'

        volw=0.0
        volf=0.0
        write(8,*)
        WRITE(8,*) '--------------------------------------------------------'
        write(8,*)'ELEM #','   D1    ','   D2   ','  D-AVG  ',' AF-AVG'
        do 999 ielem=1,nelem
        davg=dcal(ielem)/2+dcal(ielem+1)/2
            if (af(ielem).ge.af(ielem+1)) then
                  flange=af(ielem)
            else
                  flange=af(ielem+1)
            endif

            write(8,101) ielem,dcal(ielem),dcal(ielem+1),davg,flange
            volw= volw + 2*( davg * webt ) * props(ielem,8)
            volf= volf + 2*( flange ) * props(ielem,8)
 999    CONTINUE

            vol =volw + volf
            rat = volf/vol
            WRITE(8,*) 'flange volume =   ',volf
            WRITE(8,*) 'web volume =   ',volw
            WRITE(8,*) 'flange vol/ total vol =   ',rat
            WRITE(8,*) 'total volume =   ',vol
 101    FORMAT(I5,4E12.4)

        WRITE(8,*) '--------------------------------------------------------'
        WRITE(8,*) 'RISE',RISE,'theoratical VOLUME     =    ',VOL
        RETURN
        END

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

C--------------------------------------------------------------------
C                         SUBROUTINE SMOOTH
C                         ~~~~~~~~~~~~~~~~~
C--------------------------------------------------------------------
        SUBROUTINE smooth
        COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
       .            RISEM,FA,NRISES
        COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
```

```
     .              FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .              XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .              ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .              CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
       dimension x(25),y(25),b(25)
       dimension a(25,25),coef(25),dsm(25),area(25),TDSM(25),TAF(25)

C      Intializing
C      -----------
           bb=0.00
           cc=0.00
            d=0.00
       redfcr=0.00
       avgare=0.00
        pivot=0.00
         pivr=0.00
         xxxx=0.00
          vol=0.00
      DO 113 IPOIN=1, NPOIN
            x(IPOIN)  = 0.00
            y(IPOIN)  = 0.00
          dsm(IPOIN)  = 0.00
         tdsm(IPOIN)  = 0.00
           af(IPOIN)  = 0.00
          taf(IPOIN)  = 0.00
         area(IPOIN)  = 0.00
113   CONTINUE

      FB = 0.60*FY
      FV = 0.40*FY
           WRITE(7,*) 'FA,FB,WEBT',FA,FB,WEBT
      do 2001 ipoin=1,npoin
                X(iPOIN)=cleng(ipoin)
      y(ipoin) = dcal(ipoin)/2.0
2001  continue
      m=2
      m1=2+1
      do 2030 i=1,m1
         b(i) = 0.0
         do 2020 j=1, m1
              a(i,j)=0.0
2020     continue
2030  continue

c  calculating matrix a
c  --------------------
       do 3060 ipoin=1,npoin
             do 3050 i=1,m1
                 b(i)=b(i)+y(ipoin)*(cleng(ipoin))**(i-1)
                 do 3040 j=1,m1
                    a(i,j)=a(i,j)+(cleng(ipoin))**(i+j-2)
3040               continue
3050           continue
```

```
3060   continue

c  matrix inversion
c  ---------------
       do 3076 i=1,m1
       pivot=a(i,i)
       pivr=1./pivot
       do 3077 ncol=1,m1
       a(i,ncol)=a(i,ncol)/pivot
3077   continue
       do 3078 k=1,m1
       if(k.ne.i)then
       pivot=a(k,i)
       do 3079 ncol=1,m1
       a(k,ncol)=a(k,ncol)-a(i,ncol)*pivot
3079   continue
       a(k,i)=-pivot*pivr
       endif
3078   continue
       a(i,i)=pivr
3076   continue

c      matrix multiplication
c      ---------------------
       do 3080 i=1,m1
          coef(i)=0.00
       do 3080 j=1,m1
           coef(i)=coef(i)+a(i,j)*b(j)
 3080  continue
       wRITE(7,*)
       wRITE(7,*)'EQUATION FOR SMOOTHEN (d/2)'
       wRITE(7,*)'---------------------------'
       wRITE(7,*)'(d/2) = ',COEF(1),'+',coef(2),' S ',coef(3),' s^2'
       WRITE(7,*)
       WRITE(7,*) 'MODIFYING SMOOTHEN CURVE'

C      ITERATIONS TO MODIFY SMOOTH CURVE PROFILE
C      -----------------------------------------
       REDFCR= 1.0
1390   XXXX=redfcr*COEF(1)
c   smooth depth calculation
c   -----------------------
       DO 3090 I=1,NPOIN
       TDsm(i)=XXXX
       do 3090 j=2,m1
       Tdsm(i)=Tdsm(i)+coef(j)*(x(i))**(j-1)
 3090  continue


      do 3200 ipoin=1,npoin
      d =2*Tdsm(ipoin)
      BB= 4*webt*d/3-force(ipoin,1)/(2*Fa)-force(ipoin,3)/(d*fb)
      CC=((webt*d)**2)/3-WEBT*d*fORCE(ipoin,1)/(6*Fa)
```

```
     .      -WEBT*force(ipoin,3)/Fb
        TAF(ipoin)= (-BB+SQRT( BB*BB -4*CC ))/2
       if(Taf(ipoin).lt.(0.05*Tdsm(ipoin)*webt)) then
             Taf(ipoin)=0.1*d*webt
       endif
        area(ipoin)=2*webt*d + 2*TAF(ipoin)
 3200  CONTINUE

       VOLUME = 0.0
       DO 878 IELEM=1, NELEM
           AVGARE  =( AREA(IELEM) + AREA(IELEM+1) ) / 2
           VOLUME  = VOLUME + AVGARE *PROPS(IELEM,8)
  878  CONTINUE

        IF(REDFCR.EQ.1.00) TVOL = VOLUME
        IF(TVOL.GE.VOLUME)THEN
        TVOL = VOLUME
        DO 140 IPOIN =1,NPOIN
               DSM(IPOIN) =  TDSM(IPOIN)
               AF(IPOIN)  =   TAF(IPOIN)
  140   CONTINUE
        REDFCR =REDFCR -.01
        WRITE(7,*) 'REDUCTION FACTOR =  ',REDFCR,volume
        GOTO 1390
        ENDIF

        wRITE(7,*)
        wRITE(7,*)'EQUATION FOR MODIFIED SMOOTHEN (d/2)'
        wRITE(7,*)'-----------------------------------'
        wRITE(7,*)'(d/2) = ',DSM(1),'+',coef(2),' S ',coef(3),' s^2'
        WRITE(7,*)
        WRITE(7,*)

       WRITE(7,*) '     ----   ------    ---------      ------------ '
       WRITE(7,*) '     NODE   LENGTH    CALC. D/2      Smoothen d/2 '
       WRITE(7,*) '     ----   -------   ---------      ------------ '
       do 33 ipoin=1,npoin
        WRITE(7,*) IPOIN,cleng(ipoin),DCAL(IPOIN)/2,DSM(iPOIN)
   33  CONTINUE
       WRITE(7,*) '---------------------------------------------------'

       WRITE(7,*) '     ------    ----------    -------     '
       WRITE(7,*) '     PT NO:    Smoothen d    ENG AF      '
       WRITE(7,*) '     ------    ----------    -------     '
       do 32 ipoin=1,npoin
        DCAL(IPOIN)=2*DSM(IPOIN)
        WRITE(7,*) IPOIN,DCAL(IPOIN),AF(ipoin)
   32  CONTINUE
       WRITE(7,*) '---------------------------------------------------'
       WRITE(7,*) 'TOTAL VOLUME  =   ',TVOL,'  m^3'

        return
        end
```

```
 NUMBER OF WARNINGS IN PROGRAM UNIT: 0
 NUMBER OF ERRORS   IN PROGRAM UNIT: 0



C-----------------------------------------------------------------------
C                           SUBROUTINE dynmic
C                           ~~~~~~~~~~~~~~~~~
C FUNCTION :
C          1) TO CALC dynamic flange area
C-----------------------------------------------------------------------
      SUBROUTINE dynmic
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)

       dimension TL(25),flange(25)
       dimension k(0:24,25),volF(0:24,25),SAF(0:24,25)


C      Intializing
C      -----------
           XX=0.00
           cc=0.00
      DO 115 IELEM=1, NELEM
           TL(IELEM)   = 0.00
       FLANGE(IPOIN)   = 0.00
 115  CONTINUE

      WRITE(6,*)'---------------------------------------------------'
      WRITE(6,*)'      Ielem,     CUM. LENGTH            flange AREA'
      WRITE(6,*)'---------------------------------------------------'
      do 231 ielem=1,nelem
          if (ielem.EQ.1) then
                TL(ielem)=props(ielem,8)
          ELSE
                TL(ielem)=props(ielem,8) + TL(ielem - 1 )
          endif

          if (af(ielem).ge.af(ielem+1)) then
                flange(ielem)=af(ielem)
          else
                flange(ielem)=af(ielem+1)
          endif
          xx = xx +props(ielem,8)*flange(ielem)
          WRITE(6,*)Ielem,tl(ielem),flange(ielem)
 231  CONTINUE
          write(6,*)'total discrete flange area =  ',xx
```

```
c   STAGE # 1
C   ----------
      DO 232 I=0,Nelem
          K(I,1)=I
          if (i.EQ.0) then
                SAF (i,1) =0.0
          ELSE
                IF(FLANGE(I).GT.SAF(I-1,1)) THEN
                    SAF(I,1) =flange(i)
                else
                    SAF(i,1) =SAF(I-1,1)
                endif
          endif
          VOLF(I,1)=SAF(I,1) * TL(i)
232   CONTINUE



c   INTERMEDIATE STAGES
C   -------------------
      DO 249 ISTAGE=2,NSTAGE
      DO 248 IX=0,NELEM
       COMPAR=1000
       DO 247 IK=0,IX
         IK1=IX-IK+1
         PVOL=VOLF(IX-IK,ISTAGE-1)
         SL   =TL(IX)-TL(IX-IK)
         if (iK.EQ.0) then
            TVOLK=PVOL
            TAF=0.0
         ELSE
            TAF=0.0
            DO 241 I=IK1,IX
                IF(FLANGE(I).GT.TAF) THEN
                    TAF=flange(i)
                endif
241         CONTINUE
            TVOLK=TAF*SL+PVOL
         endif
           IF(TVOLK.LE.COMPAR)THEN
              COMPAR=TVOLK
              K(IX,ISTAGE)=IK
              SAF(IX,ISTAGE)=TAF
              VOLF(IX,ISTAGE)=TVOLK
           ENDIF
247   CONTINUE
248   CONTINUE
249   CONTINUE

      KOUNT=NELEM
      DO 251 ISTAGE=NSTAGE,1,-1
           IF (ISTAGE.EQ.NSTAGE) THEN
```

```
                J=K(NELEM,ISTAGE)
                I=NELEM-J
            ELSE
                J=K(I,ISTAGE)
                I=I-J
            ENDIF

            DO 250 M=KOUNT,KOUNT-J,-1
                AF(M)=SAF(KOUNT,ISTAGE)
 250        CONTINUE
        KOUNT=KOUNT-J
 251    CONTINUE
        WRITE(6,*)'-------------------------------------------------------------'
        WRITE(6,*)'         NODE #   CUM. LENGTH        AF              DYN AF'
        WRITE(6,*)'-------------------------------------------------------------'
        DO 252 I=1,NELEM
            WRITE(6,*)I,TL(I),FLANGE(I),AF(I)

 252    CONTINUE
        WRITE(6,*)'FOR # OF STAGES = ',NSTAGE
        WRITE(6,*)'REQ FLANGE AREA = ',VOLF(NELEM,NSTAGE)
        VOLUME = 0.0
        DO 1878 IELEM=1, NELEM
                d   =( dcal(IELEM) + dcal(IELEM+1) ) / 2
            avgarea = 2*webt*d
            VOLUME  = VOLUME + 2*webt*d*PROPS(IELEM,8)
 1878    CONTINUE
            VOLUME  = VOLUME + 2*volf(nelem,nstage)
        WRITE(6,*) '  TOTAL VOLUME     =     ',VOLUME,'m^3'
        WRITE(6,*) 'RISE',RISE,'VOLUME',VOLUME
         return
         end
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```
C-----------------------------------------------------------------------
C                            SUBROUTINE STIFF
C                            ~~~~~~~~~~~~~~~~
C FUNCTION :
C          1) TO CALC & WRITE TRANSFORMATION MATRIX ON THESIS.TRNM
C          2) TO CALC & WRITE ELEMENT STIFFNESS MATRIX ON THESIS.ESTM
C          3) TO CALC & WRITE TRANSF.ELEMENT STIFF MATRIX ON THESIS.TSTM

C-----------------------------------------------------------------------
        SUBROUTINE STIFF
        DIMENSION ESTIF(6,6),TRANS(6,6),XMAT(6,6)
        COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .              RISEM,FA,NRISES
        COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .              FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .              XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .              ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
```

```
     .              CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)

      REWIND 1
      REWIND 2
      REWIND 3

      DO 400 IELEM=1,NELEM

         DO 310 IEVAB=1,6
         DO 310 JEVAB=1,6
               TRANS(IEVAB,JEVAB)=0.0
               ESTIF(IEVAB,JEVAB)=0.0
  310    CONTINUE

         XAREA=PROPS(IELEM,3)
         XINRT=PROPS(IELEM,4)
         ELENG=PROPS(IELEM,8)
         CX   =PROPS(IELEM,6)/ELENG
         CY   =PROPS(IELEM,7)/ELENG

C       TRANSFORMATION MATRIX & ELEMENT LENGTH
C---------------------------------------------------
        TRANS(1,1)=CX
        TRANS(1,2)=CY
        TRANS(2,1)=-CY
        TRANS(2,2)=CX
        TRANS(3,3)=1
        TRANS(4,4)=CX
        TRANS(4,5)=CY
        TRANS(5,4)=-CY
        TRANS(5,5)=CX
        TRANS(6,6)=1
        WRITE(1,*)((TRANS(I,J),J=1,6),I=1,6)

C       ELEMENT STIFFNESS MATRIX IN LOCAL COORDINATES
C---------------------------------------------------------
         ESTIF(1,1)=XAREA*YOUNG/ELENG
         ESTIF(2,2)=12.0*YOUNG*XINRT/ELENG**3
         ESTIF(3,3)=4.0*YOUNG*XINRT/ELENG
         ESTIF(4,4)=ESTIF(1,1)
         ESTIF(5,5)=ESTIF(2,2)
         ESTIF(6,6)=ESTIF(3,3)
         ESTIF(1,4)=-ESTIF(1,1)
         ESTIF(4,1)=-ESTIF(1,1)
         ESTIF(2,3)=6.0*YOUNG*XINRT/ELENG**2
         ESTIF(3,2)=6.0*YOUNG*XINRT/ELENG**2
         ESTIF(2,5)=-ESTIF(2,2)
         ESTIF(5,2)=-ESTIF(2,2)
         ESTIF(2,6)=ESTIF(2,3)
         ESTIF(6,2)=ESTIF(2,3)
         ESTIF(3,5)=-ESTIF(2,3)
         ESTIF(5,3)=-ESTIF(2,3)
         ESTIF(3,6)=ESTIF(3,3)/2.0
```

```
            ESTIF(6,3)=ESTIF(3,3)/2.0
            ESTIF(5,6)=-ESTIF(2,3)
            ESTIF(6,5)=-ESTIF(2,3)
            WRITE(2,*)((ESTIF(I,J),J=1,6),I=1,6)

C           TRANSFORMED ELEMENT STIFFNESS MATRIX
C----------------------------------------
            DO 370 IEVAB=1,6
            DO 370 JEVAB=1,6
               XMAT(IEVAB,JEVAB)=0
            DO 370 KEVAB=1,6
               XMAT(IEVAB,JEVAB)=XMAT(IEVAB,JEVAB)+TRANS(KEVAB,IEVAB)*
     .         ESTIF(KEVAB,JEVAB)
  370       CONTINUE

            DO 380 IEVAB=1,6
            DO 380 JEVAB=1,6
               ESTIF(IEVAB,JEVAB)=0
            DO 380 KEVAB=1,6
               ESTIF(IEVAB,JEVAB)=ESTIF(IEVAB,JEVAB)+XMAT(IEVAB,KEVAB)*
     .         TRANS(KEVAB,JEVAB)
  380       CONTINUE

            WRITE(3,*)((ESTIF(I,J),J=1,6),I=1,6)

  400   CONTINUE
        RETURN
        END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```

C---------------------------------------------------------------------
C                          SUBROUTINE LOADS
C                          ~~~~~~~~~~~~~~~~
C FUNCTION :
C     * TO  CALC & WRITE  EQUIVALENT ELEMENT LOAD DUE TO UDL ON TAPE 4
C     * TO CALC TRANSFORMED EQUIVALENT ELEMENT MATRIX {TLOAD}
C---------------------------------------------------------------------
        SUBROUTINE LOADS
        COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .             RISEM,FA,NRISES
        COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .             FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .             XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .             ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .             CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
        DIMENSION TRANS(6,6),XLOAD(6),TXLOD(6)
        REWIND 1
        REWIND 4
        NNODE=2
        DO 20 IPOIN=1,NPOIN
```

```
      DO 20 IDOFN=1,3
20    TLOAD(IPOIN,IDOFN)=0.0
      DO 400 IELEM=1,NELEM
      DO 30 IEVAB=1,6
      XLOAD(IEVAB)=0.0
30    CONTINUE

C UNIFORM DEAD LOAD MATRIX
C------------------------
        P    = ULL(IELEM)
        W    = UW * PROPS(IELEM,3)
       ELENG= PROPS(IELEM,8)
        CX   = PROPS(IELEM,6)/ELENG
        CY   = PROPS(IELEM,7)/ELENG
        UL   = W + P*CX
       READ (1,*)((TRANS(I,J),J=1,6),I=1,6)

      XLOAD(1)=-ELENG*CY*UL/2
      XLOAD(2)=-ELENG*CX*UL/2
      XLOAD(3)=-ELENG*ELENG*CX*UL/12
      XLOAD(4)=-ELENG*CY*UL/2
      XLOAD(5)=-ELENG*CX*UL/2
      XLOAD(6)=+ELENG*ELENG*CX*UL/12
      WRITE(4,*)(XLOAD(I),I=1,6)

      DO 100 IEVAB=1,6
      TXLOD(IEVAB)=0
      DO 100 JEVAB=1,6
100   TXLOD(IEVAB)=TXLOD(IEVAB)+TRANS(JEVAB,IEVAB)*XLOAD(JEVAB)

C TRANSFORMED EQUIVLANT LOAD ELEMENT MATRIX
C----------------------------------------
      ICOUNT=0
      DO 40 INODE=1,2
      LNODE=IELEM+INODE-1
      DO 40 IDOFN=1,3
      ICOUNT=ICOUNT+1
      TLOAD(LNODE,IDOFN)=TLOAD(LNODE,IDOFN)+TXLOD(ICOUNT)
40    CONTINUE
400   CONTINUE
      RETURN
      END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```
C-----------------------------------------------------------------
C                          SUBROUTINE ASSEMB
C                          ~~~~~~~~~~~~~~~~~
C FUNCTION :
C            1) TO ASSEMBLE THE  LOAD MATRIX {ASLOD}
C            2) TO ASSEMBLE THE  STIFFNESS MATRIX {ASTIF}
C-----------------------------------------------------------------
```

```
      SUBROUTINE ASSEMB
      DIMENSION ESTIF(6,6)
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .               RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .               FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .               XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .               ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .               CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)

      REWIND 3
      DO 10 ISVAB=1,NSVAB
      ASLOD(ISVAB)=0.0
      DO 10 JSVAB=1,NSVAB
      ASTIF(ISVAB,JSVAB)=0.0
   10 CONTINUE

C ASSEMBLE THE ELEMENT LOADS
C -------------------------
      DO 15 IPOIN=1,NPOIN
      DO 15 IDOFN=1,3
      NROWS=(IPOIN-1)*3+IDOFN
   15 ASLOD(NROWS)=ASLOD(NROWS)+RLOAD(IPOIN,IDOFN)+TLOAD(IPOIN,IDOFN)
C      WRITE(9,*)'ASSEBLED LOAD VECTOR FOLLOWS'
C      WRITE(9,*)(ASLOD(ISVAB),ISVAB=1,NSVAB)

C ASSEMBLE THE STIFFNESS MATRIX
C ----------------------------
      DO 30 IELEM=1,NELEM
      READ (3,*)((ESTIF(I,J),J=1,6),I=1,6)
      DO 20 INODE=1,2
      NODEI=IELEM+INODE-1
      DO 20 IDOFN=1,3
      NROWS=(NODEI-1)*3+IDOFN
      NROWE=(INODE-1)*3+IDOFN
      DO 20 JNODE = 1,2
      NODEJ=IELEM+JNODE-1
      DO 20 JDOFN =1,3
      NCOLS=(NODEJ-1)*3+JDOFN
      NCOLE=(JNODE-1)*3+JDOFN
      ASTIF(NROWS,NCOLS)=ASTIF(NROWS,NCOLS)+ESTIF(NROWE,NCOLE)
 20   CONTINUE
 30   CONTINUE
 40   FORMAT(6F12.4)
      RETURN
      end
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS IN PROGRAM UNIT: 0

```
C--------------------------------------------------------------------
C                          SUBROUTINE GREDUC
C                          ~~~~~~~~~~~~~~~~~
```

```
C FUNCTION :
C         * GAUSIAN REDUCTION ROUTINE
C-----------------------------------------------------------------------
      SUBROUTINE GREDUC
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      NEQNS=NSVAB
      DO 50 IEQNS=1,NEQNS
      IF(IFPRE(IEQNS).EQ.1)GOTO 30

C REDUCE EQUATIONS
C -----------------
      PIVOT=ASTIF(IEQNS,IEQNS)
      IF(ABS(PIVOT).LT.1.0E-10)GOTO 60
      IF(IEQNS.EQ.NEQNS)GOTO 50
      IEQN1=IEQNS+1
      DO 20 IROWS=IEQN1,NEQNS
      FACTR=ASTIF(IROWS,IEQNS)/PIVOT
      IF(FACTR.EQ.0.0)GOTO 20
      DO 10 ICOLS=IEQNS,NEQNS
      ASTIF(IROWS,ICOLS)=ASTIF(IROWS,ICOLS)-FACTR*ASTIF(IEQNS,ICOLS)
 10   CONTINUE
      ASLOD(IROWS)=ASLOD(IROWS)-FACTR*ASLOD(IEQNS)
 20   CONTINUE
      GO TO 50

C ADJUST RHS(LOADS) FOR PRESCRIBED DISPLACEMENTS
C -----------------------------------------------
 30   DO 40 IROWS=IEQNS,NEQNS
      ASLOD(IROWS)=ASLOD(IROWS)-ASTIF(IROWS,IEQNS)*FIXED(IEQNS)
      ASTIF(IROWS,IEQNS)=0.0
 40   CONTINUE
      GOTO 50
 60   WRITE(9,90)PIVOT,IEQNS
 90   FORMAT(5X,18HINCORRECT PIVOT = ,E20.6,5X,13HEQUATION NO. ,I5)
      STOP
 50   CONTINUE
      RETURN
      END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS   IN PROGRAM UNIT: 0

```
C-----------------------------------------------------------------------
C                          SUBROUTINE BAKSUB
C                          ~~~~~~~~~~~~~~~~~
C
C FUNCTION :
C         * BACK-SUBSTITUTION ROUTINE
```

```
C          * SOLVE FOR GLOBAL DISPLACMENT {TDISP}
C          * SOLVE FOR SUPPORT REACTIONS {TREACT}
C-----------------------------------------------------------------------
      SUBROUTINE BAKSUB
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
     .            XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .            ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .            CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      NEQNS = NSVAB
      DO 5 IEQNS=1,NEQNS
      REACT(IEQNS)=0.0
 5    CONTINUE
      NEQN1=NEQNS+1
      DO 30 IEQNS=1,NEQNS
      NBACK=NEQN1-IEQNS
      PIVOT=ASTIF(NBACK,NBACK)
      RESID=ASLOD(NBACK)
      IF(NBACK.EQ.NEQNS)GOTO 20
      NBAC1=NBACK+1
      DO 10 ICOLS=NBAC1,NEQNS
      RESID=RESID-ASTIF(NBACK,ICOLS)*XDISP(ICOLS)
 10   CONTINUE
 20   IF(IFPRE(NBACK).EQ.0) XDISP(NBACK)=RESID/PIVOT
      IF(IFPRE(NBACK).EQ.1) XDISP(NBACK)=FIXED(NBACK)
      IF(IFPRE(NBACK).EQ.1) REACT(NBACK)=-RESID
 30   CONTINUE
      KOUNT=0
      DO 40 IPOIN=1,NPOIN
      DO 40 IDOFN=1,3
      KOUNT=KOUNT+1
      TDISP(IPOIN,IDOFN)=XDISP(KOUNT)
 40   TREAC(IPOIN,IDOFN)=REACT(KOUNT)
      RETURN
      END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0
NUMBER OF ERRORS IN PROGRAM UNIT: 0

```
C-----------------------------------------------------------------------
C                         SUBROUTINE ENDFOR
C                         ~~~~~~~~~~~~~~~~~
C FUNCTION :
C          * SOLVE FOR ELEMENT END FORCES {ENDACT}
C          * PRINT ELEMENT END FORCES
C-----------------------------------------------------------------------
      SUBROUTINE ENDFOR
      COMMON/FEM1/SPAN,NELEM,NPOIN,NSVAB,YOUNG,UW,FY,RISE,WEBT,NSTAGE,
     .            RISEM,FA,NRISES
      COMMON/FEM2/PROPS(25,9),COORD(25,2),LNODS(24,2),IFPRE(75),
     .            FIXED(75),RLOAD(25,3),MATNO(25),ULL(25),af(25),
```

```
     .               XDISP(75),TDISP(25,3),TREAC(25,3),TLOAD(25,3),
     .               ASTIF(75,75),ASLOD(75),REACT(75),ELCOD(2,2),
     .               CLENG(25),DEPTH(25),ENACT(6),dcal(25),FORCE(25,3)
      DIMENSION ESTIF(6,6),TRANS(6,6),XLOAD(6)
      DIMENSION ELDIS(3,2),ALDIS(6),EDISP(6)
      REWIND 1
      REWIND 2
      REWIND 4
      REWIND 5

       WRITE(9,51)
       WRITE(9,101)
       DO 201 IPOIN=1,NPOIN
201    WRITE(9,301) IPOIN,(TDISP(IPOIN,IDOFN),IDOFN=1,3)

       WRITE(9,351)
       WRITE(9,401)
       DO 501 IPOIN=1,NPOIN
501    WRITE(9,301) IPOIN,(TREAC(IPOIN,IDOFN),IDOFN=1,3)

      DO 200 IELEM=1,NELEM
C     --------------------
      ELENG=PROPS(IELEM,8)
      READ(1,*) ((TRANS(I,J),J=1,6),I=1,6)
      READ(2,*) ((ESTIF(I,J),J=1,6),I=1,6)
      READ(4,*) (XLOAD(I),I=1,6)
      DO 10 INODE=1,2
      LNODE=IELEM+INODE-1
      DO 10 IDOFN=1,3
      ELDIS(IDOFN,INODE)=TDISP(LNODE,IDOFN)
10    CONTINUE
      ALDIS(1)=ELDIS(1,1)
      ALDIS(2)=ELDIS(2,1)
      ALDIS(3)=ELDIS(3,1)
      ALDIS(4)=ELDIS(1,2)
      ALDIS(5)=ELDIS(2,2)
      ALDIS(6)=ELDIS(3,2)

C {ENDACT} = {ESTIF}*{TRANS}*{EDISP} - {XLOAD}
C ------------------------------------------
      DO 20 IEVAB=1,6
      EDISP(IEVAB)=0
      DO 20 JEVAB=1,6
20    EDISP(IEVAB)=EDISP(IEVAB)+TRANS(IEVAB,JEVAB)*ALDIS(JEVAB)
      DO 30 IEVAB=1,6
      ENACT(IEVAB)=0
      DO 30 JEVAB=1,6
30    ENACT(IEVAB)=ENACT(IEVAB)+ESTIF(IEVAB,JEVAB)*EDISP(JEVAB)
      DO 40 IEVAB=1,6
40    ENACT(IEVAB)=ENACT(IEVAB)-XLOAD(IEVAB)

      WRITE(9,50)IELEM
50    FORMAT(/5X,'MEMBER END ACTIONS FOR ELEMENT NO.',I4)
```

```
      WRITE(9,60)
60    FORMAT(/3X,'NODE',9X,'AXIAL',11X,'SHEAR',13X,'MOMENT'/)
      DO 70 INODE=1,2
      LNODE=IELEM+INODE-1
      IFRO=(INODE-1)*3+1
      ITO=(INODE-1)*3+3
      WRITE(9,80)LNODE,(ENACT(JEVAB),JEVAB=IFRO,ITO)
80    FORMAT(I5,3E18.4)
70    CONTINUE
      WRITE(5,*)(ENACT(I),I=1,6)
200   CONTINUE


      REWIND 5
C     Forming Joint Force Matrix
c     --------------------------
      DO 111 IPOIN=1, NPOIN
      DO 111 JFORCE=1,3
         FORCE(IPOIN,JFORCE)=0.0
111   CONTINUE

      DO 444 IELEM=1,NELEM
        READ(5,*)(ENACT(I),I=1,6)
        Do 333 IKOUNT=0,1
          KOUNT=IELEM+IKOUNT
          Do 222 I=1,3
             KOUNT2= I + 3 * IKOUNT
             IF(ABS(ENACT(KOUNT2)).GT.ABS(FORCE(KOUNT,I)))THEN

                    FORCE(KOUNT,I)=ABS(ENACT(KOUNT2))

             ENDIF
222    CONTINUE
333    CONTINUE
444    CONTINUE

      WRITE(9,*)
      WRITE(9,*)
      WRITE(9,*)'   ---------          ------------'
      WRITE(9,*)'   POINT NO:          JOINT FORCE '
      WRITE(9,*)'   ---------          ------------'
      DO 888 IPOIN=1, NPOIN
          WRITE(9,*) IPOIN,FORCE(IPOIN,1),FORCE(IPOIN,2),FORCE(IPOIN,3)
888   CONTINUE
51     FORMAT('1',30X,'NODAL DISPLACEMENTS FOLLOWS')
101    FORMAT(5X,'NODE',9X,'X-DISP',9X,'Y-DISP',9X,'Z-ROT')
301    FORMAT(/I5,3E20.6)
351    FORMAT('1',8X,'REACTIONS FOLLOWS:-')
401    FORMAT(5X,'NODE',9X,'X-REAC',9X,'Y-REAC',9X,'Z-REAC')
      RETURN
      END
```

NUMBER OF WARNINGS IN PROGRAM UNIT: 0

```
 NUMBER OF ERRORS    IN PROGRAM UNIT: 0

 01264

 NUMBER OF WARNINGS IN COMPILATION : 0
 NUMBER OF ERRORS   IN COMPILATION : 0
```

# APPENDIX E

# AN INPUT FILE

```
CASE 2: A HINGED ARCH SUBJ. TO A UNIFORMLY DISTR.LOAD OF PARTIAL LENGTH
  SPAN   NELEM     NSTAGE
  100.0   18        5
  MAXIMUM RISE    MINIMUM RISE    MAXIMUM WEB THICK.  MINIMUM THICK.
  100.000000      0.0000000000        0.025          0.08
  UNIFORM LIVE LOAD  STARTING POINT  ENDING POINT
      0.00000      1            19
  NODE      PX        PY        MZ
  7        0       -277.8     0
  8        0       -555.6     0
  9        0       -555.6     0
  10       0       -555.6     0
  11       0       -555.6      0
  12       0       -555.6     0
  13       0       -277.8     0
  19       0        0.00      0
  NODE  X-CODE VALUE   Y-CODE VALUE    R-CODE VALUE
  1  1     0.0      1   0.0     0    0
  19  1     0.0     1   0.0     0    0
   YOUNG      YEILD STRENGTH   FA       UNIT WEIGHT
   200E6       248000       100000       0.0
```

# APPENDIX F

# AN OUTPUT FILE

CASE 2: A HINGED ARCH SUBJ. TO A UNIFORMLY DISTR.LOAD OF PARTIAL LENGTH

ARCH SPAN = 100.00000000
OPT. ARCH RISE = 20.75000000
OPT. (H/L) RATIO = 0.20750000
OPT. WEB THICK. = 0.015
OPT. WEIGHT = 94.90421300

| NODE NO | Di/Dc | DEPTH |
|---|---|---|
| 1 | 0.68753254 | 1.24191058 |
| 2 | 0.76228297 | 1.37693453 |
| 3 | 0.82426053 | 1.48888636 |
| 4 | 0.87491477 | 1.58038461 |
| 5 | 0.91552663 | 1.65374303 |
| 6 | 0.94720042 | 1.71095634 |
| 7 | 0.97085714 | 1.75368822 |
| 8 | 0.98722529 | 1.78325450 |
| 9 | 0.99683303 | 1.80060923 |
| 10 | 1.00000000 | 1.80632985 |
| 11 | 0.99683386 | 1.80061078 |
| 12 | 0.98722708 | 1.78325772 |
| 13 | 0.97085983 | 1.75369310 |
| 14 | 0.94720399 | 1.71096289 |
| 15 | 0.91553110 | 1.65375113 |
| 16 | 0.87492031 | 1.58039463 |
| 17 | 0.82426703 | 1.48889816 |
| 18 | 0.76229060 | 1.37694824 |
| 19 | 0.68754113 | 1.24192607 |

| ELEMENT NO | AF | AF/AFc |
|---|---|---|
| 1 | 2.67395470E-02 | 0.84119606 |
| 2 | 2.67395470E-02 | 0.84119606 |
| 3 | 2.67395470E-02 | 0.84119606 |
| 4 | 2.67395470E-02 | 0.84119606 |
| 5 | 1.99523531E-02 | 0.62767857 |
| 6 | 1.99523531E-02 | 0.62767857 |
| 7 | 1.99523531E-02 | 0.62767857 |
| 8 | 3.17875333E-02 | 1.00000000 |
| 9 | 3.17875333E-02 | 1.00000000 |
| 10 | 3.17875333E-02 | 1.00000000 |
| 11 | 3.17875333E-02 | 1.00000000 |
| 12 | 1.99509691E-02 | 0.62763500 |
| 13 | 1.99509691E-02 | 0.62763500 |
| 14 | 1.99509691E-02 | 0.62763500 |
| 15 | 2.67415494E-02 | 0.84125900 |
| 16 | 2.67415494E-02 | 0.84125900 |
| 17 | 2.67415494E-02 | 0.84125900 |
| 18 | 2.67415494E-02 | 0.84125900 |

# REFERENCES


1. **"Manual of Steel Construction, Allowable Stress Design"**, 9th ed., American Institute of Steel Construction Inc., New York,N.Y., 1990.

2. Gallagher,R.H. and Zienkiewicz,O.C., "**Optimum Structural Design**", John Willey & Sone Book, Britain, 1973 .

3. Sawczuk, A. and Mroz, Z., "**Optimization in Structural Design**", Springer-Verlag, Berlin 1973 .

4. Kirsch,U. **"Optimum Structural Design"**, McGraw-Hill, New York 1981.

5. Atrek,E., Gallagher,R.H., Ragsdell,K.M.R. and Zienkiewicz, O.C., " **New Directions in Optimum Structural Design"**, John Wiley & Sons Ltd, New York 1984.

6. Rozvany,G. , and Karihaloo,B., " **Structural Optimization**", KLUWER Academic Publishers book, Netherland 1988.

7. Rozvany,G., " **Structural Design via Optimality criteria, the Prager Approach to Structural optimization**", KLUWER Academic Publishers book, Netherland 1989.

8. Wu,C.H., "The Strongest Circular Arch-A Perturbation Solution", **Journal of Applied Mechanics**, Vol. 35, No. 3, 1968, pp. 476-480.

9. Budiansky,B. , Frauenthal,J.C. and Hutchinson,J.W., " On Optimal Arches", **Journal of Applied Mechanics, American Society of Mechanical Engineers**, 1969, pp. 880-882.

10. Tadjbakhsh,I. and Farshad,M., "On Conservatively Loaded Funicular Arches and Their Optimal Design", **Proc. IUTAM Symp. on Structural Optimization**, Warsaw, 1973

11. Farshad,M., " On Optimal Form of Arches", **Journal of the Franklin Institute**, 1976, pp. 187-194.

12. Prager,W. and Rozvany,G., " A New Class of Structural Optimization Problems; Optimal Archgrids" **Computer Methods in Applied Mechanics and engineering**, Vol. 19, No 1, June 1979, pp.127-150.

13. Michell,A.G.M, "The Limits of Economy of Material in Frame Structures", **Philosophical Magazine**, Series 6, Vol. 8, No.47, 1904, pp. 589-597.

14. Topping, " Shape Optimization of Skeletal Structures : A Review " , **Journal of Structural Engineering**, vol. 109, No. 8, August 1983, PP 1933-1951

15. Hill,R.D., Rozvany,G., Wang,C.M. and Hwa,L.K., " Optimization, Spanning Capacity, and Cost Sensitivity of Fully Stresses Arches", **Journal of Structural Mechanics**, Vol.7, 1979, pp.375-411.

16. G.I.N. Rozvany,G., Wang,C.M. and Dow,M., " Arch optimization using Prager-Shield Criteria", **Journal of the Engineering Mechanics Division** ASCE, Vol.106(12), Dec 1980, pp.1279-1286.

17. Rozvany,G. and Wang,C.M., " On Plane Prage-Structure -I", **International Journal of Mech. Sci.** Vol.25, No 7, 1983, PP. 519-527.

18. Rozvany,G. and Wang, C.M., " On Plane Prager-Structure -II", **International Journal of Mech. Sci.** Vol.25, No 7, 1983, PP. 529-541

19. Lipson,S. and Haque,M., "Optimal Design of Arch Using the Complex Method",**Journal of the Structural Division,** Vol 106 , No. ST12, Dec 1980.

20. Box, M. " A New Method of Constrained Optimization and Comparison with other Methods", **Computer Journal,** Vol. 8, Apr., 1965, PP. 42-52.

21. Wassermann, K.," Three dimentional Shape Optimization of Arch Dams with Prescribed Shape Function", **Journal of Structural Mechanics**, Vol.11, 1983, PP. 465- 489.

22. Ermopoulos,I. and Ioannidis,S., "Optimum Rise Design of Steel Arch Bridges", **Journal of Constructional Steel Research**, vol. 5, 1985, pp. 303-310.

23. Ang,B., Teo,K., and C.Wang,C., " Optimal Shape of Arches Under Bending and Axial Compression", **Journal of Engineering Mechanics**", Vol.114, No.5, May 1988, PP. 898-905.

24. Yao,T. and Choi,K. " Shape Optimal Design of an arch dam", **Journal of Structural Engineering**, Vol.115, No. 9, 1989, pp.375-411.

25. Zhu,B., Rao,B., Jia,J. and Li,Y., " Shape Optimization of Arch Dams for Static and Dynamic Loads " , **Journal of Structural Engineering",** Vol. 118, November 1992, PP. 2996-3915.

# VITA

| | | |
|---|---|---|
| Name | : | Hani Mahmoud Mohdaly |
| Discipline | : | Structural Engineer (Designer) |
| Years of Experience | : | Seven |
| Nationality | : | Egyptian |

## Education

Finished Secondary School at Riyadh in 1984.

Bachelor of Science in Civil Engineering, 1990.
King Fahd University of Petroluem and Minerals ,Dhahran, Saudi Arabia .

M.Sc. in structural engineering, in Jan. 1997.
King Fahd University of Petroluem and Minerals ,Dhahran, Saudi Arabia

## Work Experience

He started his career as a Technical Engineer assisting users in utilising Computer Aided Design and Drafting software at Jeraisytech Apple Machintosh in Riyadh in 1990. Then, he worked in several companies as a structural designer. In 1994 he joined RANCO Precast Concrete Factory where he is involved in the design of various precast prestressed concrete projects.