FINITE ELEMENT MODEL

FOR CYCLIC LOADING OF CONCRETE FILLED STEEL TUBE PILE

by

MIN YOUNG PARK

B.Eng., CHUNG-ANG UNIVERSITY, 1993

M.Eng., CHUNG-ANG UNIVERSITY, 1995

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

Department of Civil Engineering

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

April 2001

Department of Civil Engineering

The University of British Columbia
Vancouver, Canada

Date April. 25. 2001

ABSTRACT

A non-linear finite element analysis is presented to predict the lateral response of a concrete-filled steel tube (CFT) pile subjected to cyclic loading. The computer program CFTPILE was developed as a part of this study. The pile is modeled as a beam element on a non-linear soil medium, which is able to resist compression only. Information on the soil-pile interaction and the stress-strain relationships for concrete and steel is required in order to calculate the response of the pile. The soil-pile interaction is assumed to follow a P-y curve proposed by Yan and Byrne. The formation of gaps between pile and soil is taken into account in the analysis. Tomii and Sakino's confined concrete model is adopted to determine the stress-strain relationship for the concrete inside steel tube. Steel is assumed to be elasto-perfectly plastic. Numerical examples are presented for a CFT and a Hollow section pile.

Variability of pile capacity caused by the uncertainty in soil properties is investigated using the computer program RELAN and a response surface involving six random variables. The cumulative distribution function for maximum pile capacity is calculated for cases with and without correlation between the soil variables.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to gratefully acknowledge the guidance of my supervisor, Dr. Ricardo O. Foschi. I could not have completed this thesis without his support, which have broaden my understanding of structural engineering.

Also, I would like to express my appreciation to Dr. Helmut G.L. Prion for his helpful suggestions.

I wish to thank my wife for her patience and encourgement.

1 INTRODUCTION

A pile using a concrete filled steel tube (CFT) has many advantages when compared to a hollow steel tube or a reinforced concrete member. For example, a hollow steel tube section has to meet certain limitation of diameter-thickness ratio (D/t) in order to have enough rotation capacity without triggering local bucking. On the other hand, the CFT member shows a significant yielding plateau in the load-deflection curve as a beam member. A test reported by Prion and Boehme (1994) shows a ductile behaviour of CFT members indicating the fact that neither concrete nor steel tubes are individually able to absorb significant amounts of energy under cyclic loading. The concrete inside the steel tube keeps it from buckling locally and the external steel casing prevents premature spalling and crushing of the concrete, providing a ductile behaviour that is superior to a hollow steel tube.

Research has been done with regard to the behaviour of hollow steel tube piles under earthquake load. When the pile is subjected to an earthquake excitation, it is required to have high-energy absorption capacity in order to resist the earthquake load in a ductile manner. This is provided by filling the steel tube pile with concrete.

The objective of this work is to present a non-linear finite element analysis to determine the lateral response of a CFT pile subjected to static cyclic loading. The model presented in this study is based on the analysis program HYST (Foschi, 2000), which was developed to calculate hysteresis loops of shear connectors in a non-linear medium, namely wood. Using the model, a CFT pile analysis program called CFTPILE (Appendix B), was developed as part of this work. The pile response to lateral cyclic head loading

has a non-linearity, which is caused by the elasto-plastic properties of the steel tube, the non-linear concrete behaviour, the non-linear soil-pile interaction and the formation of gaps between soil and pile.

Stress-strain relationships for steel and concrete were defined in this analysis to calculate stresses at each point over the cross-section and along the member. The soil-pile interaction was determined using P-y curves developed by Yan and Byrne (1992).

The analysis required the calculation of the tangent stiffness matrix at each step of loading. This was done numerically using a beam finite element.

2 ANALYSIS MODEL

2.1 THE FINITE ELEMENT MODEL

To find the lateral response of a CFT pile subjected to cycling loading, the lateral load associated with a given displacement of the pile cap is obtained by a non-linear finite element analysis.

A beam element model is shown in Fig1.

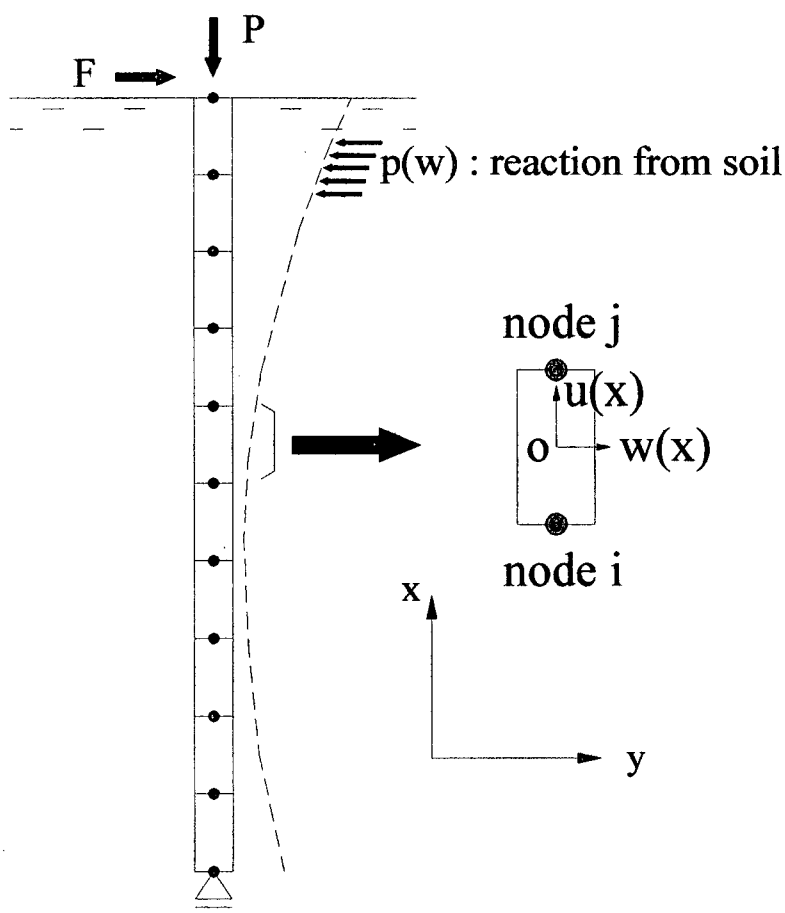


Fig.1 Beam element model

Each node at the ends of the element has five degrees of freedom: w, w', w", u and u'. u is the axial displacement in the x-direction at the centroid of the cross section, and w is the lateral displacement in the y-direction. w', w" and u' are their derivatives with respect to x. The displacement u(x) and w(x) are assumed to be, respectively, a cubic polynomial and a fifth-order polynomial. These functions can be written in matrix form using the vectors of shape functions and their derivatives, and the degrees-of-freedom vectors **a**:

$$
\begin{aligned}
u(x) &= \mathbf{N_0}^T\ \mathbf{a} \\
u'(x) &= \mathbf{N_1}^T\ \mathbf{a} \\
w(x) &= \mathbf{M_0}^T\ \mathbf{a} \\
w'(x) &= \mathbf{M_1}^T\ \mathbf{a} \\
w''(x) &= \mathbf{M_2}^T\ \mathbf{a}
\end{aligned}
\qquad \text{Eq. 1}
$$

$$
\begin{Bmatrix} u \\ w \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & n4 & n5 & 0 & 0 & 0 & n9 & n10 \\ m1 & m2 & m3 & 0 & 0 & m6 & m7 & m8 & 0 & 0 \end{bmatrix} \cdot \begin{Bmatrix} a1 \\ a2 \\ a3 \\ a4 \\ a5 \\ a6 \\ a7 \\ a8 \\ a9 \\ a10 \end{Bmatrix} \qquad \text{Eq. 2}
$$

where $a1 \sim a5$: Degrees-of-freedom at i node ($w_i, w_i', w''_i, u_i, u'_i$)

$a6 \sim a10$: Degrees-of-freedom at j node ($w_j, w_j', w''_j, u_j, u'_j$)

$n4 \sim n10$, $m1 \sim m8$: Shape functions. These are presented in Appendix A.

Using the assumption of plane section remaining plane, the strain ε along the element can be expressed as

$$\varepsilon = \frac{\partial u}{\partial x} - y \cdot \frac{\partial^2 w}{\partial x^2} + \frac{1}{2}\left(\frac{\partial w}{\partial x}\right)^2 \qquad \text{Eq. 3}$$

where y is a distance from the centroid to the point at which the strain is determined. In order to formulate the equilibrium equation the principle of virtual work is adopted.

$$\delta W = \delta W_i - \delta W_e = 0 \qquad \text{Eq. 4}$$

where the internal work is

$$\delta W_i = \int_{Vs} \sigma_s(\varepsilon) \cdot \delta\varepsilon \cdot dV + \int_{Vc} \sigma_c(\varepsilon) \cdot \delta\varepsilon \cdot dV$$

$$= \int_{Vs} \sigma_s(\varepsilon) \cdot \delta\, \mathbf{a}^{\mathbf{T}} [(\mathbf{N_1} - \mathrm{y}\, \mathbf{M_2}) + \mathbf{M_1}\, \mathbf{M_1}^{\mathbf{T}}\, \mathbf{a}]\, dV$$

$$+ \int_{Vc} \sigma_s(\varepsilon) \cdot \delta\, \mathbf{a}^{\mathbf{T}} [(\mathbf{N_1} - \mathrm{y}\, \mathbf{M_2}) + \mathbf{M_1}\, \mathbf{M_1}^{\mathbf{T}}\, \mathbf{a}]\, dV \qquad \text{Eq. 5}$$

Here, $\sigma_s(\varepsilon)$: Steel stress in the member

$\sigma_c(\varepsilon)$: Concrete stress in the member

Vs : Volume of steel tube only

Vc : Volume of concrete only

The external work is, on the other hand,

$$\delta W_e = -\int_0^L p(|w|)\cdot\left(\frac{w}{|w|}\right)\cdot\delta\ \mathbf{a}^{\mathbf{T}}\ \mathbf{M_0}\cdot dx + F\cdot\ \delta\ \mathbf{a}^{\mathbf{T}}\ \mathbf{M_{0(x=L)}} + P\cdot\delta\ \mathbf{a}^{\mathbf{T}}\ \mathbf{N_{0(x=L)}} \qquad \text{Eq. 6}$$

Here, F : Lateral load applied at x = L in the y-direction

P : Axial load applied at x = L in the x-direction

p : Soil reaction

L : Pile length

Since $\delta\ \mathbf{a}$ is arbitrary, the principle of virtual work results in a system of non-linear equations equivalent to a vector ψ being zero at the solution **a**.

$$\psi = \int_{Vs}\sigma_s(\varepsilon)\cdot[(\mathbf{N_1} - y\ \mathbf{M_2}) + \mathbf{M_1}\ \mathbf{M_1}^{\mathbf{T}}\ \mathbf{a}]\ dV$$

$$+ \int_{Vc}\sigma_c(\varepsilon)\cdot[(\mathbf{N_1} - y\ \mathbf{M_2}) + \mathbf{M_1}\ \mathbf{M_1}^{\mathbf{T}}\ \mathbf{a}]\ dV$$

$$+ \int_0^L p(|w|)\cdot\left(\frac{w}{|w|}\right)\cdot\ \mathbf{M_0}\cdot dx - F\cdot\mathbf{M_{0(x=L)}} - P\cdot\mathbf{N_{0(x=L)}} = 0 \qquad \text{Eq. 7}$$

To find the solution, an iteration procedure using the Newton-Raphson method is applied to make the vector ψ approach zero with an allowable tolerance.

Using the Newton-Raphson method the solution vector **a** is obtained by

$$\mathbf{a} = \mathbf{a}^* + [\nabla\psi^*]^{-1}\cdot\{-\psi^*\} \qquad \text{Eq. 8}$$

where $[\nabla\psi^*]$ is the tangent stiffness matrix.

In detail,

$$\{\psi_i\}=\int_0^L\int_{-R}^{R}\sigma_s(\varepsilon)\cdot[\,\mathbf{N_1}-\,y\cdot\mathbf{M_2}+\mathbf{M_1}\,\mathbf{M_1}^{\mathbf{T}}\,\mathbf{a}\,]_i\;\cdot b_s(y)\cdot dy\cdot dx$$

$$+\int_0^L\int_{-r}^{r}\sigma_c(\varepsilon)\cdot[\,\mathbf{N_1}-\,y\cdot\mathbf{M_2}+\mathbf{M_1}\,\mathbf{M_1}^{\mathbf{T}}\,\mathbf{a}\,]_i\;\cdot b_c(y)\cdot dy\cdot dx$$

$$+\int_0^L p(|w|)\cdot\left(\frac{w}{|w|}\right)\;\mathbf{M_0}\,i\cdot dx-F\cdot\;\mathbf{M_0}\,i_{\,(x=L)}-\;P\cdot\mathbf{N_0}\,i_{\,(x=L)} \qquad \text{Eq. 9}$$

$R, r, b_s(y)$ and $b_c(y)$ are shown in Fig.2.

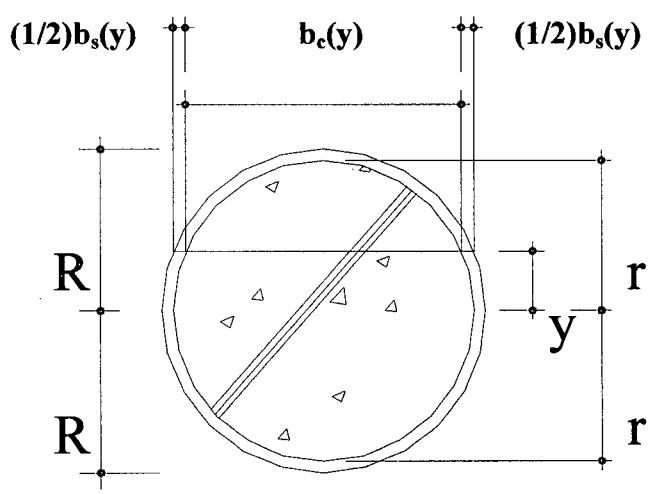


Fig.2 Pile cross-section

The element of the matrix $[\nabla\psi]$ in the i th row and j th column is represented by

$$\nabla\psi_{ij} = K_{ij} = \int_0^L \int_{-R}^{R} b_s(y) \cdot \frac{d\sigma_s}{d\varepsilon} \cdot [\,\mathbf{N_1} - y \cdot \mathbf{M_2} + \mathbf{M_1}\,\mathbf{M_1}^{\mathrm{T}}\,\mathbf{a}\,]_i \cdot [\,\mathbf{N_1} - y \cdot \mathbf{M_2} + \mathbf{M_1}\,\mathbf{M_1}^{\mathrm{T}}\,\mathbf{a}]^{\mathrm{T}}{}_j \cdot dy \cdot dx$$

$$+ \int_0^L \int_{-R}^{R} b_s(y) \cdot \sigma_s(\varepsilon) \cdot \mathbf{M_1}\,i \cdot \mathbf{M_1}^{\mathrm{T}}\,j \cdot dy \cdot dx$$

$$+ \int_0^L \int_{-r}^{r} b_c(y) \cdot \frac{d\sigma_c}{d\varepsilon} \cdot [\,\mathbf{N_1} - y \cdot \mathbf{M_2} + \mathbf{M_1}\,\mathbf{M_1}^{\mathrm{T}}\,\mathbf{a}\,]_i \cdot [\,\mathbf{N_1} - y \cdot \mathbf{M_2} + \mathbf{M_1}\,\mathbf{M_1}^{\mathrm{T}}\,\mathbf{a}]^{\mathrm{T}}{}_j \cdot dy \cdot dx$$

$$+ \int_0^L \int_{-r}^{r} b_c(y) \cdot \sigma_c(\varepsilon) \cdot \mathbf{M_1}\,i \cdot \mathbf{M_1}^{\mathrm{T}}\,j \cdot dy \cdot dx$$

$$+ \int_0^L \frac{dp}{d|w|} \cdot \mathbf{M_0}\,i \cdot \mathbf{M_0}\,j \cdot dx \qquad \text{Eq. 10}$$

Then Eq.8 is solved to find a new vector **a** for each step of displacements.

Convergence is achieved when Eq.11 and Eq.12 are satisfied.

For the residual force vector $\{\psi\}$,

$$\sum_{i=1}^{NEQ} \psi_i^{\,2} \le Tol1 \qquad \text{Eq. 11}$$

For the displacement correction vector $\{a - a^*\}$,

$$\sum_{i=1}^{NEQ} (a_i - a_i^*)^2 \le Tol2 \qquad \text{Eq. 12}$$

where NEQ is the number of equations and * indicates the values at the previous iteration.

*Tol*1 and *Tol*2 are specified tolerances.

Integrations are conducted by a Gaussian integration scheme involving a coordinate transformation from x, y to normalized coordinates ξ and η. These have a range from –1 to 1.

2.2 MODELLING OF THE SOIL

Soil-pile interaction is represented using a P-y curve which was proposed by Yan and Byrne(1992) in order to predict pile response to lateral pile head loading. This curve has a nonlinear relationship shown in Fig.3.

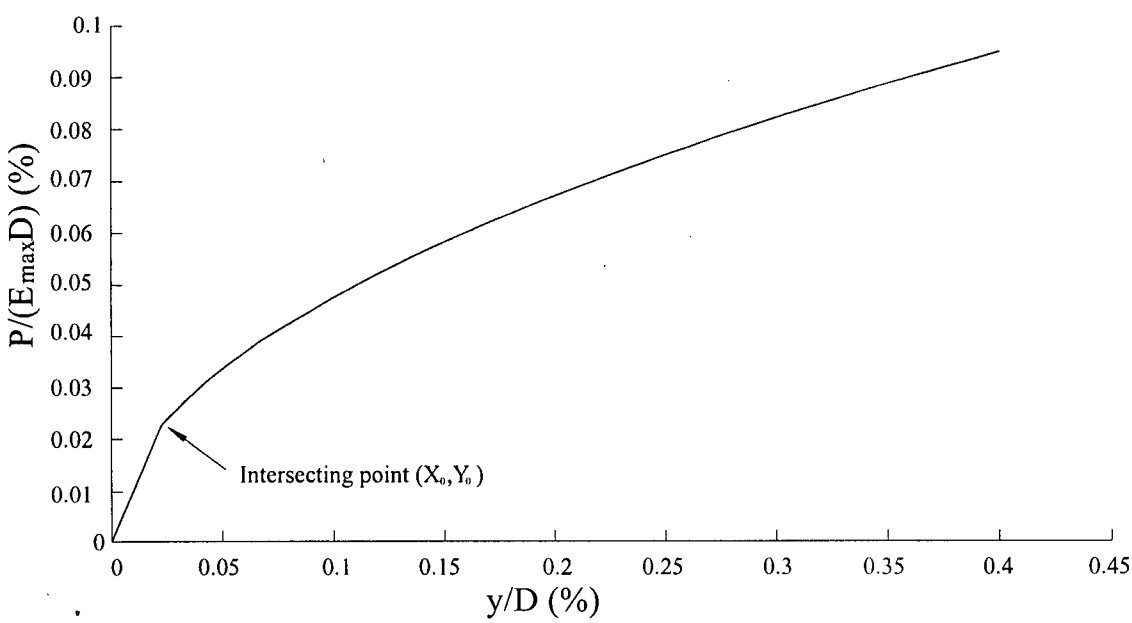


Fig.3 Normalized P-y curve

The relationship is given by

$$\frac{P}{E_{max} \cdot D} = \alpha \cdot \left(\frac{y}{D}\right)^{\beta} \qquad \text{Eq. 13}$$

where

P: Soil reaction (force/unit length)

D: Pile diameter

y: Lateral pile deflection

E_{max}: Soil maximum Young's modulus

β : Value of about 0.5

P/(E_{max}D) and y/D are percentages.

α is a function of soil relative density and can be expressed as

$$\alpha = 5 \cdot (D_r)^{-0.8} \qquad \text{Eq. 14}$$

where D_r is the relative density in percentage.

The normalized P-y curve has an initial linear portion with a slope of 45°. The intersecting point (X_0, Y_0) with the power function from Eq.13 can be found by

$$X_0 = Y_0 = \frac{y}{D}(\%) = \alpha \cdot \left(\frac{1}{1-\beta}\right) \qquad \text{Eq. 15}$$

This is required to avoid the infinite slope which Eq.13 implies at y = 0. In Eq.13, E_{max} increases with soil depth.

When the pile head displacement is reversed during cyclic loading, a gap between the pile and the soil is formed. This gap is a function of depth. It is assumed that the soil is not able to take any tensile load in order to model the soil gapping. Fig.4 shows the load-displacement relationship for soil used in the program CFTPILE.

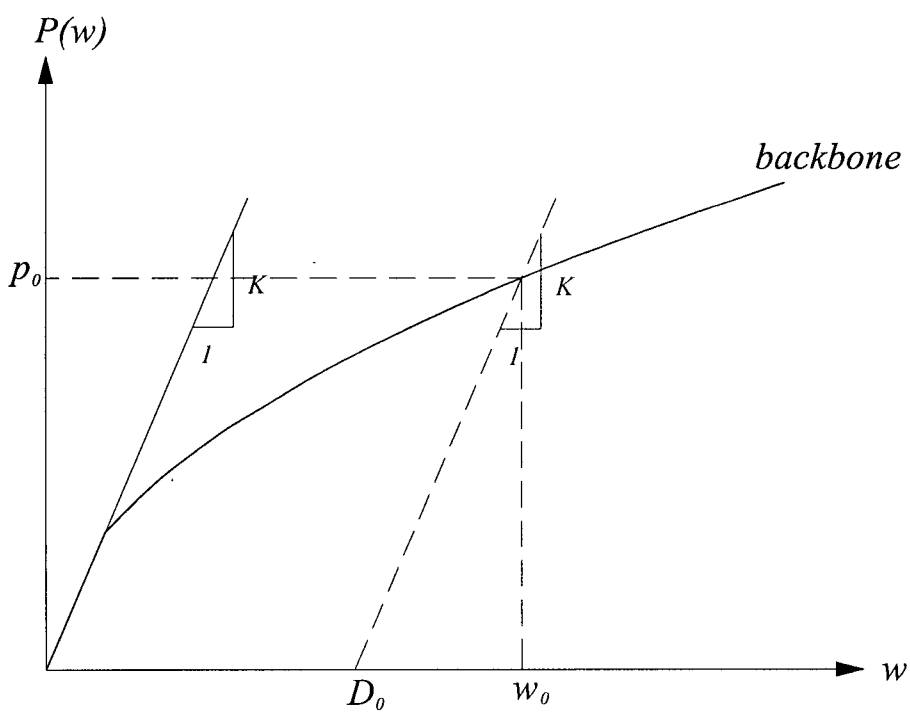


Fig.4 Load-displacement relationship for soil

In the numerical procedure, the program keeps track of a previous state soil displacement, D_0 (shown in Fig.4), at each side of the pile. D_0 is associated with the maximum displacement, w_0, achieved along the backbone curve. With an initial slope, K, D_0 is defined as

$$D_0 = w_0 - p_0 / K \qquad \text{Eq. 16}$$

Note that the initial slope K is equal to the maximum Young's modulus of soil, E_{max}.

The following algorithm can find a soil reaction p(w) for a new w.

$$\begin{aligned} &if\,(w \le D_0) \rightarrow p = 0 \\ &if\,(w > D_0) \rightarrow p = \min of\,[p_1 = K(w - D_0), p_2 = p(w) \\ &if\,(p = p_2) \rightarrow update\,D_0 : D_0 = w - p / K \\ &if\,(p = 0)\,or\,(p = p_1) \rightarrow D_0\,unchanged \end{aligned} \qquad \text{Eq. 17}$$

The displacement D_0, at either side of the pile and as a function of depth, gives the magnitude of the gap along the pile.

2.3 MODELLING OF THE CONCRETE

Confined concrete can be defined as that which is restrained in the directions at right angle to the applied stress. If the compression zone of a concrete beam or column is confined by closely spaced steel stirrup ties or steel casing, the ductility of the concrete is enhanced and large ultimate curvatures may be reached (Kent and Park, 1971). When

concrete is subjected to cyclic compressive loading it has been assumed that an "envelope" curve exists and that this envelope curve is approximately the same as the complete stress-strain curve obtained under monotonically increasing strain. This assumption was shown to be true for confined concrete, as well as for plain, unconfined concrete (Shah, Fafitis and Arnold, 1983). It was also shown that the similarity of monotonic and cyclic stress-strain envelopes indicated that the specimens subjected to unloading and reloading cycles experienced very little or no strength degradation due to cycling.

When it comes to the degradation of elastic modulus in concrete subjected to cyclic load, concrete shows a gradual decrease of its elastic modulus after it reaches its peak stress, but it is not significant when compared to the one for unconfined concrete. It is known that the degradation is a direct consequence of volumetric expansion. Confining pressure to concrete gives a lower amount of expansion and this results in a lower amount of stiffness degradation during unloading.

In a CFT pile the tube confines the concrete inside the steel tube. However, the behaviour is somewhat different from that for the concrete confined by closely spaced steel stirrup ties. The steel tube undergoes axial loads and bending moments, as well as providing confinement, while steel stirrup ties in a reinforced concrete member mainly provides confining pressure. In order to take these differences into account, Tomii and Sakino (1979) proposed a model to determine the stress-strain relationship for the concrete in CFT.

In this paper the model proposed by Tomii and Sakino, which is shown in Fig.5, is adopted to calculate the stresses in the concrete.

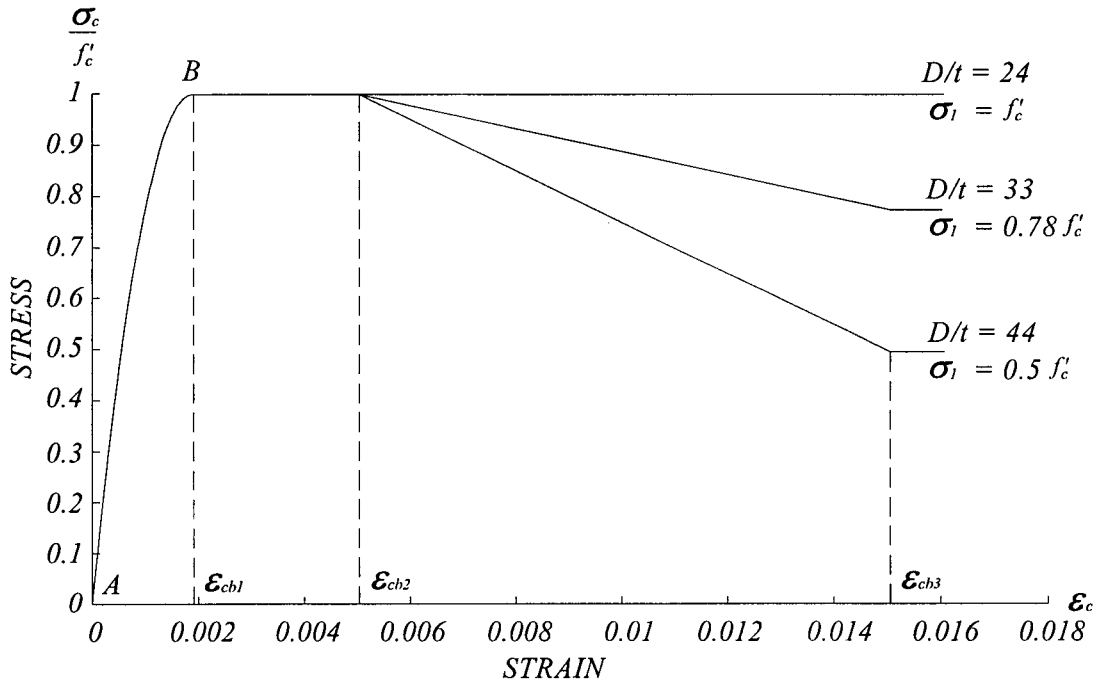


Fig.5 Tomii and Sakino's confined concrete model

A parabolic part of the curve (A-B) was represented by Eq.18,

$$\frac{\sigma_c}{f'_c} = 2 \cdot \left(\frac{\varepsilon_c}{\varepsilon_{cb1}}\right) - \left(\frac{\varepsilon_c}{\varepsilon_{cb1}}\right)^2 \qquad \text{Eq. 18}$$

where

$\varepsilon_{cb1} = 0.012354165 \times \sqrt{f'_c}$

f'_c : Compressive strength of concrete core in GPa

σ_c : Concrete stress at given strain ε_c in GPa

ε_{cb1} : The strain when the stress in concrete reaches f'c

After reaching the maximum stress f'_c, the stresses are assumed to decrease following a multi-linear function, shown in Fig.5, to a minimum σ_1. This minimum, for confined concrete, depends on the ratio D/t according to Eq.19.

$$\sigma_1 = (1.6 - 0.025 \times D/t) \cdot f'_c \qquad \text{Eq. 19}$$

where

σ_1: The stress at the strain, ε_{cb3}

D: Diameter of the CFT pile

t : Thickness of the pile tube

In Fig.5, ε_{cb2} and ε_{cb3} are constant values. ($\varepsilon_{cb2} = 0.005, \varepsilon_{cb3} = 0.015$)

The bond between the steel section and the concrete is assumed to be perfect. Tensile stresses in concrete are ignored. Therefore, the concrete stress is present only when the strain is compressive. In order to calculate the stress in concrete during unloading and reloading, the assumption is made that the unloading and reloading response is linear with a slope equal to the initial tangent modulus.

Fig.6 shows the stress-strain relationship for concrete, which is used in the program CFTPILE. Let us assume that, at a point along the member, the strain is ε_1 for the first time as shown in Fig.6. If we cut out a micro-cube at that point, as shown in Fig 7, the original cube is compressed by the amount of ε_1 as shown in Fig 6 and Fig 7.

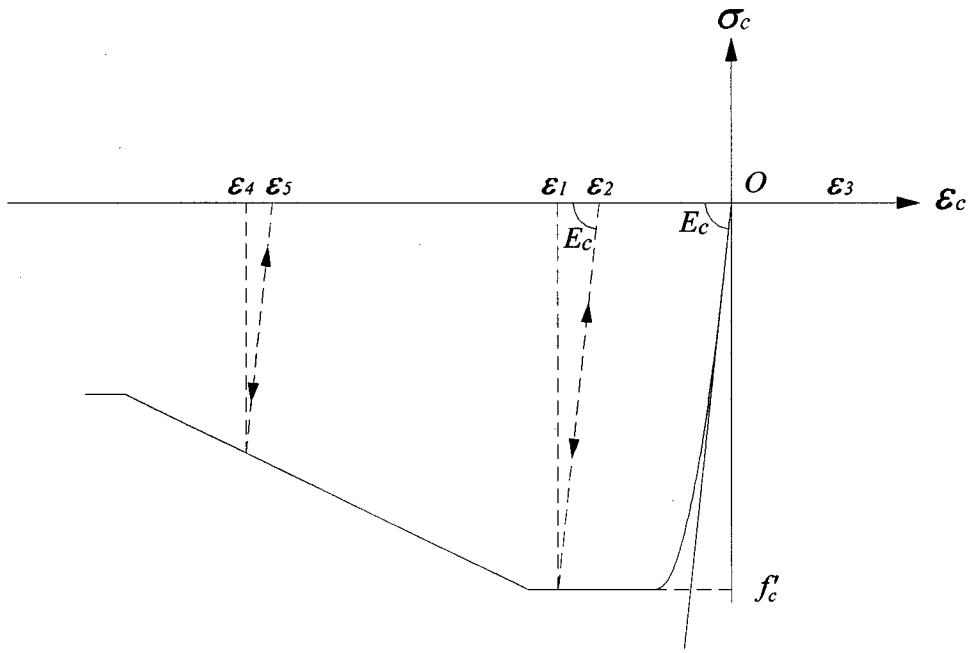


Fig.6 Stress-strain relationship for the concrete model

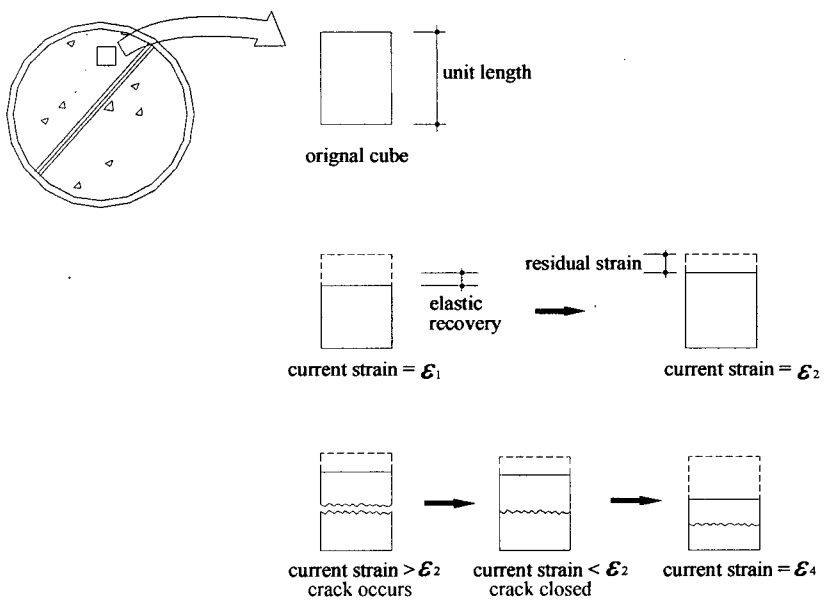


Fig.7. Behaviour of concrete micro-cube

Now let us assume that unloading starts and the current strain arrives at the value of ε_2 at which there is no stress in the concrete. As shown in Fig.6 the elastic strain recovery is $\varepsilon_1 - \varepsilon_2$ and ε_2 can be referred to as the residual strain when the load is removed. From the point of ε_2, if the strain follows the path $\varepsilon_2 \rightarrow$ O, the concrete starts to undergo tension and it develops cracks as soon as the strain becomes smaller than ε_2, in order to agree with the assumption that the tensile stress in concrete is ignored. Then the stress in the concrete will be zero through the path O $\rightarrow$ ε_3 $\rightarrow$ O $\rightarrow$ until the strain reaches ε_2 again where the cracks are closed. Now reloading starts as the strain becomes greater than ε_2. Let us assume that we reach the point ε_4. After unloading, we reach the point ε_5, which is the new residual strain. If the loading is reversed, again, the cracks which have existed since the first load reversal was made will open again and the concrete can not take any load. The whole loop will be repeated for subsequent strain cycles.

2.4 MODELLING OF THE STEEL

The stress-strain relationship in steel tube is assumed to be elasto-perfectly plastic. The relationship is shown in Fig.8, in which E is the modulus of elasticity and σ_y is the yield stress. Knowing the previous state of stress σ_0 at ε_0, the new stress, $\sigma(\varepsilon)$, in the steel tube is obtained by the following algorithm.

$$F(\varepsilon) = \sigma_0 + E \cdot (\varepsilon - \varepsilon_0)$$

$$if\,|F(\varepsilon)| \le \sigma_y \rightarrow \sigma(\varepsilon) = F(\varepsilon) \qquad \text{Eq. 20}$$

$$if\,|F(\varepsilon)| > \sigma_y \rightarrow \sigma(\varepsilon) = \sigma_y \cdot \frac{F(\varepsilon)}{|F(\varepsilon)|}$$

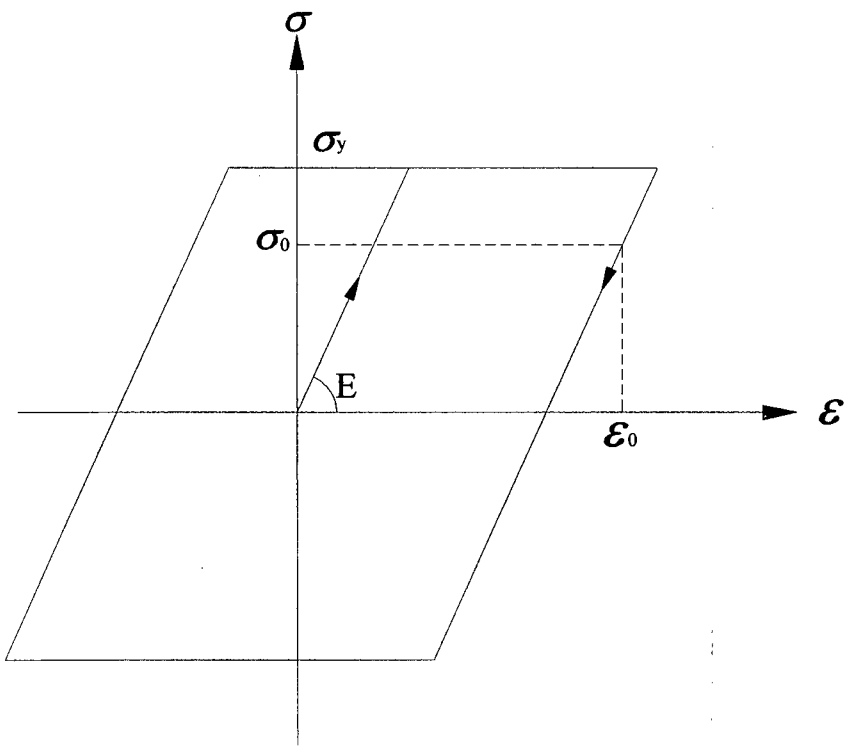


Fig.8 The stress-strain relationship for steel

3 NUMERICAL EXAMPLES

A pile with a length of 30,000mm and an outside diameter of 1500mm is considered for an example. The surrounding soil is assumed to be dense sand having a relative density of 75%. E_{max} of the soil is assumed to be 0.12 GPa at the depth of 4,000 mm, 0.2 GPa at the depth of 10,000 mm, 0.28 GPa at the depth of 20,000 mm. These values are based on the fitted curve for E_{max} shown in the Yan and Byrne (1992) reference. A linear interpolation is used for values of E_{max} at depths other than those mentioned above. The values of E_{max} at different depths are illustrated in Fig.9.

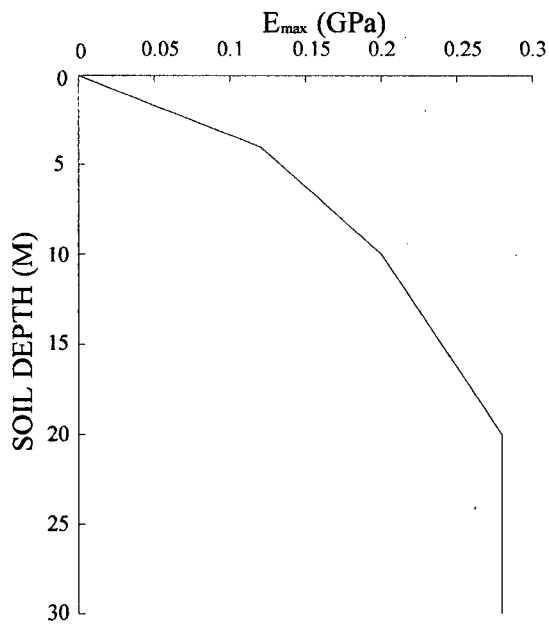


Fig.9 Maximum soil Young's modulus E_{max}

For the material properties of concrete and steel, the steel is assumed to have an elastic modulus of 200 GPa with a yield stress σ_y = 0.25 GPa, and the concrete to have an initial tangent stiffness of 30.18 GPa with a compressive strength of 0.03 GPa. A cyclic displacement with a maximum value of 15 mm is enforced to the pile cap. The displacement history is shown in Fig.10.

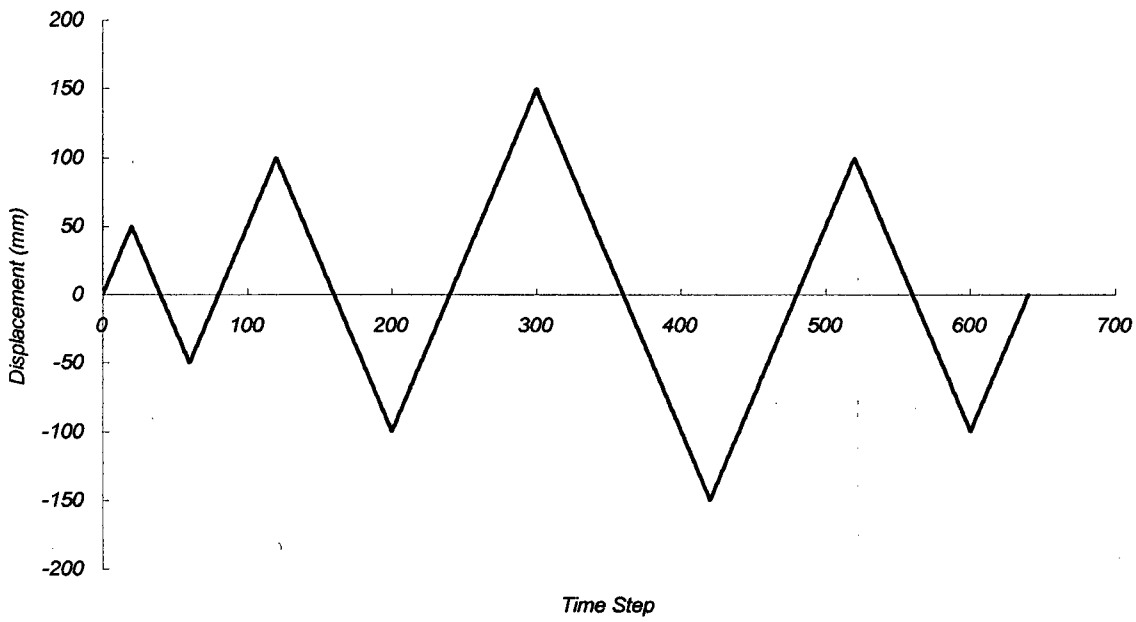


Fig.10 Displacement history

The displacements were divided in 640 individual steps. No axial load was applied and tolerances for the residual force vector and displacement correction vector were 1×10^{-3}. Numerical integration is conducted using 5 Gaussian points in the x-direction and 16 in the y-direction. Fig.11, 12 and 13 show the results of calculation for different conditions. These figures show calculated hysteresis loops for the pile-soil system. Maximum force–displacement curves for a CFT and a Hollow section with the thickness of 30 mm are illustrated in Fig.11.

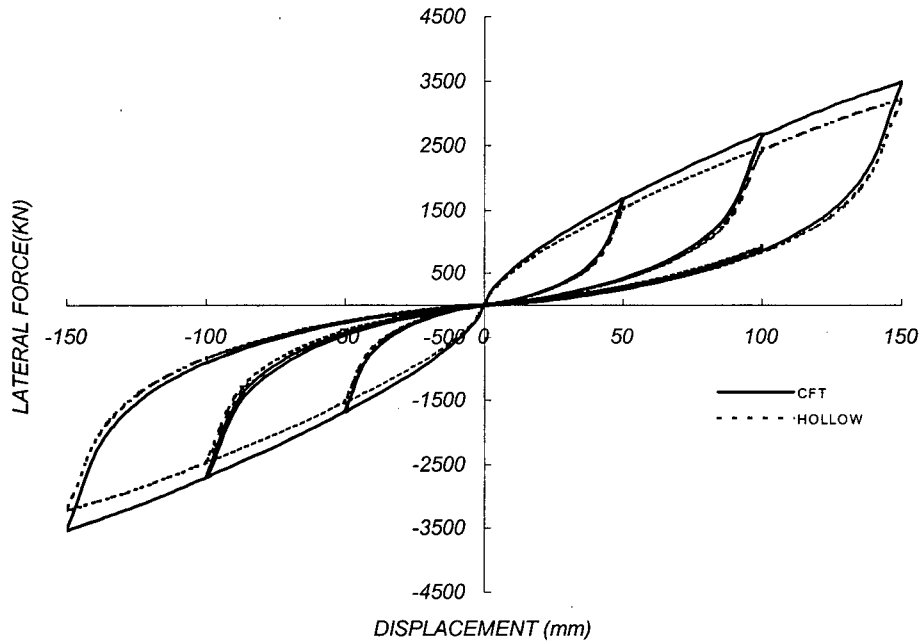


Fig.11 Comparison between CFT and Hollow section (thickness=30mm)

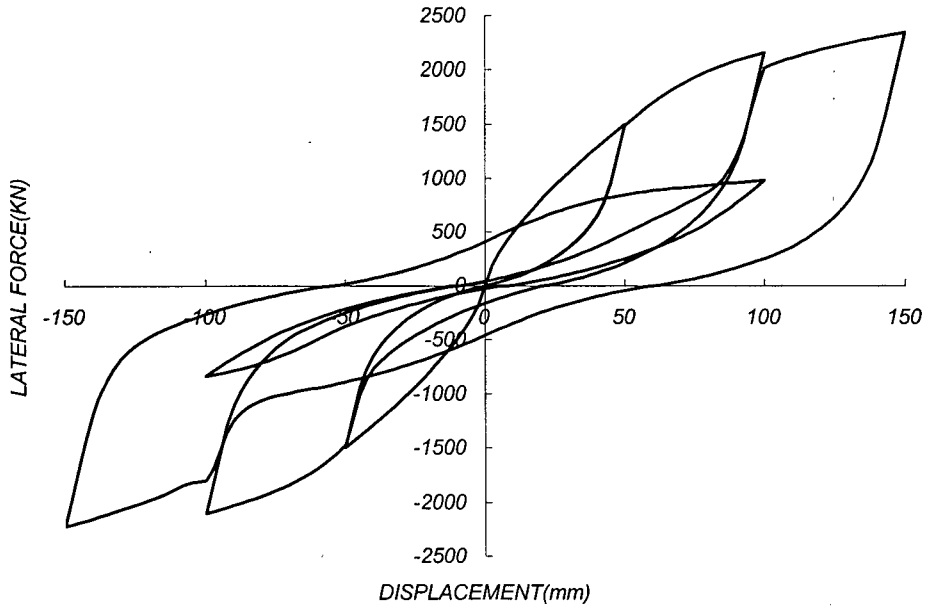


Fig.12 Force-displacement curve for CFT pile (thickness=10mm)

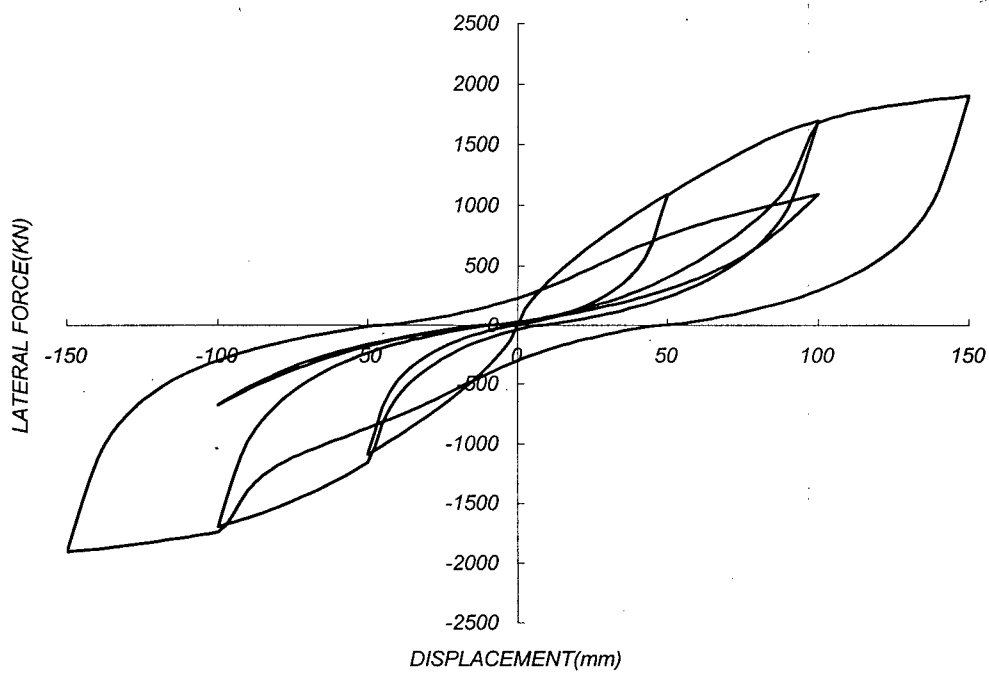
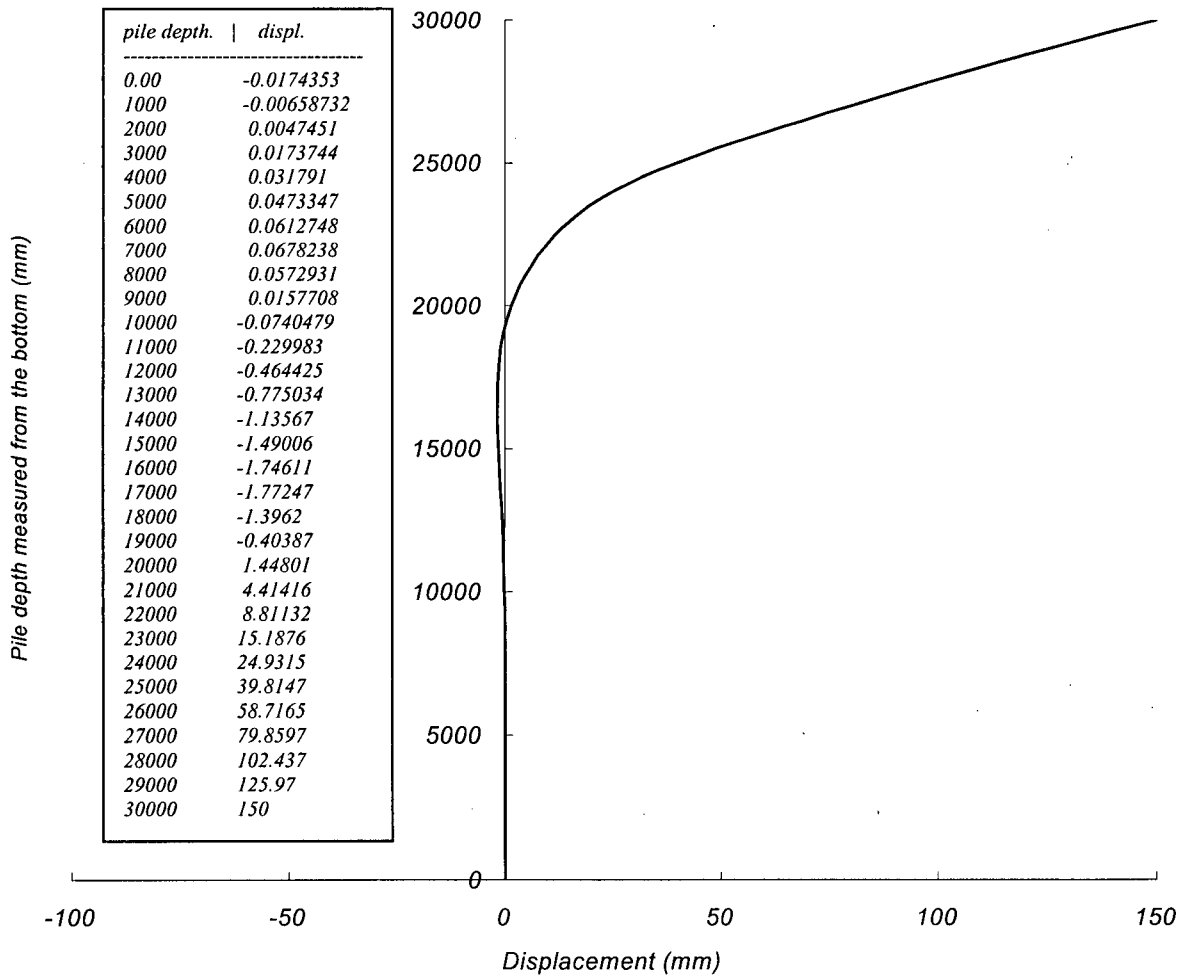


Fig.13 Force-displacement curve for Hollow pile (thickness=10mm)

Fig.12 and Fig 13 show a maximum force-displacement curve for either a CFT or a Hollow section with a wall thickness of 10mm.

It is observed that the pile shown in Fig.11, with wall thickness of 30 mm, mainly remains in the elastic region, while the one with thickness of 10 mm (Fig.12,13) goes far beyond the elastic limit, resulting in the presence of a lateral force at a displacement of zero. In other words, it can be explained that the pile in Fig.12 has a residual displacement although the lateral force is removed. Also, this accounts for the curves in Fig.11 showing that the last loading cycle follows the same force-displacement path as the previous one, resulting in overlapping. The maximum force increases up to 8 % when the pile with a 30 mm tube is filled with concrete, while there is a 23 % increment for the pile with 10 mm wall thickness. Because of the small strains developed in the section of

the 30 mm pile, the pile remains mainly elastic and the contribution of the concrete is smaller than in the case of the thinner tube. Fig.14 shows the lateral displacements along the CFT pile with the 10 mm tube when the displacement history reaches the maximum value of 15 mm.



pile depth.	displ.
0.00	-0.0174353
1000	-0.00658732
2000	0.0047451
3000	0.0173744
4000	0.031791
5000	0.0473347
6000	0.0612748
7000	0.0678238
8000	0.0572931
9000	0.0157708
10000	-0.0740479
11000	-0.229983
12000	-0.464425
13000	-0.775034
14000	-1.13567
15000	-1.49006
16000	-1.74611
17000	-1.77247
18000	-1.3962
19000	-0.40387
20000	1.44801
21000	4.41416
22000	8.81132
23000	15.1876
24000	24.9315
25000	39.8147
26000	58.7165
27000	79.8597
28000	102.437
29000	125.97
30000	150

Fig.14 Lateral displacement along the CFT pile (thickness = 10mm)

It is seen that only the upper third of the pile undergoes deformations, with largest curvatures and moments about 10m below the ground line.

1 UNCERTAINTY IN SOIL PROPERTIES AND ITS INFLUENCE ON VARIABILITY OF MAXIMUM LOAD

In this work an attempt was made to investigate how the uncertainty in soil properties can influence the variability in maximum load. For a specific displacement history, there are many factors that determine the maximum lateral force, such as pile material properties, pile size, and soil properties. These factors are considered to be uncertain, which plays a role in the reliability of the structure. Good quality control in fabrication will yield a small coefficient of variation, such as 0.05, for the geometric parameters of the structure, and 0.1 for the material (Geschwindner,1994). On the other hand, the variation in soil properties is considered to be relatively large. Also, soil shows a larger effect on the maximum force than the other variables such as material properties of the steel and the concrete.

In order to discuss the variability of maximum load applied to the pile cap, during the deformation history shown in Fig.10, a performance function is defined as

$$G = F_{max} - F$$

where

F_{max} is the maximum force, a random variable, obtained from CFTPILE for specific values of the soil and pile properties.

F is any load level.

Entering different levels of F, the performance function can be used to calculate the probability of G < 0 or $F_{max} < F$. These probabilities are the coordinates of the cumulative distribution function for F_{max}.

For the purpose of an example, let us assume that the soil properties are the only random variables to be taken into account, having a coefficient of variation (COV) of 0.25 for the soil maximum Young's modulus E_{max} and COV = 0.1 for the depth at which the E_{max} is taken. The same cyclic displacement shown in Fig.10 is applied to the pile cap. Using the same pile and soil properties (See Fig.9) used in the previous numerical example, the soil properties are given as follows.

E_{1m} = 0.12 GPa at the depth of Z_{m1} = 4,000 mm

E_{2m} = 0.2 GPa at the depth of Z_{m2} = 10,000 mm

E_{3m} = 0.28 GPa at the depth of Z_{m3} = 20,000 mm

Where the subscript m indicates the variable mean value. All variables are assumed to have a Normal distribution. These variables and their related statistics are summarized in Table 1.

Variable No.	Variables	Mean Value	COV	Mean + 2σ	Mean-2σ
1	$E_{max}1$	0.12	0.25	0.18	0.06
2	$E_{max}2$	0.2	0.25	0.3	0.1
3	$E_{max}3$	0.28	0.25	0.375	0.125
4	Z1	4,000	0.1	4,800	3,200
5	Z2	10,000	0.1	12,000	8,000
6	Z3	20,000	0.1	24,000	16,000

Table 1. Random variables and their related statistics.

Using these N = 6 random variables, a response surface for the maximum force F_{max} is constructed by

$$F_{\max} = a_1 + a_2 \cdot E_1 + a_3 \cdot E_1^{\ 2} + a_4 \cdot E_2 + a_5 \cdot E_2^{\ 2} + a_6 \cdot E_3 + a_7 \cdot E_3^{\ 2} + a_8 \cdot Z_1 + a_9 \cdot Z_1^{\ 2} + a_{10} \cdot Z_2 + a_{11} \cdot Z_2^{\ 2} + a_{12} \cdot Z_3 + a_{13} \cdot Z_3^{\ 2} \qquad \text{Eq. 1}$$

In order to find the constants, $a_1 \sim a_{13}$, 13 numerical analyses were conducted changing the variables, using the mean and mean$\pm 2 \cdot \sigma$ values for each of those six variables. First, an analysis was done using the variable means, and then each variable, in turn, was changed to mean + 2σ and then to mean - 2σ. This generated 2N+1 data sets, or 13 in this case. Since the six variables were assumed to have a Normal distribution, about 95 % of the population fell within two standard deviations on either side of the mean. As a result of the 13 analyses, 13 values of F_{max} were obtained, which are $F_{max1} \sim F_{max13}$, allowing calculation of the response surface coefficients $a_1 \sim a_{13}$.

Using the response surface for F_{max}, a cumulative distribution function was constructed by the software RELAN (2001).

The calculated cumulative distribution functions for F_{max} are shown in Fig.15 ~ 17. In each figure, two curves are drawn for different conditions. One is for the case that there is no correlation between the variables E_{max}. The other is for the case that there are certain correlations between those. These correlations are expressed using a Correlation Coefficient ρ and, for the example, they were assumed as shown in the following Table 2.

Related Variables	Correlation Coefficient
1 and 2 (ρ_{12})	0.8
2 and 3 (ρ_{23})	0.8
1 and 3 (ρ_{13})	0.7

- Variable 1, 2 and 3 are referred to in Table 1

Table 2. Correlation Coefficients

Fig.15 shows the cumulative distribution function for the CFT pile with 30 mm of thickness.

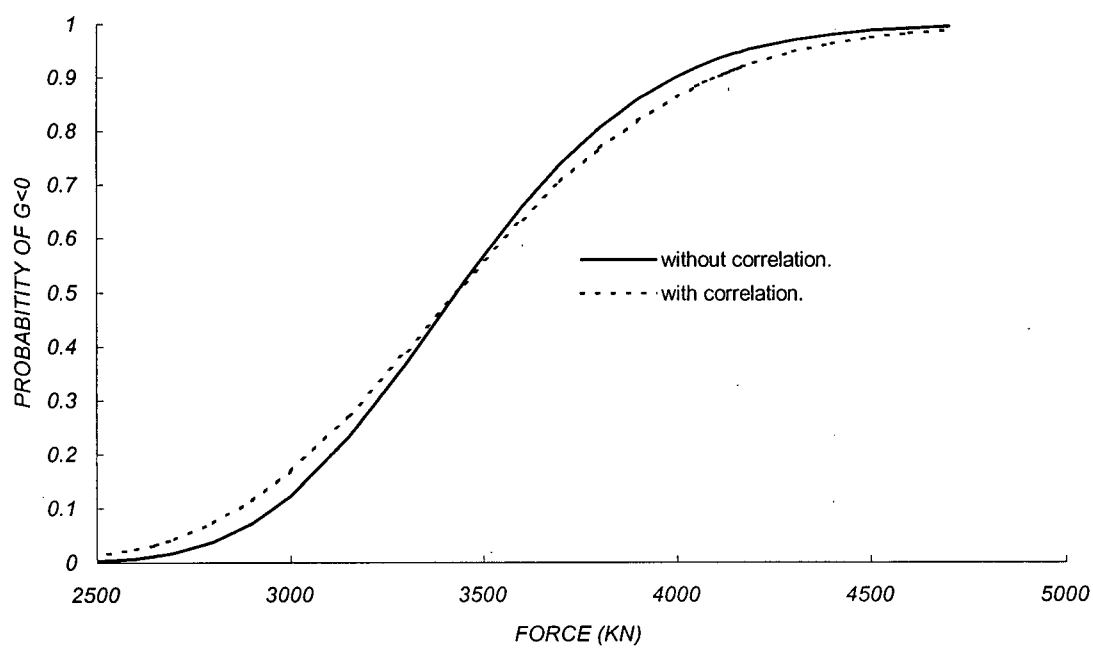


Fig.15 Cumulative distribution function (CFT, thickness=30mm)

Since the pile mainly remains in the elastic region, as explained previously, the variation in the soil properties plays the main role in the variability of maximum lateral force.

Fig.16 shows how the correlation between the variables influences the soil Young's modulus.

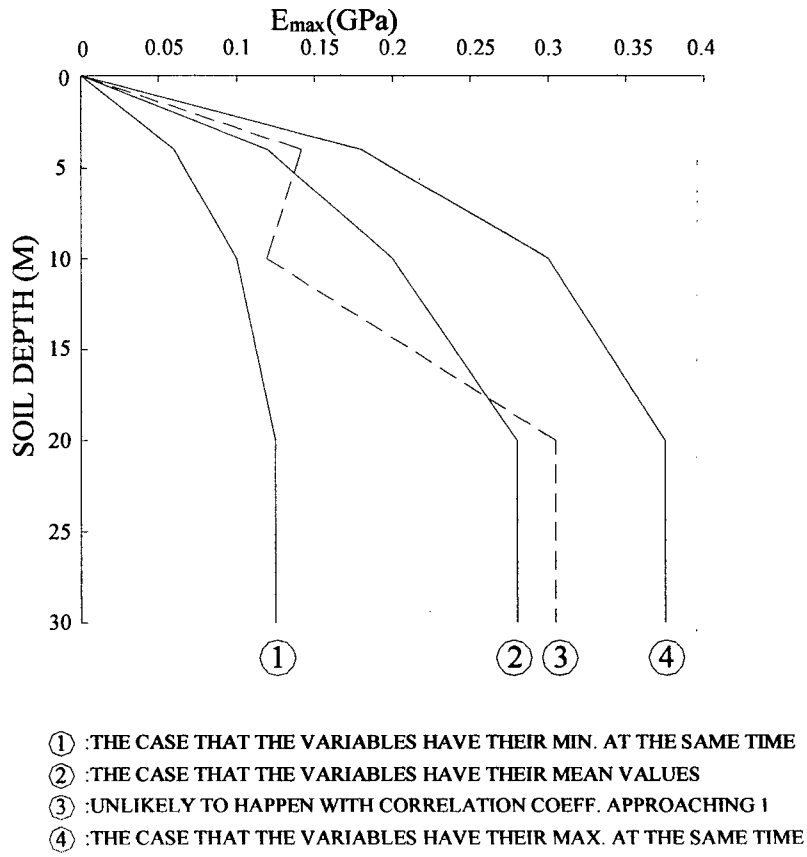


Fig.16 Variability in maximum soil Young's modulus

Due to the correlation between the variables, it is more likely to happen that the variables 1, 2 and 3 have their maximum or minimum values at the same time. This accounts for the curve with correlations in Fig.15 showing lower probabilities at higher levels of force and higher probabilities at lower levels of force. However, this trend does not appear for the thinner wall tube (Fig.17 and 18).

Fig.17 and 18 show, respectively, the cumulative distribution function for the CFT pile with 10 mm thickness and for the Hollow section with 10 mm thickness.

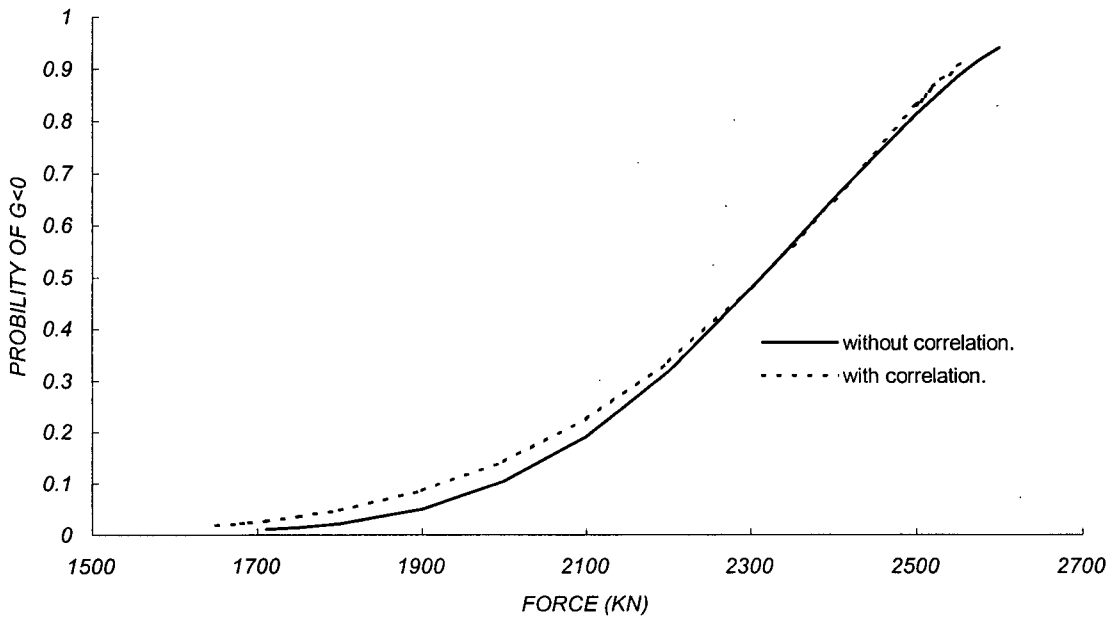


Fig.17 Cumulative distribution function (CFT, thickness=10mm)

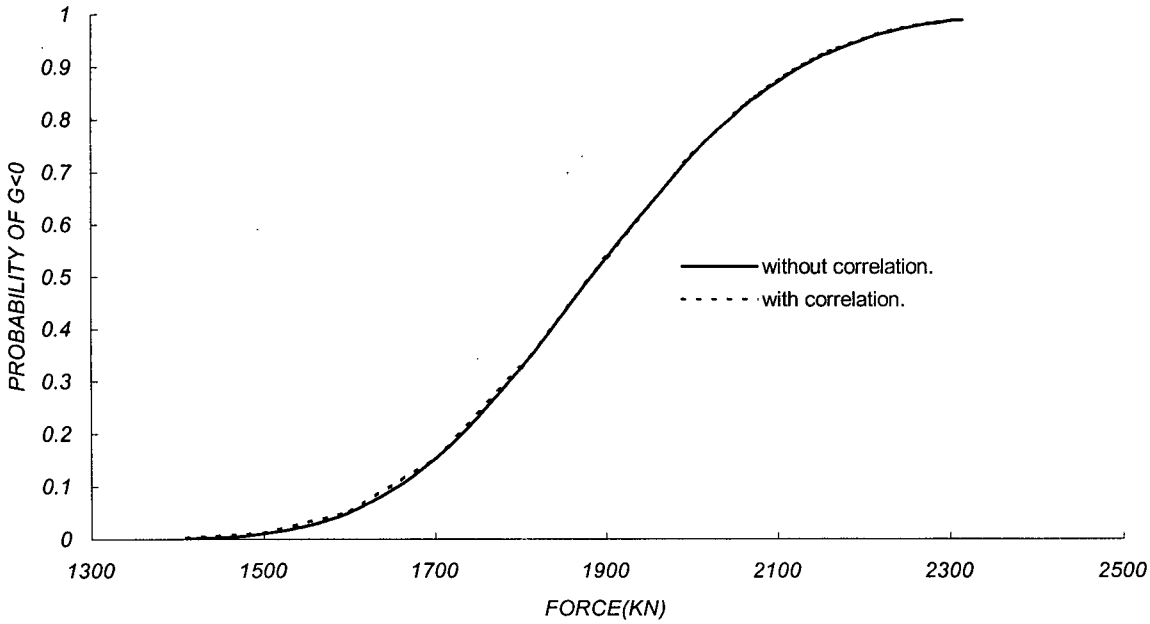


Fig.18 Cumulative distribution function (Hollow, thickness=10mm)

The maximum force-displacement relationship for the piles shown in Fig.17 and 18 involve some plastic behaviour as we have seen previously. Not only the soil property variation but also the plastic behaviour in the pile influence, in this case, the variability in maximum load.

It may be concluded that variability in soil properties has a substantial effect on the variability of the maximum load, but that correlations of soil properties with depth of the pile may not be important to the same extent.

5 CONCLUSIONS AND FUTURE RESEARCH

A non-linear finite element analysis, CFTPILE, has been presented for the calculation of CFT pile response with application of static cyclic loading to the pile cap. It was also investigated how the maximum force is influenced by the uncertainty in soil properties using the computer program RELAN.

Although only static cyclic loading was dealt with in this paper, the approach can also be used to find the pile structure response under dynamic excitation such as an earthquake. Fig.19 shows a pile subjected to an earthquake with ground acceleration a(t).

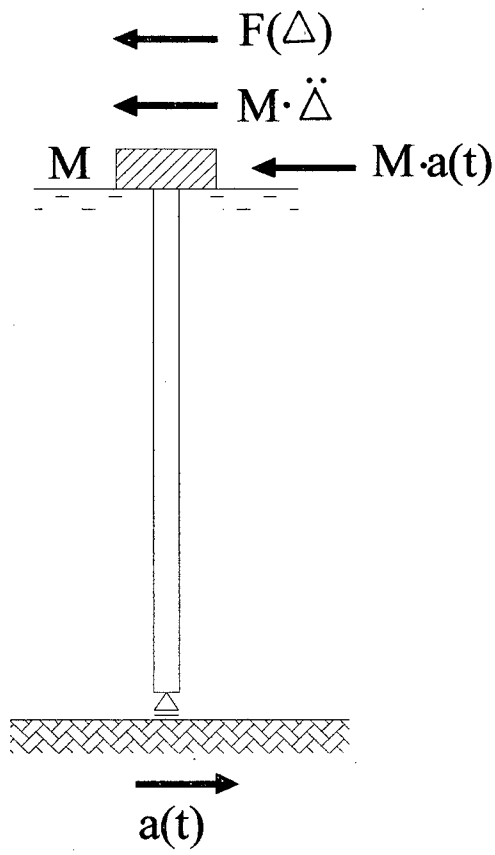


Fig.19 Horizontal forces acting on mass

If a structure is assumed to have a mass M at the pile cap and a displacement Δ caused by earthquake with acceleration a(t) at the base, the equation of motion will be

$$M \cdot \ddot{\Delta} + F(\Delta) = -M \cdot a(t) \qquad \text{Eq. 17}$$

This approach can be used only if the free-field soil displacement is assumed to be uniform or independent of depth, an assumption which may have to be corrected for long piles. This may be a topic for future research. However, this thesis has made a contribution to the calculation of the force $F(\Delta)$, which changes with the earthquake demand Δ.

REFERENCES

Alder, H.L., Roessler, E.B. (1977) Introduction to Probability and Statistics, W. H. Freeman and Company

Attard, M.M., Setunge,S. (1996) "Stress-Strain Relationship of Confined and Unconfined Concrete" ACI Material Journal, V.93, No.5, Sep.-Oct.

Benjamin, Jack R., Cornell, C. Allin, (1970) Probability, Statistics, and Decision for Civil Engineers, McGraw-Hill

Chandrupatla, T.R., Belegundu,A.D.,(1997) Introduction to Finite Elements in Engineering, Prentice-Hall, 2nd Edition.

Eisley, J. G.,(1989) Mechanics of Elastic Structures, Classical and Finite Element Methods., Prentice-Hall, Inc.

Foschi, R.O, Folz, B., Yao, F., Li, H., Baldwin, J. (2001). RELAN: Reliability Analysis Software. Department of Civil Engineering, University of British Columbia, Vancouver, B.C.

Foschi, R.O (2000). "Modeling Hysteretic Response For Reliability-Based Design in Earthquake Engineering"

Geschwindner, L.F., Disque, R.O., Bjorhovde, R.(1994) Load and Resistance Factor Design of Steel Structures, Prentice Hall

Hajjar,J.F., Gourley, B.C.,(1997) "A Cyclic Nonlinear Model for Concrete-Filled Tubes. I: Formulation",ASCE, J.of Structural Engineering, Vol.123, No.6, June

Imran, I., Pantazopoulou,S.J.,(1996) "Experimental Study of Plain Concrete under Triaxial Stress", ACI Material Journal, V.93, No.6, Nov.-Dec.

Kent, D.C., Park, R. (1971)."Flexural members with confined concrete" J. Structural Division. ASCE. No.ST7, July.1971.pp1969-1989

Prion, H.G.L, Boehme, J.,(1994) "Beam-column behaviour of steel tubes filled with high strength concrete" Can. J. Civ. Eng. 21, 207-218

Roeder , Charles W., Cameron,B.,Brown,C.B.(1999) "Composite Action in Concrete Filled Tubes" ASCE, J.of Structural Engineering, Vol. 125, No. 5, May

Shah, S. P., Fafitis, A., Arnold, R. (1983) " Cyclic Loading of Spirally Reinforced Concrete" J. Structural Engineering, Vol. 109, No. 7, July.

Tomii, M., Sakino, K. (1979). "Elasto-Plastic Behavior of Concrete Filled Square Steel Tubular Beam-Columns" Trans. of A.I.J. No.280, June.

Uy Brian (2000) "Strength of Concrete Filled Steel Box Columns Incorporating Local Buckling", ASCE, J.of Structural Engineering, Vol.126, No.3, March

Wright, H.D.(1995) "Local Stability of Filled and Encased Steel Section", ASCE, J.of Structural Engineering, Vol. 121, No. 10, Oct.

Yang, T.Y.(1986) Finite Element Analysis, Prentice-Hall

Yankelevsky, D.Z., Reinhardt, H.W.(1987) "Response of Plain Concrete to Cyclic Tension", ACI Materials Journal, Sep.- Oct.

Yan, L., and Byrne, P.M. (1992). "Lateral pile response to monotonic pile head loading" Canadian Geotechnical Journal, 29:955-970

Zhang Weizi, Shahrooz, B.M.(1999), "Strength of Short and Long Concrete-Filled Tubular Columns", ACI Structural Journal, V.96, No.2, March-April

APPENDIX A

Beam Finite Element Shape Functions

Shape functions for w

$$M_0(1,\xi) = (8 - 15\xi + 10\xi^3 - 3\xi^5)/16$$

$$M_0(2,\xi) = (5 - 7\xi - 6\xi^2 + 10\xi^3 + \xi^4 - 3\xi^5)(\Delta/32)$$

$$M_0(3,\xi) = (1 - \xi - 2\xi^2 + 2\xi^3 + \xi^4 - \xi^5)(\Delta^2/64)$$

$$M_0(4,\xi) = 0$$

$$M_0(5,\xi) = 0$$

$$M_0(6,\xi) = (8 + 15\xi - 10\xi^3 + 3\xi^5)/16$$

$$M_0(7,\xi) = (-5 - 7\xi + 6\xi^2 + 10\xi^3 - \xi^4 - 3\xi^5)(\Delta/32)$$

$$M_0(8,\xi) = (1 + \xi - 2\xi^2 - 2\xi^3 + \xi^4 + \xi^5)(\Delta^2/64)$$

$$M_0(9,\xi) = 0$$

$$M_0(10,\xi) = 0$$

Shape functions for w'

$$M_1(1,\xi) = (-15 + 30\xi^2 - 15\xi^4)(2/16\Delta)$$

$$M_1(2,\xi) = (-7 - 12\xi + 30\xi^2 + 4\xi^3 - 15\xi^4)/16$$

$$M_1(3,\xi) = (-1 - 4\xi + 6\xi^2 + 4\xi^3 - 5\xi^4)(\Delta/32)$$

$$M_1(4,\xi) = 0$$

$$M_1(5,\xi) = 0$$

$$M_1(6,\xi) = (15 - 30\xi^2 + 15\xi^4)(2/16\Delta)$$

$$M_1(7,\xi) = (-7 + 12\xi + 30\xi^2 - 4\xi^3 - 15\xi^4)/16$$

$$M_1(8,\xi) = (1 - 4\xi - 6\xi^2 + 4\xi^3 + 5\xi^4)(\Delta/32)$$

$M_1(9,\xi) = 0$

$M_1(10,\xi) = 0$

Shape functions for w"

$M_2(1,\xi) = (60\xi - 60\xi^3)/(4/16\Delta^2)$

$M_2(2,\xi) = (-12 + 60\xi + 12\xi^2 - 60\xi^3)(2/16\Delta)$

$M_2(3,\xi) = (-4 + 12\xi + 12\xi^2 - 20\xi^3)/16$

$M_2(4,\xi) = 0$

$M_2(5,\xi) = 0$

$M_2(6,\xi) = (-60\xi + 60\xi^3)/(4/16\Delta^2)$

$M_2(7,\xi) = (12 + 60\xi - 12\xi^2 - 60\xi^3)(2/16\Delta)$

$M_2(8,\xi) = (-4 - 12\xi + 12\xi^2 + 20\xi^3)/16$

$M_2(9,\xi) = 0$

$M_2(10,\xi) = 0$

Shape functions for u

$N_0(1,\xi) = 0$

$N_0(2,\xi) = 0$

$N_0(3,\xi) = 0$

$N_0(4,\xi) = (2 - 3\xi + \xi^3)/4$

$N_0(5,\xi) = (1 - \xi - \xi^2 + \xi^3)/(\Delta/8)$

$N_0(6,\xi) = 0$

$N_0(7,\xi) = 0$

$N_0(8,\xi) = 0$

$N_0(9,\xi) = (2 + 3\xi - \xi^3)/4$

$N_0(10,\xi) = (-1 - \xi + \xi^2 + \xi^3)/(\Delta/8)$

Shape functions for u'

$N_1(1,\xi) = 0$

$N_1(2,\xi) = 0$

$N_1(3,\xi) = 0$

$N_1(4,\xi) = (-3 + 3\xi^2)/(2\Delta)$

$N_1(5,\xi) = (-1 - 2\xi + 3\xi^2)/4$

$N_1(6,\xi) = 0$

$N_1(7,\xi) = 0$

$N_1(8,\xi) = 0$

$N_1(9,\xi) = (3 - 3\xi^2)/(2\Delta)$

$N_1(10,\xi) = (-1 + 2\xi + 3\xi^2)/4$

APPENDIX B

CFTPILE

Source Code

```
      DIMENSION A(10000)
C
C     ----------------------------------------------------------------
C
C  *   INPUT
C     UNIT FOR GEOMETRIC VARIABLES AND MATERIAL PROPERTIES : MM,GPA
C     SP:BEAM SPAN
C     NCROSS: BEAM CROSS-SECTION TYPE
C             IF CROSS-SECTION IS RECTANGULAR ENTER 0
C             IF CROSS-SECTION IS CIRCULAR ENTER    1
C             IF CROSS-SECTION IS ANNULAR ENTER     2
C             if cross-section is circular tube with concrete
C             inside enter 3
C     D: DIAMETER
C     DIN:INSIDE DEAMETER
C     E:STEEL YOUNG'S MODULUS
C     SY:STEEL YIELD STRESS
C     CNT:CONCRETE COMPRESSIVE STRENGTH
C     NLAYER: NO. OF LAYER
C     NPLAY:  THE NO.OF LAYERS WITH DIFFERENT PROPERTIES
C     NEMAX: SOIL MAX. YOUNG'S MODULUS
C     DPT: SOIL DEPTH WHERE THE NEMAX IS TAKEN
C     ITYP: LAYER TYPE
C     IBEG: THE NO. OF THE FIRST ELEMENT IN THE LAYER
C     IFIN: THE NO. OF THE LAST ELEMENT IN THE LAYER
C     TL: THE LAYER THICKNESS
C     NGX:THE NO. OF GAUSS POINT IN THE X-DIRECTION
C     NGY:THE NO. OF GAUSS POINT IN THE Y-DIRECTION
C     FX:AXIALLY APPLIED LOAD
C     NNBC:THE NO. OF NODES WITH SUPPORT CONDITIONS
C          NBC:THE NODE NO.
C          KBC:NO. OF SUPPORT CONDITION AT THE NODE
C          IBC:CODES FOR THE SPECIFIED SUPPORT CONDITION
C              FOR W=1
C              FOR W'=2
C              FOR W"=3
C              FOR U=4
C              FOR U'=5
C     NSPF:THE NODE NO. WITH THE SPECIFIED DISPL.
C     TOLF:TOLERANCE FOR THE OUT-OF-BALANCE VECTOR
C     TOLX:TOLERANCE FOR THE LENGTH OF THE CHANGE IN DEFORMATION VECTOR

      OPEN (UNIT=2,FILE=NAME,STATUS='OLD')
      READ (2,40) TITLE
      READ (2,*) SP
      READ (2,*) NCROSS
      IF (NCROSS.EQ.0) READ (2,*) D,B0
      IF (NCROSS.EQ.1) READ (2,*) D
      IF (NCROSS.EQ.2) READ (2,*) D,DIN
      IF (NCROSS.EQ.3) READ (2,*) D,DIN
C     if (ncross.eq.3) read (2,*) d,din
      READ (2,*) E,SY
      if (ncross.eq.3) read (2,*) cnt
      read(2,*)medium
      READ (2,*) NLAYER,NPLAY
      read(2,*) alpha,beta,nemax
      do 451 i=1,nemax
      read(2,*) emax(i),dpt(i)
 451  continue
      NELEM=0
      DO 460 I=1,NLAYER
        READ (2,*) ITYP(I),IBEG(I),IFIN(I),TL(I)
        NELEM1=IFIN(I)-IBEG(I)+1
        NELEM=NELEM+NELEM1
        DELTA(I)=TL(I)/NELEM1
 460  CONTINUE
```

```
      READ (2,*) NGX,NGY
      CALL GAUSS (NGX, EGX, HGX, IERR)
      CALL GAUSS (NGY, EGY, HGY, IERR)
      READ (2,*) FX
      READ (2,*) NNBC
      IF (NNBC.EQ.0) GO TO 480
      DO 470 I=1,NNBC
        READ (2,*) NBC(I),KBC(I),(IBC(I,J),J=1,KBC(I))
 470  CONTINUE
 480  READ (2,*) NLDIS
      IF (NLDIS.EQ.0) READ (2,*) NSPF
      IF (NLDIS.GT.0) READ (2,*) (ILAY(I),I=1,NLDIS)
      READ (2,*) TOLX, TOLF
      CLOSE (2)
 490  WRITE (*,500)
 500  FORMAT(/' ENTER NAME OF FILE WITH INPUT DISPLACEMENT HISTORY'/)
      READ (*,40) NAME1
      OPEN (UNIT=3,FILE=NAME1,STATUS='OLD')
      READ (3,*) NSTEP
      READ (3,*) (A(I),I=1,NSTEP)
      CLOSE (3)
      WRITE (*,510)
 510  FORMAT(/' ENTER STEP NUMBER AT WHICH TO SHOW SHAPE'/)
      READ (*,*) NSTP
      WRITE (*,520)
 520  FORMAT(/' ENTER 1 TO SHOW COMPREHENSIVE INTERMEDIATE RESULTS'/
     1        ' ENTER 0 TO SKIP'/)
      READ (*,*) NSCRC
      WRITE (*,521)
 521  FORMAT(/' ENTER 1 TO SHOW SUMMARY OF INTERMEDIATE RESULTS'/
     1        ' ENTER 0 TO SKIP'/)
      READ (*,*) NSCR
      DO 530 I=1,NLAYER
        DO 530 J=IBEG(I),IFIN(I)
 530    IDISP(J)=0
      IF (NLDIS.EQ.0) GO TO 550
      DO 540 I=1,NLDIS
        DO 540 J=IBEG(ILAY(I)),IFIN(ILAY(I))
 540    IDISP(J)=1
 550  DO 580 I=1,NGY
        IF (NCROSS.EQ.0) B(I)=B0
        IF (NCROSS.EQ.1) B(I)=D*DSQRT(1.0D0-EGY(I)**2)
        IF (NCROSS.EQ.2) GO TO 560
        if (ncross.eq.3) go to 560
        GO TO 580
 560    RATD=DIN/D
        IF (DABS(EGY(I)).GE.RATD) GO TO 570
        B(I)=D*(DSQRT(1.0D0-EGY(I)**2)-DSQRT(RATD**2-EGY(I)**2))
        if (ncross.eq.3) bc(i)=din*(dsqrt(1-egy(i)**2))
        GO TO 580
 570    B(I)=D*DSQRT(1.0D0-EGY(I)**2)
 580  CONTINUE
      WRITE (*,590)
 590  FORMAT(/' ENTER NAME OF OUTPUT FILE'/)
      READ (*,40) NAME2
      WRITE (*,600)
 600  FORMAT(/' ENTER MULTIPLICATION FACTOR FOR CALCULATED LOADS'/)
      READ (*,*) FLOAD
      OPEN (UNIT=4,FILE=NAME2,STATUS='UNKNOWN')
      P1 = 0.0D0
      WRITE(4,130) P1, P1
C
C    ------------------------------------------------------------
C
C  *  MAIN PROGRAM, SIZE OF VECTORS
C
```

```
      NEQ=(NELEM+1)*5
      LHB=10
      NA=NEQ*LHB
C
C  *  INITIALLY, SET CONVERGED VECTORS TO ZERO
C
      DO 610 I=1,NEQ
        R(I)=0.0D0
 610    X0C(I)=0.0D0
      R(NEQ-1)=FX
      DO 650 II=1,NLAYER
        DO 640 I=IBEG(II),IFIN(II)
          DO 630 IX=1,NGX
            D0PC(I,IX)=0.0D0
            D0NC(I,IX)=0.0D0
            DO 620 IY=1,NGY
              S0C(I,IX,IY)=0.0D0
              EPS0C(I,IX,IY)=0.0D0
              rc(i,ix,iy)=0.0d0
 620        CONTINUE
 630      CONTINUE
 640    CONTINUE
 650  CONTINUE
C
C  *  OBTAINS SHAPE FUNCTIONS AT INTEGRATION POINTS
C
      DO 670 II=1,NLAYER
        CALL SHAPES (N0L, N1L, M0L, M1L, M2L, NGX, EGX, DELTA(II))
        DO 660 I=1,10
          DO 660 J=1,NGX
          N1(II,I,J)=N1L(I,J)
          M0(II,I,J)=M0L(I,J)
          M1(II,I,J)=M1L(I,J)
          M2(II,I,J)=M2L(I,J)
 660    CONTINUE
 670  CONTINUE
C
      IF (NLDIS.EQ.0) KA=(NSPF-1)*5+1
      OPEN (UNIT=8,FILE='SHAPEP',STATUS='UNKNOWN')
      OPEN (UNIT=9,FILE='SHAPEN',STATUS='UNKNOWN')
      OPEN (UNIT=7,FILE='SHAPEW',STATUS='UNKNOWN')
C
C  *  STARTS SOLUTION FOR EACH STEP
C
      IF (NSCRC.EQ.0) WRITE (*,680)
 680  FORMAT(/' ENTER STEP AT WHICH SWITCHING TO SHOWING'/
     1        ' COMPREHENSIVE INTERMEDIATE RESULTS'/)
      IF (NSCRC.EQ.0) READ (*,*) NSTPR
C
      DELTAX = 0.0D0
      DELTAF = 0.0D0
      A0=0.0D0
      cs = 0.0d0
      cs2 = 0.0d0
      dcsdep = 0.0d0
      r0 = 0.0d0
      DO 1490 NST=1,NSTEP
        IF (NSCRC.EQ.0.AND.NST.EQ.NSTPR) NSCRC=1
        NSINT=1
        A1=A(NST)
        AA=A1
C
C
C  STARTS ITERATIONS WITHIN THE STEP A0 TO AA
C
C
```

```
C  Sets vectors to the initial values for the step, starting from
C  the last converged values
C
        NDE=0
 690    NITER=0
        IF (NSCRC.EQ.1) WRITE (*,700) NST,NDE,A0,A1,AA
 700    FORMAT(' DATA=',I4,' NDE=',I3/
     1          ' A0=',F8.3,' A1=',F8.3,' AA=',F8.3/)
        DO 710 I=1,NEQ
        X0(I)=X0C(I)
 710    CONTINUE
 718    DO 740 JL=1,NLAYER
          DO 730 IE=IBEG(JL),IFIN(JL)
            DO 720 IX=1,NGX
              D0P(IE,IX)=D0PC(IE,IX)
              D0N(IE,IX)=D0NC(IE,IX)
 720        CONTINUE
 730      CONTINUE
 740    CONTINUE
C
C OBTAIN THE REACTION FORCES AND STRESSES FOR THE CURRENT VECTOR {X0}
C
 750    DO 810 JL=1,NLAYER
          Q00=Q0(ITYP(JL))
          Q10=Q1(ITYP(JL))
          Q40=Q4(ITYP(JL))
          DMAX0=DMAX(ITYP(JL))
          XK0=XK(ITYP(JL))
          PMAX0=PMAX(ITYP(JL))
      zl=0.0d0
      if(jl.eq.1) go to 752
      do 751 i=1,jl-1
      zl=zl+tl(i)
 751  continue
 752  continue
          DO 800 IE=IBEG(JL),IFIN(JL)
            DO 760 I=1,10
              JE=(IE-1)*5+I
 760          XE(I)=X0(JE)
            DO 790 IX=1,NGX
              W=0.0D0
              EPS1=0.0D0
              EPS2=0.0D0
              EPS3(IE,IX)=0.0D0
c     xie : the depth under consideration. measured from the bottom
      xie=zl+(ie-ibeg(jl)+1)*delta(jl)-(1.0+egx(ix))*delta(jl)/2.0d0
c     interpolate to find Emax along the pile
      dpt(0)=0.0d0
      emax(0)=0.0d0
      temax=0.0d0
      dpt(nemax+1)=sp
      emax(nemax+1)=emax(nemax)
      xie=sp-xie
      do 761 i=0,nemax
      if(dpt(i).eq.dpt(i+1)) then
      write(*,*)' INPUT DATA ERROR. DPT(I) MUST BE DIFFERENT FROM DPT
     1(I+1)'
      pause
      else
      end if
      if(xie.eq.dpt(i)) then
      temax=emax(i)
      go to 762
      else
      end if
      if(xie.gt.dpt(i).and.xie.lt.dpt(i+1)) then
```

```
      slop=(emax(i+1)-emax(i))/(dpt(i+1)-dpt(i))
      temax=emax(i)+(xie-dpt(i))*slop
      go to 762
      else
      end if
 761  continue
 762            DO 770 K=1,10
               W=W+M0(JL,K,IX)*XE(K)
               EPS1=EPS1+N1(JL,K,IX)*XE(K)
               EPS2=EPS2+M2(JL,K,IX)*XE(K)
 770           EPS3(IE,IX)=EPS3(IE,IX)+M1(JL,K,IX)*XE(K)
             IF (IDISP(IE).EQ.0) WT=W
             IF (IDISP(IE).EQ.1) WT=W-AA
             AW=DABS(WT)
             IF (WT.GE.0.0D0) D0=D0PC(IE,IX)
             IF (WT.LT.0.0D0) D0=D0NC(IE,IX)
             IF (WT.GE.0.0D0) SW0(IE,IX)=1.0D0
             IF (WT.LT.0.0D0) SW0(IE,IX)=-1.0D0
             CALL PYSOIL(AW,ALPHA,BETA,D,TEMAX,D0,P,DPDW)
C
 771         IF (WT.GE.0.0D0) D0P(IE,IX)=D0
             IF (WT.LT.0.0D0) D0N(IE,IX)=D0
             P0(IE,IX)=P
             DPDW0(IE,IX)=DPDW
             DO 780 IY=1,NGY
               Y=EGY(IY)*D/2.0D0
               EPS=EPS1-Y*EPS2+(EPS3(IE,IX)**2)/2.0D0
               ST0=S0C(IE,IX,IY)
               EP0=EPS0C(IE,IX,IY)
               r0=rc(ie,ix,iy)
               CALL STRESS (ST0, EP0, EPS, E, SY, S, DSDEPS)
      IF(NCROSS.EQ.3) THEN
               CALL CONCT(NCROSS,D,DIN,R0,EPS,CNT,CS,DCSDEP)
      ELSE
      ENDIF
C
 779           rr(ie,ix,iy) = r0
               cs0(ie,ix,iy) = cs
               dcsde0(ie,ix,iy) = dcsdep
               S0(IE,IX,IY)=S
               EPS0(IE,IX,IY)=EPS
               DSDE0(IE,IX,IY)=DSDEPS
 780         CONTINUE
 790       CONTINUE
 800     CONTINUE
 810   CONTINUE
C
C  *  FOR EACH ELEMENT, CONSTRUCT TANGENT STIFFNESS MATRIX AND
C     RIGHT HAND SIDE, AND ADD TO GLOBALS
C
       DO 820 I=1,NA
         C(I)=0.0D0
 820   CONTINUE
       DO 900 JL=1,NLAYER
         DO 890 IE=IBEG(JL),IFIN(JL)
           DO 860 I=1,10
             DO 850 J=1,I
               CE(I,J)=0.0D0
               DO 840 IX=1,NGX
                 DPDW=DPDW0(IE,IX)
                 CE(I,J)=CE(I,J)+HGX(IX)*DPDW*M0(JL,I,IX)*M0(JL,J,IX)*
     1            DELTA(JL)/2.0D0
                 FACI=N1(JL,I,IX)+M1(JL,I,IX)*EPS3(IE,IX)
                 FACJ=N1(JL,J,IX)+M1(JL,J,IX)*EPS3(IE,IX)
                 DO 830 IY=1,NGY
                   Y=EGY(IY)*D/2.0D0
```

```
                    DSDEPS=DSDE0(IE,IX,IY)
                    S=S0(IE,IX,IY)
                    dcsdep = dcsde0(ie,ix,iy)
                    FAC1=FACJ-Y*M2(JL,J,IX)
                    FAC2=FACI-Y*M2(JL,I,IX)
                    CE(I,J)=CE(I,J)+HGX(IX)*HGY(IY)*(DSDEPS*FAC1*FAC2*
     1               B(IY)+S*M1(JL,I,IX)*M1(JL,J,IX)*B(IY))*(DELTA(JL)*
     2               D)/4.0D0
     3               + hgx(ix)*hgy(iy)*(dcsdep*fac1*fac2*bc(iy)+cs0(ie,
     4               ix,iy)*m1(jl,i,ix)*m1(jl,j,ix)*bc(iy))*(delta(jl)
     5               *din/4.0d0)
 830              CONTINUE
 840            CONTINUE
 850          CONTINUE
 860        CONTINUE
            DO 880 I=1,10
              II=(IE-1)*5+I
              DO 870 J=1,I
                JJ=(IE-1)*5+J
                IJ=(JJ-1)*(LHB-1)+II
 870            C(IJ)=C(IJ)+CE(I,J)
 880        CONTINUE
 890      CONTINUE
 900    CONTINUE

C
 902    DO 910 I=1,NEQ
 910      PSI(I)=0.0D0
        DO 970 JL=1,NLAYER
          DO 960 IE=IBEG(JL),IFIN(JL)
            DO 940 I=1,10
              PSIE(I)=0.0D0
              DO 930 IX=1,NGX
                SW=SW0(IE,IX)
                P=P0(IE,IX)
                PSIE(I)=PSIE(I)+HGX(IX)*P*SW*M0(JL,I,IX)*DELTA(JL)/
     1           2.0D0
                FACI=N1(JL,I,IX)+M1(JL,I,IX)*EPS3(IE,IX)
                DO 920 IY=1,NGY
                  Y=EGY(IY)*D/2.0D0
                  S=S0(IE,IX,IY)
                  FAC2=FACI-Y*M2(JL,I,IX)
                  PSIE(I)=PSIE(I)+HGX(IX)*HGY(IY)*S*FAC2*B(IY)*
     1             (DELTA(JL)*D)/4.0D0
     2             +hgx(ix)*hgy(iy)*cs0(ie,ix,iy)*fac2*bc(iy)*
     3             (delta(jl)*din)/4.0d0
 920            CONTINUE
 930          CONTINUE
 940        CONTINUE
            DO 950 I=1,10
              II=(IE-1)*5+I
              PSI(II)=PSI(II)+PSIE(I)
 950        CONTINUE
 960      CONTINUE
 970    CONTINUE
        DO 980 I=1,NEQ
          RM(I)=R(I)-PSI(I)
 980    CONTINUE
C
C
C  *  Enforces W displacement and then introduces support conditions
C
        IF (NLDIS.GT.0) GO TO 1030
C
         IF (KA.EQ.1) GO TO 1000
         K1=KA-LHB+1
```

```
          K2=KA-1
          IF (K1.LE.0) K1=1
          DO 990 J=K1,K2
           II=(J-1)*(LHB-1)+KA
           RM(J)=RM(J)-C(II)*(AA-X0(KA))
 990       C(II)=0.0D0
 1000     IF (KA.EQ.NEQ) GO TO 1020
          K1=KA+1
          K2=KA+LHB-1
          IF (K2.GT.NEQ) K2=NEQ
          DO 1010 I=K1,K2
           II=(KA-1)*(LHB-1)+I
           RM(I)=RM(I)-C(II)*(AA-X0(KA))
 1010      C(II)=0.0D0
 1020     KK=(KA-1)*(LHB-1)+KA
          C(KK)=1.0D0
          RM(KA)=AA-X0(KA)
C
 1030    IF (NNBC.EQ.0) GO TO 1100
         DO 1090 IB=1,NNBC
           DO 1080 J=1,KBC(IB)
             K=(NBC(IB)-1)*5+IBC(IB,J)
             IF (K.EQ.1) GO TO 1050
             K1=K-LHB+1
             K2=K-1
             IF (K1.LE.0) K1=1
             DO 1040 I=K1,K2
               II=(I-1)*(LHB-1)+K
               RM(I)=RM(I)+C(II)*X0(K)
 1040          C(II)=0.0D0
 1050        IF (K.EQ.NEQ) GO TO 1070
             K1=K+1
             K2=K+LHB-1
             IF (K2.GT.NEQ) K2=NEQ
             DO 1060 I=K1,K2
               II=(K-1)*(LHB-1)+I
               RM(I)=RM(I)+C(II)*X0(K)
 1060          C(II)=0.0D0
 1070        KK=(K-1)*(LHB-1)+K
             C(KK)=1.0D0
             RM(K)=-X0(K)
 1080      CONTINUE
 1090    CONTINUE
 1100    CONTINUE
C
C   Computes magnitudes of the out-of-balance vector,
C   and checks convergence if NITER > 0
C
         RMC=0.0D0
         DO 1110 I = 1, NEQ
         RMC=RMC+RM(I)**2
 1110    CONTINUE
         IF (NITER.EQ.0) then
       RMP = RMC
       ncount=0
       rmcp=rmc
       endif
         IF (NITER.EQ.0) GO TO 1113
C
C  * Checks convergence
C
       if(rmcp.lt.rmc) then
       ncount=ncount+1
c      pause
       endif
         KF=0
```

```
        KM=0
        IF (RMC.LT.TOLF) KF = 1
        IF (RMAX.LT.TOLX) KM = 1
        IF (NSCRC.EQ.1) WRITE(*,1150) NST,NSINT,NDE,NITER,
     1                                RMP,RMC,KF,
     1                                RMAX,KM
 1150 FORMAT(/' DATA =',I5,' INTERMEDIATE STEP=',I4,' NDE=',I4,
     1 ' NITER =',I3/' INIT. BAL. VECTOR=',E12.5,
     1 ' CURRENT BAL. VECTOR=',E12.5,'  KF=',I2/
     1 ' CURRENT DELTA(X)=',E12.5,'  KM=',I2)
c        IF (NSCRC.EQ.1) PAUSE
        IF (KF.EQ.1.AND.KM.EQ.1) GO TO 1190
C
C
 1113   CALL DECOMP (NEQ, LHB, C, IERROR)
        IF (IERROR.EQ.1) THEN
          IF (NSCRC.EQ.1) WRITE (*,1130)
 1130   FORMAT(/' DECOMPOSITION FAILED'/)
C
          GO TO 1170
        END IF
        CALL SOLV (NEQ, LHB, C, RM)
C
C  * Finds length of displacement correction vector
C
        RMAX=0.0D0
        DO 1140 JL=1,NLAYER
         DO 1140 IE=IBEG(JL),IFIN(JL)
          JA=(IE-1)*5+1
          RMAX = RMAX + RM(JA)**2
          IF (IE.EQ.IFIN(NLAYER)) THEN
          JA = IE*5 + 1
          RMAX = RMAX + RM(JA)**2
          END IF
 1140   CONTINUE
C
        NITER = NITER + 1
        DO 1160 I=1,NEQ
        X0(I)=X0(I)+RM(I)
 1160   CONTINUE
c        IF (NITER.EQ.1000) GO TO 1170
      if(ncount.eq.10) go to 1170
      rmcp=rmc
        GO TO 750
C
 1170   AA=(A0+AA)/2.0D0
        NDE=NDE+1
C
      IF (NDE.GT.11) THEN
          WRITE (*,1180) NST,NSINT,NDE-1,NITER,A0,AA
 1180     FORMAT(/' CANNOT FIND SOLUTION AT STEP NST=',I4/
     1            ' NSINT= ',I4,' NDE= ',I4,' NITER= ',I4/
     1           ' A0= ',E12.5,' AA= ',E12.5)
          GO TO 1495
        END IF
        GO TO 690
C
 1190   IF (AA.EQ.A1) GO TO 1260
        IF (AA.NE.A1) GO TO 1200
C
C * Convergence is achieved at AA not equal to A1
C * NSINT is a counter for the number of intermediate, converged steps
C
C
 1200   DO 1210 I=1,NEQ
 1210     X0C(I)=X0(I)
```

```
        DO 1250 II=1,NLAYER
          DO 1240 IE=IBEG(II),IFIN(II)
            DO 1230 IX=1,NGX
              D0PC(IE,IX)=D0P(IE,IX)
              D0NC(IE,IX)=D0N(IE,IX)
              DO 1220 IY=1,NGY
                S0C(IE,IX,IY)=S0(IE,IX,IY)
                EPS0C(IE,IX,IY)=EPS0(IE,IX,IY)
                rc(ie,ix,iy) = rr(ie,ix,iy)
 1220         CONTINUE
 1230       CONTINUE
 1240     CONTINUE
 1250   CONTINUE
C
        A0=AA
        AA= A1
        NSINT=NSINT+1
      if(nsint.eq.20) go to 1490
        IF (RMAX.GT.DELTAX) DELTAX = RMAX
        IF (RMC.GT.DELTAF) DELTAF = RMC
        NDE=0
        flag=0.0d0
        GO TO 690
C
C  *  Convergence has been achieved at AA = A1
C
 1260   A0=AA
        DO 1270 I=1,NEQ
 1270     X0C(I)=X0(I)
        DO 1310 II=1,NLAYER
          DO 1300 IE=IBEG(II),IFIN(II)
            DO 1290 IX=1,NGX
              D0PC(IE,IX)=D0P(IE,IX)
              D0NC(IE,IX)=D0N(IE,IX)
              DO 1280 IY=1,NGY
                S0C(IE,IX,IY)=S0(IE,IX,IY)
                EPS0C(IE,IX,IY)=EPS0(IE,IX,IY)
                rc(ie,ix,iy) = rr(ie,ix,iy)
 1280         CONTINUE
 1290       CONTINUE
 1300     CONTINUE
 1310   CONTINUE
        IF (RMAX.GT.DELTAX) DELTAX = RMAX
        IF (RMC.GT.DELTAF) DELTAF = RMC
C
C  *  Computes the load (FY0 for fixed layers, FY1 for moving layers)
C
        IF (NLDIS.GT.0) GO TO 1350
        FY0=0.0D0
        DO 1340 JL=1,NLAYER
          DO 1330 IE=IBEG(JL),IFIN(JL)
            DO 1320 IX=1,NGX
              FY0=FY0+HGX(IX)*P0(IE,IX)*SW0(IE,IX)*DELTA(JL)/2.0D0
 1320       CONTINUE
 1330     CONTINUE
 1340   CONTINUE
        GO TO 1390
C
 1350   FY1=0.0D0
        FY0=0.0D0
        DO 1380 JL=1,NLAYER
          DO 1370 IE=IBEG(JL),IFIN(JL)
            DO 1360 IX=1,NGX
              IF (IDISP(IE).EQ.1) FY1=FY1+HGX(IX)*P0(IE,IX)*SW0(IE,IX)*
     1         DELTA(JL)/2.0D0
              IF (IDISP(IE).EQ.0) FY0=FY0+HGX(IX)*P0(IE,IX)*SW0(IE,IX)*
```

```
     1          DELTA(JL)/2.0D0
 1360        CONTINUE
 1370      CONTINUE
 1380    CONTINUE
C
 1390    IF (NLDIS.EQ.0) WRITE (4,130) AA,FY0*FLOAD
         IF (NLDIS.GT.0) WRITE (4,130) AA,FY0*FLOAD
        IF (NSCR.EQ.1) WRITE (*,1400) NST,NSINT,NITER, RMAX,KM, RMC, KF
 1400   FORMAT(' FINISHED DATA No. ',I5,'  TOTAL STEPS =',I4,
     1  '      NITER LAST STEP =',I3/
     1  '      DELTA(X)=',E12.5,' KM=',I2,' DELTA(F)=',E12.5,' KF=',I2)
         IF (NLDIS.GT.0) WRITE (*,1420) NST,AA,FY0*FLOAD,FY1*FLOAD
         IF (NLDIS.EQ.0) WRITE (*,1410) NST,AA,FY0*FLOAD
 1410 FORMAT(' DATA =',I5,'   A=',E14.6,'  F=',E14.6/)
 1420 FORMAT(' DATA =',I5,'   A=',E14.6,'  F0=',E14.6,'  F1=',E14.6/)
C
C  Store shape if the step is as specified
C
         IF (NST.NE.NSTP) GO TO 1490
         DO 1480 JL=1,NLAYER
           ZL=0.0D0
           IF (JL.EQ.1) GO TO 1440
           DO 1430 I=1,JL-1
             ZL=ZL+TL(I)
 1430      CONTINUE
 1440      CONTINUE
           DO 1470 IE=IBEG(JL),IFIN(JL)
             DO 1450 IX=1,NGX
               XIE=ZL+(IE-IBEG(JL)+1)*DELTA(JL)-(1.0+EGX(IX))*DELTA(JL)/
     1          2.0D0
              WRITE (8,130) DOPC(IE,(NGX+1-IX)),XIE
               WRITE (9,130) -DONC(IE,(NGX+1-IX)),XIE
 1450        CONTINUE
             IEE=(IE-1)*5+1
             XIEE=ZL+(IE-IBEG(JL))*DELTA(JL)
             WRITE (7,130) X0(IEE),XIEE
             IF (JL.EQ.NLAYER.AND.IE.EQ.IFIN(JL)) GO TO 1460
             GO TO 1470
 1460        XIEE=ZL+(IE+1-IBEG(JL))*DELTA(JL)
             IF (JL.EQ.NLAYER.AND.IE.EQ.IFIN(JL)) IEE=IE*5+1
             IF (JL.EQ.NLAYER.AND.IE.EQ.IFIN(JL)) WRITE (7,130) X0(IEE),
     1        XIEE
 1470      CONTINUE
 1480    CONTINUE
 1490 CONTINUE
      WRITE(*,1491) DELTAX, DELTAF
1491  FORMAT(' MAXIMUM DELTA(X)=',E12.5,'  MAXIMUM DELTA(F)=',E12.5/)
C
 1495 continue
      CLOSE (4)
      CLOSE (8)
      CLOSE (9)
      CLOSE (7)
      STOP
      END
C     ------------------------------------------------------------
C
      SUBROUTINE SHAPES (N0, N1, M0, M1, M2, NG, EG, DELTA)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 M0, M1, M2, N0, N1
      DIMENSION EG(9), M0(10,9), M1(10,9), M2(10,9)
      DIMENSION N0(10,9), N1(10,9)
      DO 10 I=1,NG
        X=EG(I)
        X2=X**2
        X3=X**3
```

```
      X4=X**4
      X5=X**5
C
C     SHAPES FOR W(X)
C
      M0(1,I)=(8.0D0-15.0D0*X+10.0D0*X3-3.0D0*X5)/16.0D0
      M0(2,I)=(5.0D0-7.0D0*X-6.0D0*X2+10.0D0*X3+X4-3.0D0*X5)*(DELTA/
     1  32.0D0)
      M0(3,I)=(1.0D0-X-2.0D0*X2+2.0D0*X3+X4-X5)*(DELTA**2)/64.0D0
      M0(4,I)=0.0D0
      M0(5,I)=0.0D0
      M0(6,I)=(8.0D0+15.0D0*X-10.0D0*X3+3.0D0*X5)/16.0D0
      M0(7,I)=(-5.0D0-7.0D0*X+6.0D0*X2+10.0D0*X3-X4-3.0D0*X5)*(DELTA/
     1  32.0D0)
      M0(8,I)=(1.0D0+X-2.0D0*X2-2.0D0*X3+X4+X5)*(DELTA**2)/64.0D0
      M0(9,I)=0.0D0
      M0(10,I)=0.0D0
C
C    SHAPES FOR W'(X)
C
      M1(1,I)=(-15.0D0+30.0D0*X2-15.0D0*X4)*2.0D0/(16.0D0*DELTA)
      M1(2,I)=(-7.0D0-12.0D0*X+30.0D0*X2+4.0D0*X3-15.0D0*X4)/16.0D0
      M1(3,I)=(-1.0D0-4.0D0*X+6.0D0*X2+4.0D0*X3-5.0D0*X4)*DELTA/
     1  32.0D0
      M1(4,I)=0.0D0
      M1(5,I)=0.0D0
      M1(6,I)=(15.0D0-30.0D0*X2+15.0D0*X4)*2.0D0/(16.0D0*DELTA)
      M1(7,I)=(-7.0D0+12.0D0*X+30.0D0*X2-4.0D0*X3-15.0D0*X4)/16.0D0
      M1(8,I)=(1.0D0-4.0D0*X-6.0D0*X2+4.0D0*X3+5.0D0*X4)*DELTA/32.0D0
      M1(9,I)=0.0D0
      M1(10,I)=0.0D0
C
C     SHAPES FOR W"(X)
C
      M2(1,I)=(60.0D0*X-60.0D0*X3)*4.0D0/(16.0D0*(DELTA**2))
      M2(2,I)=(-12.0D0+60.0D0*X+12.0D0*X2-60.0D0*X3)*2.0D0/(16.0D0*
     1  DELTA)
      M2(3,I)=(-4.0D0+12.0D0*X+12.0D0*X2-20.0D0*X3)/16.0D0
      M2(4,I)=0.0D0
      M2(5,I)=0.0D0
      M2(6,I)=(-60.0D0*X+60.0D0*X3)*4.0D0/(16.0D0*(DELTA**2))
      M2(7,I)=(12.0D0+60.0D0*X-12.0D0*X2-60.0D0*X3)*2.0D0/(16.0D0*
     1  DELTA)
      M2(8,I)=(-4.0D0-12.0D0*X+12.0D0*X2+20.0D0*X3)/16.0D0
      M2(9,I)=0.0D0
      M2(10,I)=0.0D0
C
C     SHAPES FOR U(X)
C
      N0(1,I)=0.0D0
      N0(2,I)=0.0D0
      N0(3,I)=0.0D0
      N0(4,I)=(2.0D0-3.0D0*X+X3)/4.0D0
      N0(5,I)=(1.0D0-X-X2+X3)*DELTA/8.0D0
      N0(6,I)=0.0D0
      N0(7,I)=0.0D0
      N0(8,I)=0.0D0
      N0(9,I)=(2.0D0+3.0D0*X-X3)/4.0D0
      N0(10,I)=(-1.0D0-X+X2+X3)*DELTA/8.0D0
C
C     SHAPES FOR U'(X)
C
      N1(1,I)=0.0D0
      N1(2,I)=0.0D0
      N1(3,I)=0.0D0
      N1(4,I)=(-3.0D0+3.0D0*X2)/(2.0D0*DELTA)
```

```
        N1(5,I)=(-1.0D0-2.0D0*X+3.0D0*X2)/4.0D0
        N1(6,I)=0.0D0
        N1(7,I)=0.0D0
        N1(8,I)=0.0D0
        N1(9,I)=(3.0D0-3.0D0*X2)/(2.0D0*DELTA)
        N1(10,I)=(-1.0D0+2.0D0*X+3.0D0*X2)/4.0D0
 10   CONTINUE
      RETURN
      END
C
C    ---------------------------------------------------------
C
      SUBROUTINE STRESS (ST0, EP0, EPS, E, SY, S, DSDEPS)
      IMPLICIT REAL*8(A-H,O-Z)
      S=ST0+E*(EPS-EP0)
      DSDEPS=E
      S1=SY+0.001D0*E*EPS
      S2=-SY+0.001D0*E*EPS
      IF (S.GE.S1) DSDEPS=0.001D0*E
      IF (S.GE.S1) S=S1
      IF (S.LE.S2) DSDEPS=0.001D0*E
      IF (S.LE.S2) S=S2
      RETURN
      END
C    ---------------------------------------------------------
C
      subroutine conct(NCROSS,d,din,r0, eps, cnt,cs, dcsdep)
      implicit real*8(a-h,o-z)
      DT=D/T
      IF(DT.LT.64.0D0) DT=64.0D0
      IF(NCROSS.EQ.3) THEN
      t=(d-din)/2.0d0
      cnt=-cnt
      ceb1=-0.012354165d0*dsqrt(-cnt)
      ec=2.0D0*cnt/ceb1
      if(eps.gt.0.0d0) then
      cs=0.0d0
      dcsdep=0.0d0
      endif
      if (eps.le.r0) go to 11
      cs = 0.0d0
      dcsdep = 0.0d0
      go to 13
 11   if (eps.lt.r0) go to 12
      cs = 0.0d0
      dcsdep = ec
      go to 13
 12   cs = ec*(eps-r0)
      dcsdep = ec
      if(eps.ge.ceb1.and.eps.le.0.0d0) then
      cs2=2.0D0*(cnt*eps)/ceb1-(cnt/ceb1**2)*eps**2
      endif
      if(eps.gt.-0.005d0.and.eps.lt.ceb1) cs2=cnt
      if(eps.ge.-0.015d0.and.eps.le.-0.005d0) then
      cs1=(1.6d0-0.025d0*DT)*cnt
      cs2=cs1+((cnt-cs1)/(-0.015d0+0.005d0))*(eps+0.005)
      endif
      if(eps.lt.-0.015d0) then
      cs2=(1.6d0-0.025d0*DT)*cnt
      endif
      if(eps.gt.0.0d0) cs2=0.0d0
      if (cs.gt.cs2) go to 13
      cs = cs2
      r0 = eps-cs/ec
      if(eps.le.0.0d0.and.eps.ge.ceb1) then
      dcsdep = 2.0d0*cnt/ceb1-2.0d0*(cnt/ceb1**2)*eps
```

```
      endif
      if(eps.ge.-0.005d0.and.eps.lt.ceb1) dcsdep=0.0d0
      if(eps.gt.-0.015d0.and.eps.lt.-0.005d0) then
      dcsdep=(-cnt+cs1)/(-0.015+0.005)
      endif
      if(eps.lt.-0.005) dcsdep=0.0d0
   13 cnt=-cnt
      endif
      return
      end
c     ------------------------------------------------------------
c

C
      SUBROUTINE PYSOIL(AW,ALPHA,BETA,D,TEMAX,D0,P,DPDW)
      IMPLICIT REAL*8(A-H,O-Z)
      IF (AW.GE.D0) GO TO 763
      P=0.0D0
      DPDW=0.0D0
      GO TO 771
  763 IF(AW.GT.D0) GO TO 764
      P=0.0D0
      DPDW=TEMAX
      GO TO 771
  764 P=TEMAX*(AW-D0)
      dpdw=temax
      x=(d/100.0d0)*alpha**(1.0d0/(1.0d0-beta))
      if(aw.ge.x) then
      pub=(temax*d*alpha*((1.0d0/d)**beta)*aw**beta)/10.0d0
      else
      pub=temax*aw
      end if
      if(p.lt.pub) go to 771
      p=pub
      if(aw.lt.x) then
      dpdw=temax
      else
      dpdw=0.1d0*temax*d*alpha*((1.0d0/d)**beta)*beta*aw**(beta-1.0d0)
      endif
      d0=aw-p/temax
  771 return
      end
c     ------------------------------------------------------------
      SUBROUTINE DECOMP (N, LHB, A, IERROR)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(2550)
      IERROR=0
      KB=LHB-1
      TEMP=A(1)
      IF (TEMP.LE.0.0D0) IERROR=1
      IF (IERROR.EQ.1) RETURN
      TEMP=DSQRT(TEMP)
      A(1)=TEMP
      DO 10 I=2,LHB
        A(I)=A(I)/TEMP
   10 CONTINUE
      DO 60 J=2,N
        J1=J-1
        IJD=LHB*J-KB
        SUM=A(IJD)
        KO=1
        IF (J.GT.LHB) KO=J-KB
        DO 20 K=KO,J1
          JK=KB*K+J-KB
          TEMP=A(JK)
          SUM=SUM-(TEMP**2)
```

```
 20     CONTINUE
        IF (SUM.LE.0.0D0) IERROR=1
        IF (IERROR.EQ.1) RETURN
        A(IJD)=DSQRT(SUM)
        DO 50 I=1,KB
          II=J+I
          KO=1
          IF (II.GT.LHB) KO=II-KB
          SUM=A(IJD+I)
          IF (I.EQ.KB) GO TO 40
          DO 30 K=KO,J1
            JK=KB*K+J-KB
            IK=KB*K+II-KB
            TEMP=A(JK)
            SUM=SUM-A(IK)*TEMP
 30       CONTINUE
 40       A(IJD+I)=SUM/A(IJD)
 50     CONTINUE
 60   CONTINUE

      RETURN
      END
C     ------------------------------------------------------------
C
      SUBROUTINE SOLV (N, LHB, A, B)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(2550), B(255)
C
C     * FORWARD SUBSTITUTION *
C
      KB=LHB-1
      TEMP=A(1)
      B(1)=B(1)/TEMP
      DO 20 I=2,N
        I1=I-1
        KO=1
        IF (I.GT.LHB) KO=I-KB
        SUM=B(I)
        II=LHB*I-KB
        DO 10 K=KO,I1
          IK=KB*K+I-KB
          TEMP=A(IK)
          SUM=SUM-TEMP*B(K)
 10     CONTINUE
        B(I)=SUM/A(II)
 20   CONTINUE
C
C     * BACKWARD SUBSTITUTION *
C
      N1=N-1
      LB=LHB*N-KB
      TEMP=A(LB)
      B(N)=B(N)/TEMP
      DO 40 I=1,N1
        I1=N-I+1
        NI=N-I
        KO=N
        IF (I.GT.KB) KO=NI+KB
        SUM=B(NI)
        II=LHB*NI-KB
        DO 30 K=I1,KO
          IK=KB*NI+K-KB
          TEMP=A(IK)
          SUM=SUM-TEMP*B(K)
 30     CONTINUE
        B(NI)=SUM/A(II)
```

```
 40   CONTINUE
      RETURN
      END
C ------------------------------------------------------------
C
      SUBROUTINE GAUSS (N, E, H, IERR)
       REAL*8 E(32), H(32)
      M=(N-2)*(N-3)*(N-4)*(N-5)*(N-6)*(N-7)*(N-8)
      M=M*(N-9)*(N-10)*(N-11)*(N-12)*(N-15)*(N-16)*(N-32)
      IF (M.NE.0) GO TO 260
      IERR=0
      IF (N.EQ.32) GO TO 240
      IF (N.EQ.16) GO TO 220
      IF (N.EQ.15) GO TO 200
      IF (N.EQ.12) GO TO 180
      IF (N.EQ.11) GO TO 160
      IF (N.EQ.10) GO TO 140
      IF (N.EQ.9) GO TO 120
      IF (N.EQ.8) GO TO 100
      IF (N.EQ.7) GO TO 80
      IF (N.EQ.6) GO TO 60
      IF (N.EQ.5) GO TO 40
      IF (N.EQ.4) GO TO 20
      IF (N.EQ.3) GO TO 10
      E(1)=0.577350269189626D0
      E(2)=-E(1)
      H(1)=1.0D0
      H(2)=H(1)
      RETURN
 10   E(1)=0.774596669241483D0
      E(2)=0.0D0
      E(3)=-E(1)
      H(1)=0.555555555555556D0
      H(2)=0.888888888888889D0
      H(3)=H(1)
      RETURN
 20   E(1)=0.861136311594053D0
      E(2)=0.339981043584856D0
      H(1)=0.347854845137454D0
      H(2)=0.652145154862546D0
      DO 30 I=1,2
        E(5-I)=-E(I)
 30     H(5-I)=H(I)
      RETURN
 40   E(1)=0.906179845938664D0
      E(2)=0.538469310105683D0
      E(3)=0.0D0
      H(1)=0.236926885056189D0
      H(2)=0.478628670499366D0
      H(3)=0.568888888888889D0
      DO 50 I=1,2
        E(6-I)=-E(I)
 50     H(6-I)=H(I)
      RETURN
 60   E(1)=0.932469514203152D0
      E(2)=0.661209386466265D0
      E(3)=0.238619186083197D0
      H(1)=0.171324492379170D0
      H(2)=0.360761573048139D0
      H(3)=0.467913934572691D0
      DO 70 I=1,3
        E(7-I)=-E(I)
 70     H(7-I)=H(I)
      RETURN
 80   E(1)=0.949107912342759D0
      E(2)=0.741531185599394D0
```

```
      E(3)=0.405845151377397D0
      E(4)=0.0D0
      H(1)=0.129484966168870D0
      H(2)=0.279705391489277D0
      H(3)=0.381830050505119D0
      H(4)=0.417959183673469D0
      DO 90 I=1,3
        E(8-I)=-E(I)
90      H(8-I)=H(I)
      RETURN
100   E(1)=0.960289856497536D0
      E(2)=0.796666477413627D0
      E(3)=0.525532409916329D0
      E(4)=0.183434642495650D0
      H(1)=0.101228536290376D0
      H(2)=0.222381034453374D0
      H(3)=0.313706645877887D0
      H(4)=0.362683783378362D0
      DO 110 I=1,4
        E(9-I)=-E(I)
110     H(9-I)=H(I)
      RETURN
120   E(1)=0.968160239507626D0
      E(2)=0.836031107326636D0
      E(3)=0.613371432700590D0
      E(4)=0.324253423403809D0
      E(5)=0.0D0
      H(1)=0.081274388361574D0
      H(2)=0.180648160694857D0
      H(3)=0.260610696402935D0
      H(4)=0.312347077040003D0
      H(5)=0.330239355001260D0
      DO 130 I=1,4
        E(10-I)=-E(I)
130     H(10-I)=H(I)
      RETURN
140   E(1)=0.973906528517172D0
      E(2)=0.865063366688985D0
      E(3)=0.679409568299024D0
      E(4)=0.433395394129247D0
      E(5)=0.148874338981631D0
      H(1)=0.066671344308688D0
      H(2)=0.149451349150581D0
      H(3)=0.219086362515982D0
      H(4)=0.269266719309996D0
      H(5)=0.295524224714753D0
      DO 150 I=1,5
        E(11-I)=-E(I)
150     H(11-I)=H(I)
      RETURN
160   E(1)=0.978228658146057D0
      E(2)=0.887062599768095D0
      E(3)=0.730152005574049D0
      E(4)=0.519096129206812D0
      E(5)=0.269543155952345D0
      E(6)=0.0D0
      H(1)=0.055668567116174D0
      H(2)=0.125580369464905D0
      H(3)=0.186290210927734D0
      H(4)=0.233193764591990D0
      H(5)=0.262804544510247D0
      H(6)=0.272925086777901D0
      DO 170 I=1,5
        E(12-I)=-E(I)
170     H(12-I)=H(I)
      RETURN
```

```
180   E(1)=0.981560634246719D0
      E(2)=0.904117256370475D0
      E(3)=0.769902674194305D0
      E(4)=0.587317954286617D0
      E(5)=0.367831498998180D0
      E(6)=0.125233408511469D0
      H(1)=0.047175336386512D0
      H(2)=0.106939325995318D0
      H(3)=0.160078328543346D0
      H(4)=0.203167426723066D0
      H(5)=0.233492536538355D0
      H(6)=0.249147045813403D0
      DO 190 I=1,6
        E(13-I)=-E(I)
190     H(13-I)=H(I)
      RETURN
200   E(1)=0.987992518020485D0
      E(2)=0.937273392400706D0
      E(3)=0.848206583410427D0
      E(4)=0.724417731360170D0
      E(5)=0.570972172608539D0
      E(6)=0.394151347077563D0
      E(7)=0.201194093997435D0
      E(8)=0.0D0
      H(1)=0.030753241996117D0
      H(2)=0.070366047488108D0
      H(3)=0.107159220467172D0
      H(4)=0.139570677926154D0
      H(5)=0.166269205816994D0
      H(6)=0.186161000015562D0
      H(7)=0.198431485327112D0
      H(8)=0.202578241925561D0
      DO 210 I=1,7
        E(16-I)=-E(I)
210     H(16-I)=H(I)
      RETURN
220   E(1)=0.989400934991650D0
      E(2)=0.944575023073233D0
      E(3)=0.865631202387832D0
      E(4)=0.755404408355003D0
      E(5)=0.617876244402644D0
      E(6)=0.458016777657227D0
      E(7)=0.281603550779259D0
      E(8)=0.095012509837637D0
      H(1)=0.027152459411754D0
      H(2)=0.062253523938648D0
      H(3)=0.095158511682493D0
      H(4)=0.124628971255534D0
      H(5)=0.149595988816577D0
      H(6)=0.169156519395003D0
      H(7)=0.182603415044924D0
      H(8)=0.189450610455068D0
      DO 230 I=1,8
        E(17-I)=-E(I)
230     H(17-I)=H(I)
      RETURN
240   E(1)=0.997263861849482D0
      E(2)=0.985611511545268D0
      E(3)=0.964762255587506D0
      E(4)=0.934906075937740D0
      E(5)=0.896321155766052D0
      E(6)=0.849367613732570D0
      E(7)=0.794483795967942D0
      E(8)=0.732182118740290D0
      E(9)=0.663044266930215D0
      E(10)=0.587715757240762D0
```

```
      E(11)=0.506899908932229D0
      E(12)=0.421351276130635D0
      E(13)=0.331868602282128D0
      E(14)=0.239287362252137D0
      E(15)=0.144471961582796D0
      E(16)=0.048307665687738D0
      H(1)=0.007018610009471D0
      H(2)=0.016274394730906D0
      H(3)=0.025392065309262D0
      H(4)=0.034273862913021D0
      H(5)=0.042835898022227D0
      H(6)=0.050998059262376D0
      H(7)=0.058684093478536D0
      H(8)=0.065822222776362D0
      H(9)=0.072345794108849D0
      H(10)=0.078193895787070D0
      H(11)=0.083311924226947D0
      H(12)=0.087652093004404D0
      H(13)=0.091173878695764D0
      H(14)=0.093844399080805D0
      H(15)=0.095638720079275D0
      H(16)=0.096540088514728D0
      DO 250 I=1,16
        E(33-I)=-E(I)
 250    H(33-I)=H(I)
      RETURN
 260  WRITE (*,270)
 270   FORMAT(' WRONG CHOICE FOR NUMBER OF GAUSS INTEGRATION POINTS'/)
      IERR=1
      RETURN
      END
```