

格子型 二重底의 船體最適設計에 關한 研究

朴命圭* · 金成圭**

A study on the optimum ship Design of Grillaged Double Bottom.

Park Myung-kyu · Kim Sung-gyu

目 次

Abstract	4-1. Modeling
1. 序 論	4-2. 座標系
2. 格子型 구조이론	4-2-1. 전체 좌표계
2-1. 家庭	4-2-2. 국부 좌표계
2-1. 剛性行列의 구성	4-3. 境界條件
2-2-1. 部材가 굽힘을 받는 경우	4-3-1. 永久拘束
2-2-2. 部材가 비틀을 받는 경우	4-3-2. 周邊拘束
2-3. 전체 剛性行列	4-4. 荷重狀態
3. 最適化 理論	4-4-1. 海水압력
3-1. 數式化	4-4-2. 荷物압력
3-2. Hooke and Jeeves 직접 탐색법	5. 供誠船에 대한 計算缺課
3-3. Local Search	5-1. 계산결과및 비교
3-4. Pattern Move	6. 結 論
4. 二重底 構造의 格子型 Modeling.	參 考 文 獻

ABSTRACT

In spite of the widespread use of stiffened plates in ship structures, it is very difficult to analyze these directly. So, in conventional analysis of plate structures, above structures are used to be idealized as orthotropic plate or grillage structures.

Lately, the development of large computers, it is able to apply the optimum techniques to structural design.

In this paper, the double bottom structure of Bulk Carrier was idealized into flat grillage which

* 正會員, 韓國海洋大學
** DNV 검사반

is composed of intersecting beam stiffeners primarily loaded normal to its surface. And strength analysis was carried out by using the finite element method based on displacement. And further, according to variation of floor space and double bottom height, the optimum design was carried out by using Hooke and Jeeves direct search method.

1. 序 論

선박의 격벽, 갑판, 외판, 선저외판 등은 板과 보강재로 구성되어 있는 보강판 구조이고, 이들 보강판 구조의 중량이 선각중량의 60% 이상을 차지하고 있으므로, 선체구조를 해석한다는 것은, 이들 보강판 구조를 해석하는 것이라고 해도 과언은 아니다.

그러나 이들 보강판 구조는 그 형상이 매우 복잡할 뿐만 아니라 경계조건 설정이 곤란하므로 직접 해석한다는 것은 대단히 곤란하다. 따라서, 구조해석시에는 直文異方性 平板 또는 格子型(Grillage)으로 이상화하며 근사적으로 해석하고 있다.

또한 최근에는 종래의 경험이나 Rule에 의한設計에서 벗어나, 工學的, 理論的 해석에 의한 경제적이고 안전한設計를 指向하는 최적화 기법(optimization Technique)을 이용한 해석이 각광을 받고 있다.

최적화 기법을 이용한 선체구조 설계는 1960년대 부터 나타나기 시작했으며, 주로 선체의 중앙부 단면에 대한 최적설계가 수행되어 왔다.

본 論文에서는 Bulk Carrier의 이중저 구조를 格子型 구조로 모델링한 다음, 선저수압과 화물압

력을 하중조건으로 취하고, Alternate loading Condition에 대한 각 부재 재부에 발생하는 변위, 전단응력 및 굽힘응력을 계산한 후, 중량경감을 목적으로 비선형 계획법중 탐색법의 일종인 Hooke & Jeeves 직접 탐색법을 이용하여 Floor space와 이중저 높이의 변화에 따른 최적설계를 船級 Rule에 의한 실선과 比較, 검토를 행하였다.

2. 格子型 구조 이론

2-1. 假定

格子型 이중저 구조를 유한요소법으로 해석하기 위하여 다음과 같은 가정을 하였다.

- 1) 변형은 유한 변형이다.
- 2) 各 部材는 Hooke의 法則을 따른다.
- 3) 各 部材의 연결부는 完全固着 상태로 본다.
- 4) 초기 응력 및 초기 변형은 없는 것으로 한다.
- 5) 전단력에 의한 변형은 고려하지 않는다.

2-2 剛性行列의 구성

2-2-1. 部材가 굽힘을 받는 경우

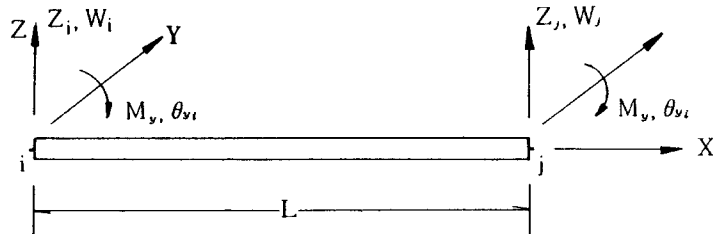


Fig. 1. Nodal forces and displacements for element in bending.

(a) 변위함수와 절점변위

이 경우 절점변위는 $W_i, W_j, \theta_{yi}, \theta_{yj}$ 인 4개의 성분으로 되며 이 매개변수에 의해 표시되는 다항식은 3次式이기 때문에 部材内の 변위는 다음과 같이 표시할 수 있다.

$$W = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \alpha_4 x^3$$

$$= [X] \{ \alpha \} \quad \dots\dots\dots (2.1)$$

여기서, $[X] = [1 \ x \ x^2 \ x^3]$; Z 방향 변위함수 matrix.

$\{ \alpha \} = \{ \alpha_1 \ \alpha_2 \ \alpha_3 \ \alpha_4 \}^T$; Z 방향의 변위에 관한 미정계수 Vector.

Beam theory에 의하여 Beam element 上에 분포 하중이 없다고 가정하면, 보의 처짐은 다음 미분 방정식을 만족한다.

$$EI \cdot \frac{d^4 W}{dx^4} = 0 \quad \dots\dots\dots (2.2)$$

따라서 (2.1)式은 (2.2)式을 만족한다. 절점 i, j에서 다음식이 성립한다.

$$\left. \begin{aligned} W_i &= \alpha_1 \\ W_j &= \alpha_1 + d_2 L + d_3 L^2 + \alpha_4 L^3 \\ \theta_{yi} &= \left(\frac{\partial w}{\partial x} \right)_{x=0} = \alpha_2 \\ \theta_{yj} &= \left(\frac{\partial w}{\partial x} \right)_{x=L} = \alpha_2 + 2\alpha_3 L + 3\alpha_4 L^2 \end{aligned} \right\} \quad \dots\dots\dots (2.3)$$

$$M = -EI \cdot \frac{d^2 w}{dx^2} = -EI (2\alpha_3 + 6\alpha_4 x)$$

$$\quad \dots\dots\dots (2.4)$$

(2.3)式을 matrix로 표현하면 다음식을 얻는다.

$$\left\{ \begin{matrix} \theta_{yi} \\ W_i \\ \theta_{yj} \\ W_j \end{matrix} \right\} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 2L & 3L^2 \\ 1 & L & L^2 & L^3 \end{bmatrix} \left\{ \begin{matrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{matrix} \right\}$$

$$\quad \dots\dots\dots (2.5)$$

(2.5)式을 Comfact form으로 표현하면 다음식을 얻는다.

$$\{ d \} = [A] \{ \alpha \} \quad \dots\dots\dots (2.5)$$

여기서 $[A]$; Z방향의 변위에 관한 절점변위의 미정계수 matrix.

또 한식(2.5)로 부터 다음식을 얻는다.

$$\{ \alpha \} = [A]^{-1} \{ d \} \quad \dots\dots\dots (2.6)$$

(2.6)式을 (2.1), (2.4)式에 각각 대입하면

$$W = [X] [A]^{-1} \{ d \} \quad \dots\dots\dots (2.7)$$

$$M = -EI [0 \ 0 \ 2 \ 6x] \left\{ \begin{matrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{matrix} \right\}$$

$$= -EI \cdot [B] [A]^{-1} \{ d \} \quad \dots\dots\dots (2.8)$$

은 matrix해법이 얻어진다.

(b) 변형률과 응력

Beam theory에서 굽힘 변형률은 다음과 같이 주어진다.

$$\epsilon_x^b = -Z \cdot \frac{\partial^2 w}{\partial x^2} \quad \dots\dots\dots (2.9)$$

(2.1)式을 이용하여 matrix로 나타내면 다음식을 얻는다.

$$\{ \epsilon_x^b \} = -Z [0 \ 0 \ 2 \ 6x] \left\{ \begin{matrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{matrix} \right\}$$

$$= [B_b] \{ d \} \quad \dots\dots\dots (2.10)$$

여기서 $[B_b]$; 굽힘 변형률의 미정계수 matrix.

(2.6)式을 (2.10)에 代入하면

$$\{ \epsilon_x^b \} = [B_b] \cdot [A]^{-1} \{ d \} \quad \dots\dots\dots (2.11)$$

Beam element의 Stress는 Hooke의 法則에 의하여 다음식을 얻는다.

$$\{ \sigma_x^b \} = E \cdot \{ \epsilon_x^b \} \quad \dots\dots\dots (2.12)$$

(c) Stiffness matrix

가상일의 원리에 의하여

$$W_{int} = W_{ext} \quad \dots\dots\dots (2.13)$$

内部일(W_{int})은 다음과 같이 표시된다.

$$W_{int} = \iiint \{\epsilon^*\}^T \{\sigma_z^*\} dx dy dz \dots (2.14)$$

식(2.14)에 (2.11), (2.12)식을 代入하면

$$W_{int} = \iiint \{d^*\}^T [A^{-1}]^T [B_0]^T E [B_0] [A]^{-1} \{d\} dx dy dz \text{ 이고} \dots (2.15)$$

식(2.15)에서 d^* 및 d 는 x, y, z 의 함수와 무관하므로

$$W_{int} = \{d^*\}^T \left(\iiint [A^{-1}]^T [B_0]^T E [B_0] [A]^{-1} dx dy dz \right) \{d\} \text{ 이다} \dots (2.16)$$

또한 외부 일(W_{ext})은 다음과 같이 표시된다.

$$W_{ext} = \{d^*\}^T \{F\} \dots (2.17)$$

여기서 $\{F\}$; 외부에 가해진 힘 Vector (2.16), (2.17)식으로 부터

$$\{d^*\}^T \left\{ \{F\} - \left(\iiint [A^{-1}]^T [B_0]^T E [B_0] [A]^{-1} dx dy dz \right) \{d\} \right\} = 0 \dots (2.18)$$

로 되고 가상변위 $\{d^*\}$ 에 관계없이 이 식이 성립하기 위해서는

$$\{F\} = \left(\iiint [A^{-1}]^T [B_0]^T E [B_0] [A]^{-1} dx dy dz \right) \{d\} \dots (2.19)$$

식(2.19)로 부터 Stiffness Matrix $[K]$ 는 다음과 같다.

$$[K] = \iiint [A^{-1}]^T [B_0]^T E [B_0] [A]^{-1} dx dy dz \dots (2.20)$$

$[A]^{-1}$ 는 x, y, z 와 무관하므로

$$[K] = [A^{-1}]^T \left(\iiint [B_0]^T E [B_0] dx dy dz \right) [A]^{-1} \dots (2.21)$$

피적분 함수를 다시 정리하면

$$[B_0]^T E [B_0] = E \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 4Z^2 & 12xz^2 \\ 0 & 0 & 12xz^2 & 36x^2z^2 \end{bmatrix}$$

으로 되고 $I_y = \iint z^2 dx dy$ 임을 고려하면 다음 식을 얻는다.

$$[K] = EI_y \begin{bmatrix} \frac{4}{L} & & & \\ & \text{Sym.} & & \\ & & -\frac{6}{L^2} & \frac{12}{L^3} \\ & & \frac{2}{L} & -\frac{6}{L} & \frac{4}{L} \\ & & \frac{6}{L^2} & -\frac{12}{L^3} & \frac{6}{L^2} & \frac{12}{L^3} \end{bmatrix} \dots (2.22)$$

2-2-2 部材가 비%을 받는 경우.

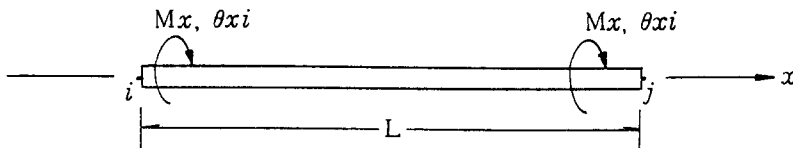


Fig. 2. Nodal forces and displacements for element in torsion.

이 경우 절점변위는 θ_{xi} , θ_{xj} 인 2개의 성분으로 되며 이 매개변수에 의하여 표시되는 다항식은 1차식이기 때문에 部材内の 변위를 다음과 같이 나타낼 수 있다.

$$\theta_x = a_1 + a_2x$$

$$= [X] \{a\} \dots\dots\dots (2.23)$$

그러므로

$$\theta_{xi} = a_1$$

$$\theta_{xj} = a_1 + a_2L \dots\dots\dots (2.24)$$

$$M = GJ \frac{d\theta_x}{dx} = GJ \begin{bmatrix} 0 & 1 \end{bmatrix} \{a\} \quad (2.25)$$

식(2.24)를 matrix로 표시하면

$$\{d\} = [A] \{a\} \dots\dots\dots (2.26)$$

$$\{a\} = [A]^{-1} \{d\} \dots\dots\dots (2.27)$$

식(2.27)을 (2.24), (2.25)에 각각 代入하면 다음식을 얻는다.

$$\theta_x = [1 \ x] \begin{bmatrix} 1 & 0 \\ -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \begin{Bmatrix} \theta_{xi} \\ \theta_{xj} \end{Bmatrix}$$

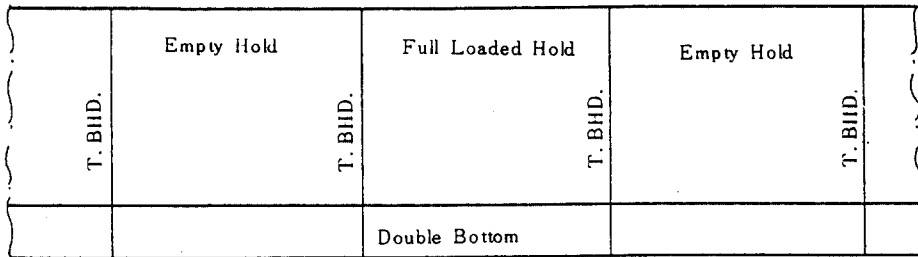
$$= [X] [A]^{-1} \{d\} \dots\dots\dots (2.28)$$

$$M = GJ [B] [A]^{-1} \{d\} \dots\dots\dots (2.29)$$

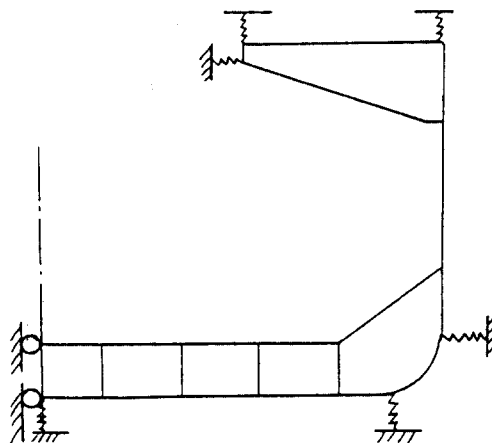
Stiffness matrix [K]를 구하기 위해 (2.21)式을 利用하면 다음식을 얻는다.

$$[K] = [A^{-1}]^T \left(\int_0^L [B]^T GJ [B] dx \right) [A]^{-1}$$

$$= GJ \begin{bmatrix} \frac{1}{L} & -\frac{1}{L} \\ -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \dots\dots\dots (2.30)$$



(a) Longitudinal Direction



(b) Transverse Direction

Fig. 4-1. Double bottom structure of Buld Carrier.

		Empty Hold					Full Loaded Hold						
B		E										C Side Shell	
6	86	12 87	18 88	24 89	30 90	36 91	42 92	48 93	54 94	60 95	66		
		5	10	15	20	25	30	35	40	45			
5	76	11 77	17 78	23 79	29 80	35 81	41 82	47 83	53 84	59 85	65	No.4 S.G	
		4	9	14	19	24	29	34	39	44			
4	66	10 67	16 68	22 69	28 70	34 71	40 72	46 73	52 74	58 75	64	No.3 S.G	
		3	8	13	18	23	28	33	38	43			
3	56	9 57	15 58	21 59	27 60	33 61	39 62	45 63	51 64	57 65	63	No.2 S.G	
		2	7	12	17	22	27	32	37	42			
2	46	8 47	14 48	20 49	26 50	32 51	38 52	44 53	50 54	56 55	62	No.1 S.G	
		1	6	11	16	21	26	31	36	41			
1		7	13	19	25	31	37	43	49	55	61	D	
	No.1 F.L		No.2 F.L		T. BHD		No.5 F.L		No.7 F.L				

Fig. 4-2. Double bottom grillage modeling.

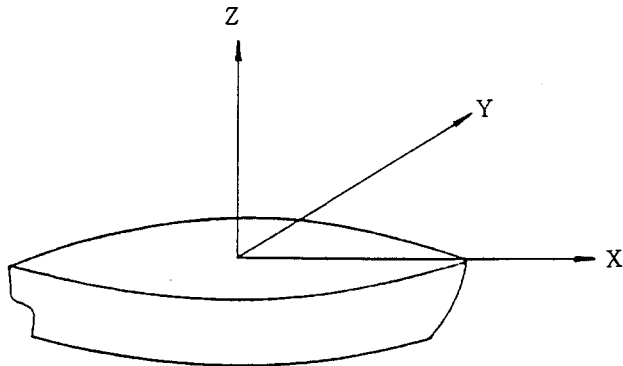


Fig. 4-3. Global Coordinate System.

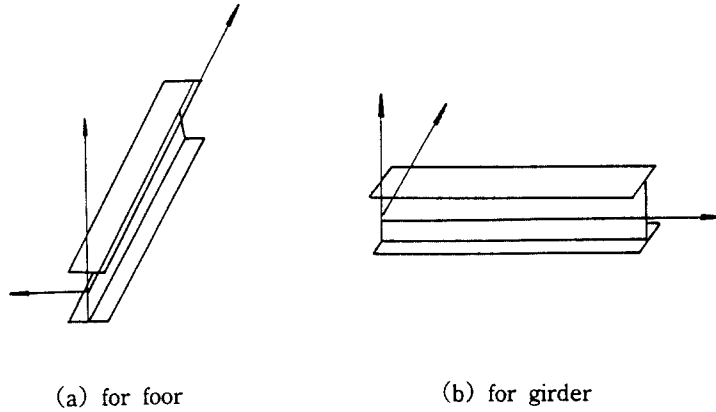


Fig. 4-4. Local Coordinate System.

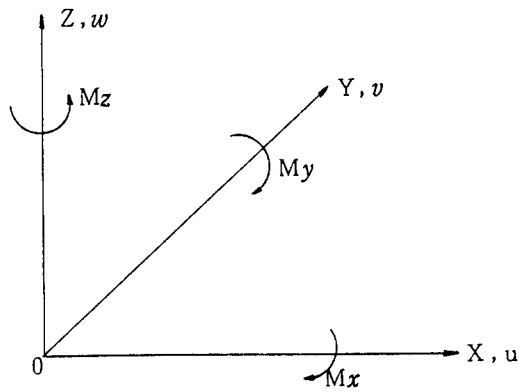


Fig. 4-5. Force and Moment Direction.

Table 4-1. Boundary Constraint.

Nodal Point	Constraint Component	Nodal Point	Constraint Component
A ~ B, C ~ D	u, v, My, Mz	B ~ C	u, v, Mx, Mz
A ~ D	u, v, Mx, Mz	E	u, v, w, Mz

2-3. 전체 강성행렬

중첩하면 전체 Stiffness matrix가 얻어진다.

부재가 굽힘을 받는 경우와 비%을 받는 경우를

$$[K_T] = \begin{bmatrix} \frac{GJ}{L} & 0 & 0 & -\frac{GJ}{L} & 0 & 0 \\ 0 & \frac{4EI_y}{L} & -\frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & \frac{6EI_y}{L^2} \\ 0 & -\frac{6EI_y}{L^2} & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & \frac{12EI_y}{L^3} \\ -\frac{GJ}{L} & 0 & 0 & \frac{GJ}{L} & 0 & 0 \\ 0 & \frac{2EI_y}{L} & -\frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & \frac{6EI_y}{L^2} \\ 0 & \frac{6EI_y}{L^2} & -\frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & \frac{12EI_y}{L^3} \end{bmatrix}$$

3. 最適化 理論

3-1. 數式化

함수F가 최대값 혹은 최소값이 되는 n次元 공간상의 Vector X Rⁿ을 찾는 문제로서 다음과 같이 표시한다.

$$\begin{aligned} & \text{Minimize} && F(x) \\ & \text{Subject to} && \left(\begin{array}{l} g_i(x) = 0 \quad i=1, 2, 3, \dots, n \\ h_j = 0 \quad j=1, 2, 3, \dots, n \end{array} \right) \end{aligned}$$

여기서 F(X) : 목적함수

X=(X₁, X₂, ···, X_n) ; 설계변수

g(x) : inequality constraints

h_j(x) : equality constraints.

최적해가 x*이면 다음식으로 표시될 수 있다.

$$\begin{aligned} x^* = \{ x \mid \min. F(x), & \quad g_i(x) \geq 0, \\ & \quad h_j(x) = 0 \} \dots\dots\dots (3-1) \end{aligned}$$

3-2. Hooke & Jeeves 직접탐색법

이 방법은 R. Hooke와 T. A. Jeeves에 의해 제안된 직접 탐색법에 바탕을 둔 것으로, 多變數이고 制約條件이 없는 非線型函數의 최소값을 구하는 방법이며, 매우 간단하고 效率的이나 目的函數의 값이 좌표축과 같이 45°방향으로 가파른 골짜기를

형성하고 있는 경우에는 최적 점에 도달하기 前에 중단되거나 반복과정이 늦어지는 단점이 있다.

그러나 이 방법은

- ① 고풀수(derivatives)가 요구되지 않고
- ② 최적 解를 찾기 위한 方法上, 각각의 有限間隔만큼 떨어져 있는 두 점들에 대한 목적함수의 값을 직접 비교함으로써 나아갈 방향이 결정되며

③ 만약 한개 以上の 最適解가 존재하거나, 목적함수의 수식형태上, 표면형상이 미지일 경우에는 여러개의 출발점을 고려해야 된다는 특징이 있다.

이 방법의 거동 형태는 Local Search와 Pattern Move로 구성되어 있으며, Fig. 3-1 및 Fig 3-2에 Flow chart와 직접탐색법을 나타내었다.

3-3. Local Search.

Base point(어떠한 한 단계에서의 기준출발점)에서 각각의 설계변수에 대해 初期에 설정한 Step size만큼 순차 누적적으로 가감한 주위의 점들에 대한 목적함수 값과, 바로 앞 단계에서 얻은 목적함수의 값을 비교함으로써, 보다 向上된 최적점을 찾아가는 과정을 말하며, 목적함수의 값이 앞 단계에서 구한 값보다 작으면 success이고 크면 Failure라고 한다.

이를 n개의 設計變數에 대해 順次的으로 수행한 후 success되는 점, 즉 새로운 Base point를 얻게

출발점이 $X_0=(x_1, x_2, \dots, x_n)$ 이고 step size가 Δi 인 목적함수가, 각각의 설계변수에 대해 정해진 step만큼 더한점에서 Success이면,

$$\begin{aligned} F(x_1+4_1, x_2, \dots, x_n) < F(x_0) \text{인 점은 } x_{01} \\ F(x_1+4_1, x_2, \dots, x_n) < F(x_0) \text{인 점은 } X_{01} \\ F(x_1+4_1, x_2+4_2, \dots, x_n) < F(x_{01}) \text{인 점은 } X_{02} \\ \vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \\ F(x_1+4_1, x_2+4_2, \dots, x_n+4_n) < F(x_{0n-1}) \text{인} \\ \text{점은 } X_{0n} \end{aligned}$$

로 된다.

이러한 경우 점 X_m 이 새로운 Base point X_0 로 되어 Pattern Move로 이동한다.

여기서, X_0 ; local search를 수행하고자 하는 점.

X_0 ; local search를 수행한 점.

3-4. pattern Move.

Local search와 결합시켜 보다 빠르게 최적점을 찾아 가는 과정으로서, Local search를 수행하여 얻어진 결과에서 最少化로 가는 方向을 찾아내었으므로, 처음 단계에서 이동한 길이와 방향만큼 옮겨 놓은 점을 다시 Base point로 하여 Local Search를 수행하고자 하는 과정이다.

Local Search를 수행한 점 X 를 Base point로 하여 式(3-2)에 의해 Pattern Move를 하면 새로운 Base point X_1 을 얻게 된다.

$$X_1 = X_0 + \alpha(X_0 - X_0) \dots \dots \dots (3-2)$$

여기서 α : acceleration factor, $\alpha=1.0$
式(3-2)를 일반적으로 식으로 표현하면 다음과 같다.

$$X_n = X_{n-1} + \alpha(X_{n-1} - X_{n-1}) \dots \dots \dots (3-3)$$

Pattern Move 단계(X_1, X_2, \dots, X_n)를 거쳐 얻어진 점을 다시 출발점으로 하여 Local Search 단계(X_1, X_2, \dots, X_n)를 교대로 이행하여 최적점을 얻을때까지 반복계산을 수행한다.

4. 二重底 構造의 格子型 Modeling

4-1. Modeling

' Fig. 4-1과 같은 Bulk Carrier의 이중저 구조를 해석시에는 이중저, bilge hopper 및 top side tank등을 포함시키는 것이 보통이며 종방향으로는 Cargo hold의 크기와 형태에 따라 다르지만, 일반적으로 Alternate loading의 경우 Full loaded hold의 중심에서 이웃한 Empty hold의 중심까지를 해석 Model로 택한다. 또 선체중심선에 대해서 대칭이므로 한 側만을 취한다.

Fig. 4-2는 Fig. 4-1에 나타난 해석범위의 grillage model을 나타낸 것이다. 그림중 A-B, D-C는 Cargo hold의 Center line, A-D는 Center girder, B-C는 Longi. BHD. (shell), A-B에 평행한 선은 floor, A-D에 평행한 선은 girder를 나타낸다.

4-2座標系

4-2-1 전체 좌표계

Fig. 4-3과 같이 선체를 1개의 Beam으로 본 좌표계로서 길이 방향을 x 축, 선폭방향을 y 축, 높이 방향을 Z 로 잡았으며 우수계를 취했다.

4-2-2 국부좌표계

선체의 이중저를 구성하고 있는 각 부재의 개별적인 좌표로서 Fig. 4-4(a), (b)에서 보는 바와 같이 각 부재의 길이방향을 X 축으로 하고, 전체좌표계와 동일하게 격자평면에 수직인 방향으로 Z 축으로 한다.

4-3 境界條件

4-3-1 永久拘束

격자형 구조해석시 하중은 격자평면($x-y$ 평면)에 수직방향으로 작용시키고, 모우멘트는 Fig. 4-5에서 보는바와 같이 x 축 및 y 축에 작용하므로 $x-y$ 평면내에서는 변위나 회전은 존재하지 않고 Z 축 방향의 범위 W 만 존재한다. 그리고 x 축 및 y 축 방향의 모우멘트 M_x, M_y 에 대한 회전이 발생하며 Z 축 방향의 모우멘트 M_z 의한 회전은 구속상태로 가정한다.

즉, Grillage 평면내의 모든 nodal point는 x 축 및 y 축 방향의 변위와 Z 축 방향의 moment에 따른

회전은 '완전구속상태로 된다.

4-3-2 周邊拘束

각 절점의 특수조건에 따라 결정되는 구속상태이며 Table 4-1과 같다.

4-4 荷重狀態

4-4-1 海水압력

하중 작용점이 하계 만재 흘수선 보다 아래에 있을 경우 선측, 선저, 그리고 폭로갑판상에 작용하느냐 해수압력은 다음과 같다.(DNV. Rule Part3, chapter 1. Section 4)

$$P = 10h_0 + [K_s - \frac{1.5h_0}{T}] \cdot C_w \text{ (KN/m}^2\text{)}$$

..... (4-1)

여기서 h_0 = vertical distance in m from the water-line at draught T to the load point.

$$K_s = 6 \text{ aft of } A \cdot P$$

$$= 3 \text{ between } 0.2L \text{ and } 0.7L \text{ from } A \cdot P$$

$$= \frac{8}{C_0 + 0.2} \text{ forward of } F \cdot P$$

$$C = 0.0792L \quad \text{for } L < 180$$

$$= 10.75 - [(300-L)100]^{3/2} \quad \text{for } 100 < L < 300$$

$$= 10.75 \quad \text{for } 300 \leq L \leq 350$$

$$= 10.75 - [(L-350)/150]^{3/2} \quad \text{for } L > 350$$

4-4-2 荷物압력

화물압력은 대상선박의 설계하중인 27T/m²를 취하였다.

5. 供試船에 대한 計算結果

선체 이중저의 강도해석 및 최적화를 수행하기 위하여 CPC-2600 16BIT INTEL8088기종으로 입력 Q10루틴을 사용하여 목록A와 같이 FORTRAN 프로그램을 구성하여 검증과 방법의 타당성을 검토할 목적으로 Table 5-1과 같은 주요치수 및 Fig 5의 중앙단면도를 가진 D/W 64,000 TON BULK Carrier을 대상 선박으로 비교하였다.

Table 5-1. Principal Dimension.

Ship Type	Bulk Carrier	D / W	64,000 Ton
Loa	224.0 m	L	214.0 m
B	32.2 m	D	18.1 m
d (design)	12.5 m	d (Scantling)	13.1 m

선체 이중저의 경우 이중저이 높이 및 선창의 치수비에 관계하느니 늑판간격의 변화가 선체중강도에 크게 기여하므로 설계변수를 다음과 같이 2가지로 나누어 계산을 수행하였다.

- 1) Case1; 늑판간격이 변하는 경우
- 2) Case2; 이 중저 높이가 변하는 경우

5-1 計算결과 및 비교

1) 늑판간격이 변하는 경우(Case 1)

Fig. 4-2와 같은 Grillage 모델에서 늑판간격이 변하는 경우의 실선 및 최적화를 수행한 후의 변위, 응력곡선은 Fig. 5-1~Fig. 5-6과 같다. 이 결과에서 목적함수의 값은 21144cm³에서 21049cm³

로 0.5%의 중량경감 효과가 있었다. 또, 늑판간격은 실선의 경우 266.7cm였으나 최적화 수행후 267.0cm로 변화하여 Hold의 길이도 각각 1200cm에서 1201.5cm로 되어 그 치수비는 1.001이었다.

2) 이중저 높이가 변하는 경우(Case 2)

이중저 높이의 변화에 대한 실선 및 최적화를 수행한 후의 변위, 응력곡선이 Fig. 5-1~Fig. 5-6에 주어져 있다.

이 결과 목적함수의 값은 21.144cm³에서 20.948cm³으로 되어 중량은 0.93%의 경감효과를 가져왔으며 이때 이중저의 높이는 160cm로 그 치수비는 0.80이다.

된다. 이러한 Local Search가 成功하면 Pattern Move로 넘어가 기준출발점에서 이동해온 거리와 방향만큼 이동시킨 점의 주위에 대해 다시 Local Search를 하는 반복작업을 계속한다.

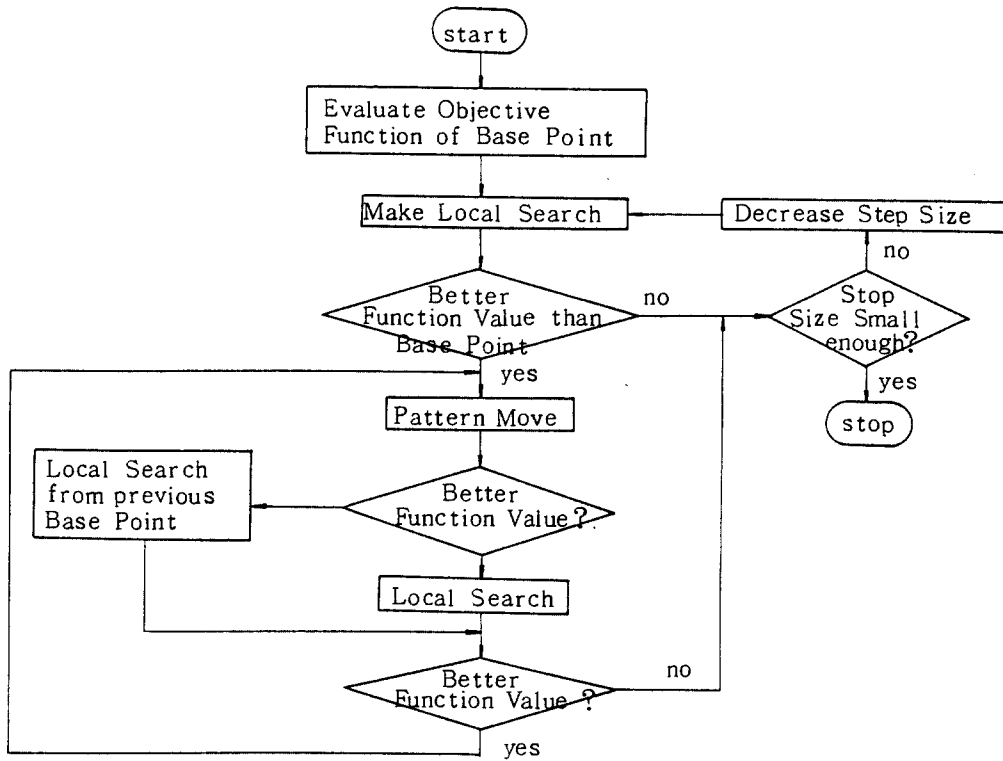


Fig. 3-1. Flow Chart(Hooke and Jeeves direct search method).

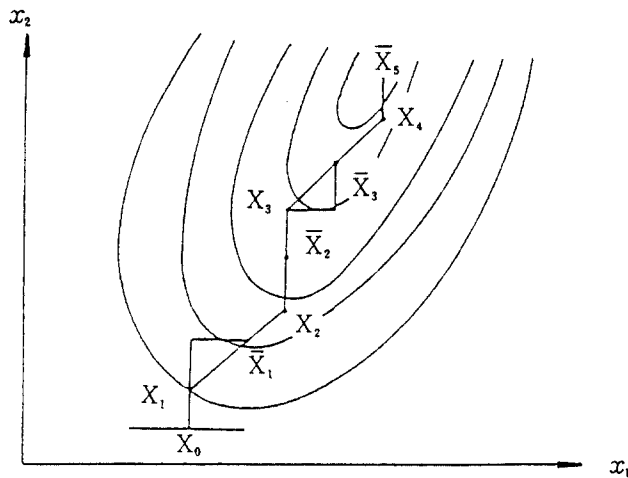


Fig. 3-2. 2-Dim. Direct Search method.

6. 結 論

실체의 이중저 구조를 格子型(Grillage)으로 모델링하여 해석하고 이를 최적화 기법을 이용하여 최적구조해석을 행한 결과는 Fig. 5-1~Fig. 5-6에 나타나 있으며 그 결론은 다음과 같다.

- 1) 선체 이중저 구조의 초기설계단계에서 적절한 늑판간격과 이중저 높이를 알수있다
- 2) 중량을 목적함수로 취하여 이중저 구조를 해석한 결과 실선의 중량보다 각각 0.50%, 0.93%의 감소가 있었다.
- 3) 굽힘응력은 Hold의 중앙과 T. BHD 부근에서 크게 발생하고 전단응력은 L. BHD 부근에서 크게 발생하므로 초기설계시 유의해야 한다.
- 4) 국부적으로 응력이 과대하게 발생하는 것으로 설계변수의 제약성 때문인 것으로 사료되며, 응력과 치수를 모두 만족하는 최적설계가 요구된다.

參 考 文 獻

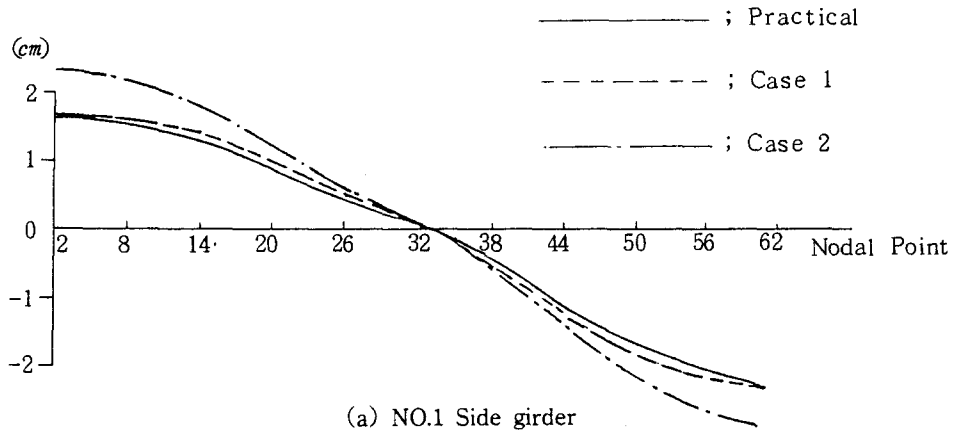
- 1) C. S. Smith, "Analysis of grillage structures by the force method", Trans. RINA, 106, 183-196(1964)
- 2) J. H. Evans, D. Khoushy, "Optimized Design of midship section structure", Trans. SNAME, 71, 144-249(1963)
- 3) J. Moe, "Fundamentals of Optimization", Computer & Structures, 4, 95-113(1974)
- 4) 北材勝英, "船體構造의最適設計에關する研究

(第2報), SNAJ, 130, 259-268(1971).

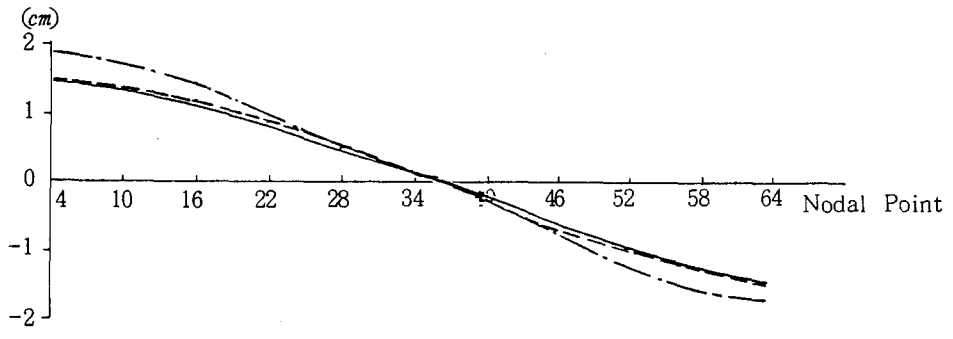
- 5) F. Moses, SUSUMU ONODA, "Minimum weight design of structures with application to elastic grillage", International Journal for Numerical method in Engineering, 1, 311-331 (1969)
- 6) D. Kavlie, J. Moe, "Application of non-linear programming to optimum grillage design with non-convex of variable", International Journal for Numerical method in Engineering, 1, 351-378 (1969).
- 7) E. Pennala, "Optimization of stiffened plates idealized as grillage structures", p. 177-182 (1976)
- 8) H. A. Schade, "Bending theory of ship bottom structure", Trans. SNAME, 46, 176-205 (1938)
- 9) 中材一郎, 阪部陽一, "ベルクキャリアの二重底構造の設計法に關する一考察", 關西造船協會誌, 109, 1-11 (1963)
- 10) M. C. Parson, "Optimization methods for use in computer-aided ship design", 1st ship technology and Yesearch symposium, SNAME, August, 13 (1975)

M. S. Troitsky, "Stiffend plates, bending, stability and vibrations", Elsevier scientific publishing company, p. 1-358 (1976)

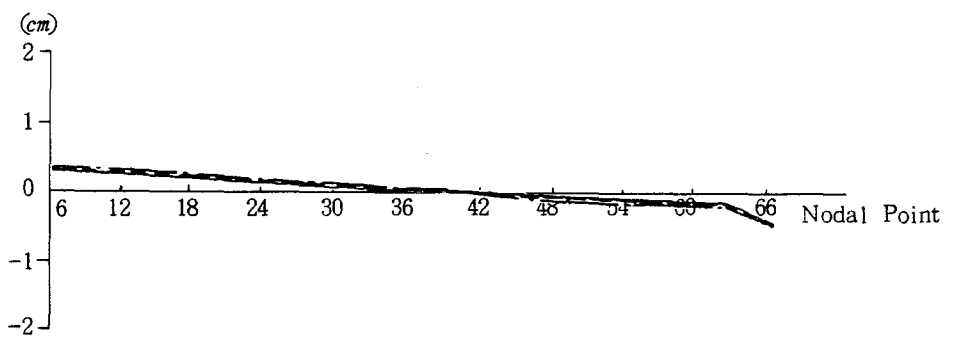
三本木茂夫, 吉材信海, "有限要素法にする構造解析プログラム", 培風館, p. 30-58 (1970).



(a) NO.1 Side girder



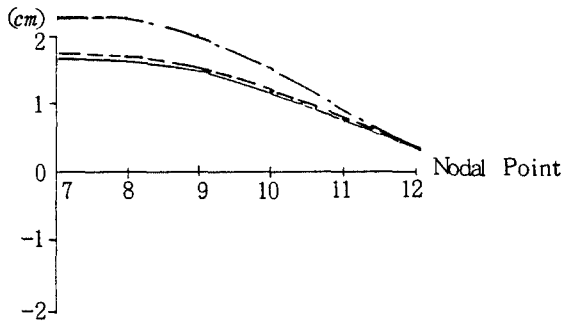
(b) NO. 3 Side girder



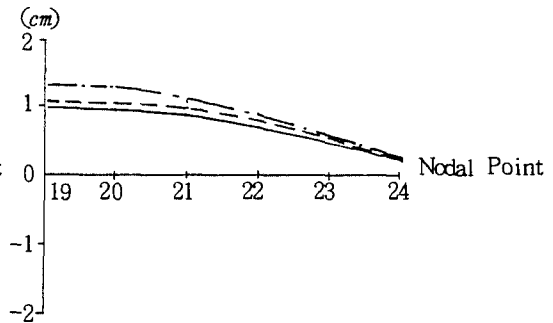
(c) Side Shell

Fig. 5-1. Deflection of girder.

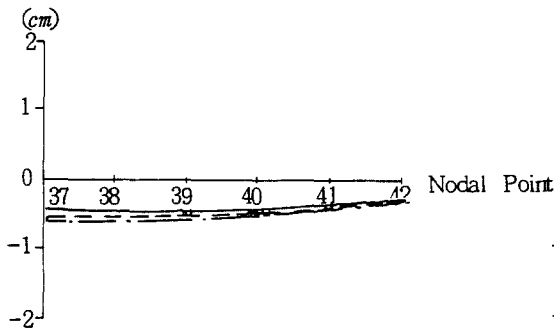
————— ; Practical
 - - - - - ; Case 1
 ———— ; Case 2



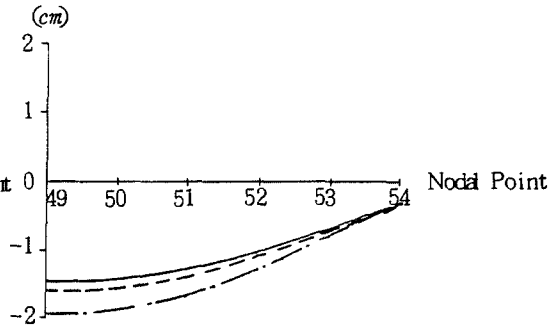
(a) NO. 1 Floor



(b) NO. 3 Floor

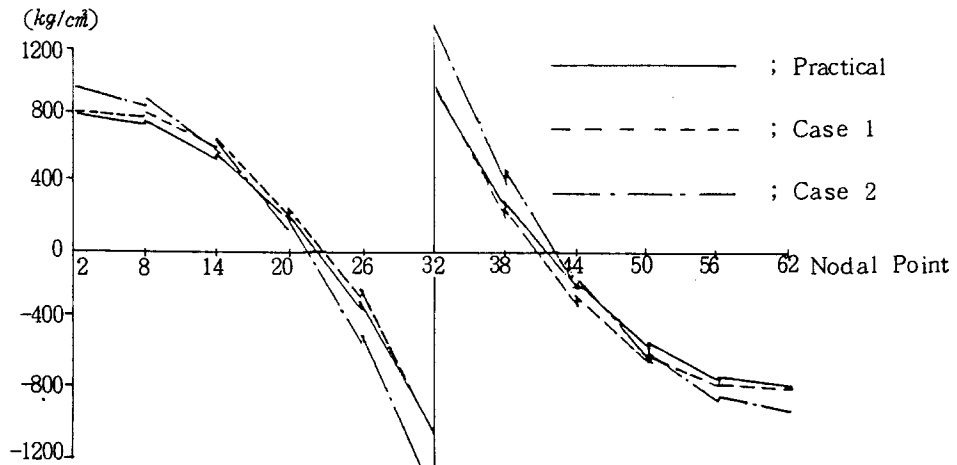


(c) NO. 5 Floor

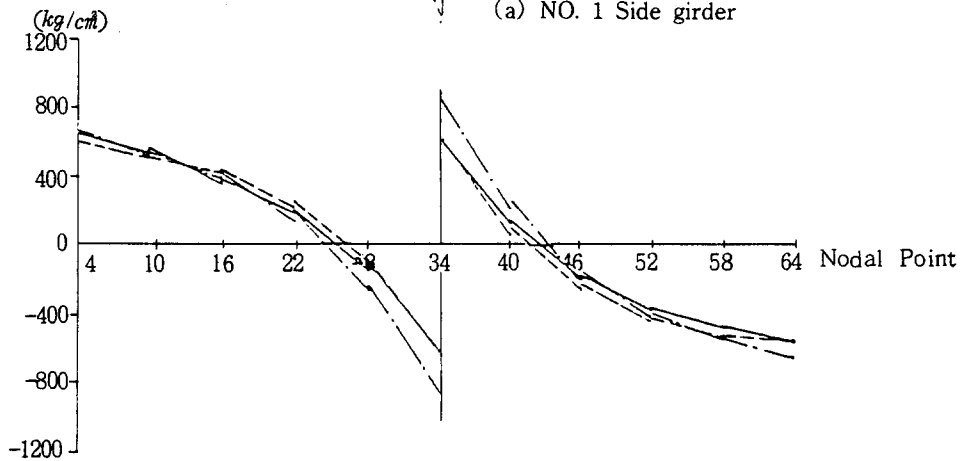


(d) NO. 7 Floor

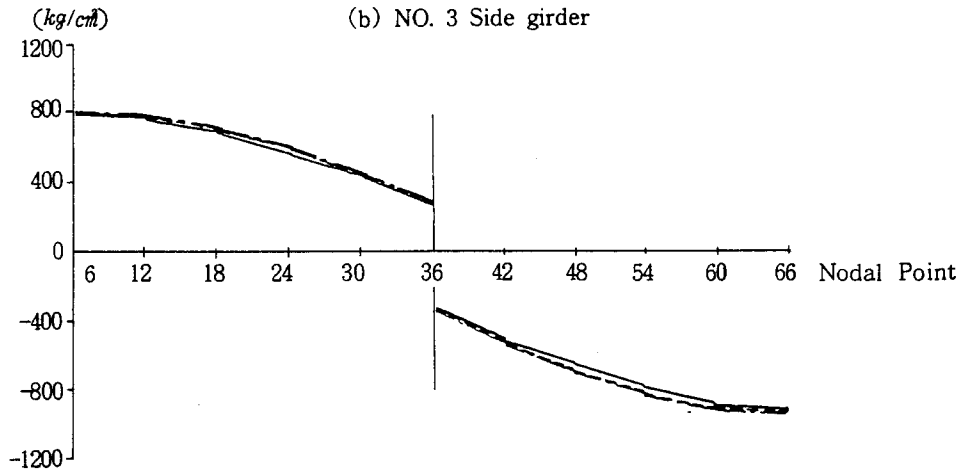
Fig. 5-2. Deflection of floor.



(a) NO. 1 Side girder



(b) NO. 3 Side girder



(c) Side shell

Fig. 5-3. Bending stress of girder.

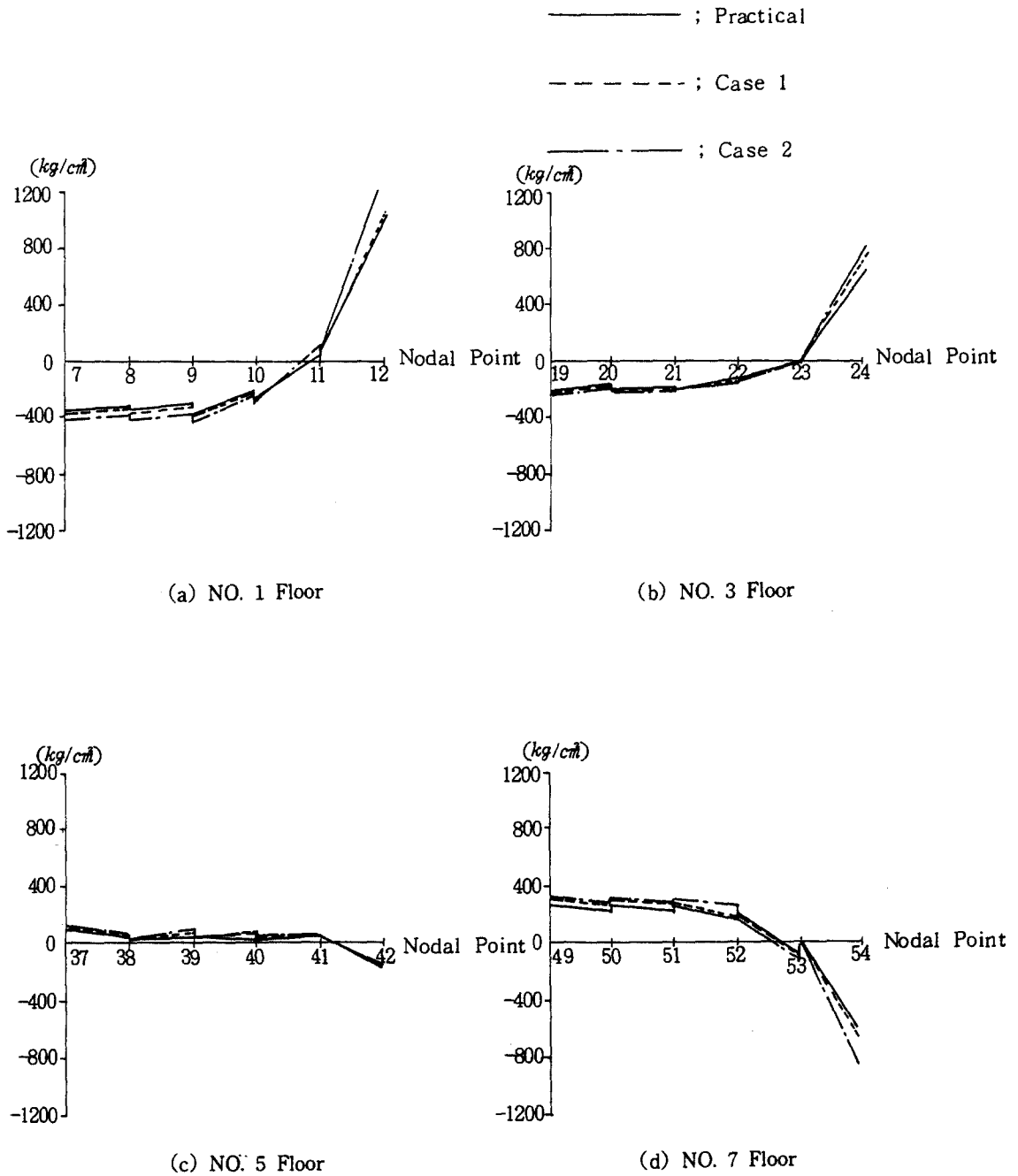
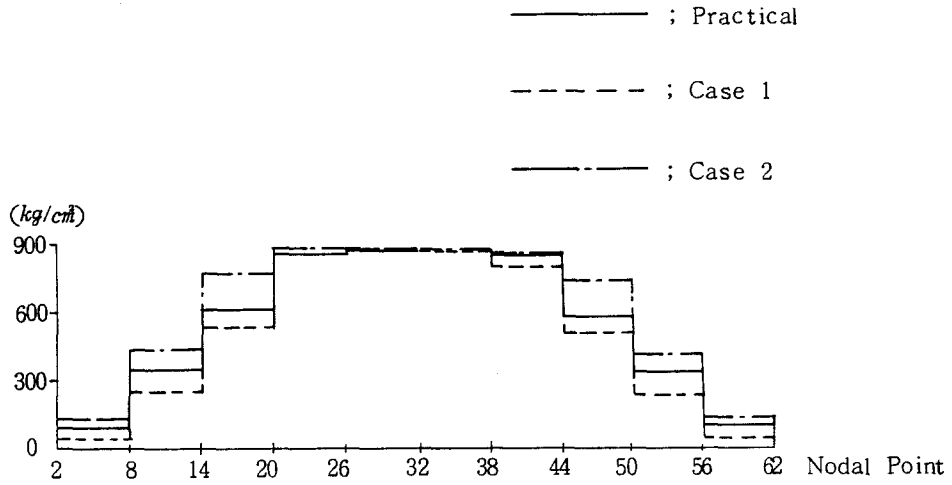
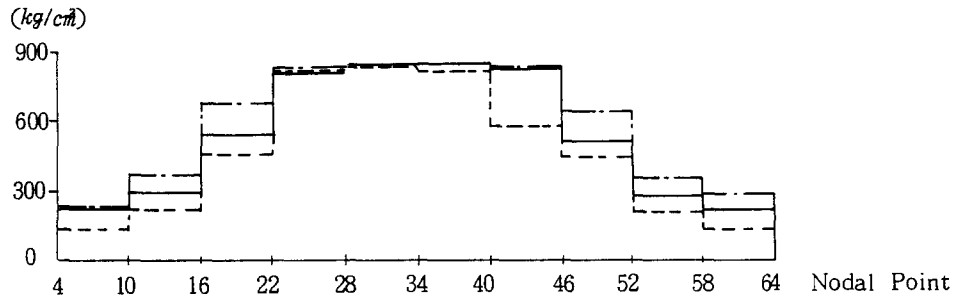


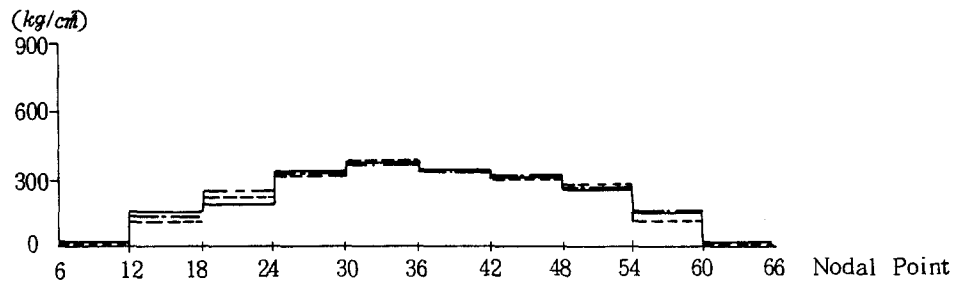
Fig. 5-4. Bending stress of floor.



(a) NO. 1 side girder



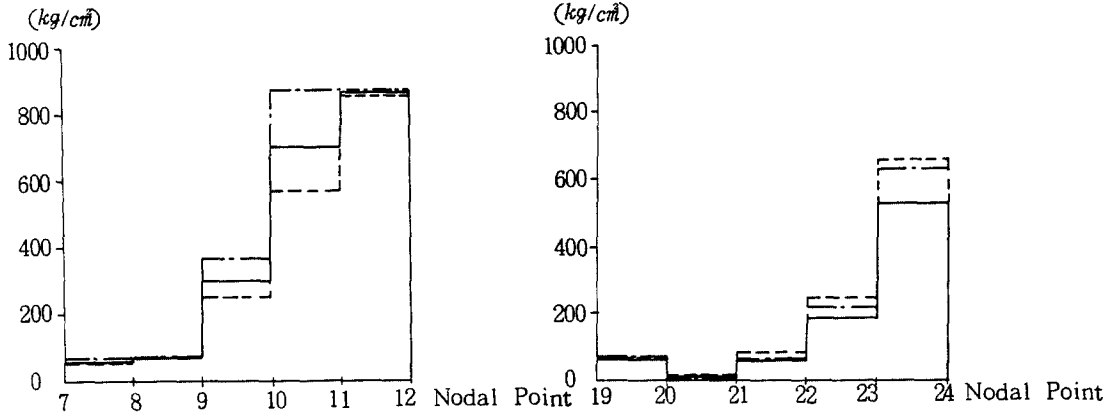
(b) NO. 3 side girder



(c) Side Shell

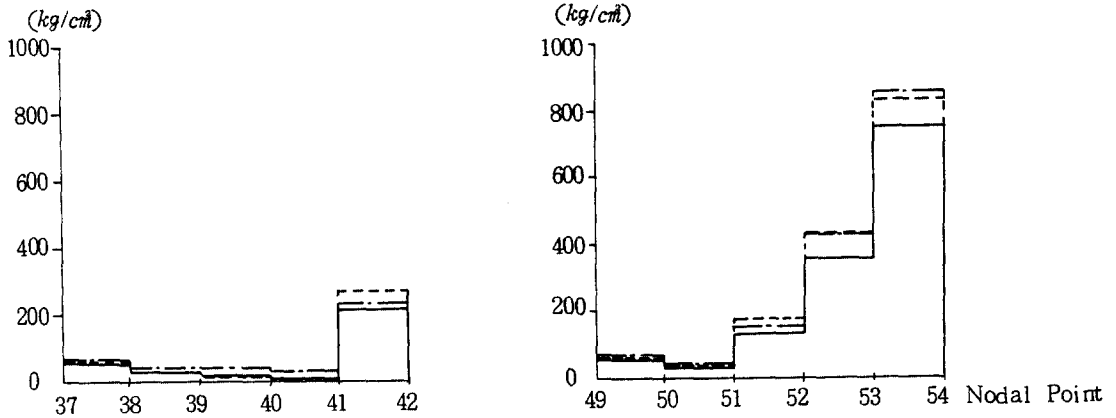
Fig. 5-5. Shear stress of girder.

———— ; Practical
 - - - - - ; Case 1
 — · — · — ; Case 2



(a) NO. 1 Floor

(b) NO. 3 Floor



(c) NO. 5 Floor

(d) NO. 7 Floor

Fig. 5-6. Shear stress of floor.

附 錄

```

PROGRAM FEGRILL (DEFGRIL, OUTPUT, TAPES=DEFGRIL, TAPES=OUTPUT)
DIMENSION X(10), DELTA(10), EPSI(10), G(20)
COMMON /IOPT/ IR, IW, IORT, IPRT, IVAR, IPNA(20)
COMMON /ISHIP/ IOCL, IDEF, IDFF, NGR, NFE, NFF, NSGR, NSFE, NSFF, NSFL,
INSFE1, ITSHD, NELF, NELG, NELT, NODT, NX, NY, NZ, NNX(40), NNY(20), NNZ(10)
COMMON /RSHIP/ HLE, HLF, HLL, BTT, DBH, BOK, UDLF, UDLF, TPB, TIF, TIF, TENX,
ITFL, TGPS, TGRC, TGRD, HTBHD, RTBHD, TTBT, TTBE, HL3HD, BLBHD, TLBT, TLBE
COMMON /MSHIP/ HLEO, HLFO, HLOO, BTTD, DBHD, BLMD, DNSFE, DNSFF, DPSFL,
LPSGR, SFL(40), SGR(10), TTITLE(13)
COMMON /GSHIP/ FORC(200,2,3), STR(200,2,3), DIS(500)
COMMON /DATA/ E, GG, XIN(100), YIN(200), YSM(200), A(200), DEL(200),
1 B1B(200), T1B(200), TB(200), HG(200), TG(200)
COMMON /COORD/ CX(200), CY(200), CZ(200), CF(600), NODE(200,2)
COMMON /STIFF/ SM(6,6), TM(6,6), G3(6,5), TSM(200,400), FSM(200,6,6)
COMMON /SJOFX/ SJOX(4,77), SJOX(8,77), STEP(10)
IR=5
IW=5
CALL DINPUT
READ (IR,100) NJOB
DO 50 KJOB=1, NJOB
WRITE(IW,1)
WRITE(IW,2) KJOB, NJOB
READ(IR,100) IVAR, IOPT, IPRT, N, NC
WRITE(IW,3) IVAR, IOPT, IPRT, N, NC
N=1
NC=0
IF (IOPT.GT.0) GOTO 20
WRITE(IW,200)
ICNT=1
CALL FUNCTN(N, X, FIN, ICNT)
WRITE(IW,220) FIN
GOTO 50
20 CONTINUE
IF (IVAR.EQ.3) N=2
NC=N*2
DO 66 I=1, NC
66 IPNA(I)=0
IF (IOPT.EQ.1) WRITE(IW,330)
WRITE(IW,350)
DO 10 I=1, N
10 READ(IR,110) X(I), DELTA(I), STEP(I), EPSI(I)
WRITE(IW,230) X(I), DELTA(I), STEP(I), EPSI(I), I=1, N)
WRITE(IW,113)
CALL HDCKE(N, NC, FOPT, X, DELTA, STEP, EPSI, ICNT)
WRITE(IW,113)
WRITE(IW,240) X(I), DELTA(I), EPSI(I), I=1, N)
WRITE(IW,250) FOPT
WRITE(IW,260) ICNT
WRITE(IW,113)
IF (IOPT.EQ.0) GOTO 60
DO 559 JJ=1,1
IWB=IW
WRITE(IWB,14) (TITLE(I), I=1,13)
WRITE(IWB,2) KJOB, NJOB
WRITE(IWB,3) IVAR, IOPT, IPRT, N, NC
WRITE(IWB,1235) ICNT, DELTA(1), EPSI(1), FOPT, (X(I), I=1, N)
WRITE(IWB,330)
WRITE(IWB,350)
WRITE(IWB,240) X(I), DELTA(I), EPSI(I), I=1, N)
WRITE(IWB,250) FOPT
IF (IVAR.EQ.1) WRITE(IWB,930)
IF (IVAR.EQ.2) WRITE(IWB,931)
IF (IVAR.EQ.3) WRITE(IWB,932)
WRITE(IWB,113)

```

```

DO 59 J=1,ICNT
WRITE(IWB,990)J,((SJOB(K,J),K=1,4),(SJCX(I,J),I=1,8)
59 CONTINUE
559 CONTINUE
60 CONTINUE
STOP
1 FORMAT(/1X,80(†††)/1X,25(†††),3X,†OPTIMIZATION   TECHNIQUE†,
13X,25(†††)/1X,80(†††))
2 FORMAT(/5X,†JOB NO †;†,I2,† OF†,I2)
3 FORMAT(10X,†IVAR=†,I2,† IOPT=†,I2,† IPRT=†,I2,† N=†,I2,† NC=†,I2
14 FORMAT(/1X,50(1H*)/1X,4(1H*),19A4,4(1H*)/1X,80(1H*))
100 FORMAT(10I5)
110 FORMAT(8F1.0)
113 FORMAT(/1X,0(†††))
200 FORMAT(/1X, ** TEST RUN AS FOR INPUT DATA †††)
220 FORMAT(/1X,† FUNCTION VALUE †;†,2X,†FIN=†,F15.3//)
230 FORMAT(/1X,† INPUT VALUES:†;†,5X,†X(I)†,9X,†DELTA(I)†,9X,†STEP(I)†
1 9X,† SI(I)†//((10X,F15.5,1X,F15.5,1X,F15.5,1X,F15.5//))
240 FORMAT(/1X,†OPTIMUM POINT:†;†,5X,†X(I)†,9X,†DELTA(I)†,9X,
1 †EPS†(I)†//((10X,F15.5,1X,F15.5,1X,F15.5//))
250 FORMAT(/1X,† OPTIMAL VALUE:†;†,5X,F15.3/1X,40(†=†))
260 FORMAT(/1X,†ITERATION(ICNT):†;†,I5/)
330 FORMAT(/10,†HOOKE AND JEEVES DIRECT SEARCH METHOD†)
350 FORMAT(1X,6(†=†))
980 FORMAT(/† CNT†,5X,†OBJ. F. ^ FOR FLOOR†S VALUE†,
1 5X,†DESIGN/REFERENCE VARIABLES†,5X,†RATIOES†)
981 FORMAT(/† CNT†,5X,†OBJ. F. ^ FOR BOTTOM†S VALUE†,
1 5X,†DESIGN/REFERENCE VARIABLES†,5X,†RATIOES†)
982 FORMAT(/† CNT†,5X,†OBJ. F. ^ FOR FLOOR ^ BOTTOM†S VALUE†,
1 5X,†DESIGN/REFERENCE VARIABLES†,5X,†RATIOES†)
990 FORMAT(1X,4,2X,4F8.0,2X,2F6.0,2F3.1,2X,4F6.3)
1230 FORMAT(† INT,DEL,ESP,FOPT =†,I5,3F15.5,
1 / † DESIG VARIABLES =†,10F10.1)
END
SUBROUTINE QINPUT
COMMON /IO TP/ IR,IX,IOPT,IPRT,IVAR
COMMON /IS IP/ IDCL,IOFE,IOFF,NGR,NFE,NFF,NSGP,NSFF,NSFL,
+ NSFEL,ITB,DEL,NEL,NELT,NDOT,NX,NY,NZ,NNX(40),NNY(20),NNZ(10)
COMMON /RS IP/ HLE,HLEF,HLE,HTT,DBH,BDK,UDLE,UOLF,TPB,TIE,TIF,TEK
+ TFL,TGRS,TGRC,TGRD,HTBHD,BTBHD,TTBT,TTBE,HLBHD,BLBHD,TLBT,TLBE
COMMON /MS IP/ HLEQ,HLEFQ,HLEQ,HTTQ,DBHQ,BLHQ,ALHQ,ONSFE,ONSFF,OPSF,
+ OPSGR,SFL(40),SGR(10),TITLE(18)
COMMON /DA A/ E,GG,XIN(200),YIN(200),YSM(200),A(200),QEL(200),
+ BTB(200),TIB(200),TBP(200),HG(200),TG(200)
READ(IR,22) (TITLE(I),I=1,18)
READ(IR,20) HLE,HLEF,HTT,DBH,BDK,UDLE,UOLF,BLMQ
READ(IR,20) TPB,TIE,TIF,TEK,TFL,TGRS,TGRC,TGRD
READ(IR,20) HTBHD,BTBHD,TTBT,TTBE,HLBHD,BLBHD,TLBT,TLBE
HLEQ=HLE+HLEF
HLEQ=HLE
HLEQ=HLE
HLEFQ=HLEF
HTTQ=HTT
DBHQ=DBH
READ(IR,10) NGR,NFE,NFF
READ(IR,10) IDCL,IOFE,IOFF
IF(IOFE.NE.0) IOFE=1
IF(IDFL.NE.0) IDFL=1
IF(IDCL.NE.0) IDCL=1
NSGR = NGR + IDCL
NSFE = NFE + IOFE
NSFF = NFF + IOFF
NSFL = NSF + NSFF + 1
SSR(1) = 0.0

```

```

SFL(1) = 0.0
READ(IR,20) (SGR(I),I=2,NSGR)
READ(IR,20) (SFL(I),I=2,NSFL)
WRITE(IW,14) (TITLE(I),I=1,18)
WRITE(IW,15)
WRITE(IW,161) HLE ,HLE ,HLE ,BTT ,DBH ,BOK ,UDLE ,UDLF ,BL40
WRITE(IW,162) TPB ,TIE ,TIF ,TEDK ,TFL ,TGRS ,TGRC ,TGRD
WRITE(IW,163) HTBHD,BTBHD,TTBT ,TTBE ,HLBHD,BLBHD,TLBT ,TLBE
WRITE(IW,17) NGR ,NFE ,NFF ,IDCL ,IDFE ,IDFF
WRITE(IW,181)NSGR
WRITE(IW,182) (SGR(I),I=1,NSGR)
WRITE(IW,183)NSFL
WRITE(IW,182) (SFL(I),I=1,NSFL)
ONSFE = FLOAT(NSFE-1) + 1.0 / (1.0+FLOAT(IDFE))
ONSFF = FLOAT(NSFF-1) + 1.0 / (1.0+FLOAT(IDFF))
QPSFL = (HLE/ONSFE + HLE/ONSFF) / 2.0
QPSFL = ROUND(QPSFL,3.0)
WRITE(IW,321) QPSFL,ONSFE,ONSFF
NSFE1= NSFE + 1
NODT = NSGR * NSFL
NELT = (NSFL-IDFE-IDFF)*(NSGR-1) + (NSGR-IDCL)*(NSFL-1)
NF = NODT
NX = NSFL
NY = NSGR + NSGR
NZ = 1
IF(IDFE.NE.0) NZ=NZ+1
IF(IDFF.NE.0) NZ=NZ+1
DO 111 I=1,NX
111 NNZ(I) = 1 + NSGR*(I-1)
DO 112 I=1,NSGR
I2 = I + NSGR
NNY(I) = I
112 NNZ(I2) = I + NODT - NSGR
NNZ(1) = NSGR * NSFE1
IF(IDFE.NE.0) NNZ(2) = 1
IF(IDFE.EQ.0 .AND. IDFF.NE.0) NNZ(2) = NODT - NSGR + 1
IF(IDFE.NE.0 .AND. IDFF.NE.0) NNZ(3) = NODT - NSGR + 1
GG = 907692.0
E = 2100000.0
WRITE(IW,200)NODT,NELT,NX,NY,NZ,NF
WRITE(IW,202) GG
IF(NX.EQ.0 .AND. NY.EQ.0 .AND. NZ.EQ.0) GO TO 218
WRITE(IW,214)
WRITE(IW,217) (NNZ(I),I=1,NX)
WRITE(IW,215)
WRITE(IW,217) (NNY(I),I=1,NY)
WRITE(IW,216)
WRITE(IW,217) (NNZ(I),I=1,NZ)
RETURN
218 WRITE(IW,219)
STOP
10 FORMAT(8I5)
20 FORMAT(8F10.0)
22 FORMAT(20A4)
14 FORMAT(//1X,80(1-*)/1X,4(1H*),18A4,4(1H*)/1X,80(1H*))
15 FORMAT(///5X,!-- SHIP INFORMATION DATA LIST AS INPUT ---+//)
161 FORMAT(1X,+ HLE =+,F8.2,+ HLE =+,F8.2,+ BTT =+,F8.2,+ DBH =+,
+F8.2,+ BOK =+,F8.2,+ UDLE =+,F8.4,+ UDLF =+,F8.4,+ BL40 =+,F8.2//)
162 FORMAT(1X,+ TPB =+,F8.2,+ TIE =+,F8.2,+ TIF =+,F8.2,+ TEDK =+,
+F8.2,+ TFL =+,F8.2,+ TGRS =+,F8.2,+ TGRC =+,F8.2,+ TGRD =+,F8.2//)
163 FORMAT(1X,+ HTBHD=+,F8.2,+ BTBHD=+,F8.2,+ TTBT =+,F8.2,+ TTBE =+,
+F8.2,+ HLBHD=+,F8.2,+ BLBHD=+,F8.2,+ TLBT =+,F8.2,+ TLBE =+,F8.2//)
17 FORMAT(1X,+ NGR=+,I2,+ NFE=+,I2,+ NFF=+,I2,+ IDCL=+,I2,+ IDFE=+,
+I2,+ IDFF=+,I2//)

```

```

181 FOPMAT(1X,↑ SP. OF GIRDER -- SGR(↑ ,I2, ↑)↑)
182 FOPMAT(5X,5F10.2)
183 FOPMAT(1X,↑ SP. UF FLOOR -- SFL(↑ ,I2, ↑)↑)
321 FOPMAT(↑↑ *** QINPUT: OPSFL, ONSFE, ONSFF=↑,F3.2,2F5.1/)
200 FOPMAT(/3X,↑NODT=↑,I4,3X,↑NELT=↑,I4,3X,↑NX=↑,I4,3X,↑NY=↑,I4,
+3X,↑NZ=↑,I4,3X,↑NF=↑,I4)
202 FOPMAT(/↑ *** MATERIAL PROPERTY ****/↑ G=↑,F12.3,↑ E=↑,F12.3)
214 FOPMAT(/↑ *** FIXED NODAL POINT IN X-DIRECTION ****)
215 FOPMAT(/↑ *** FIXED NODAL POINT IN Y-DIRECTION ****)
216 FOPMAT(/↑ *** FIXED NODAL POINT IN Z-DIRECTION ****)
217 FOPMAT(2(10I5))
219 FOPMAT(/↑↑ *** ERROR *** NO EXEC. - DUE TO NO CONSTRAINED COND.↑)
END
SUBROUTINE PINPUT(ICNT)
COMMON /IOPT/ IR, IW, IOPT, IPRT, IVAR
COMMON /ISHIP/ IDCL, IDFE, IDFF, NGR, NFE, NFF, NSGR, NSFE, NSFF, NSFL,
+NSFE1, ITBHO, NELF, NSELG, NELT, NODT, NX, NY, NZ, VNX(40), VNY(20), VNZ(10)
COMMON /RSHIP/ HLE, HLF, HOL, BT, DBH, BDK, UOLF, UOLF, TPB, TIE, TIF, TEOK,
+TFL, TGRS, TGPC, TGRD, HTBHO, BTBHO, TTBT, TTBE, HLBHO, BLBHO, TLBT, TLBE
COMMON /MSHIP/ HLEO, HLF0, HOL0, BT0, DBH0, BLM0, ONSFE, ONSFF, OPSFL,
+OPSGR, SFL(40), SGR(10), TITL(18)
COMMON /DATA/ E, GG, XIN(200), YIN(200), YSM(200), A(200), QEL(200),
+BI9(200), TIB(200), TBP(200), HG(200), TG(200)
COMMON /COORD/ CX(200), CY(200), CZ(200), CF(600), NODE(200,2)
DIMENSION FZ(200)
IF(IOPT.EQ.0 .OR. (IOPT.NE.0.AND.ICNT.EQ.1)) GO TO 55
IF(IVAR.EQ.2) GO TO 35
NSFL1 = NSFL - 1
SFL(1)=0.0
5 SFL(2)=OPSFL/(1.0+FLDAT(IDFE))
SFL(NSFL)=OPSFL/(1.0+FLDAT(IDFF))
DO 10 I=3, NSFL1
10 SFL(I)=OPSFL
HLE=OPSFL*ONSFE
HLF=OPSFL*ONSFF
HOL=HLE+HLF
VHOL=HOL-HOLO
RLE=HLE/HLEO
RLF=HLF/HLF0
ROL=HOL/HOLO
WRITE(IW,22)OPSFL, HLE, HLF, HOL, RATIO: RLE, RLF, ROL
IF(ABS(VHOL).LT.3LM0) GO TO 55
WRITE(IW,33) HOL, HOLO, VHOL, BLM0
22 FOPMAT(/1X,↑ OPSFL, HLE, HLF, HOL, RATIO: RLE, RLF, ROL↑,
+5X,↑SPINPUT↑/1X, F6.0, 3F3.2, 8X, 3F5.2)
33 FOPMAT(/1X,↑ *** WARNING ****/1X,↑MODIFIED HOLD LENGTH↑,
+F7.1,↑ IS .GT. OR .LT. ORIGINAL HOLD(↑, F8.2, ↑)↑/
+1X,↑ AS MUCH AS VHOL(↑, F7.2, ↑) EXCESS THE INPUT=LIMIT BLM0(↑,
+5.1, ↑)↑/)
55 CONTINUE
K=0
SFLI=0.0
DO 101 I=1, NSFL
SGRJ=0.0
SFLI=SFLI+SFL(I)
DO 101 J=1, NSGR
K=K+1
SGRJ=SGRJ+SGR(J)
CX(K)=SFLI
CY(K)=SGRJ
CZ(K)=0.0
101 CONTINUE
KL=0
DO 121 I=1, NSFL

```

```

SPACEF=SFL(I)/2.0
IF(I.NE.NSFL) SPACEF=(SFL(I+1)+SFL(I))/2.0
UDL=UDLF
IF(I.LT.NSFE1) UDL=UDLE
DO 121 J=1,NSGR
KL=KL+1
SPACEG=SGR(J)/2.0
IF(J.NE.NSGR) SPACEG=(SGR(J+1)+SGR(J))/2.0
FZ(KL)=UDL*SPACEF*SPACEG
IF(I.NE.NSFE1) GO TO 121
FZ(KL)=SPACEG*(UDLE*SFL(I)+UDLF*SFL(I+1))/2.0
121 CONTINUE
IF=KL
IF(NF.NE.NODT)WRITE(IW,122) NF,NODT
122 FORMAT(' ** WARNING ** <NF>=',I3,' IS NOT EQUAL TO <NODT>=',I3)
NT=NODT*3
DO 123 I=1,NT
123 CF(I)=0.0
DO 124 I=1,NF
124 CF(3*I)=FZ(I)
KN1=1
K12=NSGR
KE1=NSFL-1
KE2=NSGR-1
NEMX=NSFL-IDFE-IDFF
NEMY=NSGR-IDCL
KE=0
ITBHD=NSFE1-IDFE
DO 131 I=1,NEMX
I1=I+IDFE
I2=I1+1
IF(I2.GT.NSFL) SFL(I2)=0.0
BI3KE=(SFL(I2)+SFL(I1))/2.0
IF(I.EQ.1 .AND. IDFE.EQ.1) BI3KE=SFL(I2)/2.0+SFL(I1)
IF(I.EQ.NEMX.AND.IDFF.EQ.1)BI3KE=SFL(I2) +SFL(I1)/2.0
DO 131 J=1,KE2
KE=KE+1
IF(J.EQ.1) NODEI=KN2*(IDFE+I-1)+KN1
IF(J.NE.1) NODEI=NODEI+KN1
NODEJ=NODEI+KN1
NODE(KE,1)=NODEI
NODE(KE,2)=NODEJ
QEL(KE)=CY(NODEJ)-CY(NODEI)
BIB(KE)=BIBKE
TIB(KE)=TIE
IF(I.GT.ITBHD) ,TIB(KE)=TIF
TBP(KE)=TPB
HG(KE)=DBH
TG(KE)=TFL
IF(J.EQ.1 .AND. TEDK.GT.0.05) TG(KE)=TEDK
IF(I.EQ.1.AND.IDFE.EQ.0)TG(KE)=TG(KE)/2.0
IF(I.NE.ITBHD) GO TO 131
BI3(KE)=2TBHD
TIB(KE)=TTRT
TBP(KE)=TTBT
HG(KE) =HTBHD
TG(KE) =TTBE
131 CONTINUE
NELF=KE
DO 135 I=1,NEMY
I1=I+IDCL
I2=I1+1
IF(I2.GT.NSGR) SGR(I2)=0.0
BI3KE=(SGR(I2)+SGR(I1))/2.0

```



```

IF(I.EQ.1 .AND. IDCL.EQ.1) BIBKE=SGR(I2)/2.0+SGR(I1)
DO 135 J=1,KE1
KE=KE+1
IF(J.EQ.1) NODEI=I+IDCL
IF(J.NE.1) NODEI=NODEI+KN2
NODEJ=NODEI+KN2
NODE(KE,1)=NODEI
NODE(KE,2)=NODEJ
QEL(KE)=CX(NODEJ)-CX(NODEI)
BIB(KE)=BIBKE
TIB(KE)=TIE
IF(I.GT.NSFE) TIB(KE)=TIF
TBP(KE)=TPB
HG(KE) =DBH
TG(KE) =TGRS
IF(I.EQ.1 .AND. IDCL.EQ.0) TG(KE)=TGRC/2.0
IF(30K.LT.55.0) GO TO 135
IF(I.EQ.1 .AND.IDCL.NE.0) TG(KE)=TGRD
IF(I.EQ.2 .AND. IDCL.EQ.0) TG(KE)=TGRD
IF(I.LT.NEMY) GO TO 135
BIB(KE)=BLBHD
TIB(KE)=TLBT
TBP(KE)=TLBT
HG(KE) =HLBHD
TG(KE) =TLBE
135 CONTINUE
NELT=KE
NELG=NELT-NELF
IF(ICNT.EQ.1) WRITE(IW,136) NELF,NELG,NELT
136 FORMAT(/↑ NO. OF ELEMENTS: FLOOR=↑,I3,↑ GIR.=↑,I3,↑ TOTAL=↑,I3)
DO 139 I=1,NELT
TIBBP=TIB(I)+TBP(I)
A(I)=HG(I)*TG(I)
XIN(I)=BIB(I)*(TBP(I)*(TIB(I)*HG(I)/TIBBP)**2
+TIB(I)*(TBP(I)*HG(I)/TIBBP)**2)
YIN(I)=(BIB(I)*(HG(I)+TIBBP)**3-(BIB(I)-TG(I))*HG(I)**3)/12.
YSM(I)=YIN(I)/(HG(I)+TIBBP)*2.
139 CONTINUE
IF(ICNT.GT.1 .OR. IPRT.LT.1) RETURN
WRITE(6,210)
DO 211 I=1,NDOF
I3=I*3
211 WRITE(6,212) I,CX(I),CY(I),CZ(I),CF(I3-2),CF(I3-1),CF(I3)
WRITE(6,214)
DO 216 I=1,NELT
216 WRITE(6,218) I,(NODE(I,J),J=1,2),QEL(I),A(I),XIN(I),YIN(I),YSM(I),
+BIB(I),HG(I),TG(I),TIB(I),TBP(I)
RETURN
210 FORMAT(/↑ *** NODAL POINT COORD.-X,Y,Z AND NODAL FORCE ****/
+I,↑ NO.↑,7X,↑X↑,9X,↑Y↑,9X,↑Z↑,8X,↑FX↑,8X,↑FY↑,8X,↑FZ↑/)
212 FORMAT(1X,I5,6F10.2)
214 FORMAT(/1X,↑*** ELEMENT INPUT DATA LIST OF ALL ****/↑ EL. N
+DGE LENG. SHEAR↑,13X,↑INERTIA ^ S.M.↑,23X,↑DIMENSION↑/↑ NO.
+ I J↑,11X,↑AREA↑,8X,↑XIN↑,10X,↑YIN↑,10X,↑YSM↑,8X,↑R↑,7X,↑N↑,4X,
I↑T TIB TBP↑/)
218 FORMAT(1X,3I4,2F8.2,3E13.4,2F8.2,3F5.2)
END
SUBROUTINE OPMAIN(ICNT)
COMMON /IOPT/ IR,IW,IOPT,IPRT,IVAR
CALL PINPUT(ICNT)
CALL GRILL
CALL STRESS
RETURN
END

```

```

SUBROUTINE STRESS
COMMON /IOPTP/ IR,IW,IOPT,IPRT,IVAP
COMMON /ISHIP/ IOCL,IDFE,IDFF,NGR,NFE,NFF,NSG0,NSFE,NSFF,NSFL,
+NSFE1,ITBHD,NELF,NELG,NELT,NDDT,NX,NY,NZ,NNX(40),NNY(20),NNZ(10)
COMMON /DSHIP/ FORC(200,2,3),STR(200,2,3)
COMMON /DATA/ E,GG,XIN(200),YIN(200),YSM(200),A(200),DEL(200),
+HIB(200),TIB(200),TBP(200),HG(200),TG(200)
COMMON /COORD/ CX(200),CY(200),CZ(200),CF(600),VCDE(200,2)
DIMENSION ASTR(6)
DATA ASTR/970.0,1420.0,1800.0,870.0,1300.,1560.0/
DO 500 K=1,NELT
IFG=0
IF(K.GT.NELF) IFG=3
DO 400 I=1,2
IFG1=IFG+1
IFG2=IFG+2
IFG3=IFG+3
STR2=ABS(FORC(K,I,2))/YSM(K)
IF(STR2.GT.ASTR(IFG2)) WRITE(IW,401) K,NODE(K,I),STR2,ASTR(IFG2)
IF(STR2.LT.ASTR(IFG3)) GO TO 440
WRITE(IW,402) STR2,ASTR(IFG3)
STR1=ABS(FORC(K,I,3))/A(K)
STR3=SQRT(STR2**2+3.*STR1**2)
GO TO 450
440 CONTINUE
IDEC=0
INCR=0
300 A(K)=TG(K)+HG(K)
STR1=ABS(FORC(K,I,3))/A(K)
INCR=INCR+1
IF(INCR.GT.10) GO TO 350
IF(IDEC.GT.0) GO TO 350
PCNT=100.0*STR1/ASTR(IFG1)
IF(PCNT.LE.100.05) GO TO 350
FACTR=1.0
IF(PCNT.GT.200.0) FACTR=20.0
IF(PCNT.LE.200.0 .AND. PCNT.GT.150.0) FACTR=10.0
IF(PCNT.LE.150.0 .AND. PCNT.GT.120.0) FACTR= 6.0
IF(PCNT.LE.120.0 .AND. PCNT.GT.110.0) FACTR= 4.0
IF(PCNT.LE.110.0 .AND. PCNT.GT.100.5) FACTR= 2.0
IF(PCNT.LE. 95.0 .AND. PCNT.GT. 75.0) FACTR=-1.0
IF(PCNT.LE. 75.0) FACTR=-3.0
TG(K)=TG(K)+0.05*FACTR
GO TO 300
350 CONTINUE
IDEC=1
STR3=SQRT(STR2**2+3.*STR1**2)
PCNT=100.0*STR3/ASTR(IFG3)
IF(PCNT.LE.100.05) GO TO 460
IF(INCR.GT.10) GO TO 450
FACTR=1.0
IF(PCNT.GT.150.0) FACTR=10.0
IF(PCNT.LE.150.0 .AND. PCNT.GT.120.0) FACTR= 6.0
IF(PCNT.LE.120.0 .AND. PCNT.GT.110.0) FACTR= 4.0
IF(PCNT.LE.110.0 .AND. PCNT.GT.100.5) FACTR= 2.0
IF(PCNT.LE. 95.0 .AND. PCNT.GT. 75.0) FACTR=-1.0
IF(PCNT.LE. 75.0) FACTR=-3.0
TG(K)=TG(K)+0.05*FACTR
GO TO 300
450 CONTINUE
TG(K)=TG(K)*2.0
IF(INCR.GT.15) WRITE(IW,403) INCR,K,NODE(K,I),TG(K)
460 STR(K,I,1)=STR1
STR(K,I,2)=STR2

```

```

      STR(K,I,3)=STP3
400 CONTINUE
500 CONTINUE
      RETURN
401 FORMAT(1X,1*** WARNING *** EL. NO.,I3,↑ NODE↑,I3,
      ↑ BEND. STRESS=↑,F7.1,↑ EXCESS ALLOWABLE BEND. (↑,F6.1,↑)↑)
402 FORMAT(1X,↑$$$ ERROR OF UNREASONABLE CASE $$$↑,
      ↑ BEND. STRESS=↑,F7.1,↑ EXCESS ALLOWABLE EQUIV.(↑,F6.1,↑)↑)
409 FORMAT(/1X,↑$$$ EXCESSIVE INCREASING ITER↑↑4(↑,I2,↑) AT EL. NO.=↑,
      ↑I3,↑ NODE=↑,I3/7X,↑PENALTY INCREASE OF WEB PL. THICK.(2 TIMES)=↑
      ↑,F7.2/)
      END
      SUBROUTINE DISFST(IDS,IW)
      COMMON /ISHIP/ IDCL,IOFE,IOFF,NGR,NFE,NFF,NSG?,NSFE,NSFF,NSFL,
      ↑NSFE1,ITBHD,NELF,NELG,NELT,NODT,NX,NY,NZ,NNX(4),NNY(2),NNZ(10)
      COMMON /OSHIP/ FORC(200,2,3),STP(200,2,3),DIS(600)
      COMMON /DATA/ E,GG,XIN(200),YIN(200),YSM(200),A(200),QEL(200),
      ↑BIB(200),TIB(200),TBP(200),HG(200),TG(200)
      COMMON /COORD/ CX(200),CY(200),CZ(200),CF(600),NODE(200,2)
      GO TO (100,200,100),IDS
100 CONTINUE
      WRITE(IW,101)
101 FORMAT(/2X,↑NODE↑,9X,↑COORDINATES↑,19X,↑NODAL LOADS↑,19X,↑DISPL
      ↑ACEMENTS↑/3X,↑NO.↑,6X,↑CX↑,8X,↑CY↑,8X,↑CZ↑,8X,↑FX↑,8X,↑FY↑,8X,↑FZ↑
      ↑,5X,↑THETA(X)↑,3X,↑THETA(Y)↑,6X,↑DZ↑/1X,105(↑-↑)/)
      DD 110 KJ=1,NODT
      KL =KJ*3
      <LL =KL-2
      <LLL=KL-1
      WRITE(6,111) KJ,CX(KJ),CY(KJ),CZ(KJ),
      ↑ CF(<LLL),CF(KLL),CF(KL),DIS(<LLL),
      ↑ DIS(<LL),DIS(KL)
110 CONTINUE
111 FORMAT(1X,I5,3F10.2,3F10.0,2E11.3,F11.3)
      IF(IDS.NE.3) RETURN
200 CONTINUE
      WRITE(IW,201)
201 FORMAT(/2X,↑ELEM NODE↑,17X,↑FORCES↑,24X,↑STRESS↑,17X,↑YIN↑,7X,
      ↑↑HG↑,3X,↑QEL↑/4X,↑NO. NO.↑,6X,↑MX↑,10X,↑MY↑,13X,↑FZ↑,4X,↑SHEAR↑,
      ↑4X,↑BEND.↑,4X,↑EQUI.↑,7X,↑YSM↑,7X,↑TG↑,9X,↑AS↑/1X,105(↑-↑)/)
      DD 210 K=1,NELT
      WRITE(IW,211) K,NODE(K,1),(FORC(K,1,J),J=1,3),(STR(K,1,J),J=1,3)
      ↑,YIN(K),HG(K),QEL(K),
      ↑NODE(K,2),(FORC(K,2,J),J=1,3),(STR(K,2,J),J=1,3),
      ↑YSM(K),TG(K),A(K)
210 CONTINUE
211 FORMAT(1X,2I5,2E12.4,F13.0,3F9.0,E12.4,F9.2,F10.1/
      ↑6X,I5,2E12.4,F13.0,3F9.0,E12.4,F8.2,F10.1)
      RETURN
      END
      SUBROUTINE GRILL
      COMMON /IOPT/ IR,IW,IOPT,IPRT,IVAR
      COMMON /ISHIP/ IDCL,IOFE,IOFF,NGR,NFE,NFF,NSGR,NSFE,NSFF,NSFL,
      ↑NSFE1,ITBHD,NELF,NELG,NELT,NODT,NX,NY,NZ,NNX(4),NNY(2),NNZ(10)
      COMMON /OSHIP/ FORC(200,2,3),STR(200,2,3),DIS(600)
      COMMON /DATA/ E,GG,XIN(200),YIN(200),YSM(200),A(200),QEL(200)
      ↑,BIB(200),TIB(200),TBP(200),HG(200),TG(200)
      COMMON /COORD/ CX(200),CY(200),CZ(200),CF(600),NODE(200,2)
      COMMON /STIFF/ SM(6,6),TM(6,6),GB(6,6),TSM(200,400),FSM(200,6,6)
      DIMENSION IDENT(600),REAC(600),CFI(600),U(6),RU(6),B(2,6),T(2,3)
      WRITE(IW,1)
1 FORMAT(↑ *** GRILLAGE ANALYSIS BY F.E.M ***↑)
      NT=NODT*3
      NT2=NT*2

```

```

      DO      I= NT
      DJ 350 J=1,NT2
350  TSM(I,J)=0.0
      DO 380 NE=1,NELT
      CALL GRID(NE)
      CALL ASSEM(NE,2,3)
380  CONTINUE
      DO 381 I=1,NT
      IDENT(I)=I
381  CONTINUE
      IF(NX.EQ.0) GO TO 385
      DO 382 I=1,NX
      N=3*NX(I)-2
      IDENT(N)=0
382  CONTINUE
385  CONTINUE
      IF(NY.EQ.0) GO TO 395
      DO 392 I=1,NY
      N=3*NY(I)-1
      IDENT(N)=0
392  CONTINUE
395  CONTINUE
      IF(NZ.EQ.0) GO TO 405
      DO 402 I=1,NZ
      N=3*NZ(I)
      IDENT(N)=0
402  CONTINUE
405  CONTINUE
      MM=0
      DO 410 I=1,NT
      IF(IDENT(I).EQ.0) GO TO 410
      M1=MM+1
      IDENT(MM)=I
410  CONTINUE
      DO 420 I=1,M1
      IA=IDENT(I)
      CFI(I)=CF(IA)
      DO 420 J=1,MM
      JA=IDENT(J)
      TSM(I,J)=TSM(IA,JA)
420  CONTINUE
      IF(IPRT.LT.10) GO TO 4422
      WRITE(IW,421) MM
      DO 4421 I=1,MM
4421  WRITE(IW,422) I,(TSM(I,J),J=1,MM)
      421  FORMAT(//↑*** TOT. STIF. MATRIX BEFORE CALL-INVERT **↑,↑MM=↑,I3)
      422  FORMAT(1X,↑BEFORE: ROW=↑,I3/(10E11.3))
4422  CONTINUE
      CALL INVERT(TSM,200,400,MM)
      IF(IPRT.LT.9) GO TO 4424
      M11=M1+1
      MM2=MM+MM
      WRITE(IW,423) MM2
      DO 4423 I=1,M1
4423  WRITE(IW,424) I,(TSM(I,J),J=M11,MM2)
      423  FORMAT(//↑*** TOT. STIF. MATRIX AFTER CALL-INVERT **↑,↑MM2=↑,I3)
      424  FORMAT(1X,↑AFTER: ROW=↑,I3/(10E11.3))
4424  CONTINUE
      DO 430 I=1,NT
      DIS(I)=0.
430  CONTINUE
      DO 440 I=1,MM
      REAC(I)=0.
      DO 441 J=1,MM

```

```

      JJ=J+1M
      REAC(I)=P*EAC(I)+FSM(I,JJ)*CFI(J)
441 CONTINUE
440 CONTINUE
      DD 450 I=1,MM
      IA=IDENT(I)
      DIS(IA)=REAC(I)
450 CONTINUE
      IF(IPRT.LT.9) GO TO 489
      WRITE(IW,1450)
1450 FORMAT(/IX,†*** FOR CHECK OF LOAD, ^ DISP. MATRIX *** 1-NT†/)
      WRITE(IW,1451) (I,CF(I),DIS(I),I=1,NT)
1451 FORMAT(IX,I5,F12.2,5X,E12.4)
489 IF(IPRT.LT.3) GO TO 499
      WRITE(IW,1452)
1452 FORMAT(/IX,†*** FOR CHECK OF LOAD ^ DISP. MATRIX *** 1-1M†/)
      WRITE(IW,1451) (I,CFI(I),REAC(I),I=1,1M)
499 CONTINUE
      DD 500 NE=1,NELT
      N1=NODE(NE,1)
      N2=NODE(NE,2)
      U(1)=DIS(N1*3-2)
      U(2)=DIS(N1*3-1)
      U(3)=DIS(N1*3)
      U(4)=DIS(N2*3-2)
      U(5)=DIS(N2*3-1)
      U(6)=DIS(N2*3)
      DD 540 I=1,6
      RU(I)=0.0
      DD 540 J=1,6
      RU(I)=RU(I)+FSM(NE,I,J)*U(J)
540 CONTINUE
      DD 541 I=1,2
      I13=3*(I-1)
      DD 541 J=1,3
      IJ=I13+J
      FORC(NE,I,J)=RU(IJ)
541 CONTINUE
      IF(IPRT.LT.3 .OR. NE.GT. 5) GO TO 500
      WRITE(IW,549) NE,((FSM(NE,I,J),J=1,6),I=1,6),(U(I),I=1,6),
      +(RU(I),I=1,6)
549 FORMAT(IX,†** MOD. REAC, FORCE FSM(NE,I,J) * U(J) FOR NE=†,I3/
      †† EL.GL.ST. MATRIX FSM(6,6)=†,6E11.3/5(28X,5E11.3/),
      †† OUT GL DISP. MATRIX U(6) =†,6E11.3/
      †† EL.-I,J-REAC. MATRIX RU(6)=†,3E11.3/28X,3E11.3/)
500 CONTINUE
      RETURN
      END
      SUBROUTINE GRID(NE)
      COMMON /IOPT/ IR,IW,IOPT,IPRT,IVAR
      COMMON /DATA/ E,GG,XIN(200),YIN(200),YSM(200),A(200),QEL(200),
      + B13(200),T13(200),T3P(200),HG(200),TG(200)
      COMMON /COORD/ CX(200),CY(200),CZ(200),CF(600),N0DE(200,2)
      COMMON /STIFF/ SM(6,6),FM(6,6),GB(6,6),TSM(200,400),FSM(200,6,6)
      II=N0DE(NE,1)
      JJ=N0DE(NE,2)
      XX=CX(JJ)-CX(II)
      YY=CY(JJ)-CY(II)
      ZZ=CZ(JJ)-CZ(II)
      QL=SQRT(XX*XX+YY*YY+ZZ*ZZ)
      DD 100 I=1,6
      DD 100 J=1,6
      SM(I,J)=0.0
      TM(I,J)=0.0

```

```

100 CONTINUE
  XI=XIN(NE)
  YI=YIN(NE)
  SM(1,1)=GG*XI/QL
  SM(1,4)=-SM(1,1)
  SM(2,2)=4.0*E*YI/QL
  SM(2,3)=-6.0*E*YI/(QL*QL)
  SM(2,5)=SM(2,2)/2.0
  SM(2,6)=-SM(2,3)
  SM(3,3)=12.0*E*YI/(QL*QL*QL)
  SM(3,5)=SM(2,3)
  SM(3,6)=-SM(3,3)
  SM(4,4)=SM(1,1)
  SM(5,5)=SM(2,2)
  SM(5,6)=SM(2,6)
  SM(6,6)=SM(3,3)
  DO 150 I=1,5
    IP1=I+1
    DO 150 J=IP1,6
      SM(I,J)=SM(I,J)
150 CONTINUE
  IF(IPRT.LT.3) GO TO 155
  IF(NE .GT.5) GO TO 155
  WRITE(IW,1420) NE
  WRITE(IW,1421) ((SM(I,J),J=1,6),I=1,6)
155 CONTINUE
  SN=YY/QL
  CS=XX/QL
  TM(1,1)=CS
  TM(1,2)=SN
  TM(2,1)=-SN
  TM(2,2)=CS
  TM(3,3)=1.0
  DO 200 I=1,3
    DO 200 J=1,3
      TM(I+3,J+3)=TM(I,J)
200 CONTINUE
  DO 300 I=1,6
    DO 300 J=1,6
      GB(I,J)=0.0
    DO 300 K=1,6
      GB(I,J)=GB(I,J)+TM(K,I)*SM(K,J)
300 CONTINUE
  DO 400 I=1,6
    DO 400 J=1,6
      SM(I,J)=0.0
    DO 400 K=1,6
      SM(I,J)=SM(I,J)+GB(I,K)*TM(K,J)
400 CONTINUE
  DO 500 I=1,6
    DO 500 J=1,6
      FSM(NE,I,J)=GB(J,I)
500 CONTINUE
  IF(IPRT.LT.3) RETURN
  IF(NE .GT.5) RETURN
  WRITE(IW,1422) ((TM(I,J) ,J=1,6),I=1,6)
  WRITE(IW,1423) ((GB(I,J) ,J=1,6),I=1,6)
  WRITE(IW,1424) ((SM(I,J) ,J=1,6),I=1,6)
  WRITE(IW,1425) ((FSM(NE,I,J),J=1,6),I=1,6)
  RETURN
1420 FORMAT(/,1X,†*** ELEMENT MATRIX INFO RM FOR EL. NO.=†,I4/)
1421 FORMAT(1X,†ELEMENT LOCAL STIFF. MATRIX SM†/(3X,6E11.3))
1422 FORMAT(1X,†ELEMENT L-G TRANSFORM MATRIX TM†/(5X,6E11.3))
1423 FORMAT(1X,†ELEMENT INDUCED TRANS MATRIX GB†/(3X,6E11.3))

```

```

1424 FORMAT(IX,↑ELEMENT GLOBAL STIFF. MATRIX SM↑/(5X,6E11.3))
1425 FORMAT(IX,↑ELEMENT FORCE CAL. MATRIX FSM↑/(5X,6E11.3))
      END
      SUBROUTINE ASSEM(NE,NOD,NDGR)
      COMMON /COORD/ CX(200),CY(200),CZ(200),CF(600),NODE(200,2)
      COMMON /STIFF/ SM(6,6),TM(6,6),GB(6,6),TSM(200,400),FSM(200,6,6)
      DO 100 I=1,NOD
      DO 100 J=1,NOD
      NI=(NODE(NE,I)-1)*NDGR
      NJ=(NODE(NE,J)-1)*NDGR
      IS=(I-1)*NDGR
      JS=(J-1)*NDGR
      DO 100 K=1,NDGR
      DO 100 L=1,NDGR
      NIK=NI+K
      NJL=NJ+L
      ISK=IS+K
      JSL=JS+L
100 TSM(NIK,NJL)=TSM(NIK,NJL)+SM(ISK,JSL)
      RETURN
      END
      SUBROUTINE INVERT(TSM,IROW,JCOL,N)
      DIMENSION TSM(IROW,JCOL)
      EPSIL=0.00000001
      N1=2*N
      DO 30 I=1,N
      DO 30 J=1,N
      IF(I-J) 10,20,10
10 TSM(I,J+N1)=0.0
      GO TO 30
20 TSM(I,J+N1)=1.0
30 CONTINUE
      DO 500 IP=1,N
      IM=IP
      IST=IP+1
      IF(IP-4) 40,160,40
40 CONTINUE
      DO 100 I=IST,N
      IF(ABS(TSM(IM,IP))-ABS(TSM(I,IP))) 90,100,100
90 IM=I
100 CONTINUE
      IF(ABS(TSM(IM,IP))-EPSIL) 110,130,130
110 WRITE(I,120) TSM(IM,IP)
120 FORMAT(/10X,↑ PIVOT ELEMENT =↑,E15.5)
160 CONTINUE
      IF(TSM(IM,IP)) 130,700,130
130 CONTINUE
      IF(IM-IP) 140,200,140
140 CONTINUE
      DO 150 J=IP,N1
      AL=TSM(IP,J)
      TSM(IP,J)=TSM(IM,J)
      TSM(IM,J)=AL
150 CONTINUE
200 CONTINUE
      AL=TSM(IP,IP)
      TSM(IP,IP)=1.0
      DO 250 J=IST,N1
      TSM(IP,J)=TSM(IP,J)/AL
250 CONTINUE
      DO 400 I=1,N
      IF(I-IP) 300,400,300
300 AL=TSM(I,IP)
      DO 350 J=IP,N1

```

```

TSM(I,J)=TSM(I,J)-AL*TSM(IP,J)
350 CONTINUE
400 CONTINUE
500 CONTINUE
  NP=N+1
  GO TO 55
700 WRITE(IW,710) IM,IP,TSM(IM,IP)
710 FJRMAT(/10X,↑TSM(↑,I3,↑,↑,I3,↑)=↑,F10.3)
  55 CONTINUE
  RETURN
  END
SUBROUTINE CNTRL(N,NC,X,F,RK,ICNT,IRESET)
  DIMENSION X(N),G(30)
  COMMON /IDPTP/ IR,IW,IDPT,IPRT,IVAR,IPNA(20)
  COMMON /SJOFX/ SJOX(4,77),SJOX(8,77),STEP(10)
  ICNT=ICNT+1
  WRITE(IW,555) ICNT
  IF(ICNT.EQ.1) CALL FUNCTN(N,X,F,ICNT)
  IF(NC.EQ.0) GO TO 200
100 CALL CONSTR(N,NC,X,G)
  ISW=0
  DO 110 J=1,NC
    IF(G(J).GT.-1.0E-30) GO TO 110
    ISW=1
110 CONTINUE
    IF(ISW.EQ.0) GO TO 200
111 IF(G(1).GT.-1.0E-30) GO TO 112
    IPNA(1)=IPNA(1)+1
    X(1)=X(1)+2.0*STEP(1)*IPNA(1)
112 IF(G(2).GT.-1.0E-30) GO TO 115
    IPNA(2)=IPNA(2)+1
    X(1)=X(1)-2.0*STEP(1)*IPNA(2)
115 CONTINUE
    IF(NC.NE.2) GO TO 100
113 IF(G(3).GT.-1.0E-30) GO TO 114
    IPNA(3)=IPNA(3)+1
    X(2)=X(2)+2.0*STEP(2)*IPNA(3)
114 IF(G(4).GT.-1.0E-30) GO TO 119
    IPNA(4)=IPNA(4)+1
    X(2)=X(2)-2.0*STEP(2)*IPNA(4)
119 CONTINUE
    GO TO 100
200 IF(ICNT.GT.1) CALL FUNCTN(N,X,F,ICNT)
    IRESET=0
700 WRITE(IW,601) ICNT,IRESET,F,RK,(X(I),I=1,N)
    RETURN
555 FJRMAT(/1X,↑SUB-CONTRL CALLED AT ICNT=↑,I3)
601 FJRMAT(1X,↑SCONTRL:ICNT=↑,I3,↑ IRE=↑,I1,
  ↑↑ OBJ. F=↑,E12.5,↑ RK=↑,E10.3,↑ X(N)=↑,5F6.1/)
  END
SUBROUTINE FUNCTN(N,X,F,ICNT)
  DIMENSION X(N)
  COMMON /IDPTP/ IR,IW,IDPT,IPRT,IVAR
  COMMON /ISHIP/ IDCL,IDFE,IDFF,NGR,NFE,NFF,NSGR,NSFE,NSFF,NSFL,
  +NSFEL,ITBHD,NELF,NELG,NELT,NOOT,NX,XY,NZ,NNX(40),NNY(20),NNZ(10)
  COMMON /RSHIP/ HLE,HLF,HGL,BTT,DBH,BDK,UOLE,UOLF,TPB,TIE,TIF,TEDK,
  ITFL,TGRS,TGRG,TGRD,HTBHD,BTBHD,TTBT,TTBE,HL3HD,BL3HD,TLBT,TLBE
  COMMON /MSHIP/ HLED,HLFD,HOLD,BTTO,DB4D,BLMD,ONSFE,ONSFF,OPSF,
  +OPSGR,SFL(40),SGR(10),TITLE(18)
  COMMON /DATA/ E,GG,XIN(200),YIN(200),YSM(200),A(200),QEL(200),
  +BIB(200),TIR(200),TBP(200),HG(200),TG(200)
  COMMON /SJOFX/ SJOX(4,77),SJOX(8,77),STEP(10)
  IF(IDPT.EQ.0) GO TO 9
  GO TO (1,2,3),IVAR

```



```

1 IF(ICNT.EQ.1) X(1)=OPSF
  OPSFL=ROUND(X(1),3.0)
  GO TO 2
2 DBH=ROUND(X(1),1.0)
  GO TO 3
3 IF(ICNT.EQ.1) X(1)=OPSF
  OPSFL=ROUND(X(1),3.0)
  DBH =ROUND(X(2),1.0)
9 CONTINUE
  CALL OPMIN(ICNT)
  IF(IPRT.GE.3) CALL DISFST(3,IW)
  IF(IPRT.GE.2) CALL DISFST(3,IW)
  VTOT=0.0
  VWFL=0.0
  VWFE=0.0
  VWFF=0.0
  NFL=NFE+NFF
  KE2=NSGR-1
  KKE=NFE*KE2
  KKF=KKE+KE2+1
  DO 10 I=1,NELT
  VWB=QEL(I)*A(I)
  VFL=QEL(I)*BIR(I)*(TIB(I)+TBP(I))
  VTOT=VTOT+VWB+VFL
10 CONTINUE
  F=VTOT/HQL
  IF(IPRT.GE.3) WRITE(IW,55)F,VTOT,HJL,OPSF,HLE,HLF
  DO 20 I=1,KKE
  VWE=QEL(I)*A(I)
  VWFE=VWFEA+VWE
20 CONTINUE
  FE=VWFE/FLJAT(NFE)
  DO 30 I=KKF,NELF
  VWF=QEL(I)*A(I)
  VWFF=VWFF+VWF
30 CONTINUE
  FF=VWFF/FLJAT(NFF)
  VWFL=VWFE+VWFF
  FL=VWFL/FLJAT(NFL)
  IF(ICNT.GT.77) GO TO 900
  SJCF(1,ICNT)=F
  SJCF(2,ICNT)=FL
  SJCF(3,ICNT)=FE
  SJCF(4,ICNT)=FF
  SJOX(1,ICNT)=OPSF
  SJOX(2,ICNT)=DBH
  SJOX(3,ICNT)=HLE
  SJOX(4,ICNT)=HLF
  SJOX(5,ICNT)=HLE/HLEO
  SJOX(6,ICNT)=HLF/HLFO
  SJOX(7,ICNT)=HLE/BTT
  SJOX(8,ICNT)=HLF/BTT
  RETURN
900 CONTINUE
  WRITE(IW,901) ICNT
  RETURN
55 FORMAT(/1X,†OBJ. F(CUB. CM)=†,F9.0,† VTOT=†,F11.0,† HQL=†,F7.1,
  † OPSFL,HLE,HLF=†,3F7.1, † SUB-FUNCTN†)
901 FORMAT(1X,†SUB-CONTRL STORING RETURNED AT ICNT=†,I3,†(.GT.77)†)
  END
SUBROUTINE CONSTR(N,NC,X,G)
  DIMENSION X(N),G(NC)
  COMMON /IOPT/ IR,IW,IOPT,IPRT,IVAR
  GO TO (10,20,30),IVAR

```

```

10 CONTINUE
   G(1)=X(1)-180.0
   G(2)=320.0-X(1)
   GO TO 100
20 CONTINUE
   G(1)=X(1)-150.0
   G(2)=200.0-X(1)
   GO TO 100
30 CONTINUE
   G(1)=X(1)-180.0
   G(2)=320.0-X(1)
   G(3)=X(2)-150.0
   G(4)=200.0-X(2)
100 CONTINUE
   RETURN
   END
   SUBROUTINE HOOKE(N,NC,FDPT,X,DELTA,STEP,EPSI,ICNT)
   COMMON /IDPT/ IR,IW,IDPT,IPRT,IVAR,IPNA(20)
   DIMENSION X(N),DELTA(N),STEP(N),EPSI(N),B1(20),B2(20),T(20)
   ICNT=0
   RK=1.0E05
   IF(IPRT.GE.2) WRITE(IW,200)
200 FORMAT(21X,† FUNCTION VALUES†,10X,† INDEPENDENT VARIABLES†)
   DO 10 I=1,N
   B1(I)=X(I)
   10 CONTINUE
   IF(IPRT.GE.2) WRITE(IW,240)
240 FORMAT(/† ***INITIAL POINT***†)
   20 CALL CONTRL(N,NC,B1,FB1,RK,ICNT,IRESET)
   IF(IPRT.GE.2) WRITE(IW,210) FB1,(B1(I),I=1,N)
210 FORMAT(1X,E15.6,5X,4E15.6/(21X,4E15.6))
   30 IF(IPRT.GE.2) WRITE(IW,250)
250 FORMAT(/† ***LOCAL SEARCH***†)
   CALL JEEVES(N,NC,B1,FB1,B2,FB2,DELTA,RK,ICNT,IRESET)
   IF(IRESET.EQ.1) GO TO 110
   IF(IPRT.GE.2) WRITE(IW,210) FB2,(B2(I),I=1,N)
   IF(ABS(FB1-FB2).GT.5.0E-8) GO TO 60
   40 INDEX=0
   DO 50 I=1,N
   IF(DELTA(I).LE.EPSI(I)) GO TO 50
   INDEX=INDEX+1
   DELTA(I)=DELTA(I)-STEP(I)
   50 CONTINUE
   DO 55 I=1,NC
   55 IF(IPNA(I).GT.15) GO TO 118
   IF(ICNT.GE.50) GO TO 119
   IF(INDEX.EQ.0) GO TO 120
   IF(IPRT.GE.2) WRITE(IW,220) (DELTA(I),I=1,N)
220 FORMAT(/† *** SHRINK DELTA(N) ** †,(F14.7))
   GO TO 30
   60 DO 70 I=1,N
   T(I)=2.0*B2(I)-B1(I)
   70 CONTINUE
   CALL CONTRL(N,NC,T,FT,RK,ICNT,IRESET)
   IF(IRESET.EQ.1) GO TO 90
   IF((FT/FB2).GT.0.995) GO TO 80
   IF(IPRT.GE.2) WRITE(IW,260)
260 FORMAT(/† ***GLOBAL MOVE***†)
   CALL JEEVES(N,NC,T,FT,B2,FB2,DELTA,RK,ICNT,IRESET)
   IF(IRESET.EQ.1) GO TO 90
   IF(IPRT.GE.2) WRITE(IW,210) FB2,(B2(I),I=1,N)
   GO TO 60
   80 DO 85 I=1,N
   B1(I)=B2(I)

```

```

85 CONTINUE
F31=F32
GO TO 30
90 DO 100 I=1,N
B1(I)=T(I)
100 CONTINUE
110 IF(IPRT.GE.2) WRITE(IW,230) RK
230 FORMAT(/↑ ***RESET RK=↑,F10.2,↑ **↑)
GO TO 20
118 WRITE(IW,1233) (IPNA(I),I=1,NC)
1233 FORMAT(↑ ≤ SUB-HOOKE ADDITIONAL STOP AT IFNA(NC)=↑,10 (I3))
119 WRITE(IW,1234) ICNT
1234 FORMAT(↑ ≤ SUB-HOOKE ADDITIONAL STOP AT ICNT=↑,I3/125(1H≤))
120 DO 130 I=1,N
X(I)=B1(I)
130 CONTINUE
FOPT=F31
RETURN
END
SUBROUTINE JEEVES(N,NC,P1,FP1,P2,FP2,DELTA,RK,ICNT,IRESET)
DIMENSION P1(N),P2(N),DELTA(N),M(20)
IF(ICNT.GT.1) GO TO 20
DO 10 I=1,N
M(I)=-1
10 CONTINUE
20 DO 30 I=1,N
P2(I)=P1(I)
30 CONTINUE
SAVE=FP1
DO 60 I=1,N
P2(I)=P2(I)+M(I)*DELTA(I)
CALL CNTRL(N,NC,P2,FP2,RK,ICNT,IRESET)
IF(IRESET.EQ.1) RETURN
IF(FP2.LT.SAVE) GO TO 50
P2(I)=P2(I)-2.*M(I)*DELTA(I)
CALL CNTRL(N,NC,P2,FP2,RK,ICNT,IRESET)
IF(IRESET.EQ.1) RETURN
IF(FP2.LT.SAVE) GO TO 40
P2(I)=P2(I)+M(I)*DELTA(I)
GO TO 60
40 M(I)=-M(I)
50 SAVE=FP2
60 CONTINUE
FP2=SAVE
RETURN
END
FUNCTION ROUND(QB,TERM)
B=QB/TERM
BB=AIN(T(B))
R=B-BB
IF(R.LT.0.5) QB=BB*TERM
IF(R.GE.0.5) QB=(BB+1.0)*TERM
ROUND=QB
RETURN
END

```