

R
150
239

YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Matris Met. ve Uzay Kaf. Sist.
Mat. Dep. Met. Yan. Çöz.

YÜKSEK LİSANS TEZİ

Ahmet Büyükpoyraz

1992

Ref
İNŞ
239
1992

YILDIZ UNİVERSİTESİ
FEN BİLİMLERİ ENSTİTUSU

MATRİS METODLARI VE UZAY KAFES SİSTEMLERİNİN

MATRİS - DEPLASMAN METODU YARDIMIYLA

ÇÖZÜMÜ

YÜKSEK LİSANS TEZİ

Yöneten: TÜRKAN KÖKSAL

Hazırlayan: AHMET BÜYÜKPOYRAZ

İ S T A N B U L

1 9 9 2

YILDIZ TEKNİK ÜNİVERSİTESİ
KÜTÜPHANE DOKÜMANTASYON
DAİRE BAŞKANLIĞI

Kot : R 150
: 250
Alındığı Yer : Y.T.Ü
Tarih : 29.8.1995
Fatura : -
Fiyatı : 25.000 TL.
Ayniyat No : 1-16
Kayıt No : 51550
UDC :
Ek :

Y. T. Ü.
KÜTÜPHANE DOK. DAİRE BAŞKANLIĞI

YILDIZ ÜNİVERSİTESİ

ABSTRACT

FENBİLİMLERİ ENSTİTÜSÜ

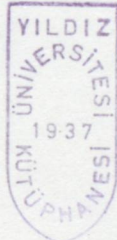
The calculation of the structure systems are somewhat long and tired, for this reason simplifying acceptances have been developed. In the last decade, computers have been used to solve engineering problems. As a result, the numerical calculations can be done very easily. And this causes the development of systematic matrix methods. By means of matrix methods, calculation of big and different systems can be done in a short time. On the other hand, by decreasing the simplifying acceptances and adding more factors to the calculation, the results which are closer to the

MATRİS METODLARI VE UZAY KAFES SİSTEMLERİN

MATRİS - DEPLASMAN METODU YARDIMIYLA

ÇÖZÜMÜ

The present method is based on the use of matrix methods. For the purpose of this study, a program directed to the market is developed and to be used to solve this type of systems. The developed program will be tested in several examples and the results will be discussed. In addition, comparisons will be made with the solutions achieved by other methods based on reference.



YÜKSEK LİSANS TEZİ

Yöneten: TÜRKAN KÖKSAL

Hazırlayan: AHMET BÜYÜKPOYRAZ

İ S T A N B U L

1 9 9 2

ABSTRACT

The calculation of the structure systems are somewhat long and tired, for this reason simplifying acceptances have been developed. In the last decade, computers have been used to solve engineering problems. As a result, the numerical calculations can be done very easily. And this causes the development of systematic matrix methods. By means of matrix methods, calculation of big and different systems can be done in a short time. On the other hand, by decreasing the simplifying acceptances and adding more factors to the calculation, the results which are closer to the reality can be achieved.

In this study, matrix methods are briefly explained and Matrix-Force and Matrix-Displacement methods are studied. A computer program which is based on Matrix-Displacements method will be developed. For the reason of 3-Dimensional truss systems are frequently used today, a program directed to the market is considered and to be used to solve this type of systems.

The developed program will be tested in several examples and the results will be discussed. In addition comparison will be made with the solutions achieved by SAP80 program taken as reference.

ÖZET

Yapı sistemlerinin hesabı çok kere uzun ve yorucu olduğundan, yıllar boyunca bazı basitleştirici kabuller yapılarak yaklaşık metodlar geliştirilmiştir. Son on-onbeş yıl içinde Mühendislik problemlerinin çözümünde bilgisayarların kullanılmaya başlaması sonucunda sayısal hesapların çok kolay yapılabilmesi olanaklarının ortaya çıkması sistematik matris metodlarının gelişmesine neden olmuştur. Matris metodları sayesinde büyük ve değişik sistemlerin kısa bir süre içinde hesabı mümkün olmaktadır. Öteyandan, idealleştirme sırasında yapılan basitleştirici kabuller azaltılarak ve daha fazla etken hesaba katılarak gerçeğe daha yakın sonuçlar elde edilebilmektedir.

Bu çalışmada matris metodları kısaca anlatılacak ve Matris-kuvvet, Matris-deplasman metodları incelenecektir. Matris-deplasman metodunu esas alan bir bilgisayar programı geliştirilecektir. Uzay kafes sistemlerin günümüzde sık sık kullanılması nedeniyle, bu tür sistemlerin çözümünde kullanılmak üzere, piyasaya yönelik bir program esas alınmıştır.

Geliştirilen program çeşitli örneklerde denenecek ve sonuçlar irdelenecektir. Ayrıca referans olarak alınan SAP80 programı ile yapılan çözümler arasında karşılaştırma yapılacaktır.

İ Ç İ N D E K İ L E R

1 -	İNGİLİZCE ÖZET	1
2 -	TÜRKÇE ÖZET	1
3 -	GİRİŞ	1
4 -	MATRİS METODLARINDA KULLANILAN TEMEL BAĞINTILAR VE TANIMLAR	2
5 -	MATRİSMETODLARI İLE HESAPTA İZLENECEK YOL	4
6 -	MATRİS - KUVVET METODU YAPI SİSTEMLERİNDE FLEKSİBİLİTE VE RİJİTLİK MATRİSLERİ ELMANDA f_p MATRİSİNİN BULUNMASI MATRİS - KUVVET METODU İLE ÇÖZÜM ÖRNEK ELEMEN ÜZERİNDE YÜK OLMASI HALİ	5 9 10 11 13 14
7 -	MATRİS - DEPLASMAN METODU TEMEL BAĞINTILAR ÖRNEK	16 18 20
8 -	UZAY KAFES SİSTEMLER	22
9 -	UZAY KAFES SİSTEMLERİN MATRİS - DEPLASMAN METODU İLE HESABI ÖRNEK	23 24
10 -	BİLGİSAYAR PROGRAMI	25
11 -	ÖRNEK ÇÖZÜMLER	36
12 -	SAP80 PROGRAMI İLE YAPILAN ÇÖZÜMLER	44
13 -	SONUÇLARIN KARŞILAŞTIRILMASI	59
14 -	KAYNAKLAR	61

GİRİŞ : Matris metodları diye adlandırdığımız sistematik hesap yöntemi ilk olarak 60' lı yılların ortalarında NASA tarafından ortaya atılmıştır. 1970' li yıllara gelindiğinde Matris metodları tam anlamıyla geliştirilmiş ve her alanda kullanılmaya başlanmıştır. 70' li yılların ortalarında bir üst aşama olan sonlu elemanlar metodu ortaya çıkmış ve en kullanışlı hesap yöntemleri arasına girmiştir. Bu iki metodu birbirinden tamamen ayırmak yanlış olur. Matris metodları sonlu elemanlar metoduna geçiş için bir zemin hazırlamıştır

Matris metodları üç ana metod olarak ele alınabilir

-- MATRİS - KUVVET METODU

-- MATRİS - DEPLASMAN METODU

-- GEÇİŞ MATRİSLERİ METODU

Matris metodlarının inşaat mühendisliğinde kullanım alanları oldukça geniştir. Büyük sistemlerin çözümünde hızlı ve pratik oluşu metodun kullanımını yaygınlaştırmaktadır.

Yapı sistemlerindeki çeşitli deformasyonlar (mesnet çökmeleri ve ısı değişimleri) kolaylıkla hesaplara dahil edilebilir. Yapılan çalışmalarla, eğri eksenli çubukların hesapları, çubuk sistemlerin serbest titreşimleri, dinamik hesaplar, elastik zemine oturan kirişlerin hesapları için Matris metodları geliştirilmiştir. Enyaygın şekilde çok sayıda çubuk elemana sahip çubuk sistemlerin hesabında kullanılmaktadır.

İleride de görüleceği gibi Matris metodlarının hesapları tamamen matris işlemlerine dayanmaktadır. Bu tür hesapların el ile yapılması oldukça zordur ve çoğu kere imkansızdır. Burada devreye bilgisayarlar girmektedir. Bir takım sistematik işlemleri gayet kolay hatasız ve hızlı bir şekilde yapan bilgisayarlar matris metodlarının pratik kullanımına imkan vermişlerdir. Günümüzde piyasada kullanılan statik programlarının temel çözüm yöntemi matris metodlarıdır.

Burada ortaya çıkan ilk problem, bilgisayarların hafıza problemi. Bilgisayarların yeni geliştirdiği sıralarda oldukça büyük problemler vardı. Eski bilgisayarlar hafıza olarak günümüze oranla oldukça yetersiz bir durumda idiler. Bu açığı kapatmak için programcılar çeşitli yöntemlere baş vurmuşlardır. Örneğin denklem takımları çözümü oldukça büyük bir problem idi. Çeşitli bilim adamları çeşitli yöntemler geliştirerek sınırlı olan hafızayı enaz şekilde kullanan programlar yazmışlardır. (Cholesky, Jordan, v.b.)

Halen günümüzde bu programlama teknikleri kullanılmaktadır. gelişen hafıza kapasitesi bu yöntemlerle daha büyük boyutlarda matris işlemlerine imkan tanımaktadır.

Matris metodları daha önce de belirtildiği gibi üç ana grupta ele alınabilir :

MATRİS - KUVVET METODU : Yapı statığında hiperstatik sistemlerin çözümü için kullanılan kuvvet metodunun matris formudur.

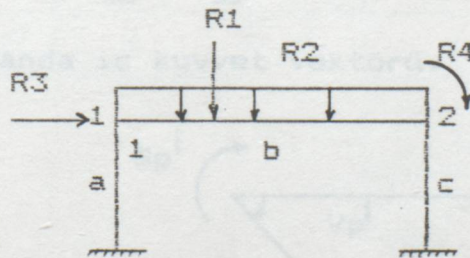
MATRİS - DEPLASMAN METODU : Yapı statığında hiperstatik sistemlerin çözümünde kullanılan açı metoduna benzer bir metoddur.

GEÇİŞ MATRİSLERİ METODU : Elastik mesnetleri olan sistemlerin, elastik zemine oturan kirişlerin hesabına pratik çözümler getiren bir metoddur.

MATRİS METODLARINDA KULLANILAN TEMEL BAĞINTILAR

VE TANIMLAR

- a -> Taşıyıcı sistem olarak doğrusal çubuk sistemler incelenecektir
- b -> Taşıyıcı sistem belli sayıdaki çubuk elemanlarının, belli sayıdaki düğüm noktalarında birleşmesinden meydana gelmiştir.
- c -> Düğüm noktaları, taşıyıcı çubuk elemanlarının birleşim noktalarıdır. Düzlemsel sistemlerde bütün düğüm noktaları aynı düzlem içindedir.
- d -> Etkiler : Düğüm noktalarına etkiyen tekil yükler elemana etkiyen yükler.
(Tekil yük, Yayılı yük, Isı değişimi)
Düğüm noktaları yer değiştirmesi.



- 1,2 Düğüm nok.
- a,b,c Elemanlar
- R1,R2 Elemana etkiyen yükler
- R3,R4 Düğüm noktasına etkiyen yükler

Sekil....1

e -> Kabuller :

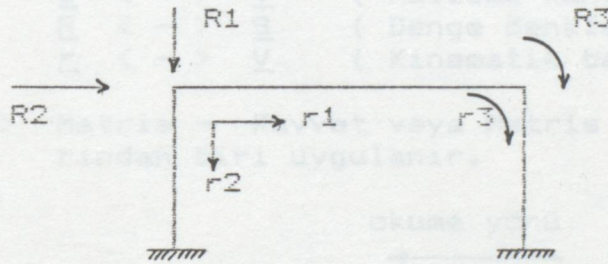
- > 1. Mertebe teorisi geçerlidir.
- > Malzeme lineer elastiktir.
- > Çubuk eksenleri doğrusaldır.
- > Sabit kesitli çubuklardır.
- > Yükler statiktir.

f -> Dış yükler ve yer değiştirmeler.

$$\underline{R} = \begin{bmatrix} R_1 \\ R_2 \\ \cdot \\ \cdot \\ R_n \end{bmatrix} ; \quad \underline{r} = \begin{bmatrix} r_1 \\ r_2 \\ \cdot \\ \cdot \\ r_n \end{bmatrix}$$

n Adet düğüm noktası
yükü

n Adet düğüm noktasına
karşı gelen \underline{r} vektörü

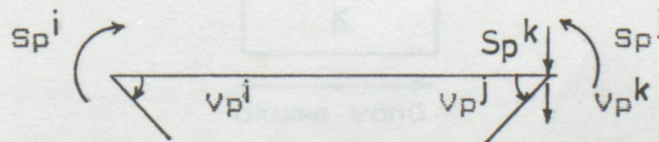


Şekil....2

f -> iç kuvvetler ve şekil değiştirmeler

$$\underline{S}_p = \begin{bmatrix} S_p^i \\ S_p^j \\ S_p^k \end{bmatrix} \quad \underline{v}_p = \begin{bmatrix} v_p^i \\ v_p^j \\ v_p^k \end{bmatrix}$$

Elemanda iç kuvvet vektörü. Elemanda şekil değiştirme vektörü.

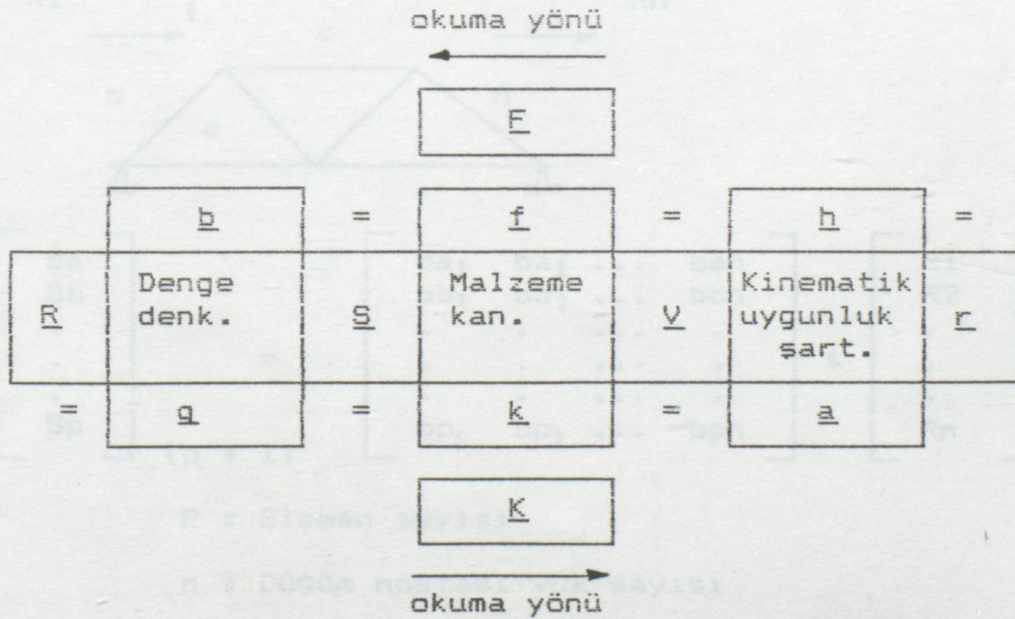


$$\underline{S} = \begin{bmatrix} S_a \\ S_b \\ \vdots \\ S_p \end{bmatrix} \quad \underline{V} = \begin{bmatrix} V_a \\ V_b \\ \vdots \\ V_p \end{bmatrix}$$

Toplam sistemde

MATRİS METODLARI İLE HESAPTA İZLENECEK YOL

- a -> Taşıyıcı sistem idealize edilerek statik sistem elde edilir.
- b -> Her elemanda $\underline{S}_p < - > \underline{V}_p$ arasındaki bağıntı belirlenir.
- c -> Elemanda iç kuvvet ve şekildeğiştirme vektörleri birer alt vektör olarak alınır ve toplam sistemde ait \underline{S} ve \underline{V} teşkil edilir. \underline{S} ile şekildeğiştirme arasındaki bağıntı belirlenir.
- d -> $\underline{R} < - > \underline{r}$
 $\underline{S} < - > \underline{V}$ (Malzeme kanunları)
 $\underline{R} < - > \underline{S}$ (Denge denklemleri)
 $\underline{r} < - > \underline{V}$ (Kinematik bağıntılar)
- e -> Matris - Kuvvet veya Matris - Deplasman metodlarından biri uygulanır.



Sekil....3

f -> Transformasyon matrisleri :

- \underline{b} : Statik transformasyon matrisi
- \underline{a} : Kinematik transformasyon matrisi
- \underline{f} : Toplam sistemin fleksibilite matrisi
- \underline{k} : Toplam sistemin rijitlik matrisi
- \underline{F} : Sistem fleksibilite matrisi
- \underline{K} : Sistem rijitlik matrisi

Matris - Deplasman metodu

$$\begin{aligned} \underline{R} &= \underline{q} * \underline{S} \\ \underline{S} &= \underline{k} * \underline{V} \\ \underline{V} &= \underline{a} * \underline{r} \\ \underline{R} &= \underline{K} * \underline{r} \end{aligned}$$

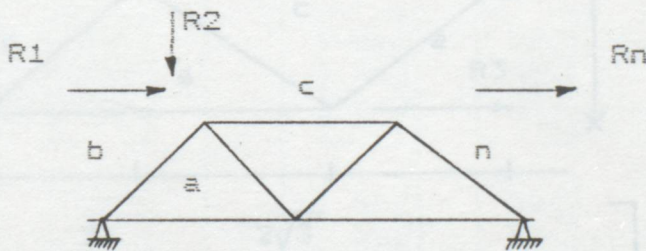
Matris - Kuvvet metodu

$$\begin{aligned} \underline{S} &= \underline{b} * \underline{R} \\ \underline{V} &= \underline{f} * \underline{S} \\ \underline{r} &= \underline{h} * \underline{V} \\ \underline{r} &= \underline{F} * \underline{R} \end{aligned}$$

MATRİS - KUVVET METODU

a -> \underline{b} Statik transformasyon matrisinin bulunması

$$\begin{aligned} S_a &= b_{a1} * R_1 + b_{a2} * R_2 + \dots + b_{an} * R_n \\ S_b &= b_{b1} * R_1 + b_{b2} * R_2 + \dots + b_{bn} * R_n \\ &\vdots \\ &\vdots \\ S_p &= b_{p1} * R_1 + b_{p2} * R_2 + \dots + b_{pn} * R_n \end{aligned}$$



$$\begin{bmatrix} S_a \\ S_b \\ \vdots \\ S_p \end{bmatrix} (p * 1) = \begin{bmatrix} b_{a1} & b_{a2} & \dots & b_{an} \\ b_{b1} & b_{b2} & \dots & b_{bn} \\ \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \dots & \vdots \\ b_{p1} & b_{p2} & \dots & b_{pn} \end{bmatrix} * \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_n \end{bmatrix}$$

P : Eleman sayısı

n : Düğüm noktası yük sayısı

$$\underline{S} = \underline{b} * \underline{r}$$

Aynı şekilde.

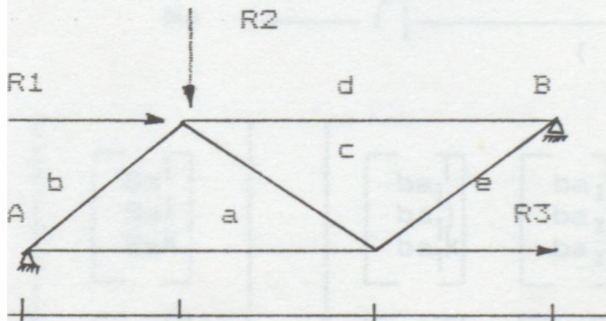
$$\begin{aligned} R_1 &= g_{1a} * S_a + g_{1b} * S_b + \dots + g_{1p} * S_p \\ R_2 &= g_{2a} * S_a + g_{2b} * S_b + \dots + g_{2p} * S_p \\ &\vdots \\ &\vdots \\ R_n &= g_{na} * S_a + g_{nb} * S_b + \dots + g_{np} * S_p \end{aligned}$$

$$\begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_n \end{bmatrix} (n * 1) = \begin{bmatrix} g_{1a} & g_{1b} & \dots & g_{1p} \\ g_{2a} & g_{2b} & \dots & g_{2p} \\ \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \dots & \vdots \\ g_{na} & g_{nb} & \dots & g_{np} \end{bmatrix} * \begin{bmatrix} S_a \\ S_b \\ \vdots \\ S_p \end{bmatrix}$$

$$\underline{R} = \underline{g} * \underline{S} \quad \text{Olduğu görülür}$$

b Matrisinin inversi her zaman g matrisi değildir.

ÖRNEK 1 :



$$\begin{aligned} H_A &= -\frac{R_1}{\sqrt{3}} - R_3 \\ V_A &= -\frac{2}{3}R_1 + \frac{1}{3}R_2 \\ V_B &= \frac{\sqrt{3}}{3}R_1 + \frac{1}{3}R_2 \end{aligned}$$

$$\begin{aligned} S_a &= \frac{2}{3}R_1 + \frac{2\sqrt{3}}{9}R_2 + R_3 \\ S_b &= \frac{2}{3}R_1 - \frac{4\sqrt{3}}{9}R_2 \\ S_c &= \frac{2}{3}R_1 - \frac{2\sqrt{3}}{9}R_2 \\ S_d &= -\frac{1}{3}R_1 - \frac{\sqrt{3}}{9}R_2 \\ S_e &= \frac{2}{3}R_1 + \frac{2\sqrt{3}}{9}R_2 \end{aligned}$$

Ritter kesim ile

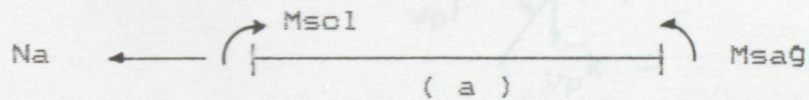
$$\begin{bmatrix} S_a \\ S_b \\ S_c \\ S_d \\ S_e \end{bmatrix}_{(5 \times 1)} = \begin{bmatrix} 2/3 & 1\sqrt{3}/9 & 1 \\ 2/3 & -4\sqrt{3}/9 & 0 \\ -2/3 & -2\sqrt{3}/9 & 0 \\ -1/3 & -\sqrt{3}/9 & 0 \\ 2/3 & 2\sqrt{3}/9 & 0 \end{bmatrix}_{(5 \times 3)} * \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}_{(3 \times 1)}$$

$$\underline{S} = \underline{q} * \underline{R}$$

$$\begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix} = \begin{bmatrix} 0 & 1/2 & -1/2 & -1 & 0 \\ 0 & -\sqrt{3}/2 & -\sqrt{3}/2 & 0 & 0 \\ 1 & 0 & 1/2 & 0 & -1/2 \end{bmatrix} * \begin{bmatrix} S_a \\ S_b \\ S_c \\ S_d \\ S_e \end{bmatrix}$$

$$\underline{R} = \underline{q} * \underline{S}$$

En genel halde $S_a = N_a$; $S_a = M_{sol}$; $S_a = M_{sa\theta}$



$$\begin{bmatrix} \begin{bmatrix} S_a^i \\ S_a^j \\ S_a^k \end{bmatrix} \\ \begin{bmatrix} S_b^i \\ S_b^j \\ S_b^k \end{bmatrix} \\ \begin{bmatrix} S_p^i \\ S_p^j \\ S_p^k \end{bmatrix} \end{bmatrix}_{(p \times 1)} = \begin{bmatrix} \begin{bmatrix} b_{a1}^i \\ b_{a1}^j \\ b_{a1}^k \end{bmatrix} & \begin{bmatrix} b_{a2}^i \\ b_{a2}^j \\ b_{a2}^k \end{bmatrix} & \dots & \begin{bmatrix} b_{an}^i \\ b_{an}^j \\ b_{an}^k \end{bmatrix} \\ \begin{bmatrix} b_{b1}^i \\ b_{b1}^j \\ b_{b1}^k \end{bmatrix} & \begin{bmatrix} b_{b2}^i \\ b_{b2}^j \\ b_{b2}^k \end{bmatrix} & \dots & \begin{bmatrix} b_{bn}^i \\ b_{bn}^j \\ b_{bn}^k \end{bmatrix} \\ \begin{bmatrix} b_{p1}^i \\ b_{p1}^j \\ b_{p1}^k \end{bmatrix} & \begin{bmatrix} b_{p2}^i \\ b_{p2}^j \\ b_{p2}^k \end{bmatrix} & \dots & \begin{bmatrix} b_{pn}^i \\ b_{pn}^j \\ b_{pn}^k \end{bmatrix} \end{bmatrix}_{(p \times n)} * \begin{bmatrix} R_1 \\ R_2 \\ \dots \\ R_n \end{bmatrix}_{(n \times 1)}$$

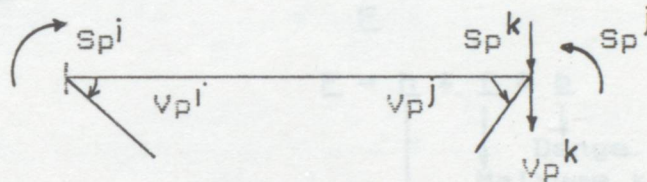
$p^x = p$ Eleman sayısı ; x Elemandaki iç kuvvet sayısı

$$\underline{b} = \begin{bmatrix} \underline{ba}_1 & \underline{ba}_2 \dots & \underline{ban} \\ \underline{bb}_1 & \underline{bb}_2 \dots & \underline{bbn} \\ \vdots & \vdots & \vdots \\ \underline{bp}_1 & \underline{bp}_2 \dots & \underline{bpn} \end{bmatrix} ; \quad \underline{S} = \begin{bmatrix} S_a \\ S_b \\ \vdots \\ S_p \end{bmatrix}$$

$$\underline{S}_a = \begin{bmatrix} S_{a_i} \\ S_{a_j} \\ S_{a_k} \end{bmatrix} ; \quad \underline{b}_{pn} = \begin{bmatrix} b_{pn}^i \\ b_{pn}^j \\ b_{pn}^k \end{bmatrix}$$

b -> f Fleksibilite matrisinin bulunması

f_p eleman fleksibilite matrisi
 $\underline{V} = \underline{f} * \underline{S}$ (Toplam sistemde)
 $\underline{V}_p = \underline{f}_p * \underline{S}_p$ (Elemanda)



$$V_p = f_p * S_p + f_p * S_p + f_p * S_p$$

$$V_p = f_p * S_p + f_p * S_p + f_p * S_p$$

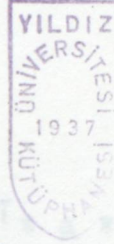
$$V_p = f_p * S_p + f_p * S_p + f_p * S_p$$

$$\underline{V}_p = \begin{bmatrix} V_{p_i} \\ V_{p_j} \\ V_{p_k} \end{bmatrix} = \begin{bmatrix} f_{p_{ii}} & f_{p_{ij}} & f_{p_{ik}} \\ f_{p_{ji}} & f_{p_{jj}} & f_{p_{jk}} \\ f_{p_{ki}} & f_{p_{kj}} & f_{p_{kk}} \end{bmatrix} * \begin{bmatrix} S_{p_i} \\ S_{p_j} \\ S_{p_k} \end{bmatrix}$$

$f_{p_{ik}} = \delta_{ik} = f_{p_{ki}}$ (Maxwell teoremi gereğince)
 Toplam sisteme adaptasyon b matrisindeki gibi yapılır.

$$\underline{V} = \begin{bmatrix} V_a \\ V_b \\ \vdots \\ V_p \end{bmatrix} = \begin{bmatrix} f_a & 0 & 0 & \dots & 0 \\ 0 & f_b & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & f_p \end{bmatrix} * \begin{bmatrix} S_a \\ S_b \\ \vdots \\ S_p \end{bmatrix}$$

Olarak elde edilir.



Şemadan,

$$\begin{aligned} \underline{V} &= \underline{f} * \underline{S} \\ \underline{S} &= \underline{k} * \underline{V} \end{aligned}$$

\underline{f} kare matris olduğundan $\underline{k} = \underline{f}^{-1}$ olduğu görülür.

Fleksibilite matrisinin özellikleri

\underline{f} kare matristir.

\underline{f} regülerdir.

Maxwel - Betti teoreminden dolayı \underline{f} simetriktir.

YAPI SİSTEMLERİNDE FLEKSİBİLİTE VE RİJİTLİK MATRİSLERİ

Şekil 3 ' den okunan değerlere göre :

$$\begin{aligned} \underline{S} &= \underline{b} * \underline{R} \\ \underline{V} &= \underline{f} * \underline{S} \\ \underline{r} &= \underline{h} * \underline{V} \end{aligned} \Rightarrow \underline{r} = \underline{h} * \underline{f} * \underline{b} * \underline{R} \Rightarrow \underline{r} = \underline{E} * \underline{R}$$

$$\underline{E} = \underline{h} * \underline{f} * \underline{b}$$

\downarrow Denge denklemleri
 \downarrow Malzeme kanunları
 \downarrow Kinematik uygunluk şartı

$$\begin{aligned} \underline{R} &= \underline{a} * \underline{S} \\ \underline{S} &= \underline{k} * \underline{V} \\ \underline{V} &= \underline{a} * \underline{r} \end{aligned} \Rightarrow \underline{R} = \underline{a} * \underline{k} * \underline{a} * \underline{r} \Rightarrow \underline{R} = \underline{K} * \underline{r}$$

$$\underline{K} = \underline{a} * \underline{k} * \underline{a}$$

\downarrow Denge denklemleri
 \downarrow Malzeme kanunları
 \downarrow Kinematik uygunluk şartı

$$\underline{K} * \underline{E} = \underline{I} \Rightarrow \underline{K} = \underline{E}^{-1} \text{ ve}$$

$$\underline{R} = \underline{K} * \underline{r}$$

$$\underline{r} = \underline{E} * \underline{R}$$

Olduğunu biliyoruz.

Virtüel iş denklemlerinden

$$A_{i\delta} = \sum_n R_n * r_n = \underline{R} * \underline{r} = \underline{r} * \underline{R} = \text{Skalerdir.}$$

$$A_{i\epsilon} = \sum_n S_n * V_n = - \underline{S} * \underline{V} = - \underline{V} * \underline{S}$$

ve $A_{i\epsilon} + A_{i\delta} = 0$ Olduğuna göre.

$$\Rightarrow \begin{matrix} \underline{R}^T * \underline{I} - \underline{S}^T * \underline{V} = 0 \\ \underline{R} = \underline{q} * \underline{S}, \underline{S}^T * \underline{q}^T * \underline{r} - \underline{S}^T * \underline{a} * \underline{r} = 0 \\ \underline{q}^T = \underline{a} \quad \text{ve} \quad \underline{q} = \underline{a}^T \end{matrix}$$

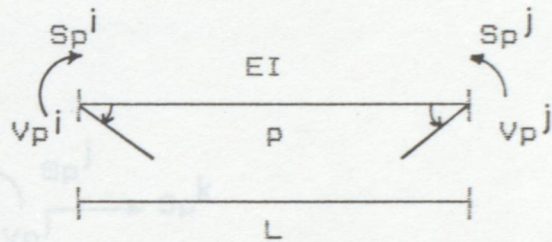
$$\Rightarrow \begin{matrix} \underline{r}^T * \underline{R} - \underline{V}^T * \underline{S} = 0 \\ \underline{r} = \underline{h} * \underline{V}, \underline{V}^T * \underline{h} * \underline{R} - \underline{V}^T * \underline{b} * \underline{R} = 0 \\ \underline{h}^T = \underline{b} \quad \text{ve} \quad \underline{h} = \underline{b}^T \end{matrix}$$

Olduğu görülür.

ELEMANDA f_p MATRİSİNİN BULUNMASI

a -> Kiriş elemanında.

$$\underline{V}_p = \underline{f}_p * \underline{S}_p$$



Mohr'a göre. $v_{p^i} = \frac{S_{p^i}}{EI} * \frac{L}{2} * \frac{2}{3} + \frac{S_{p^j}}{EI} * \frac{L}{2} * \frac{1}{3}$

$$v_{p^j} = \frac{S_{p^i}}{EI} * \frac{L}{2} * \frac{2}{3} + \frac{S_{p^j}}{EI} * \frac{L}{2} * \frac{2}{3}$$

$$v_{p^i} = \frac{L}{EI} * \left(\frac{1}{3} * S_{p^i} + \frac{1}{6} * S_{p^j} \right)$$

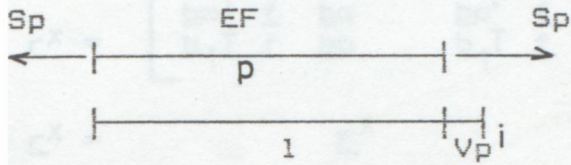
$$v_{p^j} = \frac{L}{EI} * \left(\frac{1}{6} * S_{p^i} + \frac{1}{3} * S_{p^j} \right)$$

Sonuç olarak

$$\begin{bmatrix} v_{p^i} \\ v_{p^j} \end{bmatrix} = \frac{L}{6EI} * \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} * \begin{bmatrix} S_{p^i} \\ S_{p^j} \end{bmatrix} \quad \text{olur}$$

$$\underline{f}_p = \frac{L}{6EI} * \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad \text{Olarak elde edilir.}$$

b -> Kafes kiriş elemanında.

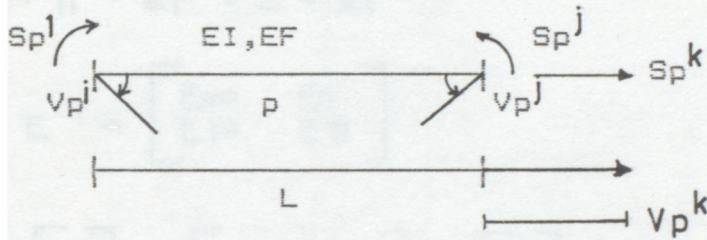


$$v_p^i = \frac{1}{EF} * S_p^i$$

(Hooke Kanunu)

$$f_p = \left[\frac{1}{EF} \right]$$

c -> Çerçeve elemanında



$$f_p = \frac{L}{6EI} * \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 6I/F \end{bmatrix} \text{ Olarak bulunur.}$$

MATRİS - KUVVET METODU İLE ÇÖZÜM

$$\underline{S} = \underline{b} * \underline{R}$$

$$\underline{r} = \underline{E} * \underline{R} \Rightarrow \underline{E} = \underline{b} * \underline{f} * \underline{b}$$

Denge şartı :

$$\underline{S} = \underline{b} * \underline{R} = \begin{bmatrix} b_0 & b_1 \end{bmatrix} * \begin{bmatrix} \underline{R} \\ \underline{X} \end{bmatrix}$$

Gecek dışyüklər
Hiperstatik bilinmeyenler.

$$\Rightarrow \underline{S} = \underline{b} * \underline{R} + b_1 * \underline{X}$$

Fleksibilitite Bağintısı :

$$\underline{r}^X = \underline{E}^X * \underline{R}^X = \underline{b}^{X^T} * \underline{f} * \underline{b}^X * \underline{R}^X$$

$$\underline{r}^X = \begin{bmatrix} b_0 & b_1 \end{bmatrix}^T * \underline{f} * \begin{bmatrix} b_0 & b_1 \end{bmatrix} * \begin{bmatrix} \underline{R} \\ \underline{X} \end{bmatrix}$$

$$\underline{r}^X = \begin{bmatrix} b_0^T & \underline{f} \\ b_1^T & \underline{f} \end{bmatrix} * \begin{bmatrix} b_0 & b_1 \end{bmatrix} * \begin{bmatrix} \underline{R} \\ \underline{X} \end{bmatrix}$$

$$\underline{r}^X = \begin{bmatrix} \underline{b}_0^T & f & \underline{b}_0 \\ \underline{b}_1^T & f & \underline{b}_1 \end{bmatrix} * \begin{bmatrix} R \\ X \end{bmatrix}$$

$$\underline{r}^X = \underline{E}^X * \underline{R}^X$$

$$F_{00} = \underline{b}_0^T * f * \underline{b}_0$$

$$F_{01} = \underline{b}_0^T * f * \underline{b}_1$$

$$F_{10} = F_{01}^T$$

$$F_{11} = \underline{b}_1^T * f * \underline{b}_1$$

$$\underline{E} = \begin{bmatrix} F_{00} & F_{01} \\ F_{10} & F_{11} \end{bmatrix}$$

$$\begin{bmatrix} I \\ r \end{bmatrix} = \begin{bmatrix} F_{00} & F_{01} \\ F_{10} & F_{11} \end{bmatrix} * \begin{bmatrix} R \\ X \end{bmatrix}$$

Elastisite Bağıntısı

$$\underline{\epsilon}_1 = 0$$

$$\underline{\epsilon}_1 = \underline{\epsilon}_{10} * R + \underline{\epsilon}_{11} * X = 0 \text{ (Geometrik uygunluk şartı)}$$

$$X = - \underline{\epsilon}_{11}^{-1} * \underline{\epsilon}_{10} * R$$

İç kuvvetlerin bulunması

$$\underline{s} = \underline{b}_0 * R + \underline{b}_1 * X \text{ idi.}$$

$$\underline{s} = \underline{b}_0 * R + \underline{b}_1 * \underline{\epsilon}_{11}^{-1} * \underline{\epsilon}_{10} * R$$

$$\underline{s} = (\underline{b}_0 - \underline{b}_1 * \underline{\epsilon}_{11}^{-1} * \underline{\epsilon}_{10}) * R \text{ ve } \underline{s} = \underline{b} * R$$

$$\Rightarrow \underline{b} = \underline{b}_0 - \underline{b}_1 * \underline{\epsilon}_{11}^{-1} * \underline{\epsilon}_{10}$$

Düğüm noktaları yerdeğiştirmeleri

$$\underline{r} = F_{00} * R + F_{01} * X$$

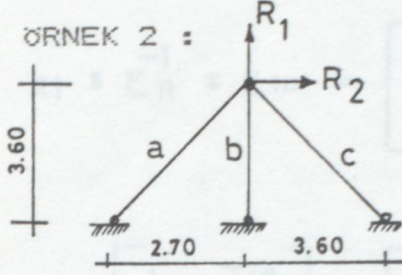
$$\underline{r} = F_{00} * R - F_{01} * \underline{\epsilon}_{11}^{-1} * \underline{\epsilon}_{10} * R$$

$$\underline{r} = (F_{00} - F_{01} * \underline{\epsilon}_{11}^{-1} * F_{10}) * R$$

ve $\underline{r} = \underline{E} * \underline{R}$

$\Rightarrow \underline{E} = \underline{E}_{00} - \underline{E}_{01} * \underline{E}_{11}^{-1} * \underline{E}_{10}$

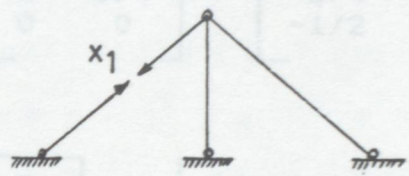
ÖRNEK 2 :



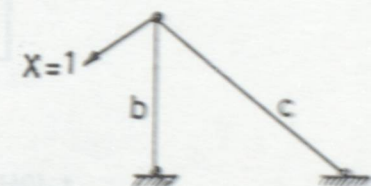
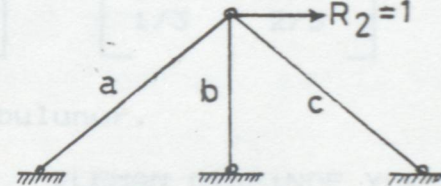
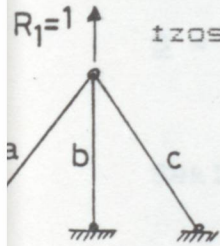
Verilenler: 1 EF

a	4.5	EF
b	3.6	EF
c	6.0	EF

Cözüm: izostatik esas sistem seçimi.



izostatik esas sistem ve hiperstatik bilinmeyen



$$\begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix} = \begin{bmatrix} 1 & 3/4 & -5/4 \\ 0 & -5/4 & 3/4 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} R_1 \\ R_2 \\ X_1 \end{bmatrix}$$

$$\underline{b}_0 = \begin{bmatrix} 1 & 3/4 \\ 0 & -5/4 \\ 0 & 0 \end{bmatrix} \quad \text{ve} \quad \underline{b}_1 = \begin{bmatrix} -5/4 \\ 3/4 \\ 1 \end{bmatrix}$$

$$\underline{S} = \underline{k}^X * \underline{R}^X = \left[\underline{b}_0 * \underline{b}_1 \right]$$

$$\underline{f} = \begin{bmatrix} \frac{f_b}{0} & 0 & 0 \\ 0 & \frac{f_c}{0} & 0 \\ 0 & 0 & \frac{f_a}{0} \end{bmatrix} = \frac{1}{EF} * \begin{bmatrix} 3.6 & 0 & 0 \\ 0 & 6.0 & 0 \\ 0 & 0 & 4.5 \end{bmatrix}$$

$$E_{11} = \underline{b}_1^T * \underline{f} * \underline{b}_1 = 27 / 2EF \quad ; \quad E_{11}^{-1} = (2/27) * EF$$

$$E_{10} = \underline{b}_1^T * \underline{f} * \underline{b}_0 = 1 / EF * \begin{bmatrix} -45/10 & -9 \end{bmatrix}$$

$$\underline{b}_1 * E_{11}^{-1} * E_{10} = \begin{bmatrix} 5/12 & 5/6 \\ -1/4 & -1/2 \\ -1/2 & -2/3 \end{bmatrix}$$

$$\underline{b} = \begin{bmatrix} 1 & 3/4 \\ 0 & -5/4 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 5/12 & 5/6 \\ -1/4 & -1/2 \\ -1/2 & -2/3 \end{bmatrix} = \begin{bmatrix} 7/12 & -1/12 \\ 1/4 & -3/4 \\ 1/3 & 2/3 \end{bmatrix}$$

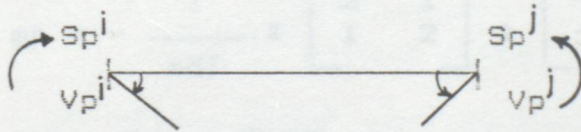
$$\underline{S} = \begin{bmatrix} S_a \\ S_c \\ S_a \end{bmatrix} = \begin{bmatrix} 7/12 & -1/12 \\ 1/4 & -3/4 \\ 1/3 & 2/3 \end{bmatrix} * \begin{bmatrix} R_1 \\ R_2 \end{bmatrix}$$

Şeklinde bulunur.

ELEMAN ÜZERİNDE YÜK OLMASI HALİ

- a -> Yayılı yük, tekil yük, moment.
- b -> Isı etkisi.
- c -> mesnet hareketleri.

Bir elemanı göz önüne alalım.



$$V_p = \underbrace{f_p * S_p}_{\text{uç iç kuvvetlerden}} + m_p \quad \text{eleman üzerindeki yükten}$$

izostatik sistemde : $V_p = f_p * S_p + m_p$

Toplam sistemde : $V = f * S + m$

$$\underline{S} = \underline{b} * \underline{R}$$

$$\underline{r} = \underline{b}^T * \underline{V} = \underline{b}^T * \underline{f} * \underline{b} * \underline{R} + \underline{b}^T * \underline{m}$$

$$\underline{r} = \underline{F} * \underline{R} + \underline{b}^T * \underline{m}$$

Hiperstatik sistemde : $\underline{V}_p = \underline{f}_p * \underline{S}_p + \underline{m}_p$

$$\underline{r}^X = \begin{bmatrix} \underline{r} \\ \underline{r} \end{bmatrix} = \begin{bmatrix} \underline{E}_{00} & \underline{E}_{01} \\ \underline{E}_{10} & \underline{E}_{11} \end{bmatrix} * \begin{bmatrix} \underline{R} \\ \underline{X} \end{bmatrix} + \begin{bmatrix} \underline{b}_0^T \\ \underline{b}_1^T \end{bmatrix} * \underline{m}$$

\underline{m} : Eleman üstü yük vektörü.

$$\underline{r}_1 = 0 \Rightarrow \underline{E}_{10} * \underline{R}_{11} + \underline{E}_{11} * \underline{X} + \underline{b}_1^T * \underline{m} = 0$$

$$\underline{X} = -\underline{E}_{11}^{-1} * (\underline{E}_{10} * \underline{R} + \underline{b}_1^T * \underline{m})$$

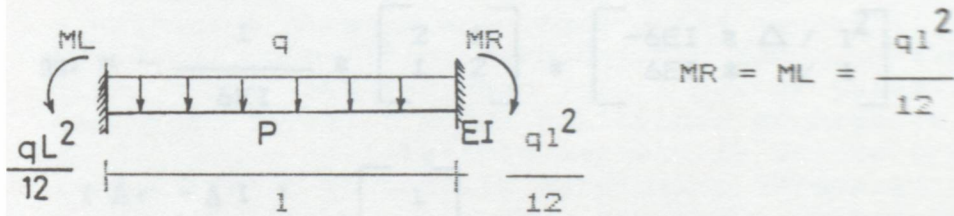
$$\underline{E} = \underline{E}_{00} - \underline{E}_{01} * \underline{E}_{11}^{-1} * \underline{E}_{10}$$

$$\underline{r} = \underline{E} * \underline{R} + \underline{b}^T * \underline{m} \Rightarrow \underline{b}^T = \underline{b}_0^T - \underline{E}_{10} * \underline{E}_{11}^{-1} * \underline{b}_1^T$$

$$\underline{S} = \underline{b} * \underline{R} \Rightarrow \underline{b} = \underline{b}_0 - \underline{b}_1 * \underline{E}_{11}^{-1} * \underline{E}_{10}$$

\underline{m} Matrisinin Hesabı

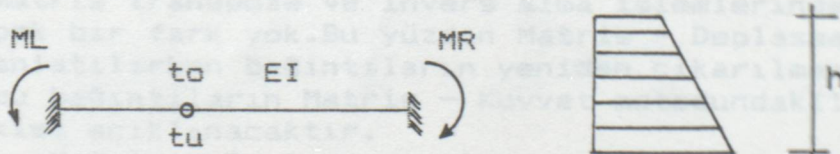
a -> Kiriş elemanında yayılı yük halinde.



$$\underline{m}_p = -\frac{1}{6EI} * \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} * \begin{bmatrix} -q l^2 / 12 \\ -q l^2 / 12 \end{bmatrix}$$

$$= \frac{q l^3}{24EI} * \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

b -> üniform olmayan ısı etkisi.



h : Kiriş yüksekliği
 t_o : Kiriş üst sıcaklığı
 t_u : Kiriş alt sıcaklığı

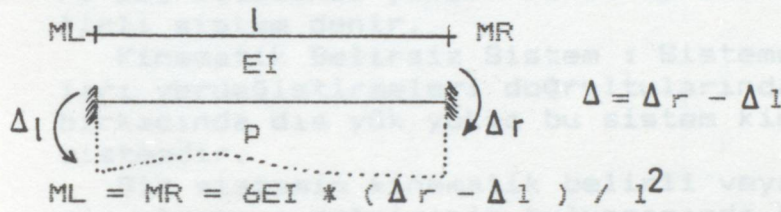
αt : Isı genleşme katsayısı
 $\Delta t = t_o - t_u$

$$M_L = M_R = EI * \alpha_T * \frac{\Delta_T}{h}$$

$$m_p = - \frac{1}{6EI} * \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} * \begin{bmatrix} -EI * \alpha_T * \Delta_T / h \\ -EI * \alpha_T * \Delta_T / h \end{bmatrix}$$

$$= \frac{\alpha_T * \Delta_T * 1}{h} * \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

c -> Mesnet çökmesi



$$m_p = - \frac{1}{6EI} * \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} * \begin{bmatrix} -6EI * \Delta / l^2 \\ 6EI * \Delta / l^2 \end{bmatrix}$$

$$m_p = \frac{(\Delta_r - \Delta_1)}{l} * \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

MATRİS DEPLASMAN METODU

Bu bölüme kadar genelde Matris - Kuvvet metodu ağırlıkta olmak koşulu ile matris metodlarının temel bağıntıları verildi. Geliştirilen bir takım matrisler ve bunlar arasındaki bağıntılar yazıldı. Genel olarak görüleceği üzere Matris - Kuvvet ve Deplasman metodları birbirine oldukça yakın, kullanılan bağıntılar arasında matris transpoze ve invers alma işlemlerinden başka pek bir fark yok. Bu yüzden Matris - Deplasman metodu anlatılırken bağıntıların yeniden çıkarılması yerine, bu bağıntıların Matris - Kuvvet metodundakilerle ilişkisi açıklanacaktır.

Matris - Deplasman metodunda kullanılan ve farklı olan matrisler, K rijitlik matrisi, a kinematik transformasyon matrisi, k_p eleman rijitlik matrisleridir. S , R , V , r , matrisleri Matris - Kuvvet metodunda-

kilerle aynı anlamları ifade etmektedirler.

\underline{K} Matrisi \underline{E} matrisinin inversidir. \underline{a} Matrisi kinematik yollarla hesaplanan bir matristir. Metodun temelini oluşturmaktadır. Aynı zamanda \underline{a} matrisi \underline{q} matrisinin tranzpozesidir. Daha önce verdiğimiz 3 numaralı şekil yardımıyla şu bağıntıları yazabiliriz.

$$\underline{r} = \underline{K}^{-1} * \underline{R} = (\underline{a}^T * \underline{k} * \underline{a})^{-1} * \underline{R}$$

$$\underline{S} = \underline{k} * \underline{a} * \underline{r} = \underline{k} * \underline{a} * \underline{K}^{-1} * \underline{R}$$

Matris - Deplasman metodunda sistemlerin kinematik belirli veya kinematik belirsiz olması oldukça önemli bir yer tutmaktadır.

Kinematik Belirli Sistem : Sistemdeki düğüm noktaları doğrultusunda yükler varsa bu sisteme kinematik belirli sistem denir.

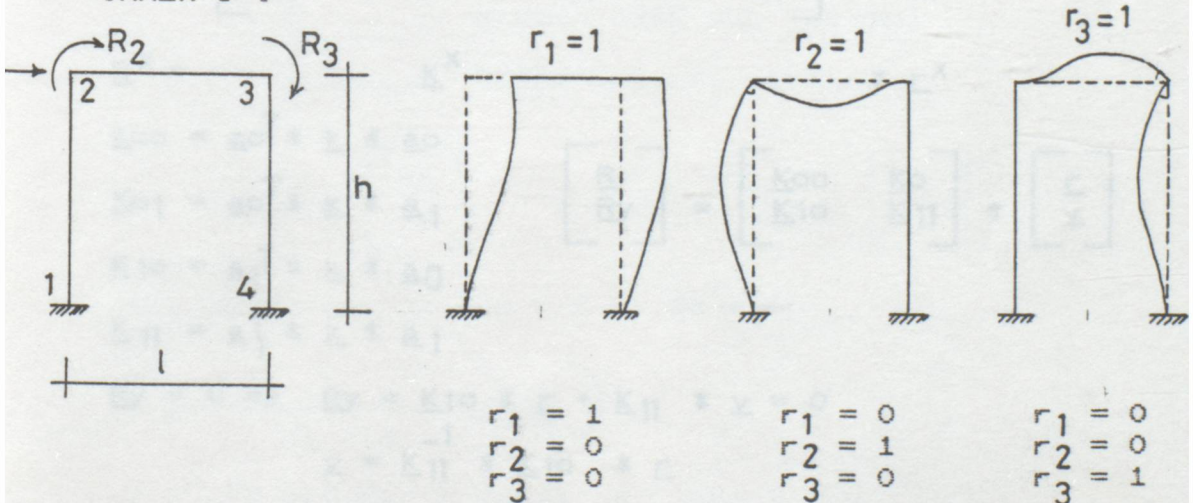
Kinematik Belirsiz Sistem : Sistemdeki düğüm noktaları yerdeğiştirmeleri doğrultularından bir veya birkaçında dış yük yoksa bu sistem kinematik belirsiz sistemdir.

Bir sistemin kinematik belirli veya kinematik belirsiz olması \underline{a} matrisinin bulunmasında önemlidir.

Kinematik belirli sistemde \underline{a} matrisinin bulunması: \underline{a} matrisini oluşturmadan önce nokta serbestliğinden bahsetmek gereklidir. Herhangi bir düğüm noktasının - diğer düğüm noktalarından farklı olarak yapabildiği deplasman sayısı o düğüm noktasının serbestlik derecesi dir. Düğüm noktalarının serbestlik derecelerinin toplamı sistemin serbestlik derecesini vermektedir.

\underline{a} Matrisini oluştururken her düğüm noktasına yapabildiği deplasmanların doğrultusunda birim deplasman verilir. Bunun sonucunda o düğüm noktasına bağlanan - elemanlardaki deplasmanlar hesaplanır.

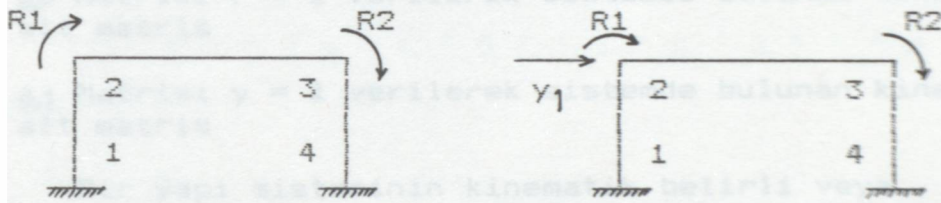
ÖRNEK 3 :



$$V = \begin{bmatrix} v_{12} \\ v_{21} \\ v_{23} \\ v_{32} \\ v_{34} \\ v_{43} \end{bmatrix} = \underbrace{\begin{bmatrix} -1/h & 0 & 0 \\ 1/h & -1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ -1/h & 0 & 1 \\ 1/h & 0 & 0 \end{bmatrix}}_{\underline{a}} * \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

Böylece \underline{a} matrisi oluşturulmuştur.

Kinematik Belirsiz Sistemlerde Matris-Deplasman Metodu



Q2 , Q3 , rx

Kinematik Belirsiz sistem Kinematik Bilinmeyen(y_1)

TEMEL BAĞINTILAR

$$\underline{Q}^X = k * \underline{V}^X$$

$$\underline{V}^X = \underline{a}^X * \underline{r}^X = \begin{bmatrix} \underline{a}_0 & \underline{a}_1 \end{bmatrix} : \begin{bmatrix} r \\ y \end{bmatrix} = \underline{a}_0 * \underline{r} * \underline{a} * \underline{y}$$

$$\underline{R}^X = \begin{bmatrix} \underline{a}_0^T * k * \underline{a}_0 & \underline{a}_0 * k * \underline{x} \\ \underline{a}_1^T * k * \underline{a}_0 & \underline{a}_1 * k * \underline{a} \end{bmatrix} * \underline{r}^X$$

$$\underline{R}^X = \underline{K}^X * \underline{r}^X$$

$$K_{00} = \underline{a}_0^T * k * \underline{a}_0$$

$$K_{01} = \underline{a}_0^T * k * \underline{a}_1$$

$$K_{10} = \underline{a}_1^T * k * \underline{a}_0$$

$$K_{11} = \underline{a}_1^T * k * \underline{a}_1$$

$$\begin{bmatrix} R \\ R_y \end{bmatrix} = \begin{bmatrix} K_{00} & K_{01} \\ K_{10} & K_{11} \end{bmatrix} * \begin{bmatrix} r \\ y \end{bmatrix}$$

$$R_y = 0 \Rightarrow R_y = K_{10} * r + K_{11} * y = 0$$

$$y = \frac{-1}{K_{11}} * K_{10} * r$$

$$\underline{R} = \underline{K}_{00} * \underline{r} + \underline{K}_{01} * \underline{y} = \underline{K}_{00} * \underline{r} * \underline{K}_{01}^{-1} * \underline{K}_{11}^{-1} * \underline{K}_{10} * \underline{r}$$

$$\underline{R} = (\underline{K}_{00} - \underline{K}_{01} * \underline{K}_{11}^{-1} * \underline{K}_{10}) * \underline{r}$$

$$\underline{K} = (\underline{K}_{00} - \underline{K}_{01} * \underline{K}_{11}^{-1} * \underline{K}_{10})$$

$$\underline{V}^X = (\underline{a}_0 * \underline{r} - \underline{a}_1 * \underline{K}_{11}^{-1} * \underline{K}_{10}) * \underline{r}$$

$$\underline{V}^X = (\underline{a}_0 - \underline{a}_1 * \underline{K}_{11}^{-1} * \underline{K}_{10}) * \underline{r}$$

$$\underline{V} = \underline{a} * \underline{r} \quad \Rightarrow \quad \underline{a} = (\underline{a}_0 - \underline{a}_1 * \underline{K}_{11}^{-1} * \underline{K}_{10})$$

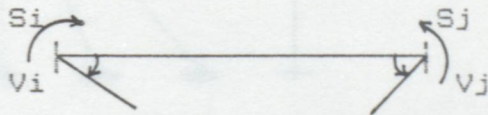
Bağıntıları bulunur.

\underline{a}_0 Matrisi $r = 1$ verilerek sistemde bulunan kinematik alt matris

\underline{a}_1 Matrisi $y = 1$ verilerek sistemde bulunan kinematik alt matris

Bir yapı sisteminin kinematik belirli veya kinematik belirsiz oluşuna bakılmaksızın her sistem kinematik belirli olarak çözülebilir. Durum böyle olunca kinematik bilinmeyen doğrultusundaki dış yük değerini sıfır almak gerekir.

a -> Eleman Üzerinde Yük Olması Durumu



$$\underline{S}_p = \underline{k}_p * \underline{V}_p + \underline{R}_p$$

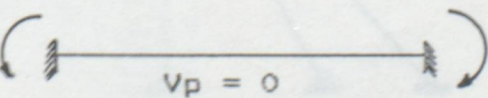
↓ Eleman üstü yük vektörü

→ Düğüm noktası yükleri

$$\underline{V}_p = \begin{bmatrix} v_i \\ v_j \end{bmatrix}$$

$$; \quad \underline{S}_p = \begin{bmatrix} s_i \\ s_j \end{bmatrix}$$

iki uç ankastre. $v_p = 0 \Rightarrow s_p = n_p$



$$\underline{S} = \underline{k} * \underline{V} + \underline{\Delta} \quad (\text{şekil 3 ' den })$$

$$\underline{R} = \underline{a}^T * \underline{S}$$

$$\underline{V} = \underline{a} * \underline{r} \Rightarrow \underline{R} = \underline{a}^T * \underline{k} * \underline{a} * \underline{r} + \underline{a}^T * \underline{n}$$

$$\underline{a}^T * \underline{k} * \underline{a} = \underline{K}$$

$$\underline{R} = \underline{K} * \underline{r} + \underline{a}^T * \underline{n}$$

$$\underline{r} = \underline{K}^{-1} * (\underline{R} - \underline{a}^T * \underline{n})$$

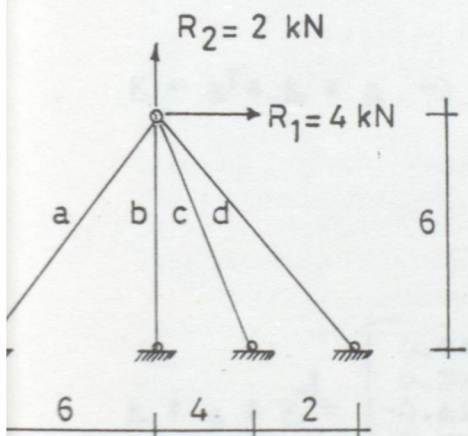
$$\underline{S} = \underline{k} * \underline{a} * \underline{r}$$

$$\underline{S} = \underline{k} * \underline{a} * \underline{r} + \underline{n}$$

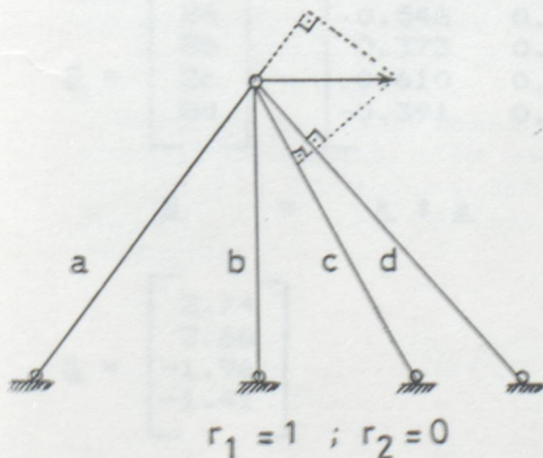
$$\underline{S} = \underline{k} * \underline{a} * \underline{K}^{-1} * \underline{R} - \underline{k} * \underline{a} * \underline{K}^{-1} * \underline{a}^T * \underline{n} + \underline{n}$$

$$\text{ve } \underline{S} = \underline{k} * \underline{a} * \underline{K}^{-1} * \underline{R} + (\underline{I} - \underline{k} * \underline{a} * \underline{K}^{-1} * \underline{a}^T) * \underline{n}$$

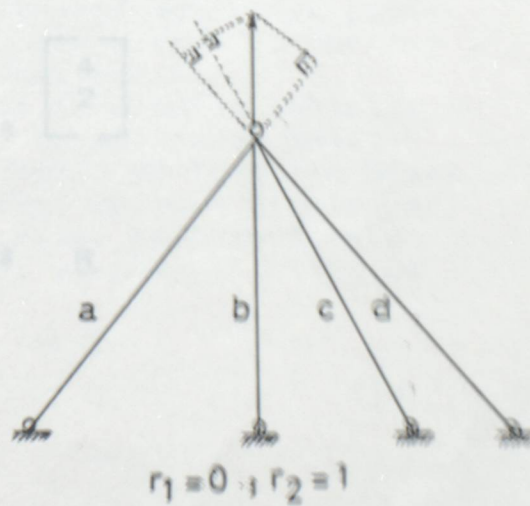
ÖRNEK 4 :



	a	b	c	d
Boy	3.49	6.00	7.21	8.49
Kesit	EI	EI	EI	EI



ve



$$\begin{bmatrix} V_a \\ V_b \\ V_c \\ V_d \end{bmatrix} = \begin{bmatrix} 0.707 & 0.707 \\ 0 & 1 \\ -0.555 & 0.831 \\ -0.707 & 0.707 \end{bmatrix} * \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} \quad k_p = \frac{EF}{1}$$

$$\Rightarrow \begin{aligned} k_a &= 0.1179 \\ k_b &= 0.3330 \\ k_c &= 0.2780 \\ k_d &= 0.1179 \end{aligned}$$

$$\underline{k} = \begin{bmatrix} 0.1179 & 0 & 0 & 0 \\ 0 & 0.3330 & 0 & 0 \\ 0 & 0 & 0.2780 & 0 \\ 0 & 0 & 0 & 0.1179 \end{bmatrix} * EF$$

$$\underline{K} = \underline{a}^T * \underline{k} * \underline{a} \Rightarrow \underline{K} = \begin{bmatrix} 0.203 & -0.128 \\ -0.128 & 0.643 \end{bmatrix} * EF$$

$$\underline{K}^{-1} = \begin{bmatrix} 5.64 & 1.12 \\ 1.12 & 1.79 \end{bmatrix} * \frac{1}{EF}$$

$$\underline{k} * \underline{a} * \underline{K}^{-1} = \begin{bmatrix} 0.546 & 0.244 \\ 0.373 & 0.594 \\ -0.610 & 0.239 \\ -0.391 & 0.056 \end{bmatrix}$$

$$\underline{S} = \begin{bmatrix} S_a \\ S_b \\ S_c \\ S_d \end{bmatrix} = \begin{bmatrix} 0.546 & 0.244 \\ 0.373 & 0.594 \\ -0.610 & 0.239 \\ -0.391 & 0.056 \end{bmatrix} * \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

$$\underline{S} = \underline{k} * \underline{a} * \underline{R}$$

$$\underline{S} = \begin{bmatrix} 2.74 \\ 2.68 \\ -1.96 \\ -1.41 \end{bmatrix}$$

UZAY KAFES SİSTEMLER

Günümüzde hayat standartlarının yükselmesi, kişilerin yaşadıkları ortamlarda birtakım özellikler aramalarına nedem olmuştur. Geniş mekanlar, orjinal mimari tarzları hep bunun sonucudur. Birtakım teknik gereksinimlerin dışında, (uçak hangarları, fabrika binaları) bazı sosyal faaliyetlerin yürütüldüğü binalarda, mimari tekniğin elverdiği ölçüde, geniş açıklıklara ve geniş alanların kapatılmasına yönelmiştir. Artık günümüzde hiç kimse hertarafı kolonlarla kaplı bir spor salonunda çalışmaktan veya bir futbol karşılaşmasını kolon arkasında seyretmekten hoşlanmamaktadır. Bu gibi hoşnutsuzlukları giderme işi doğal olarak mühendislere düşmüştür. Ülkemizde oldukça çok kullanılan bir malzeme olan betonarme ile bu tür problemleri çözmek oldukça güç olmuştur. Beton ve kalitesini artırmak ve hatta yeni bir yöntem olan öngerilmeli beton sistemleri kullanmak dahi bu probleme yeterince kesin çözümler getirmemiştir. Öte yandan betonarmenin yetmediği veya ekonomik olmaktan çıktığı durumlarda eskiden beri kullanılan malzeme çeliktir. Buna rağmen çağımızın ihtiyaçları veya daha çok lüksü diyebiliriz, öyle boyutlara ulaşmıştır ki bu sağlam malzeme dahi bazı durumlarda yetmemektedir. O zaman daha başka çözümler, kompozit kesitler kullanılmakta hatta malzeme olarak branda dan faydalanılabilmektedir.

Genelde, çelik, kafes giriş dediğimiz düzlemsel taşıyıcı elemanlarda oldukça ekonomik olmaktadır. (büyük açıklıklarda) Ancak açıklıklar arttıkça kesitler büyümekte, kesitler büyüdükçe ağırlıklar artmakta ve bu kısır döngü içerisinde düzlemsel taşıyıcı elemanlar ekonomik avantajlarını yitirmektedirler. Bu duruma mühendislerce getirilen çözüm ise uzay taşıyıcı sistemlerdir. Üçboyutlu olarak teşkil edilen ve çelik çubukların birbirine bağlanması ile oluşan bu tür sistemlere uzay kafes girişler adı verilir.

Uzay kafes sistemlerin hesabı da oldukça zahmetlidir. Eski yöntemlerle (el ile) hesaplanması doğal olarak mümkün değildir. Ritter kesim yöntemi ile hesap yapmayı düşünmek bile yanlış olur. Gelişen bilgisayar imkanları, hesapları kolaylaştırmış, kesinleştirmiş bununla birlikte kullanım imkanı ve güvenilirliği de artmıştır.

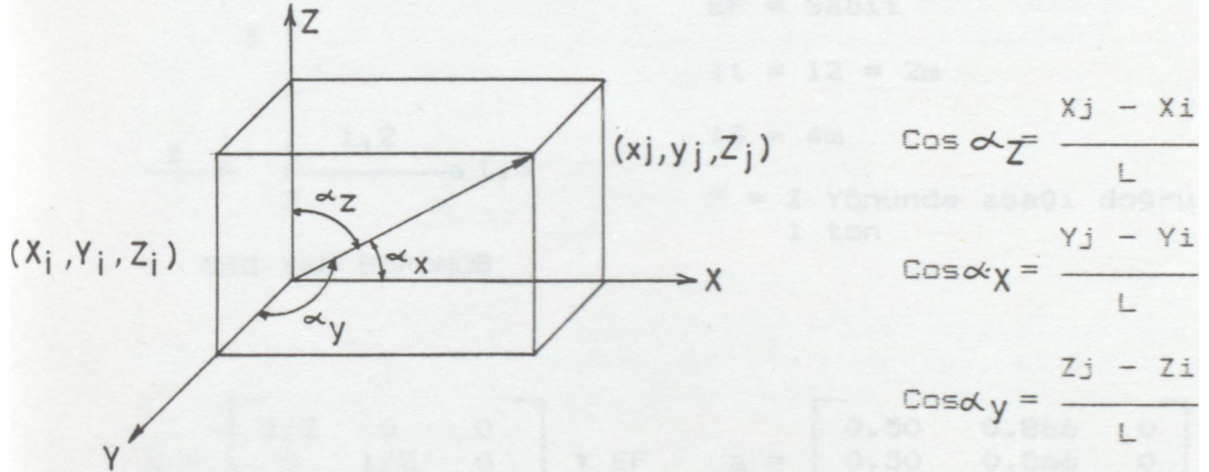
UZAY KAFES SİSTEMLERİN MATRİS - DEPLASMAN METODU İLE HESABI

Daha önce açıklandığı gibi Matris metodları birtakım sistematik hesap yöntemleridir. Bağıntılar ve hesap yolu açıklanmıştır. Bu bölümde bağıntılar üç boyutlu sisteme adapte edilecek ve örnekler verilecektir.

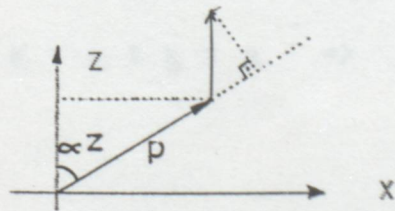
Matris - Deplasman metodunda en önemli işlem a matrisinin oluşturulmasıdır. Diğer işlemler ise sadece matrisiyel işlemlerdir. Bilgisayar ile hesap yaparken gerekli olan ilk bilgiler uzaydaki noktaların koordinatlarıdır. Bu koordinatlardan çubuk boyları ve çubuk doğrultu kosinüsleri hesaplanır.

Doğrultu kosinüsleri :

$$L = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2 + (Z_j - Z_i)^2}$$

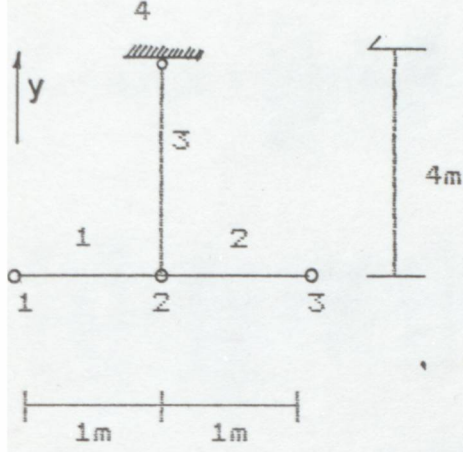


a Matrisi hesaplanırken düğüm noktalarına verilen deplasmanlar çubukların i , j uçları için ayrı ayrı hesaplanan X, Y, Z, doğrultu kosinüsleri ile çarpılınca çubuklardaki boy değişimini verir.

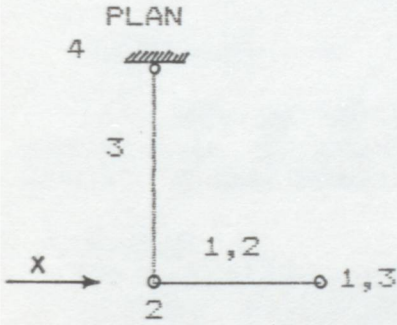


$$r_z = 1 \Rightarrow v_{pz} = \cos \alpha_Z \text{ olur.}$$

ÖRNEK 5 :



AYRIT	X	Y	Z
1	0	0	0
2	1	0	1.73
3	2	0	0
4	1	4	1.73



SAG YAN GÖRÜNÜŞÜ

EF = Sabit

11 = 12 = 2m

13 = 4m

F = Z Yönünde aşağı doğru
1 ton

$$\underline{k} = \begin{bmatrix} 1/2 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/2 \end{bmatrix} * EF \quad \underline{a} = \begin{bmatrix} 0.50 & 0.866 & 0 \\ 0.50 & 0.866 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$\underline{K} = \underline{a} * \underline{k} * \underline{a} \Rightarrow \underline{K} = \begin{bmatrix} 0.25 & 0 & 0 \\ 0 & 0.25 & 0 \\ 0 & 0 & 0.25 \end{bmatrix} * EF$$

$$\underline{K} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 1.34 & 0 \\ 0 & 0 & 4 \end{bmatrix}$$

$$\underline{k} * \underline{a} * \underline{K} = \begin{bmatrix} 1 & 0.58 & 0 \\ -1 & 0.58 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} S \\ S \\ S \end{bmatrix} = \begin{bmatrix} 1 & 0.58 & 0 \\ -1 & 0.58 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} S \\ S \\ S \end{bmatrix} = \begin{bmatrix} 0.58 \\ 0.58 \\ 0 \end{bmatrix}$$

BİLGİSAYAR PROGRAMI

Özellikleri :

1 -> Bir editör program yardımı ile data dosyası hazırlanır. Bu dosya çözüm için gerekli bilgileri içerir. 5 ana başlıktan oluşur.

SİSTEM :

N= Sistemdeki düğüm noktası sayısı

ELEMAN= Sistemdeki eleman sayısı

AYRITLAR :

Bu başlık altında düğüm noktalarının koordinatları yer alır. ilk değer düğüm noktası numarası, sonraki "X=" "Y=" "Z=" ifadeleriyle tanımlanan değerler ise X,Y,Z koordinatlarıdır.

MESNETLER :

Bu başlık altında sistemin mesnetlenme durumu girilir. ilk değer mesnet olarak verilen ayrıtın numarasıdır. "SERBESTLIK=1,1,1" şeklinde girilen veriler ise o ayrıtın X,Y,Z yönlerinden herhangi birinde kayıcı mesnet teşkil edilebilir.

ELEMANLAR :

Bu başlık altında çubuk kesit değerleri ve sistem içindeki konumları girilir.

TS= Sistemdeki farklı tipteki çubuk sayısı.
Hemen altındaki satıra "TS=" ile tanımlanan sayıda farklı kesit değerleri girilir. A= Alan; E= Elastisite modülüdür.

Çubuk kesitleri girildikten sonra tek tek elemanlar girilir.

1. Değer : Eleman numarası.
 2. Değer : Elemanın bir ucunun bağlı olduğu ayrıt.
 3. Değer : Elemanın diğer ucunun bağlı olduğu ayrıt.
- TS= Yukarıda girilen farklı kesit tiplerinden birisi.

YÜKLEMELER :

bu başlık altında, sistemdeki yükler girilir. İlk değer yükün etki ettiği düğüm noktası, daha sonraki F=1,1,1 değeri ise X Y Z doğrultusundaki yüklerdir.

Burada mesnet olarak tanımladığımız noktalara yük vermemek gerekir.

Her başlık altındaki datalar bittikten sonra " : " işareti konmalıdır. MESNET Başlığı hariç.

Sonuç değerler birkaç programın arka arkaya çalışması sonucunda elde edilir. İlk olarak TEZ1.BAS çalıştırılır. Program kendisinden sonra gelmesi gereken programı işleme koyar.

İŞLEM SIRASI :

- 1 -> TEZ1.BAS : Data dosyasını düzenler ve yeni bir data dosyası yazar. bu dosya ASCII formundadır ve ilk dosyaya oranla daha basit bir şekildedir. SYS.COZ
- 2 -> MESNET.BAS : Bu program mesnet datalarını derler. Sistemin eksiği olan mesnet reaksiyonlarını bulmak için bir yaklaşımda bulunulur. Mesnet olarak verilmiş her nokta için (eğer verilmiş ise) X Y Z doğrultularında birer eleman tayin eder. Bu elemanlar (x y z) doğrultularına paraleldir ve Sp değerleri mesnet reaksiyonlarıdır. Ayrıca bu tip birkaç eleman varsa bu sayıyı sistem programında kullanılmak üzere saklar. Mesnet numaraları ve yönleride belirlidir.
- 3 -> SISTEM.BAS : Bu program sistem datalarını oluşturur. Bu bilgiler daha sonra HAKAN2.BAS programınca kullanılmak üzere saklanır. SOUC1.DAT
- 4 -> YUK.BAS : Bu program yükler başlığı altında verilen dataları R matrisine dönüştürür.
- 5 -> RIJIT.BAS : Bu programda hesaplanan değerler oldukça fazladır. Eleman boyları, doğrultu kosinüsleri, a kinematik transformasyon matrisi bu programda hesaplanır.
- 6 -> HAKAN2.BAS : Kendisinden önceki programların

basit hale indirgediđi datalardan Matri - Deplasman metoduna gre sistemin zmn yapar. Sonu olarak ıktılar SONUC.OUT dosyasında saklanır.

7 -> Programın Dskete yazdıđı Dosyalar :

M.DAT : Mesnet noktası adedi
M1.DAT : Mesnat numaraları
M2.DAT : Fiktif mesnet elemanlarının sayısı
A.DAT : Sistemdeki nokta sayısı
F.DAT : Ykl noktaların sayısı
E.DAT : Sistemdeki ubuk sayısı
SONUC1.DAT : Sistem dataları (Nokta, Eleman sayısı)
SONUC2.DAT : K Rijitlik matrisi
SONUC3.DAT : R Dıř yk vektr
SONUC4.DAT : a Kinematik transformasyon matrisi
SONUC.OUT : S ubuk kuvvetleri matrisi

```
10 CLS :
20 PRINT "#####"
30 PRINT "#"
40 PRINT "#          MATRIS METODLARI TEZ ÇALIŞMASI          #"
50 PRINT "#"
60 PRINT "#          HAZIRLAYAN AHMET BÜYÜKPOYRAZ   YÜK.LtS 893211          #"
70 PRINT "#"
80 PRINT "#          UZAY KAFES SİSTEMLERİN ÇÖZÜMÜ          #"
90 PRINT "#"
100 PRINT "#####"
115 PRINT "DOSYA": INPUT A$
110 DIM EL$(20), FKL$(20): WAIT 1000, 1000
120 REM BU PROGRAM SATIR İŞLER
130 OPEN A$ FOR INPUT AS #1: OPEN "SYS.COZ" FOR OUTPUT AS #2
140 LINE INPUT #1, SATIR$: PRINT SATIR$
150 IF SATIR$ = "SİSTEM" THEN PRINT #2, "S": GOSUB 220
160 IF SATIR$ = "AYRITLAR" THEN PRINT #2, "A": GOSUB 310
170 IF SATIR$ = "MESNETLER" THEN PRINT #2, "M": GOSUB 480
180 IF SATIR$ = "ELEMENLAR" THEN PRINT #2, "E": GOSUB 630
190 IF SATIR$ = "YUKLEMELER" THEN PRINT #2, "Y": GOSUB 1040
200 IF EOF(1) THEN CLOSE #1: CLOSE #2: CHAIN "MESNET"
210 GOTO 140
220 REM SİSTEM DATALARI
230 LINE INPUT #1, SATIR$: PRINT SATIR$
240 LET PAT$ = "N=": LET KIT$ = "ELEMEN="
250 POT = INSTR(SATIR$, PAT$): PD = POT + 2: KAT = INSTR(SATIR$, KIT$): KI = KAT + 7
260 LET N$ = MID$(SATIR$, PD, 5): LET ELN$ = MID$(SATIR$, KI, 3)
270 N = VAL(N$): ELN = VAL(ELN$)
280 WRITE #2, N, ELN, ELN
290 PRINT N,ELN,ELN
300 RETURN
310 REM AYRITLARIN DATALARI
320 X$ = "X=": Y$ = "Y=": Z$ = "Z="
330 FOR I = 1 TO N
340 LINE INPUT #1, SATIR$: PRINT SATIR$
350 X1 = INSTR(SATIR$, X$)
360 Y1 = INSTR(SATIR$, Y$)
370 Z1 = INSTR(SATIR$, Z$)
380 LET X1$ = MID$(SATIR$, (X1 + 2), 5)
390 LET Y1$ = MID$(SATIR$, (Y1 + 2), 5)
400 LET Z1$ = MID$(SATIR$, (Z1 + 2), 5)
410 X1 = VAL(X1$): Y1 = VAL(Y1$): Z1 = VAL(Z1$)
420 JN = VAL(SATIR$)
430 WRITE #2, JN, X1, Y1, Z1: KONT = KONT + 1
440 PRINT JN,X1,Y1,Z1
450 NEXT I
460 LINE INPUT #1, SATIR$
470 OPEN "A.DAT" FOR OUTPUT AS #3: WRITE #3, KONT: CLOSE #3: KONT = 0: RETURN
```

```
480 REM MESNETLER ALT PROGRAMI
490 KONT = 0
500 LINE INPUT #1, SATIR$: PRINT SATIR$
510 IF SATIR$ = ":" THEN GOTO 620
520 MN = VAL(SATIR$)
530 LET SER$ = "SERBESTLIK"
540 K = INSTR(SATIR$, SER$)
550 LET K$ = MID$(SATIR$, (K + 1), 5)
560 LET SX$ = MID$(K$, 1, 1): SX = VAL(SX$)
570 LET SY$ = MID$(K$, 3, 1): SY = VAL(SY$)
580 LET SZ$ = MID$(K$, 5, 1): SZ = VAL(SZ$)
590 WRITE #2, MN, SX, SY, SZ: KONT = KONT + 1
600 'PRINT MN,SX,SY,SZ
610 GOTO 500
620 OPEN "M.DAT" FOR OUTPUT AS #3: WRITE #3, KONT: CLOSE #3: KONT = 0: RETURN
630 REM ELEMENLAR ALT PROGRAMI
640 LINE INPUT #1, SATIR$: PRINT SATIR$
650 IF SATIR$ = ":" THEN GOTO 1120
660 IF EOF(1) THEN GOTO 1120
670 LET T$ = "T="
680 TI = INSTR(SATIR$, T$)
690 LET TL$ = MID$(SATIR$, (TI + 3), 5): T = VAL(TL$)
700 WRITE #2, T
710 PRINT T
720 FOR I = 1 TO T
730 LINE INPUT #1, SATIR$: PRINT SATIR$
740 EN = VAL(SATIR$)
750 E$ = "E=": A$ = "A="
760 K = INSTR(SATIR$, E$): L = INSTR(SATIR$, A$)
770 LET E$ = MID$(SATIR$, (K + 2), 10)
780 LET A$ = MID$(SATIR$, (L + 2), 10)
790 E = VAL(E$): A = VAL(A$)
800 WRITE #2, EN, A, E:
810 'PRINT EN,A,E
820 NEXT I
830 LINE INPUT #1, SATIR$: PRINT SATIR$
840 IF EOF(1) THEN GOTO 1030
850 IF SATIR$ = ":" THEN GOTO 1030
860 LET T$ = "T="
870 TI = INSTR(SATIR$, T$)
880 LET TK$ = MID$(SATIR$, (TI + 2), 2): TK = VAL(TK$)
890 LET ARA$ = MID$(SATIR$, 1, TI)
900 FOR I = 1 TO TI
910 LET EL$(I) = MID$(ARA$, I, 1): NEXT I
920 KONTROL = 1
930 FOR I = 1 TO TI
940 IF EL$(I) = "," THEN KONTROL = KONTROL + 1: GOTO 990
950 IF KONTROL = 1 THEN LET B1$ = B1$ + EL$(I)
```

```
960 IF KONTROL = 2 THEN LET B2$ = B2$ + EL$(I)
970 IF KONTROL = 3 THEN LET B3$ = B3$ + EL$(I)
980 B1 = VAL(B1$): B2 = VAL(B2$): B3 = VAL(B3$)
990 NEXT I
1000 WRITE #2, B1, B2, B3, TK: LET B1$ = "": LET B2$ = "": LET B3$ = "": KONT = KONT + 1
1010 *PRINT B1,B2,B3,TK
1020 GOTO B30
1030 OPEN "E.DAT" FOR OUTPUT AS #3: WRITE #3, KONT: CLOSE #3: KONT = 0: RETURN
1040 REM YUKLEMELER ALT PROGRAMI
1050 LINE INPUT #1, SATIR$: PRINT SATIR$
1060 IF EOF(1) THEN GOTO 1250
1070 IF SATIR$ = "" THEN GOTO 1250
1080 NKK = VAL(SATIR$)
1090 LET F$ = "F="
1100 QW = INSTR(SATIR$, F$)
1110 LET QW$ = MID$(SATIR$, (QW + 2), 20)
1120 FOR I = 1 TO 10
1130 LET FKL$(I) = MID$(QW$, I, 1): NEXT I
1140 KONTROL = 1
1150 FOR I = 1 TO 10
1160 IF FKL$(I) = "," THEN KONTROL = KONTROL + 1: GOTO 1200
1170 IF KONTROL = 1 THEN LET F1$ = F1$ + FKL$(I)
1180 IF KONTROL = 2 THEN LET F2$ = F2$ + FKL$(I)
1190 IF KONTROL = 3 THEN LET F3$ = F3$ + FKL$(I)
1200 NEXT I
1210 F1 = VAL(F1$): F2 = VAL(F2$): F3 = VAL(F3$)
1220 WRITE #2, NKK, F1, F2, F3: LET F1$ = "": LET F2$ = "": LET F3$ = "": KONT = KONT + 1
1230 *PRINT NKK,F1,F2,F3
1240 GOTO 1040
1250 OPEN "F.DAT" FOR OUTPUT AS #3: WRITE #3, KONT: CLOSE #3: KONT = 0: RETURN

20 REM BU PROGRAM MESNETLERIN BERBESTLILIKLERINI YAZAR
21 GOSUB 100: PRINT "DATA DIRMASI ISLENDI"
30 OPEN "M1.DAT" FOR INPUT AS #1: KONT = 0
40 INPUT #1, ME: CLOSE #1
50 OPEN "SMB.DAT" FOR INPUT AS #1: OPEN "M1.DAT" FOR OUTPUT AS #2: OPEN "M2.DAT" FOR OUTPUT AS #3
60 LINE INPUT #1, SATIR$
70 IF SATIR$ = "M" THEN GOSUB 100
80 IF EOF(1) THEN CLOSE #1: CLOSE #2: CLOSE #3: PRINT "MESNETLER BELIRLENDI": CHAIN "SYSTEM"
90 GOTO 60
100 FOR I = 1 TO ME
110 INPUT #1, AN, MX, MY, MZ
120 IF MX = 1 THEN LET AN$ = STR$(AN): LET KAL$ = "X": KONT = KONT + 1: WRITE #2, KAL$, AN: PRINT KAL$, AN
130 IF MY = 1 THEN LET AN$ = STR$(AN): LET KAL$ = "Y": KONT = KONT + 1: WRITE #2, KAL$, AN: PRINT KAL$, AN
140 IF MZ = 1 THEN LET AN$ = STR$(AN): LET KAL$ = "Z": KONT = KONT + 1: WRITE #2, KAL$, AN: PRINT KAL$, AN
150 NEXT I
160 WRITE #3, KONT: PRINT KONT
170 RETURN
```

```
20 REM BU PROGRAM SISTEM DATALARINI YAZAR
30 OPEN "M2.DAT" FOR INPUT AS #1
40 INPUT #1, MES: CLOSE #1
45 OPEN "SYS.COZ" FOR INPUT AS #1
50 LINE INPUT #1, SATIR#
55 IF EOF(1) THEN CLOSE #1: PRINT "SISTEM DEGERLERI OKUNDU": CHAIN "YUK"
80 IF SATIR# = "S" THEN GOSUB 110
90 GOTO 50
110 INPUT #1, ASU, ASA
120 OPEN "SONUC1.DAT" FOR OUTPUT AS #3
130 ASA = ASA + MES
140 WRITE #3, ASU, ASA, ASA: CLOSE #3
160 RETURN

20 REM BU PROGRAM [R] MATRISINI YAZAR
30 OPEN "SONUC1.DAT" FOR INPUT AS #1: OPEN "F.DAT" FOR INPUT AS #2
40 INPUT #1, A1, A2, A3: INPUT #2, A4: CLOSE #1: CLOSE #2
50 DIM F(A1 * 3)
60 OPEN "SYS.COZ" FOR INPUT AS #1: OPEN "SONUC3.DAT" FOR OUTPUT AS #2
70 INPUT #1, SATIR#
80 IF SATIR# = "Y" THEN GOSUB 110
90 IF EOF(1) THEN CLOSE #1: PRINT "YUK DEGERLERI OKUNDU": CHAIN "RIJIT"
100 GOTO 70
110 FOR I = 1 TO A4
120 INPUT #1, B1, B2, B3, B4
130 QA = (B1 - 1) * 3 + 1
140 F(QA) = B2: F(QA + 1) = B3: F(QA + 2) = B4
150 NEXT I
160 FOR I = 1 TO A1 * 3
170 WRITE #2, F(I): PRINT F(I);
180 NEXT I
190 CLOSE #2
200 RETURN

10 OPEN "SONUC1.DAT" FOR INPUT AS #1: OPEN "M2.DAT" FOR INPUT AS #2
20 INPUT #1, A1, A2: INPUT #2, A3: CLOSE #1: CLOSE #2
30 DIM AYRI(A1, 4), ELAY(A2, 4), K(A2, A2), BOY(A2), A(A2, (A3 * 3))
40 OPEN "SYS.COZ" FOR INPUT AS #1
50 INPUT #1, SATIR#
60 IF SATIR# = "A" THEN GOSUB 100
70 IF SATIR# = "E" THEN GOSUB 170
80 IF EOF(1) THEN CLOSE #1: GOTO 350
90 GOTO 50
100 FOR I = 1 TO A1
110 INPUT #1, B1, B2, B3, B4
120 AYRI(B1, 1) = B1: AYRI(B1, 2) = B2: AYRI(B1, 3) = B3: AYRI(B1, 4) = B4
130 ' PRINT AYRI(B1, 1), AYRI(B1, 2), AYRI(B1, 3), AYRI(B1, 4)
```

```
140 IF B1 = 0 THEN PRINT "SISTEMDE HATA VAR"; I; "SATIRDA"
150 NEXT I
160 RETURN
170 INPUT #1, ADET: DIM TI(ADET), AD(ADET), E(ADET)
180 FOR I = 1 TO ADET
190 INPUT #1, TI(I), AD(I), E(I)
200 ' PRINT TI(I), AD(I), E(I)
210 NEXT I
220 FOR I = 1 TO A2
230 IF I > (A2 - A3) THEN GOTO 270
240 INPUT #1, B1, B2, B3, B4
250 ELAY(B1, 1) = B1: ELAY(B1, 2) = B2: ELAY(B1, 3) = B3: ELAY(B1, 4) = E(B4) * AD(B4)
260 ' PRINT ELAY(B1, 1), ELAY(B1, 2), ELAY(B1, 3), ELAY(B1, 4)
270 NEXT I
280 OPEN "M1.DAT" FOR INPUT AS #2
290 FOR I = 1 TO A2
300 IF I <= (A2 - A3) THEN GOTO 330
310 INPUT #2, EL$: INPUT #2, ELAY(I, 3): ELAY(I, 2) = 0: ELAY(I, 4) = E(I) * 10000 * AD(I) * 100000
320 ' PRINT EL$, ELAY(I, 2), ELAY(I, 3), ELAY(I, 4)
330 NEXT I: CLOSE #2
340 RETURN
350 ERASE TI, A, E
360 FOR I = 1 TO A2
370 IF I > (A2 - A3) THEN BOY(I) = .01: GOTO 400
380 Z1 = ELAY(I, 2): Z2 = ELAY(I, 3)
390 BOY(I) = SQR((AYRI(Z1, 2) - AYRI(Z2, 2)) ^ 2 + (AYRI(Z1, 3) - AYRI(Z2, 3)) ^ 2 + (AYRI(Z1, 4) - AYRI(Z2, 4)) ^ 2): PRINT BOY(I)
400 NEXT I
410 OPEN "SONUC2.DAT" FOR OUTPUT AS #2
420 FOR I = 1 TO A2
430 FOR J = 1 TO A2
440 IF I = J THEN K(I, J) = ELAY(J, 4) / BOY(I): WRITE #2, K(I, J): GOTO 460
450 WRITE #2, K(I, J): PRINT K(I, J);
460 NEXT J: NEXT I: CLOSE #2
470 REM [A] MATRISININ YAZILMASI
480 DIM AYD(A2, 7)
490 FOR I = 1 TO A2
500 IF I > (A2 - A3) THEN AYD(I, 1) = AYD(I, 1) + I - A2 + A3: GOTO 590
510 AYD(I, 1) = ELAY(I, 1)
520 UCI = ELAY(I, 2): UCJ = ELAY(I, 3)
530 K1 = AYRI(UCJ, 2) - AYRI(UCI, 2): IF K1=0 THEN K1=1E-08
540 K2 = AYRI(UCJ, 3) - AYRI(UCI, 3): IF K2=0 THEN K2=1E-08
550 K3 = AYRI(UCJ, 4) - AYRI(UCI, 4): IF K3=0 THEN K3=1E-08
560 AYD(I, 2) = K1 / BOY(I)
570 AYD(I, 3) = K2 / BOY(I)
580 AYD(I, 4) = K3 / BOY(I)
590 NEXT I
600 FOR I = 1 TO A2
```

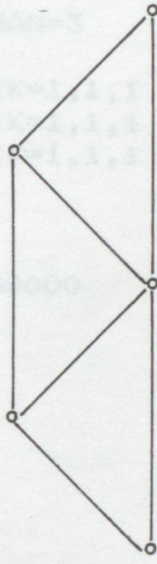
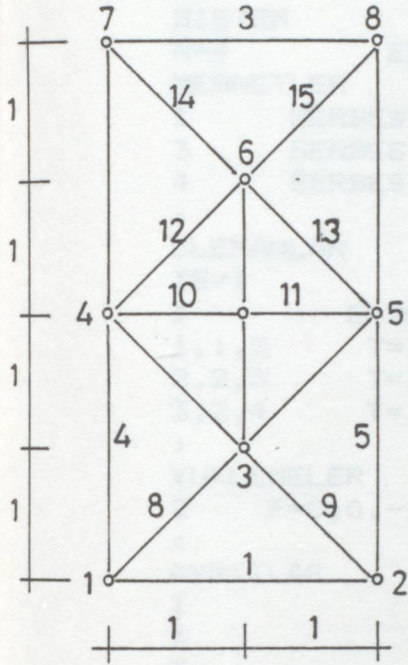
```
610 FOR J = 1 TO 4
620 IF AYD(I, J) < 0 THEN AYD(I, J) = AYD(I, J) * (-1)
630 NEXT J: NEXT I
640 OPEN "M1.DAT" FOR INPUT AS #2: OPEN "M2.DAT" FOR INPUT AS #3
650 INPUT #3, MN: CLOSE #3
660 FOR I = 1 TO MN
670 INPUT #2, BOK$, JN
680 IF BOK$ = "X" THEN AYD(A2 - A3 + I, 2) = 1: AYD(A2 - A3 + I, 5) = 1: GOTO 710
690 IF BOK$ = "Y" THEN AYD(A2 - A3 + I, 3) = 1: AYD(A2 - A3 + I, 6) = 1: GOTO 710
700 IF BOK$ = "Z" THEN AYD(A2 - A3 + I, 4) = 1: AYD(A2 - A3 + I, 7) = 1: GOTO 710
710 NEXT I: CLOSE #2
720 REM AYD MATRISI DUZENLENIYOR
730 FOR I = 1 TO A2 - A3
740 UCI = ELAY(I, 2): UCJ = ELAY(I, 3)
750 IF AYRI(UCJ, 4) - AYRI(UCI, 4) >= 0 THEN GOTO 790
760 IF AYRI(UCJ, 4) - AYRI(UCI, 4) < 0 THEN GOTO 830
770 NEXT I
780 GOTO 870
790 IF AYRI(UCJ, 2) - AYRI(UCI, 2) >= 0 AND AYRI(UCJ, 3) - AYRI(UCI, 3) >= 0 THEN AYD(I, 2) = AYD(I, 2) *
(-1): AYD(I, 3) = AYD(I, 3) * (-1): AYD(I, 4) = AYD(I, 4) * (-1): GOTO 770 800 IF AYRI(UCJ, 2) - AYRI(UCI,
2) >= 0 AND AYRI(UCJ, 3) - AYRI(UCI, 3) <= 0 THEN AYD(I, 2) = AYD(I, 2) * (-1): AYD(I, 3) = AYD(I, 3) *
(1): AYD(I, 4) = AYD(I, 4) * (-1): GOTO 770 810 IF AYRI(UCJ, 2) - AYRI(UCI, 2) <= 0 AND AYRI(UCJ, 3) -
AYRI(UCI, 3) >= 0 THEN AYD(I, 2) = AYD(I, 2) * (1): AYD(I, 3) = AYD(I, 3) * (-1): AYD(I, 4) = AYD(I, 4) *
(-1): GOTO 770 820 IF AYRI(UCJ, 2) - AYRI(UCI, 2) <= 0 AND AYRI(UCJ, 3) - AYRI(UCI, 3) <= 0 THEN AYD(I, 2)
= AYD(I, 2) * (1): AYD(I, 3) = AYD(I, 3) * (1): AYD(I, 4) = AYD(I, 4) * (-1): GOTO 770 830 IF AYRI(UCJ, 2)
- AYRI(UCI, 2) >= 0 AND AYRI(UCJ, 3) - AYRI(UCI, 3) >= 0 THEN AYD(I, 2) = AYD(I, 2) * (-1): AYD(I, 3) =
AYD(I, 3) * (-1): GOTO 770
840 IF AYRI(UCJ, 2) - AYRI(UCI, 2) >= 0 AND AYRI(UCJ, 3) - AYRI(UCI, 3) <= 0 THEN AYD(I, 2) = AYD(I, 2) *
(-1): AYD(I, 3) = AYD(I, 3) * (1): GOTO 770
850 IF AYRI(UCJ, 2) - AYRI(UCI, 2) <= 0 AND AYRI(UCJ, 3) - AYRI(UCI, 3) >= 0 THEN AYD(I, 2) = AYD(I, 2) *
(1): AYD(I, 3) = AYD(I, 3) * (-1): GOTO 770
860 IF AYRI(UCJ, 2) - AYRI(UCI, 2) <= 0 AND AYRI(UCJ, 3) - AYRI(UCI, 3) <= 0 THEN AYD(I, 2) = AYD(I, 2) *
(1): AYD(I, 3) = AYD(I, 3) * (1): GOTO 770
870 FOR I = 1 TO A2 - A3
880 AYD(I, 5) = AYD(I, 2) * (-1): AYD(I, 6) = AYD(I, 3) * (-1): AYD(I, 7) = AYD(I, 4) * (-1)
890 NEXT I
900 FOR I = 1 TO A2
910 FOR J = 1 TO 4
920 PRINT ELAY(I, J);
930 NEXT J: PRINT
940 NEXT I: PRINT
950 REM NIHAYET [A] MATRISI
960 DIM A(A2, A1 * 3)
970 FOR I = 1 TO A1
980 FOR J = 1 TO A2
990 FOR S = 2 TO 3
1000 IF I = ELAY(J, S) THEN GOTO 1040
1010 NEXT S
```

```
1020 NEXT J
1030 NEXT I: GOTO 1060
1040 IF S = 2 THEN A(J, 3 * I - 2) = AYD(J, 2): A(J, 3 * I - 1) = AYD(J, 3): A(J, 3 * I) = AYD(J, 4): GOTO
1010
1050 A(J, 3 * I - 2) = AYD(J, 5): A(J, 3 * I - 1) = AYD(J, 6): A(J, 3 * I) = AYD(J, 7): GOTO 1010
1060 OPEN "SONUC4.DAT" FOR OUTPUT AS #1
1070 FOR I = 1 TO A2
1080 FOR J = 1 TO A1 * 3
1090 WRITE #1, A(I, J): PRINT A(I, J)
1100 NEXT J
1110 NEXT I
1120 CLOSE #1: PRINT "SISTEM MATRISLERI HAZIRLANDI": CHAIN "HAKAND"

10 OPTION BASE 1: PRINT "HESAPLAR YAPILYOR"
20 OPEN "SONUC1.DAT" FOR INPUT AS #1: INPUT #1, B, M: B = B * 3: CLOSE #1
30 DIM A(M, B), KA(M, B), BA(M, B), K1(M, M), K(B, B), INVK(B, B), AT(B, M), S(M, 1), S1(M, 1), R(B, 1)
40 REM DATATALARIN OKUTULMASI
70 OPEN "SONUC4.DAT" FOR INPUT AS #1
80 FOR I = 1 TO M
90 FOR J = 1 TO B
110 INPUT #1, A(I, J)
140 NEXT J: NEXT I: CLOSE #1
170 OPEN "SONUC2.DAT" FOR INPUT AS #1
180 FOR I = 1 TO M
190 FOR J = 1 TO M
210 INPUT #1, K1(I, J)
230 NEXT J
240 NEXT I: CLOSE #1
260 OPEN "SONUC3.DAT" FOR INPUT AS #1
270 FOR I = 1 TO B
280 INPUT #1, R(I, 1): NEXT: CLOSE #1
290 REM A MATRISININ TRANSPOZESI
300 FOR I = 1 TO M
310 FOR J = 1 TO B
320 AT(J, I) = A(I, J): NEXT J: NEXT I
330 REM K RIJITLIK MATRISI HESABI
340 FOR I = 1 TO M
350 FOR J = 1 TO B
360 FOR Z = 1 TO M
370 KA(I, J) = KA(I, J) + K1(I, Z) * A(Z, J)
380 NEXT Z: NEXT J: NEXT I
390 FOR I = 1 TO B
400 FOR J = 1 TO B
410 FOR Z = 1 TO M
420 K(I, J) = K(I, J) + AT(I, Z) * KA(Z, J): NEXT Z: NEXT J: NEXT I
421 PRINT "SISTEM RIJITLIK MATRISI [K] HESAPLANDI"
430 REM K MATRISININ INVERSI DIREKT YONTEM
440 FOR I = 1 TO B
```

```
450 FOR J = 1 TO B
460 INVK(I, J) = K(I, J) / 100000: NEXT J: NEXT I
470 FOR I = 1 TO B
480 FOR J = 1 TO B
490 FOR K = 1 TO B
500 IF J = I OR K = I THEN GOTO 520
510 LET INVK(J, K) = INVK(J, K) - INVK(J, I) * INVK(I, K) / INVK(I, I)
520 NEXT K: NEXT J
530 LET INVK(I, I) = -1 / INVK(I, I)
540 FOR J = 1 TO B
550 IF J = I THEN GOTO 570
560 LET INVK(J, I) = INVK(J, I) * INVK(I, I)
570 NEXT J
580 FOR J = 1 TO B
581 IF J = I THEN GOTO 583
582 INVK(I, J) = INVK(I, J) * INVK(I, I)
583 NEXT J: NEXT I
584 FOR I = 1 TO B
585 FOR J = 1 TO B
586 INVK(I, J) = -INVK(I, J): NEXT J: NEXT I
620 FOR I = 1 TO B
630 FOR J = 1 TO B
641 INVK(I, J) = INVK(I, J) / 100000
650 NEXT J: NEXT I
659 PRINT "[K] MATRISININ INVERSI ALINDI"
660 REM BA MATRISININ HESABI
670 FOR I = 1 TO M
680 FOR J = 1 TO B
690 FOR Z = 1 TO B
700 BA(I, J) = BA(I, J) + KA(I, Z) * INVK(Z, J): NEXT Z: NEXT J: NEXT I
710 REM S ICKUVVET MATRISI
720 FOR I = 1 TO M
730 FOR J = 1 TO B
740 S1(I, 1) = S1(I, 1) + BA(I, J) * R(J, 1): NEXT J: NEXT I
750 OPEN "SONUC.OUT" FOR OUTPUT AS #1: OPEN "M1.DAT" FOR INPUT AS #2: OPEN "M2.DAT" FOR INPUT AS #3: INPUT
#3, MES: CLOSE #3
760 DIM AN$(MES), MAS(MES)
770 FOR I = 1 TO MES
780 INPUT #2, AN$(I), MAS(I): NEXT I
790 PRINT #1, "##### CUBUK IC KUVVETLERI #####"
800 PRINT #1,
810 KONT = 0
820 FOR I = 1 TO M
830 IF I > M - MES THEN : KONT = KONT + 1: AN$(KONT) = STR$(MAS(KONT)) + AN$(KONT): PRINT #1, AN$(KONT);
"="; USING "###.###"; S1(I, 1): GOTO 850
840 PRINT #1, "S("; I; " )="; USING "+###.###"; S1(I, 1)
850 NEXT I: CLOSE #1
860 PRINT "HESAP TAMAMLANDI SONUCLAR [SONUC.OUT] DOSYASINDA": END
```

ÖRNEK 6 :



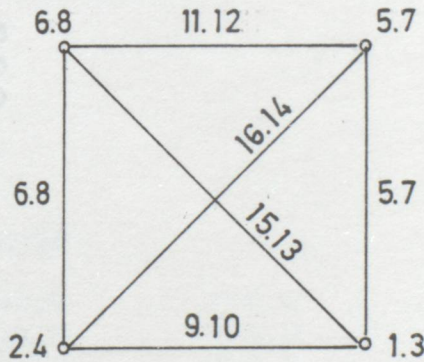
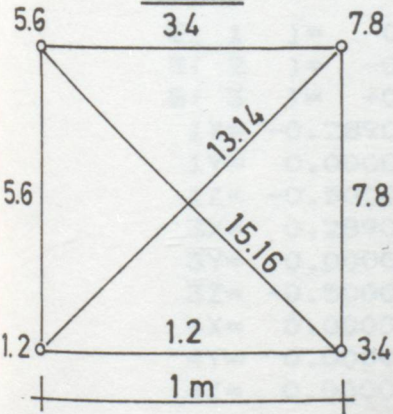
AYRIT	X	Y	Z
1	0	0	0
2	2	0	0
3	1	1	1
4	0	2	0
5	2	2	0
6	1	3	1
7	0	4	0
8	2	4	0

Yükler : (3.....6) Ayrıtlarında aşağı doğru 1 ton EF Sabit

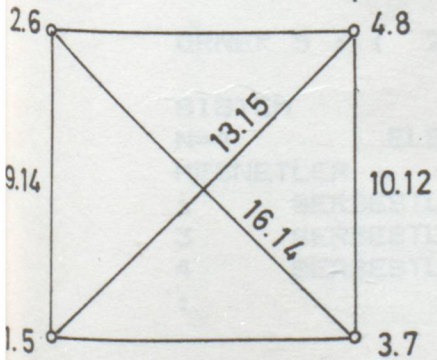
ÇÖZÜM : SAP80 Programı ile yapılacak.

ÖRNEK 7 :

PLAN



YAN GÖRÜNÜŞ



ÖN GÖRÜNÜŞ

1.2.3.4 Noktaları mesnet, 5.6.7.8 Noktaları serbest, 5.7 noktaları Aşağı doğru 1er ton yükleniyor.

ÖRNEK ÇÖZÜMLER

ÖRNEK 5 :

SISTEM

N=4 ELEMEN=3

MESNETLER

1 SERBESTLIK=1,1,1
3 SERBESTLIK=1,1,1
4 SERBESTLIK=1,1,1

ELEMANLAR

TS=1

1 E=28500000 A=0.01
1,1,2 T=1
2,2,3 T=1
3,2,4 T=1

YUKLEMELER

2 F=0,0,-1

AYRITLAR

1	X=0.0	Y=0.0	Z=0.0
2	X=1.0	Y=0.0	Z=1.73
3	X=2.0	Y=0.0	Z=0.0
4	X=1.0	Y=4.0	Z=1.73

CUBUK IC KUVVETLERI

S(1)= -0.5775

S(2)= -0.5775

S(3)= +0.0000

1X= -0.2890

1Y= 0.0000

1Z= -0.5000

3X= 0.2890

3Y= 0.0000

3Z= -0.5000

4X= 0.0000

4Y= 0.0000

4Z= 0.0000

ÖRNEK 5 : (2 Noktasında X Y Z yönlerinde 1 ton)

SISTEM

N=4 ELEMEN=3

MESNETLER

1 SERBESTLIK=1,1,1
3 SERBESTLIK=1,1,1
4 SERBESTLIK=1,1,1

ELEMANLAR

TS=1

1 E=28500000 A=0.01
1,1,2 T=1
2,2,3 T=1
3,2,4 T=1

:
YUKLEMELER

2 F=1,1,1

:
AYRITLAR

1	X=0.0	Y=0.0	Z=0.0
2	X=1.0	Y=0.0	Z=1.73
3	X=2.0	Y=0.0	Z=0.0
4	X=1.0	Y=4.0	Z=1.73

CUBUK IC KUVVETLERI

S(1)= +1.5766

S(2)= -0.4216

S(3)= -1.0000

1X= 0.7890

1Y= 0.0000

1Z= 1.3650

3X= 0.2110

3Y= 0.0000

3Z= -0.3650

4X= 0.0000

4Y= 1.0000

4Z= 0.0000

ORNEK 6 :

SISTEM

N=8 ELEMAN=16

AYRITLAR

1	X=0.0	Y=0.0	Z=0.0
2	X=2.0	Y=0.0	Z=0.0
3	X=1.0	Y=1.0	Z=1.0
4	X=0.0	Y=2.0	Z=0.0
5	X=2.0	Y=2.0	Z=0.0
6	X=1.0	Y=3.0	Z=1.0
7	X=0.0	Y=4.0	Z=0.0
8	X=2.0	Y=4.0	Z=0.0

:
MESNETLER

1 SERBESTLIK=1,1,1

2 SERBESTLIK=1,1,1

7 SERBESTLIK=1,1,1

8 SERBESTLIK=1,1,1

```
:
ELEMANLAR
TS=1
1          E=28500000          A=0.01
1,1,2      T=1
2,4,5      T=1
3,7,8      T=1
4,1,4      T=1
5,2,5      T=1
6,4,7      T=1
7,5,8      T=1
8,1,3      T=1
9,2,3      T=1
10,3,4     T=1
11,3,5     T=1
12,4,6     T=1
13,6,5     T=1
14,6,7     T=1
15,6,8     T=1
16,3,6     T=1
:
YUKLEMELER
3          F=0,0,-1
4          F=0,0,-1
5          F=0,0,-1
6          F=0,0,-1
:
#####          CUBUK  IC  KUVVETLERI          #####
S( 1  )=  +0.0000
S( 2  )=  -1.0872
S( 3  )=  +0.0000
S( 4  )=  +0.0000
S( 5  )=  +0.0000
S( 6  )=  -0.0000
S( 7  )=  -0.0000
S( 8  )=  -1.7321
S( 9  )=  -1.7321
S( 10 )=  +1.0804
S( 11 )=  +0.8052
S( 12 )=  +0.8810
S( 13 )=  +0.8558
S( 14 )=  -1.7321
S( 15 )=  -1.7321
S( 16 )=  -3.0000
1X= -1.0000
1Y= -1.0000
1Z= -1.0000
2X=  1.0000
2Y= -1.0000
```

ZZ= -1.0000
7X= -1.0000
7Y= 1.0000
7Z= -1.0000
8X= 1.0000
8Y= 1.0000
8Z= -1.0000

ÖNEK 6 : (3 ile 6 Noktasında 1 ton)

SISTEM Y2

N=8 ELEMEN=16

AYRITLAR

1	X=0.0	Y=0.0	Z=0.0
2	X=2.0	Y=0.0	Z=0.0
3	X=1.0	Y=1.0	Z=1.0
4	X=0.0	Y=2.0	Z=0.0
5	X=2.0	Y=2.0	Z=0.0
6	X=1.0	Y=3.0	Z=1.0
7	X=0.0	Y=4.0	Z=0.0
8	X=2.0	Y=4.0	Z=0.0

:

MESNETLER

1	SERBESTLIK=1,1,1
2	SERBESTLIK=1,1,1
7	SERBESTLIK=1,1,1
8	SERBESTLIK=1,1,1

:

ELEMENLAR

TS=1

1	E=28500000	A=0.01
1,1,2	T=1	
2,4,5	T=1	
3,7,8	T=1	
4,1,4	T=1	
5,2,5	T=1	
6,4,7	T=1	
7,5,8	T=1	
8,1,3	T=1	
9,2,3	T=1	
10,3,4	T=1	
11,3,5	T=1	
12,4,6	T=1	
13,6,5	T=1	A=0.01
14,6,7	T=1	
15,6,8	T=1	
16,3,6	T=1	

:

YUKLEMELER

3	F=0,0,-1
6	F=0,0,-1

```
##### CUBUK IC KUVVETLERI #####
S( 1 )= +0.0000
S( 2 )= +0.0000
S( 3 )= +0.0000
S( 4 )= +0.0000
S( 5 )= +0.0000
S( 6 )= -0.0000
S( 7 )= -0.0000
S( 8 )= -0.8660
S( 9 )= -0.8660
S( 10 )= +0.0000
S( 11 )= +0.0000
S( 12 )= +0.0000
S( 13 )= +0.0000
S( 14 )= -0.8660
S( 15 )= -0.8660
S( 16 )= -1.0000
1X= -0.5000
1Y= -0.5000
1Z= -0.5000
2X= 0.5000
2Y= -0.5000
2Z= -0.5000
7X= -0.5000
7Y= 0.5000
7Z= -0.5000
8X= 0.5000
8Y= 0.5000
8Z= -0.5000
```

ORNEK 7 :

SISTEM

N=8 ELEMEN=16

MESNETLER

- 1 10 SERBESTLIK=1,1,1
- 2 11 SERBESTLIK=1,1,1
- 3 12 SERBESTLIK=1,1,1
- 4 13 SERBESTLIK=1,1,1

ELEMANLAR

TS=1

- 1 E=28500000 A=0.01
- 1,1,3 T=1
- 2,2,4 T=1
- 3,5,7 T=1
- 4,6,8 T=1
- 5,1,5 T=1
- 6,2,6 T=1
- 7,3,7 T=1

8,4,8 0.4 T=1
9,1,2 0.3 T=1
10,3,4 0.0 T=1
11,5,6 0.5 T=1
12,7,8 T=1
13,1,8 T=1
14,2,7 T=1
15,3,6 T=1
16,4,5 T=1

:

YUKLEMELER

5 F=0,0,-1

7 F=0,0,-1

:

AYRITLAR

1	X=0.0	Y=0.0	Z=0.0
2	X=0.0	Y=0.0	Z=1.0
3	X=1.0	Y=0.0	Z=0.0
4	X=1.0	Y=0.0	Z=1.0
5	X=0.0	Y=1.0	Z=0.0
6	X=0.0	Y=1.0	Z=1.0
7	X=1.0	Y=1.0	Z=0.0
8	X=1.0	Y=1.0	Z=1.0

:

CUBUK IC KUVVETLERI

S(1)= +0.0000
S(2)= -0.0000
S(3)= -0.3859
S(4)= +0.7718
S(5)= -0.5347
S(6)= +0.4653
S(7)= -0.5347
S(8)= +0.4653
S(9)= -0.0000
S(10)= -0.0000
S(11)= +0.6243
S(12)= +0.5102
S(13)= -0.7466
S(14)= +0.9450
S(15)= -0.7294
S(16)= +1.0184

1X= -0.4653

1Y= -1.0000

1Z= -0.4653

2X= 0.5347

2Y= 1.0000

2Z= -0.5347

3X= 0.4653

3Y= -1.0000

3Z= -0.4653
4X= -0.5347
4Y= 1.0000
4Z= -0.5347

SAP80 PROGRAMI DOZUMLERI

SAP80 V03.01

***** CHG OF SAP INPUT DATA *****

TOTAL NUMBER OF JOINTS = 4
TOTAL NUMBER OF LOAD CONDITIONS = 1

PRELIMINARY SCAN OF FRAME D

MEM= 1=1

SECTION PROPERTY DATA

1. E=28300000 A=0.01

FRAME ELEMENT DATA

1,1,2 M=1
2,2,3
3,3,4

RESTRAINT INFORMATION

1,3,1 R=0,0,0,1,1,1
2 R=1,1,1,1,1
3 R=1,1,1,1,1
4 R=1,1,1,1,1

SAP80 PROGRAMI ÇÖZÜMLERİ

ÖRNEK 5 :

SYSTEM
SAP80 V85.01

***** ECHO OF SAP INPUT DATA *****

TOTAL NUMBER OF JOINTS = 4
TOTAL NUMBER OF LOAD CONDITIONS = 1

PRELIMINARY SCAN OF FRAME DATA

NM=1 Z=-1

SECTION PROPERTY DATA
1 E=28500000 A=0.01

FRAME ELEMENT DATA
1,1,2 M=1
2,2,3
3,2,4

RESTRAINT INFORMATION

1,3,1 R=0,0,0,1,1,1
1 R=1,1,1,1,1,1
3 R=1,1,1,1,1,1
4 R=1,1,1,1,1,1

EQUILIBRIUM EQUATION NUMBERS
(ONE EQUATION FOR EACH UNKNOWN DISPLACEMENT)

JOINT #	U(X)	U(Y)	U(Z)	R(X)	R(Y)	R(Z)
1	0	0	0	0	0	0
2	1	2	3	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0

INPUT JOINT DATA

1	NEW LOAD	X=0.0	DIST	Y=0.0	1-2 PL	Z=0.0	PLANE
2	AXIAL	X=1.0		Y=0.0		Z=1.73	
3		X=2.0		Y=0.0		Z=0.0	JOINT SHEAR
4	MOMENT TORQUE	X=1.0		Y=4.0		Z=1.73	

GENERATED JOINT COORDINATES

JOINT #	X	Y	Z
1	.000	.000	.000
2	1.000	.000	1.730
3	2.000	.000	.000
4	1.000	4.000	1.730

SYSTEM
SAP80 V85.03

***** JOINT DISPLACEMENTS *****

LOAD CONDITION 1 - DISPLACEMENTS "U" AND ROTATIONS "R"

JOINT	U(X)	U(Y)	U(Z)
1	.0000E+00	.0000E+00	.0000E+00
2	.0000E+00	.0000E+00	-.4677E-05
3	.0000E+00	.0000E+00	.0000E+00
4	.0000E+00	.0000E+00	.0000E+00

REACTIONS AND APPLIED FORCES

LOAD CONDITION 1 - FORCES "F" AND MOMENTS "M"

JOINT	F(X)	F(Z)
1	.2890	.5000
2	.0000	-1.0000
3	-.2890	.5000
4	.0000	.0000

TOTAL .0000E+00 .0000E+00

SYSTEM
SAP80 V85.03

***** FRAME MEMBER FORCES *****

MEMBERS WITH NUMBERS BETWEEN 1 & 32000

MEM	LOAD	(X)	AXIAL	DIST	R(X)	1-2 PLANE	1-3 PLANE
AXIAL	#	#	FORCE	I		SHEAR	MOMENT SHEAR
MOMENT	3	TORQUE					
1	1						
2	1						
3	1						

ÖRNEK 5 : (2 Noktasında X Y Z yönlerinde 1 ton)

SYSTEM
SAP80 V85.01

GENERATED JOINT COORDINATES

***** ECHO OF SAP INPUT DATA *****

TOTAL NUMBER OF JOINTS = 4
TOTAL NUMBER OF LOAD CONDITIONS = 1

SYSTEM
SAP80 V85.01

PRELIMINARY SCAN OF FRAME DATA

NM=1 Z=-1 X=1 Y=1

SECTION PROPERTY DATA

1 JOINT E=28500000 A=0.01 U(1)
1 .0000E+00 .0000E+00 .0000E+00

FRAME ELEMENT DATA

1,1,2 M=1 .0000E+00 .0000E+00 .0000E+00
2,2,3 .0000E+00 .0000E+00 .0000E+00
3,2,4

REACTIONS AND APPLIED FORCE

RESTRAINT INFORMATION

LOAD CONDITION 1 - FORCES "F" AND MOMENTS "M"

1 R=1,1,1,1,1,1
2 R=0,0,0,1,1,1
3 R=1,1,1,1,1,1
4 R=1,1,1,1,1,1

EQUILIBRIUM EQUATION NUMBERS
(ONE EQUATION FOR EACH UNKNOWN DISPLACEMENT)

JOINT #	U(X)	U(Y)	U(Z)	R(X)	R(Y)	R(Z)
1	0	0	0	0	0	0
2	1	2	3	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0

INPUT JOINT DATA

1	X=0.0	Y=0.0	Z=0.0
2	X=1.0	Y=0.0	Z=1.73
3	X=2.0	Y=0.0	Z=0.0
4	X=1.0	Y=4.0	Z=1.73

GENERATED JOINT COORDINATES

JOINT #	X	Y	Z
1	.000	.000	.000
2	1.000	.000	1.730
3	2.000	.000	.000
4	1.000	4.000	1.730

SYSTEM
SAP80 V85.03

***** JOINT DISPLACEMENTS *****

LOAD CONDITION 1 - DISPLACEMENTS "U" AND ROTATIONS "R"

JOINT	U(X)	U(Y)	U(Z)
1	.0000E+00	.0000E+00	.0000E+00
2	.1400E-04	.1404E-04	.4677E-05
3	.0000E+00	.0000E+00	.0000E+00
4	.0000E+00	.0000E+00	.0000E+00

REACTIONS AND APPLIED FORCES

LOAD CONDITION 1 - FORCES "F" AND MOMENTS "M"

JOINT	F(X)	F(Y)	F(Z)
1	-.7890	.0000	-1.3650
2	1.0000	1.0000	1.0000
3	-.2110	.0000	.3650
4	.0000	-1.0000	.0000

TOTAL -.8327E-16 .0000E+00 .0000E+00
SYSTEM

SAP80 V85.03

***** FRAME MEMBER FORCES *****

MEMBERS WITH NUMBERS BETWEEN 1 & 32000

MEM	LOAD	AXIAL	DIST	1-2 PLANE	1-3 PLANE
#	#	FORCE	I	SHEAR	MOMENT SHEAR
1	1	1.58			
2	1	-.42			
3	1	-1.00			

ORNEK 6 : (3 ile 6 arasinda asa91 do9ru 1 ton)

SYSTEM
SAP80 V85.01

***** ECHO OF SAP INPUT DATA *****

TOTAL NUMBER OF JOINTS = 8
TOTAL NUMBER OF LOAD CONDITIONS = 1

PRELIMINARY SCAN OF FRAME DATA

NM=1 Z=-1

SECTION PROPERTY DATA
1 E=28500000 A=0.01

FRAME ELEMENT DATA

- 1,1,2 M=1
- 2,4,5
- 3,7,8
- 4,1,4
- 5,2,5
- 6,4,7
- 7,5,8

8,1,3	2	2.000	.000	.000
9,2,3	5	1.000	1.000	1.000
10,3,4	1	.000	2.000	.000
11,3,5	8	2.000	2.000	.000
12,4,6	6	1.000	3.000	1.000
13,6,5	7	.000	4.000	.000
14,6,7	3	2.000	5.000	.000
15,6,8	4	.000	6.000	.000
16,3,6	785.03			

RESTRAINT INFORMATION

1,8,1		R=0,0,0,1,1,1
LOAD CONDITION		R=1,1,1,1,1,1
2 JOINT		R=1,1,1,1,1,1
7 1		R=1,1,1,1,1,1
8 2		R=1,1,1,1,1,1

EQUILIBRIUM EQUATION NUMBERS
(ONE EQUATION FOR EACH UNKNOWN DISPLACEMENT)

JOINT #	U(X)	U(Y)	U(Z)	R(X)	R(Y)	R(Z)
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	10	11	12	0	0	0
4	4	5	6	0	0	0
5	7	8	9	0	0	0
6	1	2	3	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0

INPUT JOINT DATA

1	3	X=0.0	Y=0.0	Z=0.0
2	6	X=2.0	Y=0.0	Z=0.0
3	7	X=1.0	Y=1.0	Z=1.0
4	9	X=0.0	Y=2.0	Z=0.0
5		X=2.0	Y=2.0	Z=0.0
6	TAL	X=1.0	Y=3.0	Z=1.0
7	STEM	X=0.0	Y=4.0	Z=0.0
8	W85.03	X=2.0	Y=4.0	Z=0.0

GENERATED JOINT COORDINATES

JOINT #	X	Y	Z
1	.000	.000	.000

MEMBER	2	WITH	2.000	.000	.000
	3		1.000	1.000	1.000
MEM	4	AXIAL	.000	2.000	.000
	5		2.000	2.000	.000
	6		1.000	3.000	1.000
MOMENT	7	TORQUE	.000	4.000	.000
	8		2.000	4.000	.000

SYSTEM
SAP80 V85.03

***** JOINT DISPLACEMENTS *****

LOAD CONDITION 1 - DISPLACEMENTS "U" AND ROTATIONS "R"

JOINT	U(X)	U(Y)	U(Z)
1	.0000E+00	.0000E+00	.0000E+00
2	.0000E+00	.0000E+00	.0000E+00
3	.2511E-20	.1053E-04	-.2876E-04
4	.5787E-05	.2163E-21	-.5419E-04
5	-.1231E-05	-.2584E-21	-.4963E-04
6	.1906E-20	-.1053E-04	-.2876E-04
7	.0000E+00	.0000E+00	.0000E+00
8	.0000E+00	.0000E+00	.0000E+00

REACTIONS AND APPLIED FORCES

LOAD CONDITION 1 - FORCES "F" AND MOMENTS "M"

JOINT	F(X)	F(Y)	F(Z)
1	1.0000	1.0000	1.0000
2	-1.0000	1.0000	1.0000
3	.0000	.0000	-1.0000
4	.0000	.0000	-1.0000
5	.0000	.0000	-1.0000
6	.0000	.0000	-1.0000
7	1.0000	-1.0000	1.0000
8	-1.0000	-1.0000	1.0000

TOTAL .0000E+00 -.8882E-15 .4441E-15

SYSTEM
SAP80 V85.03

***** FRAME MEMBER FORCES *****

MEMBERS WITH NUMBERS BETWEEN 1 & 32000

PRELIMINARY SCAN OF FRAME DATA

MEM	LOAD	AXIAL	DIST	1-2 PLANE	1-3 PLANE
#	#	FORCE	I	SHEAR	MOMENT SHEAR
1	1	-----			
2	1	.00			
3	1	-1.00			
4	1	.00			
5	1	.00			
6	1	.00			
7	1	.00			
8	1	.00			
9	1	.00			
10	1	.00			
11	1	.00			
12	1	.00			
13	1	-1.73			
14	1	-1.73			
15	1	.87			
16	1	.87			
17	1	.87			
18	1	.87			
19	1	.87			
20	1	-1.73			
21	1	-1.73			
22	1	-3.00			

ORNEK 6 : (3 ile 6 noktasında aşağı doğru 1 ton)

SYSTEM
SAP80 V85.01

***** ECHO OF SAP INPUT DATA *****

TOTAL NUMBER OF JOINTS = 8
TOTAL NUMBER OF LOAD CONDITIONS = 1

INPUT JOINT DATA
PRELIMINARY SCAN OF FRAME DATA

1 X=0.0 Y=0.0 Z=0.0
NM=1 Z=-1 X=2.0 Y=0.0 Z=0.0
2 X=1.0 Y=1.0 Z=1.0

SECTION PROPERTY DATA
1 E=28500000 A=0.01

FRAME ELEMENT DATA
1,1,2 M=1 X=2.0 Y=4.0 Z=0.0
2,4,5
3,7,8

GENERATED JOINT COORDINATES

	X	Y	Z
1	.000	.000	.000
2	2.000	.000	.000
3	1.000	1.000	1.000
4	.000	2.000	.000
5	2.000	2.000	.000
6	1.000	2.000	1.000
7	.000	4.000	.000
8	2.000	4.000	.000
9	.000	.000	.000
10	.000	.000	.000
11	.000	.000	.000
12	.000	.000	.000
13	.000	.000	.000
14	.000	.000	.000
15	.000	.000	.000
16	.000	.000	.000

RESTRAINT INFORMATION

1,8,1 R=0,0,0,1,1,1
1 R=1,1,1,1,1,1
2 R=1,1,1,1,1,1
7 R=1,1,1,1,1,1
8 R=1,1,1,1,1,1

EQUILIBRIUM EQUATION NUMBERS
(ONE EQUATION FOR EACH UNKNOWN DISPLACEMENT)

JOINT #	U(X)	U(Y)	U(Z)	R(X)	R(Y)	R(Z)
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	10	11	12	0	0	0
4	4	5	6	0	0	0
5	7	8	9	0	0	0
6	1	2	3	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0

INPUT JOINT DATA

JOINT #	X	Y	Z
1	0.0	0.0	0.0
2	2.0	0.0	0.0
3	1.0	1.0	1.0
4	0.0	2.0	0.0
5	2.0	2.0	0.0
6	1.0	3.0	1.0
7	0.0	4.0	0.0
8	2.0	4.0	0.0

GENERATED JOINT COORDINATES

JOINT #	X	Y	Z
1	.000	.000	.000
2	2.000	.000	.000
3	1.000	1.000	1.000
4	.000	2.000	.000
5	2.000	2.000	.000
6	1.000	3.000	1.000
7	.000	4.000	.000
8	2.000	4.000	.000

SYSTEM
SAP80 V85.03

***** JOINT DISPLACEMENTS *****

LOAD CONDITION 1 - DISPLACEMENTS "U" AND ROTATIONS "R"

JOINT	U(X)	U(Y)	U(Z)
1	.0000E+00	.0000E+00	.0000E+00
2	.0000E+00	.0000E+00	.0000E+00
3	.9765E-21	.3509E-05	-.1262E-04
4	.1506E-05	-.3174E-21	-.1764E-04
5	.1506E-05	-.2077E-21	-.1463E-04
6	.5294E-22	-.3509E-05	-.1262E-04
7	.0000E+00	.0000E+00	.0000E+00
8	.0000E+00	.0000E+00	.0000E+00

REACTIONS AND APPLIED FORCES

LOAD CONDITION 1 - FORCES "F" AND MOMENTS "M"

JOINT	F(X)	F(Y)	F(Z)
1	.5000	.5000	.5000
2	-.5000	.5000	.5000

3	.0000	.0000	-1.0000
4	.0000	.0000	.0000
5	.0000	.0000	.0000
6	.0000	.0000	-1.0000
7	.5000	-.5000	.5000
8	-.5000	-.5000	.5000

TOTAL .0000E+00 -.2220E-15 .1110E-15
 SYSTEM
 SAP80 V85.03

 ***** FRAME MEMBER FORCES *****

TOTAL NUMBER OF JOINTS
 TOTAL NUMBER OF LOAD CONDITIONS
 MEMBERS WITH NUMBERS BETWEEN 1 & 32000

MEM	LOAD	AXIAL	DIST	1-2 PLANE	1-3 PLANE
#	#	FORCE	I	SHEAR	MOMENT SHEAR
1	1	.00			
2	1	.00			
3	1	.00			
4	1	.00			
5	1	.00			
6	1	.00			
7	1	.00			
8	1	.00			
9	1	-.87			
10	1	-.87			
11	1	.00			
12	1	.00			
13	1	.00			
14	1	-.87			

```

15 -----
1      -.87
16 -----
1      -1.00

```

ORNEK 7 :

SYSTEM
SAP80 V85.01

```

*****
***** ECHO OF SAP INPUT DATA *****
*****

```

```

TOTAL NUMBER OF JOINTS          = 8
TOTAL NUMBER OF LOAD CONDITIONS = 1

```

INPUT JOINT DATA

PRELIMINARY SCAN OF FRAME DATA

NM=1 Z=-1

```

SECTION PROPERTY DATA
1      E=28500000      A=0.01

```

FRAME ELEMENT DATA

```

1,1,3      M=1
2,2,4
3,5,7      CREATED JOINT COORDINATES
4,6,8
5,1,5
6,2,6      .000      .000      .000
7,3,7      .000      .000      1.000
8,4,8      1.000      .000      .000
9,1,2      1.000      .000      1.000
10,3,4     .000      1.000      .000
11,5,6     .000      1.000      1.000
12,7,8     1.000      1.000      .000
13,1,8     1.000      1.000      1.000
14,2,7
15,3,6     V85.01
16,4,5

```

RESTRAINT INFORMATION

1,8,1 R=0,0,0,1,1,1

1 R=1,1,1,1,1,1
2 R=1,1,1,1,1,1
3 R=1,1,1,1,1,1
4 R=1,1,1,1,1,1

EQUILIBRIUM EQUATION NUMBERS
(ONE EQUATION FOR EACH UNKNOWN DISPLACEMENT)

JOINT #	U(X)	U(Y)	U(Z)	R(X)	R(Y)	R(Z)
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	4	5	6	0	0	0
6	1	2	3	0	0	0
7	7	8	9	0	0	0
8	10	11	12	0	0	0

INPUT JOINT DATA

1	X=0.0	Y=0.0	Z=0.0
2	X=0.0	Y=0.0	Z=1.0
3	X=1.0	Y=0.0	Z=0.0
4	X=1.0	Y=0.0	Z=1.0
5	X=0.0	Y=1.0	Z=0.0
6	X=0.0	Y=1.0	Z=1.0
7	X=1.0	Y=1.0	Z=0.0
8	X=1.0	Y=1.0	Z=1.0

GENERATED JOINT COORDINATES

JOINT #	X	Y	Z
1	.000	.000	.000
2	.000	.000	1.000
3	1.000	.000	.000
4	1.000	.000	1.000
5	.000	1.000	.000
6	.000	1.000	1.000
7	1.000	1.000	.000
8	1.000	1.000	1.000

SYSTEM
SAP80 V85.03

***** JOINT DISPLACEMENTS *****

LOAD CONDITION 1 - DISPLACEMENTS "U" AND ROTATIONS "R"

JOINT	U(X)	U(Y)	U(Z)
1	.0000E+00	.0000E+00	.0000E+00
2	.0000E+00	.0000E+00	.0000E+00
3	.0000E+00	.0000E+00	.0000E+00
4	.0000E+00	.0000E+00	.0000E+00
5	.1130E-05	-.1876E-05	-.1276E-04
6	-.1008E-05	.1632E-05	-.1112E-04
7	-.7461E-06	-.1876E-05	-.1237E-04
8	.6242E-06	.1632E-05	-.1074E-04

REACTIONS AND APPLIED FORCES

LOAD CONDITION 1 - FORCES "F" AND MOMENTS "M"

JOINT	F(X)	F(Y)	F(Z)
1	.4653	1.0000	.4653
2	-.5347	-1.0000	.5347
3	-.4653	1.0000	.4653
4	.5347	-1.0000	.5347
5	.0000	.0000	-1.0000
6	.0000	.0000	.0000
7	.0000	.0000	-1.0000
8	.0000	.0000	.0000

TOTAL -.1110E-15 -.2326E-15 .2110E-15

SYSTEM
SAP80 V85.03

***** FRAME MEMBER FORCES *****

MEMBERS WITH NUMBERS BETWEEN 1 & 32000

MEM	LOAD	AXIAL	DIST	1-2 PLANE	1-3 PLANE
#	#	FORCE	I	SHEAR	MOMENT SHEAR
MOMENT	TORQUE				
1	-----				
1		.00			
2	-----				
1		.00			
3	-----				
1		-.53			
4	-----				
1		.47			

5	-----	
1		-.53
6	-----	
1		.47
7	-----	
1		-.53
8	-----	
1		.47
9	-----	
1		.00
10	-----	
1		.00
11	-----	
1		.47
12	-----	
1		.47
13	-----	
1		-.81
14	-----	
1		.93
15	-----	
1		-.81
16	-----	
1		.93

olması da ihtimal S O N U Ç Çünkü, a matrisi belli
11 bir mantık sıralaması içinde 1 her bir

Örneklerdeki çubuk kuvvetlerinde basınç (-) çekme (+) olarak çıkmaktadır. Mesnet reaksiyonları ise seçilen yönde bir reaksiyon varsa (+), aksi yönde bir reaksiyon varsa (-) olarak bulunur.

3 Sonuç değerler referans program ile karşılaştırıldığında bir takım farklar görülmüştür. 5 numaralı örneğin çözümü tamamen doğru olarak yapılmıştır. 2 noktasında alınan 1 tonluk aşağı doğru yük ve ikinci olarak 2 noktasında X Y Z yönlerinde alınan 1'er tonluk yüklerin sisteme etkisi doğru olarak hesaplanabilmiştir.

6 numaralı örneğe gelindiğinde 3 ile 6 noktası arasındaki noktalara aşağı doğru yük uygulandığı durum da referans program ile farklılıklar ortaya çıkmaktadır. Çubuk kuvvetleri tek tek karşılaştırıldığında 1, 2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 16, numaralı çubukların kuvvetleri tamamen uyumludur. sistemin simetri durumları göz önüne alındığında, kendi içlerinde eşittirler. 1 ile 3, 8 ile 9, 14 ile 15, birbirleri ile simetriktir ve değerleri eşittir. Aynı şekilde 4, 5, 6, 7, simetri gereği eşittir ve 0'dır. Ancak 10, 11, 12 ve 13, numaralı çubuklara bakıldığında, referans program ile farklılıklar vardır. Örneğin, 10 numaralı çubuğun hesaplanan değeri 1.0804 ton iken, referans programın verdiği değer 0.8700 ton'dur. 11 numaralı çubuk 0.805 ton olarak bulunmuş, referans programda ise 0.8700 ton'dur. Simetri durumlarından da görüleceği üzere 10, 11, 12, 13, numaralı çubukların kuvvetleri eşit olmalıdır. Referans programda bu eşitlik mevcuttur. Ancak geliştirilen programda bu eşitlik sağlanamamıştır.

Mesnet reaksiyonları arasında karşılaştırma yapıldığında reaksiyonların tamamen aynı olduğu görülür.

Bununla birlikte, yük durumları değiştiği zaman, yani örnek 6'da 3 ve 6 numaralı noktalarda aşağı doğru 1'er tonluk yükler alındığı zaman tüm çubuk kuvvetleri ve mesnet reaksiyonları referans programla aynı olmaktadır.

7 Numaralı örneğin çözümünde de aynı durum ortaya çıkmıştır. Mesnet reaksiyonları tamamen doğru, ancak birtakım çubukların kuvvetleri hatalı olarak bulunmuştur.

Bu hataların sebepleri sırasıyla şunlar olabilir.

a -> Programda bir hata vardır ve a matrisinin tamamı veya bir bölümü yanlış hesaplanıyor olabilir. Ancak, yaptığım incelemelerde programın a matrisini doğru olarak hesapladığını tesbit etmiş bulunmaktayım. Matrisin bir bölümünün yanlış

KAYNAKLAR

- 1 -> ÇUBUK SİSTEMLERİN ELEKTRONİK HESAP MAKİNALARI İLE
ÇÖZÜMÜ Prof. Dr. Semih Tezcan
- 2 -> NUMERICAL AND MATRIX METHODS IN STRUCTURAL
MECHANICS Prof. Chun Wang
- 3 -> YAPI SİSTEMLERİNİN HESABI I, II,
Prof. Adnan Çakiroğlu
- 4 -> MATRICES SCHAUM'S OUTLINE SERIES
Frank Ayres
- 5 -> YILDIZ ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS PROGRAMI MATRİS METODLARI I, II,
DERS NOTLARI Prof. İrdesel Gögüş

ÖZGEÇMİŞ

1968 Yılında Ankara'da doğdu. İlkokul tahsilini Ankara'da tamamladı. Ortaokul ve lise tahsilini İstanbul'da yaptı. 1985 senesinde Yıldız Üniversitesi Mühendislik Fakültesi İnşaat Mühendisliği bölümünü kazandı. 1989 yılında mezun oldu. Aynı yıl aynı üniversitede yüksek lisans çalışmalarına başladı. Halen çalışmaları devam etmektedir.



