İ T Ü

NÜKLEER ENERJİ ENSTİTÜSÜ

YÜKSEK DERECELI SONLU ELEMANLAR YÖNTEMİYLE İ T Ü TRIGA-MARK II REAKTÖRU YETKİNLİK HESAPLARI

(Yüksek lisans tezi)

Tez öğrencisi : HİKMET ÇAĞLAR

Tez yöneticisi : Yrd.Doç.Dr ATİLLA ÖZGENER

OCAK 1985

Ö Z E T

Yapılan tez çalışmasında silindirik geometride tek boyutlu(radyal yönde), çok bölgeli ve çok guruplu difüzyon denklemlerinin sayısal çözümleri için bir bilgisayar kodu yazılmıştır Bilgisayar kodu ile kritikalite hesabı, serbest nötron kaynak problemleri ve nötron akı dağılımı hesabı yapılabilir.

Çok guruplu difüzyon denklemlerinin sayısal çözümleri sonlu elemanlar metoduyla gerçeklenmiştir.

İ.T.Ü TRIGA MARK II reaktörü nötronik hesabı iki bölgeli ve iki guruplu, 10 bölgeli ve iki guruplu eşdeğer homojenleştirme yapılarak yazılan kod yardımıyla gerçeklenmiştir.Yüksek dereceli sonlu elemanlar kullanılarak bilgisayar zamanının azaltılmasına çalışılmıştır.

Kullanılan semböller

σ	Mikroskopik tesir kesiti
$\sum$	Makroskopik tesir kesiti
$S(\vec{r})$	Kaynak terimi
$D_g(\vec{r})$	$\vec{r}$ vektörü dolayındaki g. enerjili gurup difüzyon sabiti.
$\phi_g(\vec{r})$	$\vec{r}$ vektörü dolayındaki g enerji gurup numaralı nötron akısı.
$\sigma_s^{g' \to g}$	g'. uncü enerji gurubundan g. enerjili guruba saçılma tesir kesiti.
σ_{rg}	Çıkartma tesir kesiti
σ_{ag}	g inci gurup absorbsiyon tesir kesiti.
χ_g	Bir fisyon nötronunun g. enerji aralığında doğma olasılığı
ν^g	Herbir fisyondan g.inci gurupta doğan nötron sayısı
F	Fonksiyonel
ν	İterasyon sayısı.
ξ	Lokal koordinatlar bağımsız değişkeni.
k	Çoğaltma katsayısı
$\underline{\underline{A}}^T$	A matrisinin transpozesi.
$k^{(i)}$	Çoğaltma katsayısının i.inci iterasyondaki değeri.

İÇİNDEKİLER

I. SONLU ELEMANLAR

1950 lerde inşaat mühendisliğinin karmaşık gerilme problemlerinin çözümü için bir sayısal yöntem geliştirilmiştir. Bu yöntemde klasik sonlu fark yaklaşımının gerektirdiği dikdörtgensel ızgara yapısı gerekliliği ortadan kalkıyordu. Karmaşık geometrideki sistemlerde üçgensel ızgara yapısı ve doğal ızgara yapısını olanaklı kılıyordu.

Yöntem matematiksel bir analiz sonucu doğmamıştır. İnşaat mühendislerinin kendi problemlerine yönelik sezgisel bir düşünüş tarzı sonucu ortaya çıkmıştır.

Bilgisayarların gelişmesine bağlı olarak büyük boyutlarda mühendislik problemleri bu yöntemle çözülmüştür.

Bu arada ısı transferi, akışkanlar mekaniği, nötron difüzyon transport analizinde bu yöntem kullanılmıştır.

Daha sonraları bu yöntemin gerçekte varyasyonel problemlerin çözümü için önerilen klasik Ritz metodunun bir uygulaması olduğu anlaşıldı.

Kendine ek(Self adjoint) bir lineer diferansiyel denklemin belirli sınır koşulları altında çözümü olan fonksiyon, aynı zamanda bir fonksiyonelin minumum değerini sağlayan fonksiyondur. Varyasyonel matematiğin bir prensibine göre uygulamalı bilim dallarında rastladığımız kısmi diferansiyel denklem - lerin direkt çözümü yerine varyasyonel bir prensipten har - eketle bir fonksiyonelin minumumunu bulabiliriz.Bu minumum aradığımız çözüm olacaktır.Eğer diferansiyel denklemin ana-litik çözümü bulunamıyorsa, fonksiyonelin minumumu da anal-itik olarak bulunamaz.

Analitik çözümün olanaksız olması durumunda: tek seçenek sayısal yöntemlere başvurmaktır.Taylor serileri kullanılarak belirli ızgara yapısı içinde problemi lineer sistem çözümüne indirgemek sonlu fark yöntemini oluşturur.Bu yöntemin temel eksikleri:

a) Dikdörtgensel ızgara yapısı ve bunun açtığı zorluklar.

b) Yüksek mertebe doğruluklara ulaşmada rastlanan güçlükler.

Yukarıda belirtilen zorluklar bazı durumlarda sonlu fark yak-laşımıyle çözüm yapmayı olanaksız kılmakta, bazı hallerde ise bilgisayar süresini uzatmaktadır.

İzlenebilecek diğer bir yolda diferansiyel denkleme say-ısal yaklaşım arama yerine fonksiyonelin minumuna sayısal bir yaklaşım aramaktır.Sonsuz boyutlu sürekli fonksiyonlar uzayının sonlu boyutlu bir alt uzayında fonksiyonelin min-umumunu sağlayan fonksiyon aranır.Bulunan fonksiyon minum-umu sağlayan yaklaşık bir fonksiyon olacaktır.Fakat yapıl-acak yaklaşımın mertebesini öngörebiliyorsak sorun çıkmaz. Fonksiyonelin minumumunu aramak bilinen Ritz metodudur.

Sonlu elemanlar yönteminde parçasal polinomlar uzayı seçilerek Ritz yönteminin sayısal analize en uygun varyantı oluşturulur. Dikdörtgen ızgara zorunluluğu ortadan kalkmakta ve istenilen doğruluk mertebesine ulaşılabilmekte.

Sonlu elemanlar yönteminin getirdiği en önemli yenilik olarak dikdörtgensel ızgara zorunluğunun ortadan kalkışı gösterilebilir. Geometrik modellemeyi olanaksız kılan durumlarda üçgenlerden oluşan ızgara, izo-parametrik eleman kavramına geçilerek kenarları konik olanızgaralar oluşturulabilir. Diğer bir avantaj ise yüksek mertebeli polinomlar kullanarak istenilen doğruluk mertebesine ulaşabilmemiz.

Nükleer sahada, nükleer reaktör hücresinin geometrisi modellenebilmekte. Reaktör eşdeğer homojenleştirme sonunda geometrisi belirlenebilmekte. Bu konuda tek sorun bilgisayar süresini azaltmaktır.

Sonuçta sonlu elemanlar yönteminin bilgisayar-etkin bir yöntem olduğunu söyleyebiliriz.

II. SONLU ELEMANLAR YÖNTEMİ VE DİFÜZYON DENKLEMİNİN ÇÖZÜMÜ

Lineer self-adjoint problemlerde, varyasyonel sonlu elemanlar yöntemi kullanılır. Sistemin diferansiyel denkleminin çözümünü bulmak için fonksiyonun kararlı değerinin bulunması şartı kullanılır.

Reaktör fizik hesaplarında problemimiz self-adjoint olduğundan varyasyonel sonlu elemanlar yöntemi kullanılabilir.

Difüzyon denklemi,

$$-\vec{\nabla} D(\vec{r})\vec{\nabla}\phi(\vec{r}) + \sigma(\vec{r})\phi(\vec{r}) = S(\vec{r}) \qquad \vec{r} \in V \qquad (II.1)$$

Sınır koşuları,

$$\vec{n}.D(\vec{r})\vec{\nabla}\phi(\vec{r}) + \frac{1}{2}\phi(\vec{r}) = 0 \qquad \vec{r} \in \Gamma_v \qquad (II.2)$$

$$\vec{n}.D(\vec{r})\vec{\nabla}\phi(\vec{r}) = 0 \qquad \vec{r} \in \Gamma_r \qquad (II.3)$$

Γ_r yansıtıcı dış yüzey

Γ_v boşluk dış yüzeyi

Difüzyon denkleminin fonksiyoneli:

$$F\left[\psi(\vec{r})\right] = \int_V \left[D(\vec{r})(\vec{\nabla}\psi(\vec{r}))^2 + \sigma(\vec{r})\psi^2(\vec{r}) - 2\psi(\vec{r})S(\vec{r})\right] dV + \frac{1}{2}\int_{\Gamma_v} \psi^2(\vec{r}) \qquad (II$$

Difüzyon denkleminin fonksiyonelinin minumumunu sağlayan fonksiyon ayni zamanda difüzyon denklemini sınır koşularıyle sağlayan fonksiyonla eşdeğerdir.Difüzyon denkleminin çözümü ile fonksiyonelin minumumunu arama probleminin eşdeğer olduğu /1/ numaralı kaynakta gösterilmiştir.

RİTZ METODU

Difüzyon denkleminin çözümü varyasyonel metotla (II.4) deki varyasyonelin minumumunu sağlayan $\psi(r)$ fonksiyonunu arama problemine dönüştürülmüştü.

Varyasyonel formdaki (II.4) denklemi sonsuz boyutlu fonksiyonlar uzayında (H^1) sürekli, türevleri parçasal sürekli fonksiyonlar uzayında minimize edilebilir.(H^1) uzayında fonksiyonelin minumumunu aramak zor ve genellikle imkansızdır.Ayrıca minumumu sağlayacak fonksiyondan elde edilecek sonsuz boyutlu lineer denklem sistemini çözmek olanak dışıdır.

Ritz metodu minumum arama işlemine yaklaşık çözüm getiren bir yöntemdir.

Bu yöntemde sonsuz boyutlu fonksiyonlar uzayında fonksiyonelin minumumunu aramak yerine bu uzayın sonlu boyutlu bir alt uzayında fonksiyonelin minumumunu sağlayacak fonksiyon aranır. Bu yaklaşım şüphesiz aradığımız minumum fonksiyonu bize vermeyecektir.Fakat bizim hedefimiz difüzyon denklemine yaklaşık çözüm getirmek ve sistemin davranışları hakkında bilgi sahibi olmaktır.

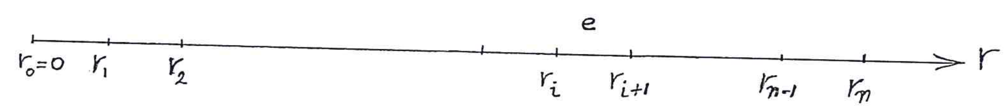
Ritz yönteminde seçilen alt uzayda baz oluşturan sonlu sayıda fonksiyon seçilir.Bu fonksiyonlar deneme fonksiyonu olarak adlandırılır.

Meydana gelen lineer denklem sisteminin boyutu kullanılan alt uzayın boyutuna eşittir.

Ritz yöntemini açıklayan bir örnek /1/ numaralı kaynakta 11. sayfada verilmiştir.

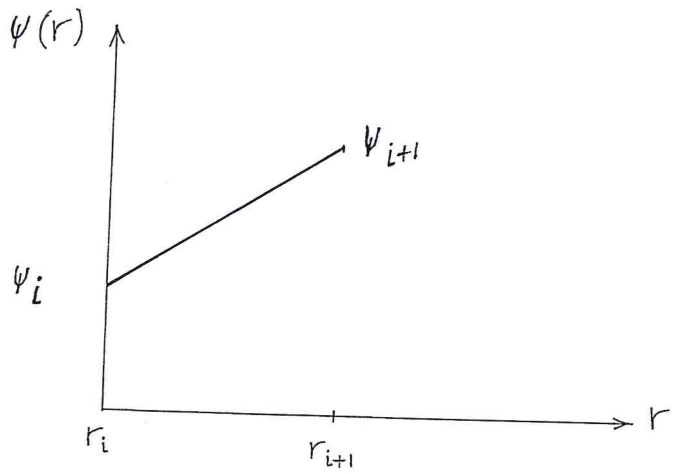
II.1. SONLU ELEMANLAR YÖNTEMİNDE DENEME FONKSİYONLARI

Tek boyutlu r uzunluğunda bir sistemi n parçaya bölelim. Bu parçaları eleman olarak tanımlayalım.



$r_0, r_1, \ldots, r_n$ noktalarına nod adı verilir.

e numaralı eleman içinde deneme fonksiyonlarını belirleyelim,

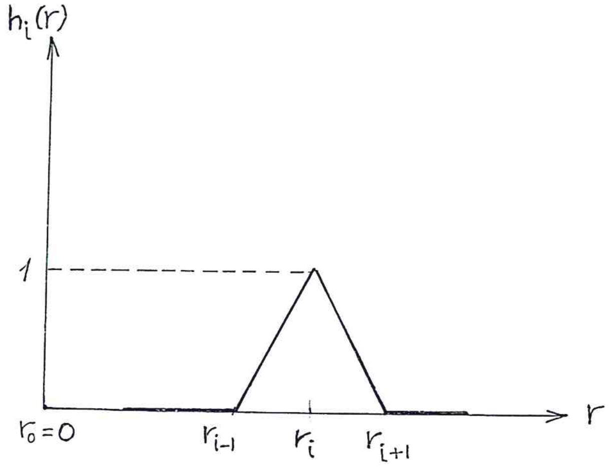


Fonksiyonumuz lineer olduğundan lagrange polinomu $\psi(r)$

$$\psi(r) = \frac{r - r_{i+1}}{r_i - r_{i+1}} \psi_i + \frac{r - r_i}{r_{i+1} - r_i} \psi_{i+1} \qquad r_{i+1} \geqslant r \geqslant r_i$$

şeklinde yazılır.

Lagrange polinomu ile çalışmak işlem zorlukları getirdiğinden deneme fonksiyonları ile çalışmak uygun olacaktır.

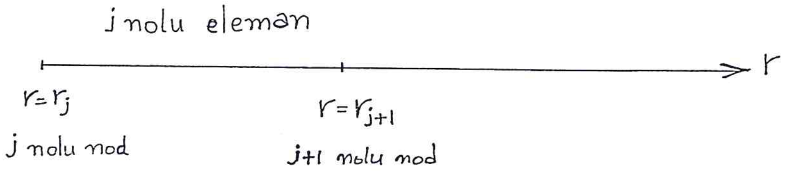


Fonksiyonlar her elemanda lineer ve nodlarda sürekli olmak üzere deneme fonksiyonları aşağıdaki şekilde yazılır.

$$h_i(r)=\begin{cases} 0 & r>r_{i+1} \\[2ex] \dfrac{r-r_{i+1}}{r_i-r_{i+1}} & r_i<r<r_{i+1} \\[2ex] \dfrac{r-r_{i-1}}{r_i-r_{i-1}} & r_{i-1}<r<r_i \\[2ex] 0 & r<r_{i-1} \end{cases}$$

II.2. TEK BOYUTLU SİLİNDİRİK GEOMETRİ VE LİNEER SONLU ELEMANLAR.

Tek boyutlu R uzunluğundaki bir sistemi n elemana bölelim. Bu sistemde deneme fonksiyonları eleman nodlarında süreklilik şartını sağlamaktadır.



(II.4) deki fonksiyonelimizi silindirik geometride yazalım.

$$F\left[\psi(r)\right] = \int_0^{R_J} \left[D(r)\left(\frac{\partial \psi}{\partial r}\right)^2 + \sigma(r)\psi^2(r) - 2\psi(r)\,S(r) \right] r\,dr + \frac{1}{2}\, R\psi^2(r$$

fonksiyonelin eleman içindeki değeri,

$$F_j\left[\psi(r)\right] = \int_{r_j}^{r_{j+1}} \left[D_j\left(\frac{\partial \psi}{\partial r}\right)^2 + \sigma_j\psi^2(r) - 2\psi(r)S(r) \right] r\,dr + \frac{1}{2}\,\delta_{j,J-1}\, R_{J,V}\psi^2(R_J)$$

$$R_{J,V} = \begin{cases} R_J & \text{boşluk sınır koşulu} \\ 0 & \text{yansıtıcı sınır koşulu} \end{cases}$$

$$r = r_j + \Delta_j \xi \qquad \Delta_j = r_{j+1} - r_j$$

$$\xi = \frac{r - r_j}{\Delta_j} \qquad d\xi = \frac{dr}{\Delta_j}$$

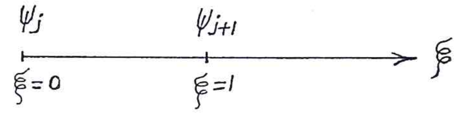
dönüşümü yapılarak lokal koordinatlarda fonksiyonel aşağıdaki şekilde yazılabilir.

Lokal koordinatlarda deneme fonksiyonlarımız,

$$h_1 = 1 - \xi$$

$$h_2 = \xi$$

$\psi(\xi)$ fonksiyonu deneme fonksiyonlarının lineer kombinozonu olarak ifade edilebilir.



$\underline{\psi}_j$ j inci nodda $\underline{\psi}(\xi)$ fonksiyonunun aldığı değer

ψ_{j+1} j+1 inci nodda $\psi(\xi)$ fonksiyonunun aldığı değer

$$\underline{\psi}(\xi) = (1 - \xi)\,\psi_j + \xi\,\psi_{j+1}$$

$$\underline{\psi}(\xi) = \begin{bmatrix} h_1 & h_2 \end{bmatrix} \begin{bmatrix} \psi_j \\ \psi_{j+1} \end{bmatrix}$$

$$\underline{f}(\xi) = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix}$$

$$\underline{\psi}(\xi) = \underline{\phi}_j^T \, \underline{f}(\xi) = \underline{f}^T(\xi) \, \underline{\phi}_j$$

$$F\left[\underline{\psi}(\xi)\right] = \int_0^1 \Big[r_j \frac{D_j}{\Delta_j} \underline{\phi}_j^T \, \underline{f}'(\xi) \, \underline{f}'^T(\xi) \, \underline{\phi}_j + \Delta_j \, r_j \, \sigma_j \, \underline{\phi}_j^T \, \underline{f}(\xi) \, \underline{f}^T(\xi) \, \underline{\phi}_j$$

$$-2 \Delta_j \, r_j \, \underline{\phi}_j^T \, \underline{f}(\xi) \, \underline{S}(\xi) \Big] d\xi + \int_0^1 \Big[\xi D_j \, \underline{\phi}_j^T \, \underline{f}'(\xi) \, \underline{f}'^T(\xi) \, \underline{\phi}_j$$

$$\Delta_j^2 \, \xi \, \sigma_j \, \underline{\phi}_j^T \, f(\xi) \, \underline{f}^T(\xi) \, \underline{\phi}_j - 2 \Delta_j^2 \, \xi \, \underline{\phi}_j^T \, \underline{f}(\xi) \, \underline{S}(\xi) \Big] d\xi$$

$$+ \frac{1}{2} \, \delta_{j,J-1} \, R_{J,V} \, \psi^2(R_{J,V}) \qquad \text{(II.2.1)}$$

Son bulunan fonksiyonelde gerekli işlemler yapılarak j numaralı eleman matrisi ve kaynak vektörü aşağıdaki şekilde bulunur.

Lineer sonlu elemanlar j numaralı eleman matrisi.

$$\underline{\underline{A}}_j = \begin{bmatrix} \dfrac{D_j r_j}{\Delta_j} + \dfrac{D_j}{2} + \dfrac{\Delta_j r_j \sigma_j}{3} + \dfrac{\Delta_j^2 \sigma_j}{12} & \dfrac{-D_j r_j}{\Delta_j} + \dfrac{-D_j}{2} + \dfrac{\Delta_j r_j \sigma_j}{6} + \dfrac{\Delta_j \sigma_j}{12} \\ \\ \dfrac{-D_j r_j}{\Delta_j} + \dfrac{-D_j}{2} + \dfrac{\Delta_j r_j \sigma_j}{6} + \dfrac{\Delta_j^2 \sigma_j}{12} & \dfrac{D_j r_j}{\Delta_j} + \dfrac{D_j}{2} + \dfrac{\Delta_j r_j \sigma_j}{3} + \dfrac{\Delta_j \sigma_j}{4} + \dots \end{bmatrix}$$

Yığılmış kaynak yaklaşımı yapılarak,

$$\underline{S}(\xi) = \sigma \underline{\emptyset}(\xi)$$

Lineer sonlu elemanlar j numaralı kaynak vektörü.

$$\underline{S}_j = \begin{bmatrix} \dfrac{r_j \Delta_j \sigma_j}{2} + \dfrac{\Delta_j^2 \sigma_j}{6} & 0 \\ \\ 0 & \dfrac{r_j \Delta_j \sigma_j}{2} + \dfrac{\Delta_j^2 \sigma_j}{3} \end{bmatrix} \begin{bmatrix} \emptyset_j \\ \\ \emptyset_{j+1} \end{bmatrix}$$

Global sisteme "Boole transformasyon matrisi" kullanılarak geçili

$$
\underline{\underline{\exists}}_j^T = \begin{array}{c} \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{array}\right] \end{array} \begin{array}{l} \\ \\ \\ \leftarrow j \\ \leftarrow j+1 \\ \\ \\ \\ \end{array}
$$

$$\underline{\underline{\tilde{A}}}_j = \underline{\underline{\exists}}_j^T \, \underline{\underline{A}}_j \, \underline{\underline{\exists}}_j \qquad \underline{\underline{A}} = \sum_{j=1}^{J-1} \underline{\underline{\tilde{A}}}_j$$

$$\underline{\tilde{S}}_j = \underline{\underline{\exists}}_j^T \underline{S}_j \qquad \underline{S} = \sum_{j=1}^{J-1} \underline{\tilde{S}}_j$$

$$F_j[\underline{\psi}] = \underline{\psi}^T \underline{\underline{A}}_j \underline{\psi} - 2 \underline{\psi}^T \underline{S}_j$$

$$\underline{\phi}_j = \underline{\underline{\exists}}_j \underline{\psi} \qquad F[\underline{\psi}] = \sum_{j=1}^{J-1} F_j[\underline{\psi}]$$

A matrisi pozitif kesin bir matristir.Dolayısıyle $\underline{\underline{A}} = \underline{\underline{A}}^T$ yazılabil

Fonksiyonelimizi minumum kılan $\underline{\phi}$ fonksiyonunu arayalım.

$$\underline{\psi} = \underline{\phi} + \epsilon \underline{\delta}$$

$$F[\underline{\psi}] = \underline{\psi}^T \underline{\underline{A}} \underline{\psi} - 2 \underline{\psi}^T \underline{S}$$

$$F[\underline{\psi}] = \left[\underline{\phi}^T + \epsilon \underline{\delta}^T\right] \underline{\underline{A}} \left[\underline{\phi} + \epsilon \underline{\delta}\right] - 2\left[\underline{\phi}^T + \epsilon \underline{\delta}^T\right] \underline{S}$$

$$F[\underline{\psi}] = \overbrace{\left[\underline{\phi}^T \underline{\underline{A}}\, \underline{\phi} - 2\, \underline{\phi}^T \underline{S}\right]}^{F[\underline{\phi}]} + \epsilon \overbrace{\left[\underline{\delta}^T \underline{\underline{A}}\, \underline{\phi} + \underline{\phi}^T \underline{\underline{A}}\, \underline{\delta} - 2\, \underline{\delta}^T \underline{S}\right]}^{\delta F[\underline{\phi}]}$$

$$+ \epsilon^2 \underbrace{\left[\underline{\delta}^T \underline{\underline{A}}\, \underline{\delta}\right]}_{\delta^2 F[\underline{\phi}]}$$

$\delta^2 F[\underline{\phi}]$ ikinci varyasyonel sıfırdan büyük olduğundan $F[\underline{\psi}] \geq F[\underline{\phi}]$ şartının sağlanabilmesi için birinci varyasyonelin sıfır olması gerekir.

$$\underline{\delta}^T \underline{\underline{A}}\, \underline{\phi} + \underline{\phi}^T \cdot \underline{\underline{A}}\underline{\delta} - 2\, \underline{\delta}^T \underline{S} = 0$$

$\underline{\underline{A}} = \underline{\underline{A}}^T$ olduğundan dolayı

$$2\, \underline{\delta}^T (\underline{\underline{A}}\, \underline{\phi} - \underline{S}) = 0$$

$$\underline{\underline{A}}\, \underline{\phi} = \underline{S}$$

lineer denklem sistemi elde edilir.

Bu denklem sisteminin çözümü sonucu $\underline{\phi}$ bulunur.

II.3. TEK BOYUTLU SİLİNDİRİK GEOMETRİ VE KUADRATİK SONLU ELEMANLAR

Eleman içinde akı dağılımı ikinci derece fonksiyon ile gösterilmekte ve fonksiyon nodlarda süreklilik şartını sağlamakta.

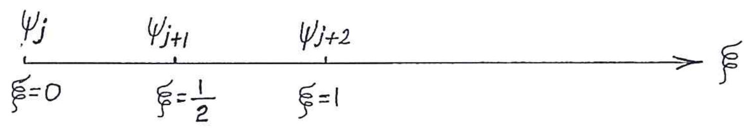
Kuadratik deneme fonksiyonlarımız.

$$h_1 = (1 - 2\xi)(1 - \xi)$$

$$h_2 = 4\xi(1 - \xi)$$

$$h_3 = \xi(2\xi - 1)$$

$\psi(\xi)$ fonksiyonu deneme fonksiyonlarının lineer kombinozonu olarak ifade edilebilir.



$$\underline{\psi}(\xi) = (1 - 2\xi)(1 - \xi)\psi_j + (1 - \xi)4\xi\psi_{j+1} + \xi(2\xi - 1)\psi_{j+2}$$

$$f(\xi) = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}$$

$$\underline{\psi}(\xi) = \underline{f}^T(\xi)\underline{\emptyset}_j$$

fonksiyonu (II.2.1) de yerine yazılarak $\underline{\underline{A}}_j$ ve $\underline{S}_j$ belirlenir.

Kuadratik sonlu elemanlar j numaralı eleman matrisi.

$$
\underline{\underline{A}}_j = \frac{D_j r_j}{3\Delta_j}
\begin{bmatrix} 7 & -8 & 1 \\ -8 & 16 & -8 \\ 1 & -8 & 7 \end{bmatrix}
+ \frac{D_j}{6}
\begin{bmatrix} 3 & -4 & 1 \\ -4 & 16 & -12 \\ 1 & -12 & 11 \end{bmatrix}
+ \frac{r_j \Delta_j \zeta_j}{30}
\begin{bmatrix} 4 & 2 & -1 \\ 2 & 16 & 2 \\ -1 & 2 & 4 \end{bmatrix}
+ \frac{\Delta_j^2 \zeta_j}{60}
\begin{bmatrix} 1 & 0 & -1 \\ 0 & 16 & 4 \\ -1 & 4 & 7 \end{bmatrix}
$$

Kuadratik sonlu elemanlar j numaralı kaynak vektörü.

$$\underline{S}_j = \begin{bmatrix} \dfrac{r_j \Delta_j \sigma_j}{6} & 0 & 0 \\ 0 & \dfrac{2r_j \Delta_j \sigma_j}{3} + \dfrac{\Delta_j^2 \sigma_j}{3} & 0 \\ 0 & 0 & \dfrac{r_j \Delta_j \sigma_j}{6} + \dfrac{\Delta_j^2 \sigma_j}{6} \end{bmatrix} \begin{bmatrix} \phi_j \\ \phi_{j+1} \\ \phi_{j+2} \end{bmatrix}$$

II.4. TEK BOYUTLU SİLİNDİRİK GEOMETRİ VE KÜBİK SONLU ELEMANLAR

Eleman içinde akı dağılımı üçüncü derece fonksiyon ile gösterilmekte ve fonksiyon nodlarda süreklilik şartını sağlamakta.

Kübik deneme fonksiyonlarımız.

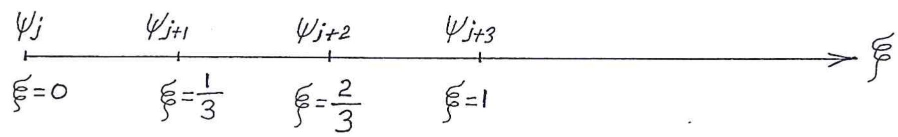
$$h_1 = \frac{1}{2}(1 - 3\xi)(2 - 3\xi)(1 - \xi)$$

$$h_2 = \frac{9}{2}\xi(2 - 3\xi)(1 - \xi)$$

$$h_3 = \frac{9}{2}\xi(3\xi - 1)(1 - \xi)$$

$$h_4 = \frac{1}{2}\xi(3\xi - 1)(3\xi - 2)$$

$\Psi(\xi)$ fonksiyonu deneme fonksiyonlarının lineer kombinozonu olarak ifade edilebilir.



$$\underline{\Psi}(\xi) = \frac{1}{2}(1 - 3\xi)(2 - 3\xi)(1 - \xi)\,\Psi_j + \frac{9}{2}\xi(2 - 3\xi)(1 - \xi)\,\Psi_{j+1}$$

$$+ \frac{9}{2}\xi(3\xi - 1)(1 - \xi)\,\Psi_{j+2} + \frac{1}{2}\xi(3\xi - 1)(3\xi - 2)\,\Psi_{j+3}$$

$$\underline{f}(\xi) = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix}$$

$$\underline{\Psi}(\xi) = \underline{f}^T(\xi)\underline{\emptyset}_j$$

fonksiyonu (II.2.1) de yerine yazılarak $\underline{\underline{A}}_j$ ve $\underline{S}_j$ belirlenir.

Kübik sonlu elemanlar j numaralı eleman matrisi.

$$\underline{\underline{A}}_j = \frac{D_j r_j}{40\Delta_j}\begin{bmatrix} 148 & -189 & 54 & -13 \\ -189 & 432 & -297 & 54 \\ 54 & -297 & 432 & -189 \\ -13 & 54 & -189 & 148 \end{bmatrix}$$

$$+\frac{D_j}{80}\begin{bmatrix} 34 & -51 & 30 & -13 \\ -51 & 270 & -297 & 78 \\ 30 & -297 & 594 & -327 \\ -13 & 78 & -327 & 262 \end{bmatrix}$$

$$+\frac{r_j\Delta_j\zeta_j}{1680}\begin{bmatrix} 128 & 99 & -36 & 19 \\ 99 & 648 & -81 & -36 \\ -36 & -81 & 648 & 99 \\ 19 & -36 & 99 & 128 \end{bmatrix}$$

$$+\frac{\Delta_j^2\zeta_j}{3360}\begin{bmatrix} 19 & 9 & 9 & 19 \\ 9 & 405 & -81 & -81 \\ 9 & -81 & 891 & 189 \\ 19 & -81 & 189 & 237 \end{bmatrix}$$

Kübik sonlu elemanlar j numaralı kaynak vektörü.

$$\underline{S}_j = \begin{bmatrix} \frac{r_j \Delta_j \sigma_j}{8} + \frac{\Delta_j^2 \sigma_j}{60} & 0 & 0 & 0 \\ 0 & \frac{3 r_j \Delta_j \sigma_j}{8} + \frac{3 \Delta_j^2 \sigma_j}{40} & 0 & 0 \\ 0 & 0 & \frac{3 r_j \Delta_j \sigma_j}{8} + \frac{3 \Delta_j^2 \sigma_j}{10} & 0 \\ 0 & 0 & 0 & \frac{r_j \Delta_j \sigma_j}{8} + \frac{13 \Delta_j^2 \sigma_j}{120} \end{bmatrix} \begin{bmatrix} \phi_j \\ \phi_{j+1} \\ \phi_{j+2} \\ \phi_{j+3} \end{bmatrix}$$

II.5. TEK BOYUTLU SİLİNDİRİK GEOMETRİ VE KUARTİK SONLU ELEMANLAR

Eleman içinde akı dağılımı dördüncü derece fonksiyon ile gösterilmekte ve fonksiyon nodlarda süreklilik şartını sağlamakta.

$$h_1 = \frac{1}{3}(1 - 4\xi)(1 - 2\xi)(3 - 4\xi)(1 - \xi)$$

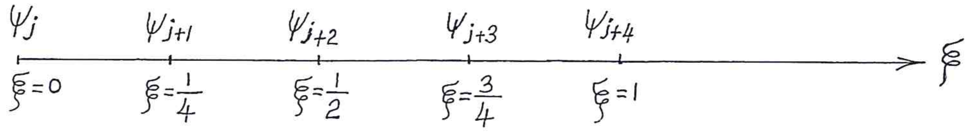
$$h_2 = \frac{16}{3}\,\xi(1 - 2\xi)(3 - 4\xi)(1 - \xi)$$

$$h_3 = -4\,\xi(1 - 4\xi)(3 - 4\xi)(1 - \xi)$$

$$h_4 = \frac{16}{3}\,\xi(1 - 4\xi)(1 - 2\xi)(1 - \xi)$$

$$h_5 = -\frac{1}{3}\,\xi(1 - 4\xi)(1 - 2\xi)(3 - 4\xi)$$

$\underline{\psi}(\xi)$ fonksiyonu deneme fonksiyonlarının lineer kombinozonu olarak ifade edilebilir.



$$\underline{\psi}(\xi) = \frac{1}{3}(1-4\xi)(1-2\xi)(3-4\xi)(1-\xi)\,\psi_j + \frac{16}{3}\,\xi(1-2\xi)(3-4\xi)(1-\xi)\,\psi_{j+1}$$

$$- \xi(1-4\xi)(3-4\xi)(1-\xi)\,\psi_{j+2} + \frac{16}{3}\,\xi(1-4\xi)(1-2\xi)(1-\xi)\,\psi_{j+3}$$

$$-\frac{1}{3}\,\xi(1-4\xi)(1-2\xi)(3-4\xi)\,\psi_{j+4}$$

$$\underline{f}(\xi) = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{bmatrix}$$

$$\underline{\psi}(\xi) = \underline{f}^T(\xi)\underline{\emptyset}_j$$

fonksiyonu (II.2.1) de yerine yazılarak $\underline{\underline{A}}_j$ ve $\underline{S}_j$ belirlenir.

4 derece sonlu elemanlar j numaralı eleman matrisi.

$$
\underline{\underline{A}}_j = \frac{D_j r_j}{945\,\Delta_j}
\begin{bmatrix}
4925 & -6848 & 3048 & -1472 & 347 \\
-6848 & 16640 & -14208 & 5888 & -1472 \\
3048 & -14208 & 22320 & -14208 & 3048 \\
-1472 & 5888 & -14208 & 16640 & -6848 \\
347 & -1472 & 3048 & -6848 & 4925
\end{bmatrix}
$$

$$
+ \frac{D_j}{1890}
\begin{bmatrix}
705 & -992 & 804 & -864 & 347 \\
-992 & 6400 & -9216 & 5888 & -2080 \\
804 & -9216 & 22320 & -19200 & 5292 \\
-864 & 5888 & -19200 & 26880 & -12704 \\
347 & -2080 & 5292 & -12704 & 9145
\end{bmatrix}
$$

$$
+ \frac{r_j \Delta_j \zeta_j}{11340}
\begin{bmatrix}
584 & 592 & -348 & 112 & -58 \\
592 & 3584 & -768 & 512 & 112 \\
-348 & -768 & 3744 & -768 & -348 \\
112 & 512 & -768 & 3584 & 592 \\
-58 & 112 & -348 & 592 & 584
\end{bmatrix}
$$

$$
+ \frac{\Delta_j^2 \zeta_j}{11340}
\begin{bmatrix}
29 & 32 & 0 & -32 & -29 \\
32 & 768 & -192 & 256 & 144 \\
0 & -192 & 1872 & -576 & -348 \\
-32 & 256 & -576 & 2816 & 560 \\
-29 & 144 & -348 & 560 & 555
\end{bmatrix}
$$

$$
\underline{S}_j = \begin{bmatrix}
\frac{7r_j \Delta_j \zeta_j}{90} & 0 & 0 & 0 & 0 \\
0 & \frac{16r_j \Delta_j \zeta_j}{45} + \frac{4 \Delta_j^2 \zeta_j}{45} & 0 & 0 & 0 \\
0 & 0 & \frac{2r_j \Delta_j \zeta_j}{15} + \frac{\Delta_j^2 \zeta_j}{15} & 0 & 0 \\
0 & 0 & 0 & \frac{16r_j \Delta_j \zeta_j}{45} + \frac{4 \Delta_j^2 \zeta_j}{15} & 0 \\
0 & 0 & 0 & 0 & \frac{7r_j \Delta_j \zeta_j}{90} + \frac{7 \Delta_j^2 \zeta_j}{90}
\end{bmatrix}
\begin{bmatrix}
\phi_j \\ \phi_{j+1} \\ \phi_{j+2} \\ \phi_{j+3} \\ \phi_{j+4}
\end{bmatrix}
$$

4 derece elemanlar j numaralı kaynak vektörü

III. ÇOK GURUPLU DİFÜZYON DENKLEMLERİNİN SAYISAL ÇÖZÜMÜ

Çok guruplu difüzyon denklemlerini yazarak, denklemleri bilgisayarda çözebilecek hale getirelim.

$$-\vec{\nabla} D_1 \vec{\nabla} \phi_1 + \Sigma_{r1} \phi_1 = \frac{1}{k} \chi_1 S$$

$$-\vec{\nabla} D_2 \vec{\nabla} \phi_2 + \Sigma_{r2} \phi_2 = \frac{1}{k} \chi_2 S + \Sigma_{s1,2} \phi_1$$

$$\vdots \qquad\qquad \vdots$$

$$-\vec{\nabla} D_G \vec{\nabla} \phi_G + \Sigma_{rG} \phi_G = \frac{1}{k} \chi_G S + {}_{s1,G} \phi_1 + \cdots + \Sigma_{sG-1,G} \phi_{G-1}$$

Denklemleri yazarken düşük enerji guruplarından yukarı enerji guruplarına saçılma olmadığını varsaydık. Ayrıca fisyon kaynağının konuma bağımlılığı her enerji gurubunda ayni olmakta.

$$S(\vec{r}) = \sum_{g'=1}^{G} \nu_{g'} \Sigma_{fg'} \phi_{g'}(\vec{r})$$

Fisyon kaynağına ve çoğaltma katsayısına başlangıç değeri vererek gurup difüzyon denklemlerini birbirine bağlı olarak çözelim.

$$S(r) \sim \overset{(0)}{S}(r)$$

$$k \sim k^{(0)}$$

$$-\vec{\nabla} D_1 \vec{\nabla} \phi_1^{(1)} + \Sigma_{r1} \phi_1^{(1)} = \frac{1}{k^{(0)}} \cdot \chi_1 \cdot \overset{(0)}{S}(r)$$

$$-\vec{\nabla} D_2 \vec{\nabla} \phi_2^{(1)} + \Sigma_{r2} \phi_2^{(1)} = \frac{1}{k^{(0)}} \cdot \chi_2 \cdot \overset{(0)}{S}(r) + \Sigma_{s1,2} \cdot \phi_1^{(1)}$$

$$\vdots$$

$$-\vec{\nabla} D_G \vec{\nabla} \phi_G^{(1)} + \Sigma_{rG} \phi_G^{(1)} = \frac{1}{k^{(0)}} \cdot \chi_G \cdot \overset{(0)}{S}(r) + \Sigma_{s1,G} \cdot \phi_1^{(1)} \cdots\cdots \Sigma_{sG-1,G} \phi_{G-1}^{(1)}$$

Tüm gurup akıları bulunduktan sonra yeni fisyon kaynağı ve çoğaltma katsayısı hesaplanır.

$$\overset{(1)}{S}(r) = \sum_{g'=1}^{G} \nu_{g'} \cdot \Sigma_{fg'} \cdot \phi_{g'}^{(1)}$$

$$k^{(1)} = \frac{\int d^3r\, S^{(1)}(r)}{\frac{1}{k^{(0)}} \int d^3r\, S^{(0)}(r)}$$

Her kaynak iterasyonunda, çok guruplu denklemler düşük enerjili denklemlere doğru çözülür.Bu çözüm düşük enerji gurup - larından yukarı enerji guruplarına saçılma olmadığını varsaymakla mümükün olur.Dolayısıyle yüksek enerji guruplarındaki akı düşük enerji guruplarına kaynak oluşturur.

Çok guruplu denklemlerde, termal enerjileri içeren denklemler birden fazlaysa üst enerjilere saçılma mümkün olacağından denklemleri bu şekilde çözmek mümkün değildir.

Çok guruplu difüzyon denklemlerini tekrar yazalım.

$$-\vec{\nabla} D(\vec{r}) \vec{\nabla} \phi(\vec{r}) + \sigma(\vec{r})\, \phi(\vec{r}) = S(\vec{r})$$

$$-\vec{\nabla} D^{1}(\vec{r}) \nabla \phi^{1}(\vec{r}) + \Sigma_{r}^{1}(\vec{r})\ \phi^{1}(\vec{r}) = \frac{\chi^{1}}{k} \sum_{g'=1}^{G} \nu\Sigma_{f}^{g'}(\vec{r})\ \phi^{g'}(\vec{r})$$

$$-\nabla D^{2}(\vec{r}) \vec{\nabla} \phi^{2}(\vec{r}) + \Sigma_{r}^{2}(\vec{r})\ \phi^{2}(\vec{r}) = \frac{\chi^{2}}{k} \sum_{g'=1}^{G} \nu\Sigma_{f}^{g'}(\vec{r})\ \phi^{g'}(\vec{r}) + \Sigma_{s}^{1\to 2}\ \phi^{1}(\vec{r})$$

$$\vdots \qquad\qquad\qquad\qquad \vdots$$

$$-\vec{\nabla} D^{g}(\vec{r}) \vec{\nabla} \phi^{g}(\vec{r}) + \Sigma_{r}^{g}(\vec{r})\ \phi^{g}(\vec{r}) = \frac{\chi^{g}}{k} \sum_{g'=1}^{G} \nu\Sigma_{f}(\vec{r})\ \phi^{g'}(\vec{r}) + \sum_{g'=1}^{g-1} \Sigma_{s}^{g'\to g}(\vec{r})\ \phi^{g'}(\vec{r})$$

$$\vdots \qquad\qquad\qquad\qquad \vdots$$

$$-\vec{\nabla} D^{G}(\vec{r}) \vec{\nabla} \phi^{G}(\vec{r}) + \Sigma_{r}^{G}(\vec{r})\ \phi^{G}(\vec{r}) = \frac{\chi^{G}}{k} \sum_{g'=1}^{G} \nu\Sigma_{f}(\vec{r})\ \phi^{g'}(\vec{r}) + \sum_{g'=1}^{G-1} \Sigma_{s}^{g'\to G}(\vec{r})\ \phi^{g'}(\vec{r})$$

Yukarıdaki denklemlerde g, enerji gurup numarasını göstermekte.

Σ_r^g ise g. inci enerji gurubu çıkartma tesir kesitidir.

Denklemleri daha basit ifade edebilmek için matris notasyon - unda yazalım.

$$\begin{bmatrix} -\vec{\nabla} D^{1}(\vec{r})\vec{\nabla} + \Sigma_r^{1}(\vec{r}) & & & \\ -\Sigma_s^{1\to 2}(\vec{r}) & -\vec{\nabla} D^{2}(\vec{r})\vec{\nabla} + \Sigma_r^{2}(\vec{r}) & & \\ \vdots & & \ddots & \\ -\Sigma_s^{1\to G}(\vec{r}) & -\Sigma_s^{2\to G}(\vec{r}) & & -\vec{\nabla} D^{G}(\vec{r})\vec{\nabla} + \Sigma_r^{G}(\vec{r}) \end{bmatrix} \begin{bmatrix} \phi^{1}(\vec{r}) \\ \phi^{2}(\vec{r}) \\ \vdots \\ \phi^{G}(\vec{r}) \end{bmatrix}$$

$$= \frac{1}{k} \begin{bmatrix} \chi_1 \nu \Sigma_f^{1}(\vec{r}) & \chi_1 \nu \Sigma_f^{2}(\vec{r}) & \cdots & \chi_1 \nu \Sigma_f^{G}(\vec{r}) \\ \chi_2 \nu \Sigma_f^{1}(\vec{r}) & \chi_2 \nu \Sigma_f^{2}(\vec{r}) & \cdots & \chi_2 \nu \Sigma_f^{G}(\vec{r}) \\ \vdots & & & \\ \chi_G \nu \Sigma_f^{1}(\vec{r}) & \chi_G \nu \Sigma_f^{2}(\vec{r}) & \cdots & \chi_G \nu \Sigma_f^{G}(\vec{r}) \end{bmatrix} \begin{bmatrix} \phi^{1}(\vec{r}) \\ \phi^{2}(\vec{r}) \\ \vdots \\ \phi^{G}(\vec{r}) \end{bmatrix}$$

$$\underline{\underline{M}}\,\underline{\emptyset} = \frac{1}{k}\,\underline{F}\,\underline{\emptyset}$$

$$\underline{\underline{M}}^{-1}\,\underline{F}\,\underline{\emptyset} = k\,\underline{\emptyset}$$

matrisyel eşitliği yazılabilir.

Bu eşitlikten k ve $\emptyset$ yi bulmak istiyoruz.

Yukarıdaki eşitlik bir özdeğer-özvektör problemidir.k ve $\underline{\emptyset}$ özdeğer-özvektör çiftinin bulunması için sayısal bir yöntem sunalım.

$\lambda^{(0)}$, $\underline{\emptyset}^{(0)}$ yapay seçilmiş

$\nu=1$

$$\underline{\psi}^{(\nu)} = \underline{\underline{M}}^{-1}\,\underline{F}\,\underline{\emptyset}^{(\nu-1)}$$

$\nu=\nu+1$

$$\lambda^{(\nu)} = \frac{\left\| \int_V \underline{F}\,\underline{\psi}^{(\nu)}\,dv \right\|}{\left\| \int_V \underline{F}\,\underline{\emptyset}^{(\nu-1)}\,dv \right\|}$$

$$\underline{\emptyset}^{(\nu)} = \frac{1}{\lambda^{(\nu)}}\,\psi^{(\nu)}$$

HAYIR $\left| \dfrac{\lambda^{\nu} - \lambda^{\nu-1}}{\lambda^{\nu}} \right| < \epsilon$ ise $\xrightarrow{\text{EVET}}$ $\underline{\emptyset} \cong \underline{\emptyset}^{\nu}$

ν Iterasyon parametresi

$\|U\|_1$ Doğal vektör normu

Her iterasyonda G tane kısmi türevli diferansiyel denklemin çözülmesi gerekmektedir.

$$
\begin{bmatrix}
-\vec{\nabla} D^{1}(\vec{r})\vec{\nabla} + \Sigma_r^{1}(\vec{r}) & 0 & & 0 & & 0 \\
-\Sigma_s^{1\to 2}(\vec{r}) & -\vec{\nabla} D^{2}(\vec{r})\vec{\nabla} + \Sigma_r^{2}(\vec{r}) & & 0 & & 0 \\
\vdots & & \ddots & \vdots & & \vdots \\
-\Sigma_s^{1\to g}(\vec{r}) & -\Sigma_s^{2\to g}(\vec{r}) & \cdots & -\vec{\nabla} D^{g}(\vec{r})\vec{\nabla} + \Sigma_r^{g}(\vec{r}) & & 0 \\
\vdots & & & & \ddots & \\
-\Sigma_s^{1\to G}(\vec{r}) & -\Sigma_s^{2\to G}(\vec{r}) & \cdots & -\Sigma_s^{g\to G}(\vec{r}) & \cdots & -\vec{\nabla} D^{G}(\vec{r})\vec{\nabla} + \Sigma_r^{G}(\vec{r})
\end{bmatrix}
\begin{bmatrix}
\psi^{1(\nu)}(\vec{r}) \\
\psi^{2(\nu)}(\vec{r}) \\
\vdots \\
\psi^{g(\nu)}(\vec{r}) \\
\vdots \\
\psi^{G(\nu)}(\vec{r})
\end{bmatrix}
=
$$

$$
\begin{bmatrix}
\chi^{1}\nu^{1}\zeta_f^{1}(\vec{r}) & \chi^{1}\nu^{2}\zeta_f^{2}(\vec{r}) & \cdots & \chi^{1}\nu^{g}\zeta_f^{g}(\vec{r}) & \cdots & \chi^{1}\nu^{G}\zeta_f^{G}(\vec{r}) \\
\chi^{2}\nu^{1}\zeta_f^{1}(\vec{r}) & \chi^{g}\nu^{2}\zeta_f^{2}(\vec{r}) & \cdots & \chi^{2}\nu^{g}\zeta_f^{g}(\vec{r}) & \cdots & \chi^{g}\nu^{G}\zeta_f^{G}(\vec{r}) \\
\vdots & & & & & \\
\chi^{g}\nu^{1}\zeta_f^{1}(\vec{r}) & \chi^{g}\nu^{2}\zeta_f^{2}(\vec{r}) & \cdots & \chi^{g}\nu^{g}\zeta_f^{g}(\vec{r}) & \cdots & \chi^{g}\nu^{G}\zeta_f^{G}(\vec{r}) \\
\vdots & & & & & \\
\chi^{G}\nu^{1}\zeta_f^{1}(\vec{r}) & \chi^{G}\nu^{2}\zeta_f^{2}(\vec{r}) & \cdots & \chi^{G}\nu^{g}\zeta_f^{g}(\vec{r}) & \cdots & \chi^{G}\nu^{G}\zeta_f^{G}(\vec{r})
\end{bmatrix}
\begin{bmatrix}
\phi^{(\nu-1)} \\
\phi^{(\nu-1)} \\
\vdots \\
\phi^{(\nu-1)} \\
\vdots \\
\phi^{(\nu-1)}
\end{bmatrix}
$$

Yukarı enerji guruplarına saçılma olmadığı varsayıldığından dolayı M matrisi alt üçgen bir matris olmakta.

$$- \bar{\nabla} D^{g}(\bar{r})\bar{\nabla}\psi^{g(\nu)}(\bar{r}) + \Sigma_r^{g}(\bar{r})\ \psi^{g(\nu)}(\bar{r}) = S^{g(\nu)}(\bar{r})$$

Kısmi türevli diferansiyel denklemde S kaynak vektörü

$$S^{g(\nu)}(\bar{r}) = \sum_{g'=1}^{g-1} \Sigma_s^{g'\to g}(\bar{r})\ \psi^{g'(\nu)}(\bar{r}) + \chi^{g'} \sum_{g'=1}^{G} \nu \Sigma_f^{g'}(\bar{r})\ \phi^{g'(\nu-1)}(\bar{r})$$

$g=1,2\ \ldots\ G$

g guruplu denklemde tek bilinmeyen saçılma ve fisyon kaynağından oluşan kaynak terimidir.Bu terim fisyon kaynağı iterasyonu ile belirlenir.

Tüm elemanları belirlenen g inci gurup difüzyon denklemi klasik sonlu fark yöntemiyle çözülebileceği gibi varyasyonel matematikten kaynaklanan sonlu elemanlar yöntemiylede çözülebilir.

Problem 1

R=98.74 cmyarıçapında silindirik bir reaktörün yetkinlik analizini, tek boyutlu tek enerji guruplu ve tek bölgeli verilere göre çözümleyelim.

$D = 9.210$ cm

$\Sigma_a = 0.152149$ cm^{-1}

$\nu\Sigma_f = 0.155952$ cm^{-1}

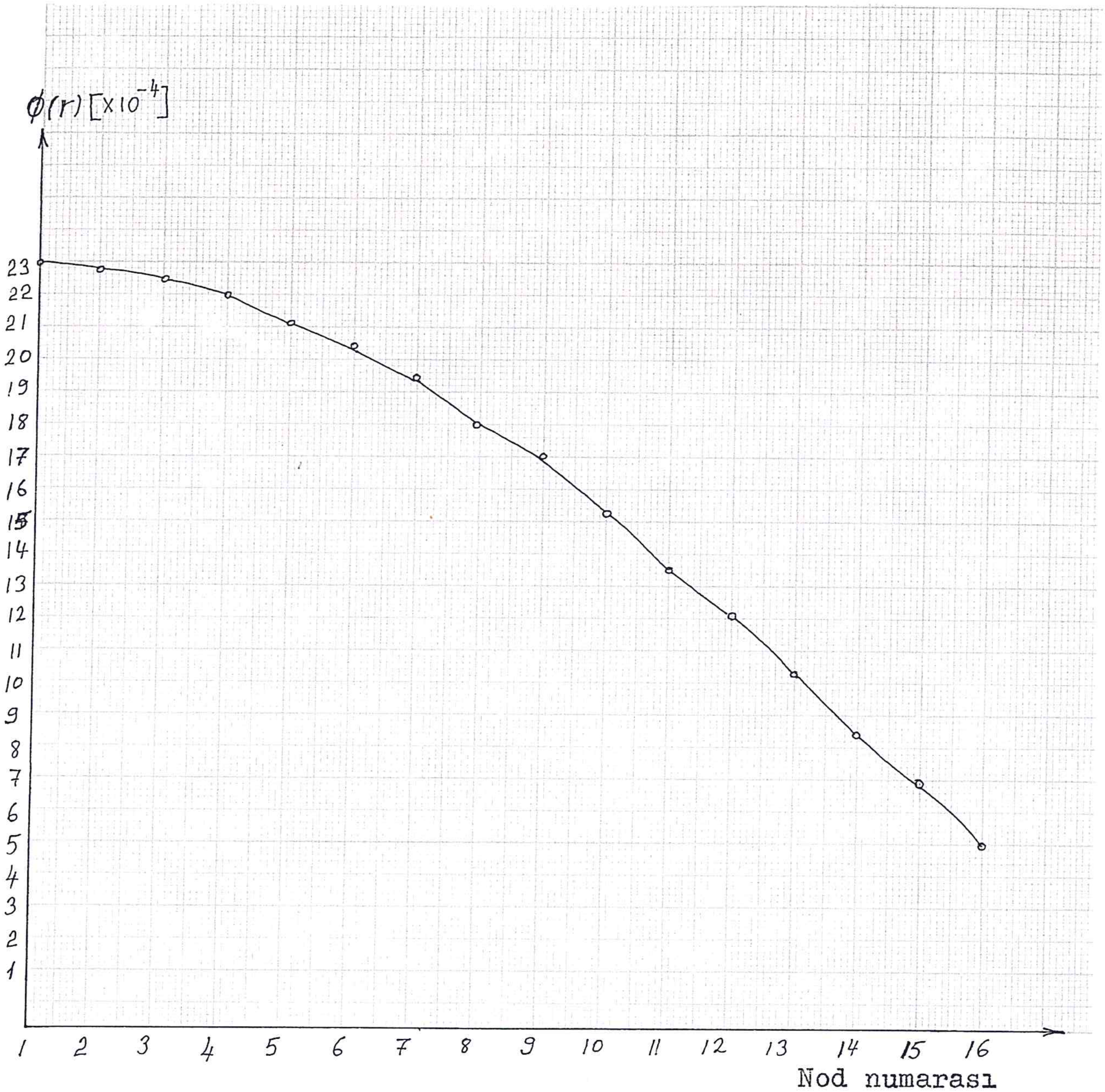
Bilgisayar giriş parametreleri:

ITMAX	100
EPS	0.00001
ENGEN1	1.0
FINOR	1.0
NRBCT	1
SFIS(1)	1.0
NFISOR	1
NETSOR	0

Problemin analitik çözümünde $k_{ef} = 1.000006$ bulunmaktadır. Bilgisayar çıkışları Tablo.1 de verilmektedir.

Eleman derecesi	Eleman sayısı	Nod sayısı	Yakınsama	İterasyon sayısı	k_{ef}	CPU zamanı(dk
1	25	26	9.655E-6	48	1.00048	4.05
1	14	15	9.298E-6	52	1.00217	3.01
2	7	15	8.940E-6	48	0.99977	2.56
3	5	16	9.059E-6	48	0.99993	3.26
4	6	25	9.775E-6	47	1.00121	5.34
4	4	17	8.821E-6	48	1.00053	3.58
4	3	13	9.298E-6	48	1.0004	3.07
4	2	9	9.775E-6	55	1.0012	2.37

Tablo.1



Şekil.1

Eleman derecesi : 3

Eleman sayısı : 5

Nod sayısı : 16

Problem 2

İ.T.Ü TRIGA MARK-II reaktörü tek boyutlu iki bölgeli ve iki guruplu kritikalite hesabı.

Reaktör (Şekil.2) deki gibi eşdeğer homojen bölgelere ayrılmıştır.Eşdeğer homojen veriler (Tablo.2) de verilmektedir. x

Yutulma tesir kesitlerine eksenel sızma terimi (DB_g^2) eklenmiştir. xx

x Eşdeğer homojenleştirme ve eşdeğer homojen veriler "A.Anacan,A.Yücel,T.Yarman İ.T.Ü TRIGA MARK-II reaktörü fizik hesapları" tez çalışmasından alınmıştır.

xx /4/ kaynaktan alındı.

Bilgisayar programı farklı eleman dareceleri ve eleman sayılarında çalıştırılmıştır.Bilgisayar çıkışları Tablo.3 de özetlenmiştir.

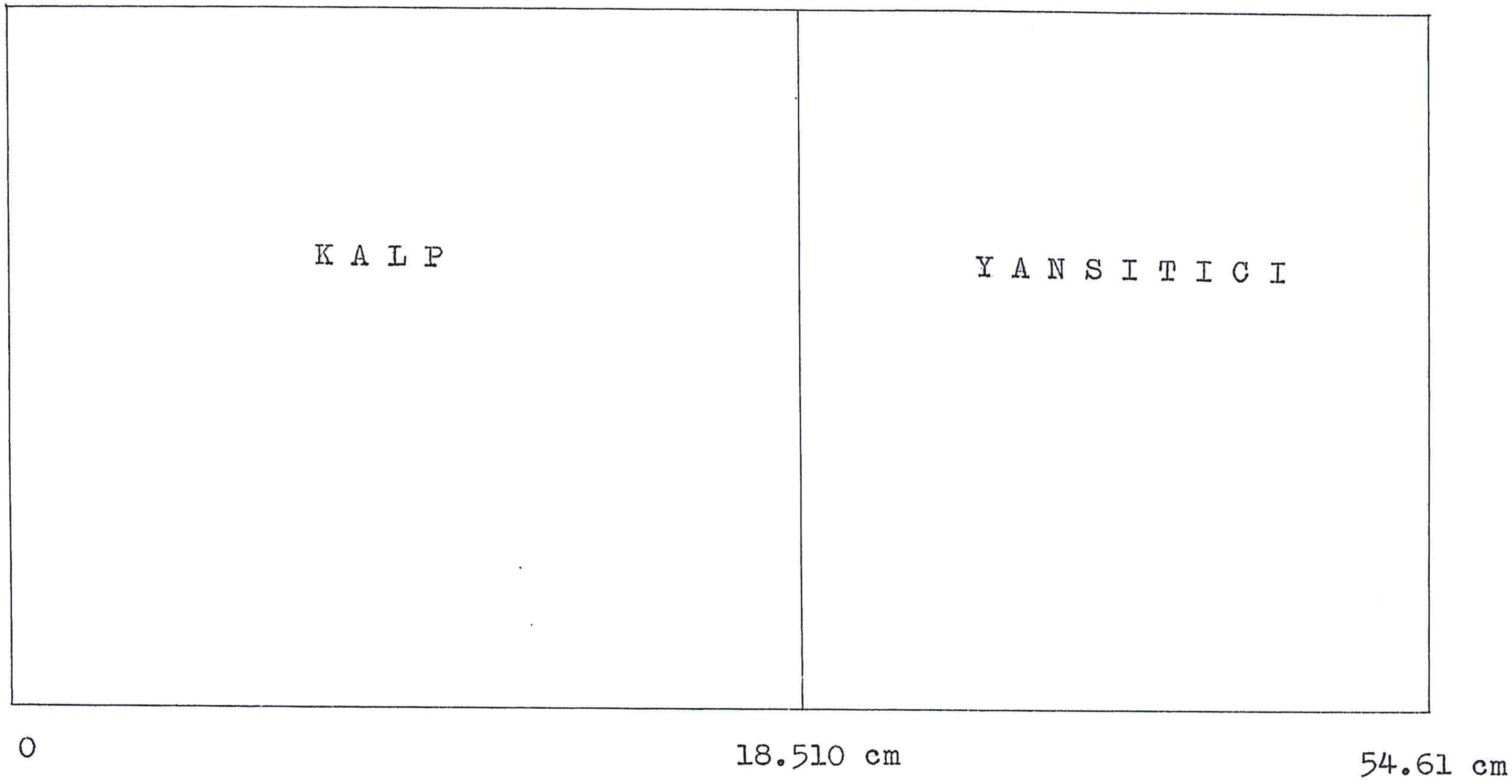
Yüksek mertebeli elemanlarla çalışılırken gerçek çözüme yaklaşılabilinmektedir./1/

Gerçek çözüme en yakın k_{ef} değeri olarak, 4 derece elemanlar ve 69 nod sayısında çalışılırken elde edilen k_{ef} 1.11623 değeri alınabilir.Diğer eleman derecelerinde elde edilen k_{ef} değerleri bu değerlerle mukayese edilirse 2 derece elemanlarda yanılgı 0 olmakta.En iyi yakınsamanın 2 derece elemanlarda gerçekleştiği söylenebilir.(Bilgisayar süresi dikkate alınarak)

Birinci derece sonlu elemanlarda yüksek yanılgı yüzde - leri görülmekte.

Bilgisayar giriş parametreleri:

ITMAX	100
EPS	0.00001
ENGEN1	1.0
FINOR	1.0
NRBCT	1
SFIS(1)	1.0
SFIS(2)	0.0
NFISOR	1
NETSOR	0



Şekil·2 Tek boyutlu iki bölgeli düzenek

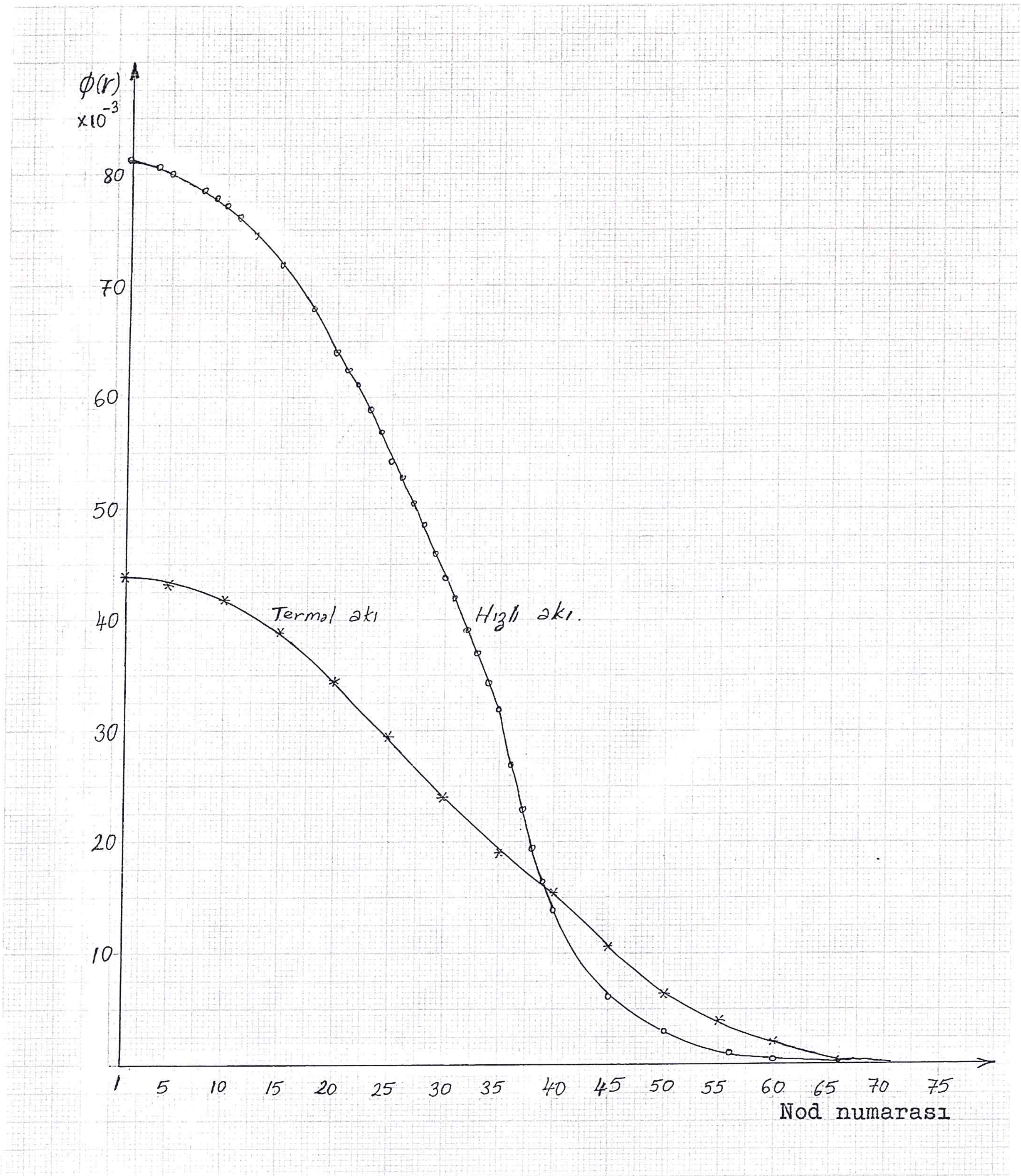
Bölge	Σ_{a1}	$\nu\Sigma_{f1}$	Σ_{tr_1}	Σ_{21}	Σ_{a2}	$\nu\Sigma_{f2}$	Σ_{tr_2}
1	0.016550	0.00813	0.371972	0.063567	0.11484	0.17843	1.91662
2	0.000160		0.426023	0.010050	0.00267		0.54318

Tablo.2 Tek boyut, iki bölgeli homojenleştirilmiş makroskopik tesir kesitleri (cm^{-1})

Tablo.3

Eleman derecesi	Eleman sayısı	Nod sayısı	Yakınsama	İterasyon sayısı	k_{ef}	CPU zamanı(dk)	% yanıl
1	20	21	8.106E-6	19	1.12185	3.19	0.50
1	30	31	8.702E-6	18	1.11866	4.31	0.21
1	70	71	6.675E-6	18	1.11668	9.50	0.040
2	6	13	8.583E-6	18	1.11731	3.33	0.090
2	8	17	7.033E-6	18	1.11654	2.56	0.027
2	10	21	8.106E-6	18	1.11636	3.34	0.016
2	15	31	9.775E-6	17	1.11624	4.52	0.0008
2	20	41	9.894E-6	17	1.11623	6.20	0.0
2	35	71	9.775E-6	17	1.11623	10.40	0.0
3	4	13	7.629E-6	20	1.12243	2.46	0.55
3	5	16	7.390E-6	20	1.12174	3.21	0.49
3	7	22	7.629E-6	18	1.11753	5.07	0.16
3	10	31	6.794E-6	18	1.11646	5.41	0.021

Eleman derecesi	Eleman sayısı	Nod sayısı	Yakınsama	İterasyon sayısı	k_{ef}	CPU zamanı	% yanılgı
3	13	40	6.556E-6	18	1.11631	7.16	0.0071
3	23	70	6.198E-6	18	1.11625	12.27	0.0018
4	4	17	8.225E-6	18	1.11687	3.33	0.057
4	5	21	8.463E-6	18	1.11686	4.19	0.0056
4	12	29	6.556E-6	18	1.11630	5.53	0.0063
4	10	41	6.318E-6	18	1.11624	8.16	0.00089
4	17	69	9.894E-6	17	1.11623	13.07	0.0



$\phi(r)$
$\times 10^{-3}$
80
70
60
50
40
30
20
10
Termal akı
Hızlı akı.
1
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
Nod numarası

Şekil.3

Problem 3

İ.T.Ü TRIGA MARK-II reaktörü tek boyutlu on bölgeli ve iki guruplu kritikalite hesabı.

Reaktör (Şekil.4) deki gibi eşdeğer homojen bölgelere ayrılmıştır.Eşdeğer homojen veriler (Tablo.4) de veril - mektedir. x

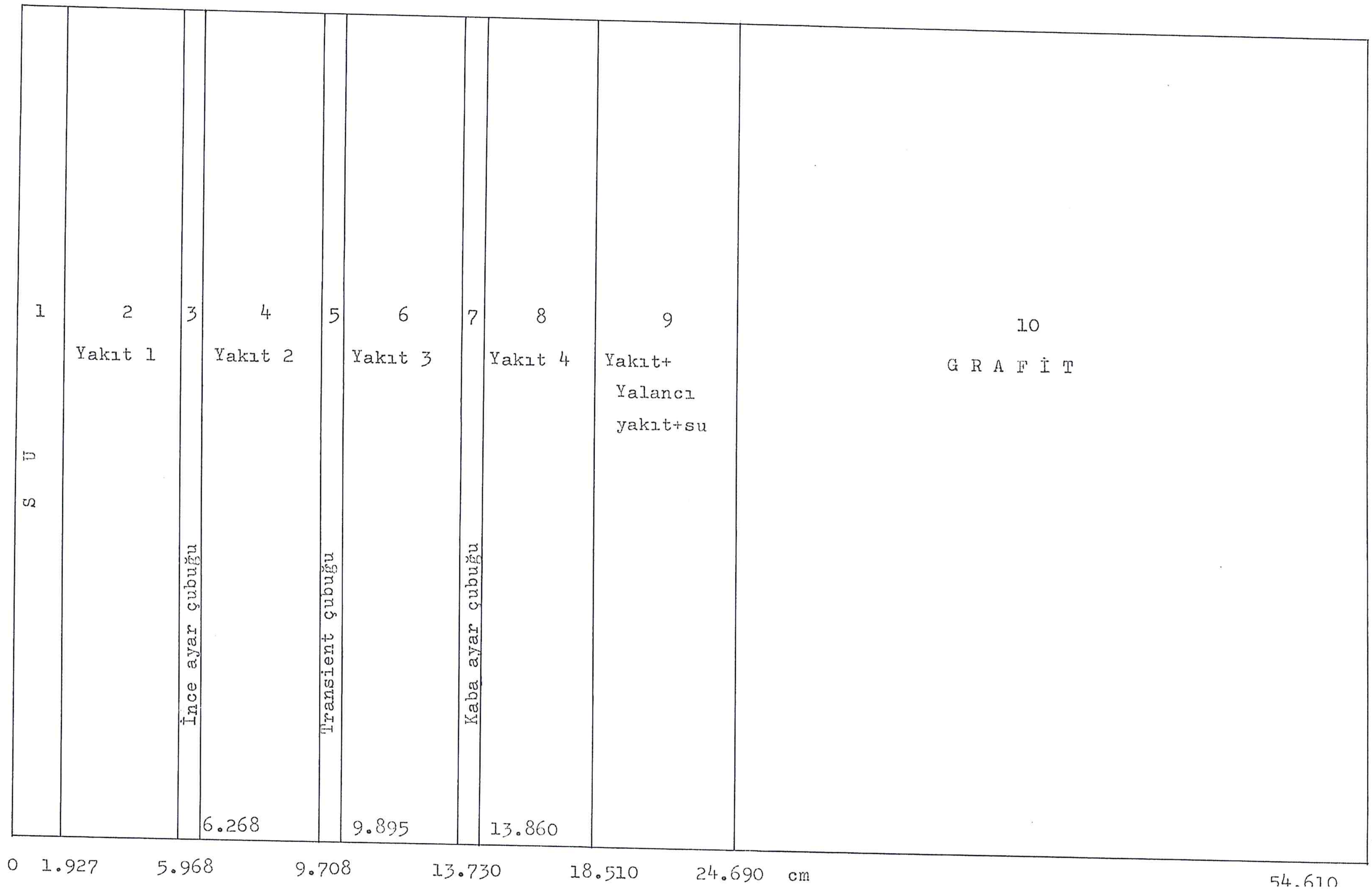
Çıkartma tesir kesitlerine eksenel sızma terimi (DB_g^2) eklenmiştir. xx

x Eşdeğer homojenleştirme ve eşdeğer homojen veriler "A.Anacan,A.Yücel,T.Yarman İ.T.Ü TRIGA MARK-II reaktörü fizik hesapları" tez çalışmasından alınmıştır.

xx /4/ kaynaktan alındı.

TRIGA reaktörü Şekil.4 de görüldüğü gibi eşdeğer homojen bölgelere ayrılmıştır./5/ Eşdeğer homojen veriler Tablo.4 de verilmektedir.Bilgisayar programı farklı eleman derecelerinde ve farklı nod sayılarında çalıştırıl - mıştır.4 derece elemanlar ile çalışılırken elde ettiğimiz sonuç gerçeğe enyakın değerdir.(k_{ef}= 1.08990) Reaktördeki termal ve hızlı akı dağılımı Şekil.5 de görülmektedir. x

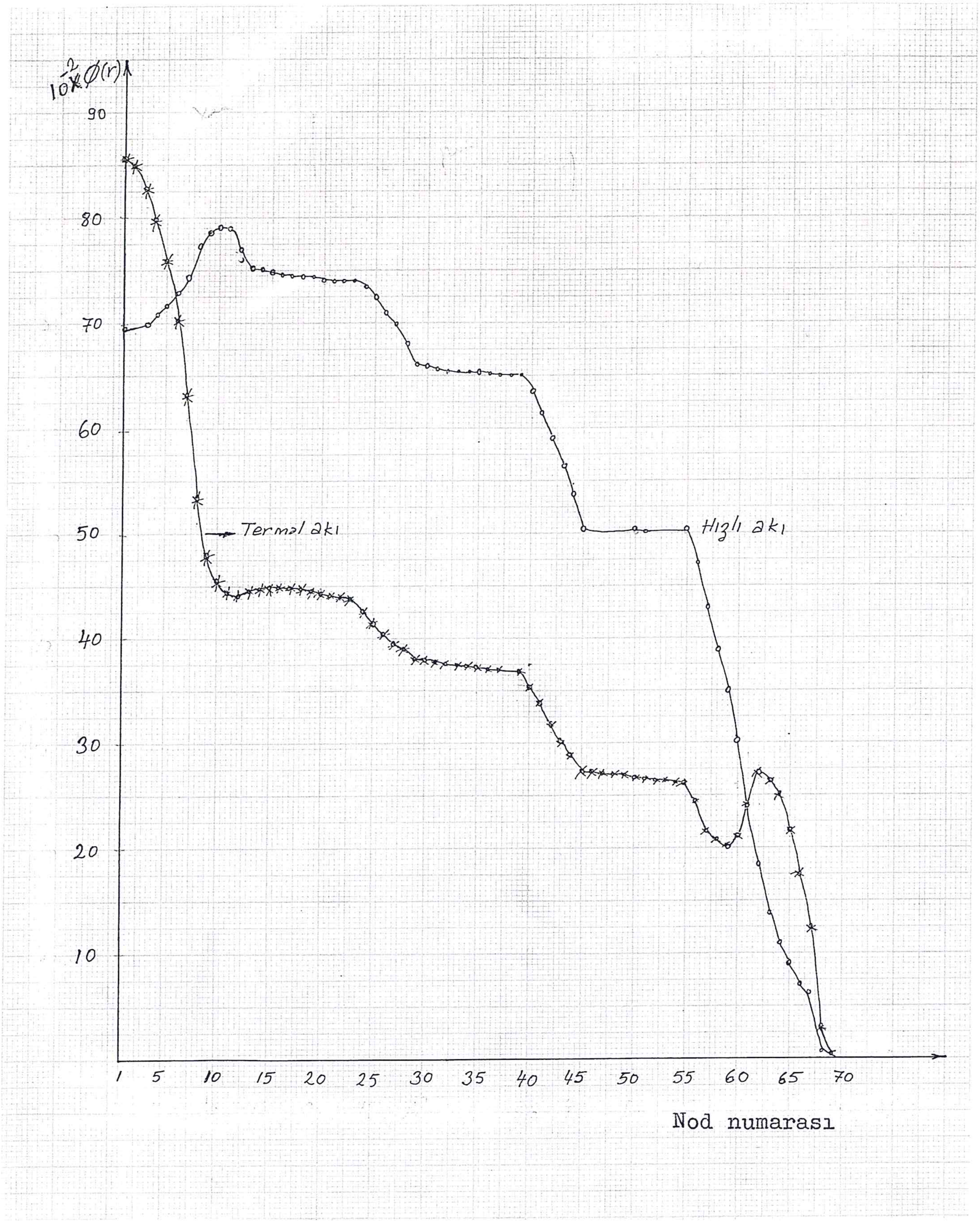
x Kontrol çubukları kalp dışında ve bulundukları yer su dolu.


1
SU
2
Yakıt 1
3
İnce ayar çubuğu
4
Yakıt 2
5
Transient çubuğu
6
Yakıt 3
7
Kaba ayar çubuğu
8
Yakıt 4
9
Yakıt+
Yalancı
yakıt+su
10
G R A F İ T
6.268
9.895
13.860
0
1.927
5.968
9.708
13.730
18.510
24.690
cm
54.610

Tek boyut, çok bölgeli homojenleştirilmiş makroskopik tesir kesitleri (cm^{-1})

Bölge	Σ_{a1}	$\nu\Sigma_{f1}$	Σ_{tr1}	Σ_{21}	Σ_{a2}	$\nu\Sigma_{f2}$	Σ_{tr2}
1	0.00112		0.29498	0.07751	0.01967		2.09210
2	0.01678	0.00826	0.37184	0.06360	0.11630	0.18138	1.91904
3	0.00106		0.24325	0.05344	0.01755		1.46625
4	0.01628	0.00800	0.36940	0.06407	0.11317	0.17553	1.92540
5	0.00106		0.24325	0.05344	0.01755		1.46625
6	0.01680	0.00825	0.37582	0.064304	0.11641	0.18113	1.93910
7	0.001064		0.24207	0.05288	0.01750		1.45184
8	0.01831	0.00905	0.38365	0.06302	0.12577	0.19864	1.92411
9	0.000893		0.32380	0.06045	0.01554		1.69738
10	0.000077		0.43752	0.00437	0.00122		0.41330

Tablo 4


$10^{-2} \times \phi(r)$
90
80
70
60
50
40
30
20
10
1
5
10
15
20
25
30
35
40
45
50
55
60
65
70
Termal akı
Hızlı akı
Nod numarası

Şekil.5

SONUÇ

Yüksek dereceli sonlu elemanlar kullanarak tek boyutlu silindirsel geometride çok guruplu ve çok bölgeli difüzyon denklemlerini çözen bilgisayar programı yazılmıştır. Bilgisayar programı ile İTÜ TRIGA MARK-II reaktörü iki bölgeli iki guruplu ve on bölgeli iki guruplu kritikalite hesapları gerçeklenmiştir.

Bilgisayar programında alt enerji guruplarından üst enerji guruplarına saçılma olmadığı varsayılmıştır. Eksenel sızma dikkate alınmıştır.

Yüksek dereceli elemanlar kullanılarak gerçek değerlere yakın sonuçlar bulunabilir. Yüksek dereceli elemanlar ile bilgisayar zamanı azaltılarak iyi bir yaklaşım sağlanmıştır. Lineer elemanlarla elde edilen sonuçlarda yanılgı büyük olmakta ve yüksek dereceli elemanların kullanılmasını zorunlu hale getirmektedir.

İki bölgeli ve iki guruplu hesap sonuçu TRIGA MARK-II reaktörü çoğaltma katsayısı 1.11623 bulunmuştur. On bölgeli ve iki guruplu hesap sonucu 1.08990 bulunmuştur. Bulunan sonuçlar birbirleriyle uyumludur.

EK.1

SONLU ELEMANLAR YÖNTEMİYLE ÇOK GURUPLU DİFÜZYON DENKLEMLERİNİ ÇÖZEN BİLGİSAYAR PROGRAMI:

Bilgisayar programı "FORTRAN" programlama dili ile yazılmıştır. Bir ana program ve 9 altprogramdan oluşmaktadır.

Ana program: HIKMET

Tüm altprogramlara dallanmayı gerçekleştirir.Çözülecek problemin parametreleri ana programda girdilenir.
COMMON deyimi ile ana programdaki sahalar ile altprogramdaki sahalar ortak kullanılmaktadır.COMMON deyiminde ayrılan sahalar maksimum 2 guruplu, 10 bölgeli ve maksimum 70 nod kullanılan problemlerin çözümü için yeterlidir.COMMON deyiminde sahaların boyutları genişletilerek çok guruplu çok bölgeli ve istenilen nod ile çalışılabilir.COMMON deyimindeki değişken listesi altprogramlardada ayni sırada ve boyutlarda tekrarlanmalıdır.

DEĞİŞKEN LİSTESİ

NGT	Enerji gurup sayısı.
MAT	Malzeme sayısı.
NETSOR	Serbest nötron kaynak parametresi.
NFISOR	Fısyon kaynak parametresi.
NDEG	Eleman derecesi.
NELT	Eleman sayısı.
NRBCT	Boşluk sınır koşulu.
NSP	Nod sayısı.
NHBW	Eleman matrisi bant genişliği.
ITMAX	Maksimum iterasyon sayısı.
EPS	Yakınsama parametresi.
ENGEN1	k_{ef} öngörüsü.
FINOR	Fısyon normalizasyon parametresi.
ITT	Iterasyon sayısı.

Alt program: SKINP

SKINP altprogramında serbest nötron kaynağı girdilenir.

DEĞİŞKEN LİSTESİ

SKAY(NGT,MAT)	Serbest nötron kaynağı.

Alt program: INPUT

Kullanılan malzemenin tesir kesitleri girdilenir

DEĞİŞKEN LİSTESİ

D(NGT,MAT)	Difüzyon sabitleri.
CEKES(NGT,MAT)	Çıkartma tesir kesitleri.
SEKES(NGT,NGT-1,MAT)	Saçılma tesir kesitleri.
FEKES(NGT,MAT)	Fisyon tesir kesitleri.
SFIS(NGT)	Fisyon spekturumu.

Alt program IZJEN

Eleman matrisi ve kaynak vektörünün oluşturulması için gerekli ızgarayı hazırlar.Ayrıca nod koordinatları belirlenir.

DEĞİŞKEN LİSTESİ

RM(MAT)	Malzeme dış yarıçapları.
RMEL(MAT)	Malzeme eleman sayıları.
REL(NSP,NDEG+1)	Izgara.
R(NSP)	Nod koordinatları.

Alt program: MATRIS

Sonlu elemanlar yöntemindeki eleman matrisini oluşturur. Çözülecek problemin parametrelerine bağlı kalarak fisyon kaynağı, serbest nötron kaynağı, saçılma vektörünü oluşturur.

DEĞİŞKEN LİSTESİ

ZE(NSP,NGT)	Serbest nötron kaynak vektörü.
SSS(NSP,NGT)	Saçılma vektörü.
FSS(NSP,NGT)	Fisyon kaynak vektörü.
AA(NSP,NDEG+1,NGT)	Eleman matrisi.

Alt program CHOLS1

Eleman matrisi Ek.3 de açıklanan yöntemle alt üçgen matris haline getirilir.

Alt program SCASOR

Kaynak vektörü belirlenir.

Alt program: FISSOR

32 inci sayfada açıklanan güç iterasyonu ile fisyon kaynağı belirlenir.

DEĞİŞKEN LİSTESİ

STARK(NSP,NDEG+1)	Iterasyon için gerekli ek bir saha. REL değişkenine ait sahayı kullanmakta.
CHKEG	Herhangibir iterasyon adımında yakınsama parametresi.

Alt program CHOLS2

CHOLS1 de hesaplanan alt üçgen matris ve kaynak vektörü kullanılarak lineer denklem sistemi çözülür

Alt program OUTPUT

Her enerji gurubuna ait akı dağılımı istenilen çıkış ortamına yazılır.

DEĞİŞKEN LİSTESI

AKI(NSP,MAT)	Akı dağılımı

BİLGİSAYAR PROGRAMI GİRDİ LİSTESİ

MAIN program.

Değişkenler	Tanımı
NGT	Enerji gurup sayısı.
MAT	Malzeme sayısı.
NETSOR	Serbest nötron kaynak parametresi. (0 ise kaynak yok, 1 ise var)
NFİSOR	Fisyon kaynak parametresi. (0 ise kaynak yok, 1 ise var)
NDEG	Sonlu eleman derecesi.
NELT	Sonlu eleman sayısı.
NRBCT	Boşluk sınır koşulu parametresi. (0 ise boşluk yok, 1 ise var)
ITMAX	Fisyon kaynağını belirleyen güç iterasyonunda maksimum iterasyon sayısı.
EPS	Yakınsama parametresi.
ENGEN1	k_{eff} öngörüsü.
FINOR	Fisyon normalizasyon parametresi. (Normal olarak 1 girilir)

1.Kart NGT,MAT,NETSOR,NFİSOR,NDEG,NELT,NRBCT

değişkenlerini 7I8 formatıyle giriniz.

2.Kart ITMAX,EPS,ENGEN1,FINOR

değişkenlerini (I8,3F10.6) formatıyle giriniz.

SUBROUTINE SKINP

Değişkenler	Tanımı
SKAY(NGT,MAT)	Serbest nötron kaynağı NETSOR 0 durumunda bu kart atlanır

3.Kart SKAY(NGT,MAT)

değişkenini 6F12.6 formatıyle giriniz.

SUBROUTİNE INPUT

Değişkenler	Tanımı
D(NGT,MAT)	Difüzyon sabiti.
CEKES(NGT,MAT)	Çıkartma tesir kesiti.
SEKES(NGT,NGT-1,MAT)	Saçılma tesir kesitleri. NGT 1 durumunda bu kart atlanır.
FEKES(NGT,MAT)	Fisyon tesir kesiti.
SFIS(NGT)	Fisyon spekturumu.

4.Kart D(NGT,MAT),CEKES(NGT,MAT),SEKES(NGT,NGT-1,MAT)

FEKES(NGT,MAT)

değişkenlerini 6F12.6 formatıyle giriniz.

5.Kart SFIS(NGT)

değişkenini 6F12.6 formatıyle giriniz.

SUBROUTINE IZJEN

Değişkenler	Tanımı
RM(MAT)	Malzeme dış yarıçapları.
RMEL(MAT)	Her malzemedeki eleman sayıları.

6.Kart RM(MAT)

değişkenini 8F10.5 formatıyle giriniz.

7.Kart RMEL(MAT)

değişkenini 8F10.5 formatıyle giriniz.

```
  $ *^DTYPE HIKMET.FOR
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),CEKES(2,10),FEKES(2,10),SEKES(2,2,10),SFIS(2)
     * ,SKAV(2,10),AM(10),AMEL(10),REL(70,5),R(70),AA(70,5,2),
     * SAC(2),ZE(70,2),SSS(70,2),FSS(70,2),AKI(70,2)
      READ(7,1) NGT,MAT,NETSOR,NFISOR,NDEG,NELT,NABCT
 1    FORMAT(7I8)
      WRITE(2,100)
 100  FORMAT(20X,'GIRDI LISTESI',//,15X)
      WRITE(2,101) NGT,MAT,NETSOR,NFISOR,NDEG,NELT,NABCT
 101  FORMAT(//,10X,'GURUP SAYISI=',I5,//,10X,'MALZEME SAYISI=',
     * I5,//,10X,'SERBEST NOTRON KAYNAGI=',I5,//,10X,'FISIL MALZEME='
     * ,I5,//,10X,'ELEMAN MERTEBESI=',I5,//,10X,'ELEMAN SAYISI=',I5,
     * //,10X,'SOGLUM SINIR KOSULU=',I5,//)
      NSP=NELT*NDEG+1
      NHBW=NDEG+1
      NOM=NDEG+1
      NSOV=NGT*(NGT-1)/2
      IF(NFISOR.EQ.0) GO TO 3
      READ(7,4) ITMAX,EPS,ENSEN1,FINOR
 4    FORMAT(I8,3F10.6)
      WRITE(2,102) ITMAX,EPS,ENSEN1,FINOR
 102  FORMAT(//,10X,'MAXIMUM ITERASYON SAYISI=',I8,//,10X,
     * 'YAKINSAMA PARAMETRESI=',F10.6,//,10X,'KEFF ONGORUSU=',F10.6,
     * //,10X,'FISYON NORMALIZASYON PARAMETRESI=',F10.6,////////)
 3    IF(NETSOR.EQ.0) GO TO 5
      CALL SKINP(NGT,MAT)
 5    CALL INPUT(NGT,MAT,NFISOR)
      CALL IZGEN(NDEG,NELT,MAT,NOM,NSP)
      DO 6 IEG=1,NGT
      CALL MATRIS(MAT,NGT,NSOV,IEG,NDEG,NELT,NOM,NETSOR,NFISOR,
     * NABCT,NSP,NHBW)
      CALL CHOLS1(NSP,NHBW,NGT,IEG)
 6    CONTINUE
      ITT=0
 12   IF(NFISOR.EQ.0) GO TO 7
      CALL FISSOR(NGT,ITT,NSP,FINOR,CHKES,ENSEN1)
      IF(ITT.EQ.0) GO TO 7
      IF(CHKES.LT.EPS) GO TO 9
      IF(ITT.GE.ITMAX) GO TO 9
 7    ITT=ITT+1
      WRITE(2,10)ITT
 10   FORMAT(15X,'ITERASYON SAYISI=',I10)
      DO 11 IEG=1,NGT
      CALL SCASOR(IEG,NGT,NSP,NSOV,NFISOR,NETSOR)
      CALL CHOLS2(IEG,NGT,NSP,NHBW)
 11   CONTINUE
      IF(NFISOR.EQ.0) GO TO 9
      GO TO 12
 9    CONTINUE
      CALL OUTPUT(NGT,NSP)
      STOP
      END
```

```
A>TYPE IZJEN.FOR
      SUBROUTINE IZJEN(NDEG,NELT,MAT,NOM,NSP)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),DEKES(2,10),FEKES(2,10),SEKES(2,2,10),SFIS(2)
     * ,SKAV(2,10),RM(10),RMEL(10),REL(70,5),R(70),AA(70,5,2),
     * SAD(2),ZE(70,2),SSS(70,2),FSS(70,2),AKI(70,2)
      NDEG1=NDEG+1
      READ(7,3)  (RM(I),I=1,MAT)
  3   FORMAT(8F10.5)
      RMAX=RM(MAT)
      READ(7,3)  (RMEL(I),I=1,MAT)
      WRITE(2,7)
      DO 8 I=1,MAT
      NMEL=RMEL(I)
  8   WRITE(2,9)  I,RM(I),NMEL
  9   FORMAT(7X,I5,13X,F12.6,17X,I5)
      RNDEG=NDEG
      NN=1
      NNN=RMEL(1)
      RIC=0.
      NOD=0
      DO 10 I=1,MAT
      RNN=NNN-NN+1
      RDIS=RM(I)
      REL=(RDIS-RIC)/RNN
      DO 11 J=NN,NNN
      REL(J,NOM)=I
      RJ=J-NN
      RR=RIC+RJ*REL
      DO 18 K=1,NDEG1
      NOD=NOD+1
      REL(J,K)=NOD
      RK=K
      R(NOD)=RR+(RK-1.)*REL/RNDEG
 18   CONTINUE
      NOD=NOD-1
 11   CONTINUE
      IF(I.EQ.MAT)  GO TO 10
      NN=NNN+1
      MMEL=RMEL(I+1)
      NNN=NNN+MMEL
      RIC=RDIS
 10   CONTINUE
      WRITE(2,12)
      DO 13 I=1,NELT
      WRITE(2,14)  (REL(I,J),J=1,NOM)
 13   CONTINUE
 14   FORMAT(2X,5(F10.0,2X))
      WRITE(2,15)
      DO 16  I=1,NSP
 16   WRITE(2,17)  I,R(I)
 17   FORMAT(10X,I5,10X,F12.5)
 7    FORMAT(15(/),20X,'MALZEME YARICAPI VE MALZEMEDEKI ELEMAN
     *  SAYILARI',///,5X,'MALZEME NOSU',5X,'MALZEME DIS YARICAPI'
     *  ,5X,'MALZEMEDEKI ELEMAN SAYISI',///)
 12   FORMAT(///,20X,'ELEMANLARA AIT BILGILER',//,2X,'1 INCI NOD
     *  ',4X,'2 INCI NOD',4X,'3 UNCU NOD',4X,'4 UNCU NOD',4X,
     *  '5 INCI NOD',///)
 15   FORMAT(///,15X,'NOD KOORDINATLARI',///,10X,'NOD NOSU'12X,///)
      WRITE(2,777)
777   FORMAT(14(/))
      RETURN
      END
```

```
A>TYPE SBKINP.FOR
      SUBROUTINE SBKINP(NBT,MAT)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),DEKEB(2,10),FEKEB(2,10),BEKEB(2,2,10),BFIB(2)
     *,SBKAY(2,10),RM(10),RMEL(10),REL(70,5),R(70),AA(70,5,2),
     * BAD(2),ZE(70,2),BBB(70,2),FBB(70,2),AKI(70,2)
      WRITE(2,3)
      DO 1 J=1,MAT
      WRITE(2,4) J
      READ(7,2) (SBKAY(I,J),I=1,NBT)
      WRITE(2,5) (SBKAY(I,J),I=1,NBT)
 1    CONTINUE
 3    FORMAT(////,20X,'SERBEST KAYNAK GIRDISI',///)
 4    FORMAT(15X,'MALZEME NOSU',2X,I5,//)
 5    FORMAT(10X,6(6F12.6,2X))
 2    FORMAT(6F12.6)
      RETURN
      END
```

```
A) TYPE INPUT.FOR
      SUBROUTINE INPUT(NGT,MAT,NFISOR)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),CEKES(2,10),FEKES(2,10),SEKES(2,2,10),SFIS(2)
     * ,SKAV(2,10),RM(10),RMEL(10),REL(70,5),R(70),AA(70,5,2),
     * SAD(2),ZE(70,2),SSS(70,2),FSS(70,2),AKI(70,2)
      WRITE(2,100)
      DO 1 J=1,MAT
      WRITE(2,101) J
      WRITE(2,102)
      READ(7,2) (D(I,J),I=1,NGT)
      WRITE(2,103) (D(I,J),I=1,NGT)
      WRITE(2,104)
      READ(7,2) (CEKES(I,J),I=1,NGT)
      WRITE(2,103) (CEKES(I,J),I=1,NGT)
      IF(NGT.EQ.1) GO TO 5
      WRITE(2,105)
      DO 3 I=1,NGT
      IF(I.EQ.1) GO TO 3
      WRITE(2,106) I
      I1=I-1
      READ(7,2) (SEKES(I,K,J),K=1,I1)
      WRITE(2,103) (SEKES(I,K,J),K=1,I1)
3     CONTINUE
5     IF(NFISOR.EQ.0) GO TO 1
      WRITE(2,107)
      READ(7,2)(FEKES(I,J),I=1,NGT)
      WRITE(2,103)(FEKES(I,J),I=1,NGT)
1     CONTINUE
      IF(NFISOR.EQ.0) GO TO 6
      WRITE(2,108)
      READ(7,2) (SFIS(I),I=1,NGT)
      WRITE(2,103)(SFIS(I),I=1,NGT)
2     FORMAT(4F12.6)
100   FORMAT(20(/),20X,'TESIR KESITLERI LISTESI',///)
101   FORMAT(/,10X,'MALZEME NOSU',2X,I5,/)
102   FORMAT(10X,'DIFUZYON SABITLERI',/)
103   FORMAT(10X,4(F12.6,2X))
104   FORMAT(/,10X,'CIKARTMA TESIR KESITLERI',/)
105   FORMAT(/,10X,'SACILMA TESIR KESITLERI',/)
106   FORMAT(10X,'ENERJI GRUBU',2X,I5)
107   FORMAT(/,10X,'FISYON TESIR KESITLERI',/)
108   FORMAT(//////,20X,'FISYON SPEKTRUMU',/)
6     RETURN
      END
```

```
A)^CTYPE MARTRIS.FOER)A
      SUBROUTINE MATRIS(MAT,NST,NSOV,IEG,NDEG,NELT,NOM,NETGOR,
     * NFIGOR,NRSCT,NGP,NHBW)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),CEKEG(2,10),FEKEG(2,10),GEKEG(2,2,10),BFIG(2)
     * ,GKAV(2,10),RM(10),RMEL(10),REL(70,5),R(70),AA(70,5,2),
     *  GAO(2),ZE(70,2),SGG(70,2),FGG(70,2),AKI(70,2)
      NN=1
      NNN=RMEL(1)
      NDEG1=NDEG+1
      DO 101 I=1,NGP
      DO 101 J=1,NHBW
 101  AA(I,J,IEG)=0.
      IF(NFIGOR.EQ.0) GO TO 105
      DO 102 I=1,NGP
 102  FGG(I,IEG)=0.
 105  IF(IEG.EQ.1) GO TO 106
      IBAS=(IEG-1)*(IEG-2)/2+1
      IGON=IBAS+IEG-2
      DO 103 I=IBAS,IGON
      DO 103 J=1,NGP
 103  SGG(J,I)=0.
 106  IF(NETGOR.EQ.0) GO TO 552
      DO 104 I=1,NGP
 104  ZE(I,IEG)=0.
 552  DO 1 II=1,MAT
      DIFG=D(IEG,II)
      DTKG=DEKEG(IEG,II)
      IF(NETGOR.EQ.0) GO TO  2
      GKVN=GKAV(IEG,II)
 2    IF(IEG.EQ.0) GO TO 3
      JEG=IEG-1
      DO 4 JJ=1,JEG
 4    GAO(JJ)=GEKEG(IEG,JJ,II)
 3    IF(NFIGOR.EQ.0) GO TO 5
      FTKG=FEKEG(IEG,II)
 5    DO 6 JJ=NN,NNN
      I=REL(JJ,1)
      J=REL(JJ,2)
      IF(NDEG.LE.1) GO TO 7
      K=REL(JJ,3)
      IF(NDEG.LE.2) GO TO 7
      L=REL(JJ,4)
      IF(NDEG.LE.3) GO TO 7
      M=REL(JJ,5)
 7    IF(NDEG.GT.1) GO TO 9
      A=DIFG*(R(I)/(R(J)-R(I))+0.5)
      B=(R(J)-R(I))*R(I)*DTKG/6.
      C=DTKG*(R(J)-R(I))**2/12.
      IF(NFIGOR.EQ.0) GO TO 11
      E=FTKG*(R(J)-R(I))*R(I)/2.
      F=FTKG*(R(J)-R(I))**2/6.
      FGG(I,IEG)=FGG(I,IEG)+E+F
      FGG(J,IEG)=FGG(J,IEG)+E+2.*F
```

```
11    IF(NETGOR.EQ.0) GO TO 12
      E=GKVN*(R(J)-R(I))*R(I)/2.
      F=GKVN*(R(J)-R(I))**2/6.
      ZE(I,IEG)=ZE(I,IEG)+E+F
      ZE(J,IEG)=ZE(J,IEG)+E+2.*F
12    IF(IEG.EQ.1) GO TO 13
      IEG1=IEG-1
      DO 14 IBOV=1,IEG1
      E=GAC(IBOV)*(R(J)-R(I))*R(I)/2.
      F=GAC(IBOV)*(R(J)-R(I))**2/6.
      JBOV=(IEG-1)*(IEG-2)/2+IBOV
      GGG(I,JBOV)=GGG(I,JBOV)+E+F
      GGG(J,JBOV)=GGG(J,JBOV)+E+2.*F
14    CONTINUE
13    AA(I,NHBW,IEG)=AA(I,NHBW,IEG)+A+2.*B+C
      JKL=NHBW-J+I
      AA(J,JKL,IEG)=AA(J,JKL,IEG)-A+B+C
      AA(J,NHBW,IEG)=AA(J,NHBW,IEG)+A+2.*B+2.*C
      IF(JJ.NE.NELT) GO TO 15
      IF(NRGOT.NE.1) GO TO 15
      AA(J,NHBW,IEG)=AA(J,NHBW,IEG)+R(NGP)/2.
15    GO TO 6
8     IF(NDEG.GT.2) GO TO 16
      A=DIFG*R(I)/(3.*(R(K)-R(I)))
      B=(R(K)-R(I))*R(I)*CTKG/20.
      D=CTKG*(R(K)-R(I))**2/60.
      P=DIFG/6.
      IF(NFIGOR.EQ.0) GO TO 17
      E=FTKG*(R(K)-R(I))*R(I)/6.
      F=FTKG*(R(K)-R(I))**2/6.
      FGG(I,IEG)=FGG(I,IEG)+E
      FGG(J,IEG)=FGG(J,IEG)+4.*E+2.*F
      FGG(K,IEG)=FGG(K,IEG)+E+F
17    IF(NETGOR.EQ.0) GO TO 18
      E=GKVN*(R(K)-R(I))*R(I)/6.
      F=GKVN*(R(K)-R(I))**2/6.
      ZE(I,IEG)=ZE(I,IEG)+E
      ZE(J,IEG)=ZE(J,IEG)+4.*E+2.*F
      ZE(K,IEG)=ZE(K,IEG)+E+F
18    IF(IEG.EQ.1) GO TO 19
      IEG1=IEG-1
      DO 20 IBOV=1,IEG1
      E=GAC(IBOV)*(R(K)-R(I))*R(I)/6.
      F=GAC(IBOV)*(R(K)-R(I))**2/6.
      JBOV=(IEG-1)*(IEG-2)/2+IBOV
      GGG(I,JBOV)=GGG(I,JBOV)+E
      GGG(J,JBOV)=GGG(J,JBOV)+4.*E+2.*F
      GGG(K,JBOV)=GGG(K,JBOV)+E+F
20    CONTINUE
19    AA(I,NHBW,IEG)=AA(I,NHBW,IEG)+7.*A+3.*P+4.*B+C
      JKL=NHBW-J+I
      AA(J,JKL,IEG)=AA(J,JKL,IEG)-8.*A-4.*P+2.*B
      AA(J,NHBW,IEG)=AA(J,NHBW,IEG)+16.*(A+P+C+B)
      JKL=NHBW-K+I
```

```
      AA(K,JKL,IEG)=AA(K,JKL,IEG)+A+P-B-C
      JKL=NHBW-K+J
      AA(K,JKL,IEG)=AA(K,JKL,IEG)-8.*A-12.*P+2.*B+4.*C
      AA(K,NHBW,IEG)=AA(K,NHBW,IEG)+7.*A+11.*P+4.*B+7.*C
      IF(JJ.NE.NELT) GO TO 21
      IF(NRBCT.NE.1) GO TO 21
      AA(K,NHBW,IEG)=AA(K,NHBW,IEG)+R(NBP)/2.
21    GO TO 6
16    IF(NDEG.GT.3) GO TO 31
      A=DIFB*R(I)/(40.*(R(L)-R(I)))
      P=DIFB/80.
      B=(R(L)-R(I))*R(I)*CTKB/1680.
      C=CTKB*(R(L)-R(I))**2/3360.
      IF(NFIBOR.EQ.0) GO TO 22
      E=FTKB*(R(L)-R(I))*R(I)/8.
      F=FTKB*(R(L)-R(I))**2/120.
      FBB(I,IEG)=FBB(I,IEG)+E+2.*F
      FBB(J,IEG)=FBB(J,IEG)+3.*E+9.*F
      FBB(K,IEG)=FBB(K,IEG)+3.*E+26.*F
      FBB(L,IEG)=FBB(L,IEG)+E+13.*F
22    IF(NETBOR.EQ.0) GO TO 23
      E=BHYN*(R(L)-R(I))*R(I)/8.
      F=BHYN*(R(L)-R(I))**2/120.
      ZE(I,IEG)=ZE(I,IEG)+E+2.*F
      ZE(J,IEG)=ZE(J,IEG)+3.*E+9.*F
      ZE(K,IEG)=ZE(K,IEG)+3.*E+26.*F
      ZE(L,IEG)=ZE(L,IEG)+E+13.*F
23    IF(IEG.EQ.1) GO TO 24
      IEG1=IEG-1
      DO 25 IBOV=1,IEG1
      E=BAD(IBOV)*(R(L)-R(I))*R(I)/8.
      F=BAD(IBOV)*(R(L)-R(I))**2/120.
      JBOV=(IEG-1)*(IEG-2)/2+IBOV
      BBB(I,JBOV)=BBB(I,JBOV)+E+2.*F
      BBB(J,JBOV)=BBB(J,JBOV)+3.*E+9.*F
      BBB(K,JBOV)=BBB(K,JBOV)+3.*E+26.*F
      BBB(L,JBOV)=BBB(L,JBOV)+E+13.*F
25    CONTINUE
24    AA(I,NHBW,IEG)=AA(I,NHBW,IEG)+148.*A+24.*P+128.*B+19.*C
      JKL=NHBW-J+I
      AA(J,JKL,IEG)=AA(J,JKL,IEG)-189.*A-81.*P+99.*B+9.*C
      AA(J,NHBW,IEG)=AA(J,NHBW,IEG)+432.*A+270.*P+648.*B+405.*C
      JKL=NHBW-K+I
      AA(K,JKL,IEG)=AA(K,JKL,IEG)+54.*A+28.*P-36.*B+9.*C
      JKL=NHBW-K+J
      AA(K,JKL,IEG)=AA(K,JKL,IEG)-297.*A-297.*P-81.*B-81.*C
      AA(K,NHBW,IEG)=AA(K,NHBW,IEG)+432.*A+594.*P+648.*B+891.*C
      JKL=NHBW-L+I
      AA(L,JKL,IEG)=AA(L,JKL,IEG)-13.*A-13.*P+19.*B+19.*C
      JKL=NHBW-L+J
      AA(L,JKL,IEG)=AA(L,JKL,IEG)+54.*A+72.*P-36.*B-81.*C
      JKL=NHBW-L+K
      AA(L,JKL,IEG)=AA(L,JKL,IEG)-189.*A-297.*P+99.*B+189.*C
      AA(L,NHBW,IEG)=AA(L,NHBW,IEG)+148.*A+262.*P+128.*B+297.*C
      IF(JJ.NE.NELT) GO TO 6
      IF(NRBCT.NE.1) GO TO 6
      AA(L,NHBW,IEG)=AA(L,NHBW,IEG)+R(NBP)/2.
31    A=DIFB*R(I)/((R(M)-R(I))*945.)
      B=(R(M)-R(I))*R(I)*CTKB/11340.
      C=CTKB*(R(M)-R(I))**2/11340.
      P=DIFB/1890.
```

```
      IF(NFIBDR.EQ.0) GO TO 23
      E=FTKG*(R(M)-R(I))*R(I)/90.
      F=FTKG*(R(M)-R(I))**2/90.
      FGG(I,IEG)=FGG(I,IEG)+7.*E
      FGG(J,IEG)=FGG(J,IEG)+32.*E+8.*F
      FGG(K,IEG)=FGG(K,IEG)+12.*E+6.*F
      FGG(L,IEG)=FGG(L,IEG)+32.*E+24.*F
      FGG(M,IEG)=FGG(M,IEG)+7.*(E+F)
23    IF(NETBDR.EQ.0) GO TO 24
      E=GKVN*(R(M)-R(I))*R(I)/90.
      F=GKVN*(R(M)-R(I))**2/90.
      ZE(I,IEG)=ZE(I,IEG)+7.*E
      ZE(J,IEG)=ZE(J,IEG)+32.*E+8.*F
      ZE(K,IEG)=ZE(K,IEG)+12.*E+6.*F
      ZE(L,IEG)=ZE(L,IEG)+32.*E+24.*F
      ZE(M,IEG)=ZE(M,IEG)+7.*(E+F)
24    IF(IEG.EQ.1) GO TO 25
      IEG1=IEG-1
      DO 29 IBDY=1,IEG1
      JBDY=(IEG-1)*(IEG-2)/2+IBDY
      E=GAD(IBDY)*(R(M)-R(I))*R(I)/90.
      F=GAD(IBDY)*(R(M)-R(I))**2/90.
      GGG(I,JBDY)=GGG(I,JBDY)+7.*E
      GGG(J,JBDY)=GGG(J,JBDY)+32.*E+8.*F
      GGG(K,JBDY)=GGG(K,JBDY)+12.*E+6.*F
      GGG(L,JBDY)=GGG(L,JBDY)+32.*E+24.*F
      GGG(M,JBDY)=GGG(M,JBDY)+7.*(E+F)
29    CONTINUE
25    AA(I,NHBW,IEG)=AA(I,NHBW,IEG)+4925.*A+705.*P+564.*B+28.*C
      JKL=NHBW-J+I
      AA(J,JKL,IEG)=AA(J,JKL,IEG)-6048.*A-992.*P+592.*B+32.*C
      AA(J,NHBW,IEG)=AA(J,NHBW,IEG)+16640.*A+6400.*P+2884.*B+768.*C
      JKL=NHBW-K+I
      AA(K,JKL,IEG)=AA(K,JKL,IEG)+2048.*A+224.*P-348.*B
      JKL=NHBW-K+J
      AA(K,JKL,IEG)=AA(K,JKL,IEG)-14208.*A-8216*P-768.*B-192*C
      AA(K,NHBW,IEG)=AA(K,NHBW,IEG)+22328*A+22228*P+3744*B+1672*C
      JKL=NHBW-L+I
      AA(L,JKL,IEG)=AA(L,JKL,IEG)-1472*A-864*P+112*B-32*C
      JKL=NHBW-L+J
      AA(L,JKL,IEG)=AA(L,JKL,IEG)+5888*A+5888*P+512*B+256*C
      JKL=NHBW-L+K
      AA(L,JKL,IEG)=AA(L,JKL,IEG)-14208*A-19200*P-768*B-576*C
      AA(L,NHBW,IEG)=AA(L,NHBW,IEG)+16640*A+26800*P+2584*B+2816*C
      JKL=NHBW-M+I
      AA(M,JKL,IEG)=AA(M,JKL,IEG)+247*A+247*P-50*B-29*C
      JKL=NHBW-M+J
      AA(M,JKL,IEG)=AA(M,JKL,IEG)-1472*A-2080*P+112*B+144*C
      JKL=NHBW-M+K
      AA(M,JKL,IEG)=AA(M,JKL,IEG)+2048*A+5292*P-348*B-348*C
      JKL=NHBW-M+L
      AA(M,JKL,IEG)=AA(M,JKL,IEG)-6848*A-12704*P+592*B+560*C
      AA(M,NHBW,IEG)=AA(M,NHBW,IEG)+4925*A+9145*P+564*B+555*C
      IF(JJ.NE.NELT) GO TO 6
      IF(NABOT.NE.1) GO TO 6
      AA(M,NHBW,IEG)=AA(M,NHBW,IEG)+R(NBP)/2
6     CONTINUE
      IF(II.EQ.MAT) GO TO 1
      NN=NN+1
      ILJ=II+1
      MMEL=AMEL(ILJ)
      NNN=NNN+MMEL
1     CONTINUE
      RETURN
      END
```

```
A>TYPE CHOLS1.FOR
      SUBROUTINE CHOLS1(NBP,NHBW,NGT,IEG)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),DEKEG(2,10),FEKEG(2,10),SEKEG(2,2,10),GFIG(2)
     * ,GKAV(2,10),RM(10),RMEL(10),REL(70,5),R(70),AA(70,5,2),
     * GAD(2),ZE(70,2),GGG(70,2),FGG(70,2),AKI(70,2)
      WRITE(2,888)IEG
 888  FORMAT(5X,I2,'   INCI GURUP ELEMAN MATRISI',//)
      WRITE(2,999)((AA(I,J,IEG),J=1,NHBW),I=1,NBP)
 999  FORMAT(2X,5F15.4)
      WRITE(2,111)
 111  FORMAT(44(/))
      DO 4 J=1,NBP
      SUM=0.
      IF(J.EQ.1) GO TO 5
      L=J-NHBW+1
      IF(L.LT.1) L=1
      KKK=J-1
      DO 6 K=L,KKK
      JJ=NHBW-J+K
  6   SUM=SUM+AA(J,JJ,IEG)**2
 5    TS=AA(J,NHBW,IEG)-SUM
      AA(J,NHBW,IEG)=TS**0.5
      IF(J.EQ.NBP) GO TO 4
      L=J+NHBW-1
      IF(L.GT.NBP) L=NBP
      III=J+1
      DO 7 I=III,L
      SUM=0.
      IF(J.EQ.1) GO TO 8
      LL=I-NHBW+1
      IF(LL.LT.1) LL=1
      MMM=J-1
      IF(LL.GT.MMM) GO TO 8
      DO 9 M=LL,MMM
      KK=NHBW-J+M
      KKK=NHBW-I+M
  9   SUM=SUM+AA(J,KK,IEG)*AA(I,KKK,IEG)
  8   KK=NHBW-I+J
      AA(I,KK,IEG)=(AA(I,KK,IEG)-SUM)/AA(J,NHBW,IEG)
  7   CONTINUE
  4   CONTINUE
      RETURN
      END
```

```
A>TYPE FISSDR.FOR
      SUBROUTINE FISSDR(NST,ITT,NSP,FINOR,CHKES,ENGEN1)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),DEKES(2,10),FEKES(2,10),SEKES(2,2,10),SFIS(2)
     *,SKAV(2,10),RM(10),RMEL(10),STARK(70,5),R(70),AA(70,5,2),
     *SAD(2),ZE(70,2),SSS(70,2),FSS(70,2),AKI(70,2)
      IF(ITT.NE.0) GO TO 11
      DO 12 IES=1,NST
      DO 12 I=1,NSP
12    AKI(I,IES)=1.
11    FINT=0.
      DO 1 IES=1,NST
      DO 2 I=1,NSP
2     STARK(I,IES)=0.
      DO 3 JES=1,NST
      DO 4 I=1,NSP
4     STARK(I,IES)=STARK(I,IES)+SFIS(IES)*FSS(I,JES)*AKI(I,JES)
3     CONTINUE
      DO 5 I=1,NSP
5     FINT=FINT+STARK(I,IES)
1     CONTINUE
      FACTRN=FINOR/FINT
      IF(ITT.EQ.0) GO TO 6
      ENGEN=FINT*ENGEN1/FINOR
      CHKES=ENGEN/ENGEN1-1
      IF(CHKES.LT.0) CHKES=CHKES*(-1)
      ENGEN1=ENGEN
      WRITE(2,7) ENGEN1,CHKES
7     FORMAT(/,5X,'KEFF=',E12.5,5X,'VAKINGAKA PARAMETREBI=',E12.5)
6     FSOR=FACTRN/ENGEN1
      DO 8 IES=1,NST
      DO 9 I=1,NSP
9     AKI(I,IES)=FACTRN*AKI(I,IES)
      DO 10 I=1,NSP
10    STARK(I,IES)=FSOR*STARK(I,IES)
8     CONTINUE
      RETURN
      END
```

```
A>TYPE SCAGGR.FOR
      SUBROUTINE SCAGGR(IEG,NGT,NGP,NGGV,NFIGGR,NETGGR)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),DEKEG(2,10),FEKEG(2,10),GEKEG(2,2,10),GFIG(2)
     *,EKAV(2,10),RM(10),RMEL(10),STARK(70,5),R(70),AA(70,5,2),
     *GAD(2),ZE(70,2),GGG(70,2),FGG(70,2),AKI(70,2)
      IF(NFIGGR.NE.0) GO TO 1
      DO 2 I=1,NGP
2     STARK(I,IEG)=0.
1     IF(IEG.EQ.1)GO TO 3
      KG=IEG-1
      DO 4 KEG=1,KG
      IGGV=(IEG-1)*(IEG-2)/2.+KEG
      DO 5 I=1,NGP
5     STARK(I,IEG)=STARK(I,IEG)+GGG(I,IGGV)*AKI(I,KEG)
4     CONTINUE
3     CONTINUE
      IF(NETGGR.EQ.0) GO TO 6
      DO 7 I=1,NGP
7     STARK(I,IEG)=STARK(I,IEG)+ZE(I,IEG)
6     RETURN
      END
```

```
A>TYPE CHOLS2.FOR
        SUBROUTINE CHOLS2(IES,NST,NSP,NHBW)
        IMPLICIT REAL*8(A-H,O-Z)
        COMMON D(2,1Ø),DEKES(2,1Ø),FEKES(2,1Ø),SEKES(2,2,1Ø),SFIS(2)
     *  ,SHAY(2,1Ø),RM(1Ø),RMEL(1Ø),STARK(7Ø,5),R(7Ø),AA(7Ø,5,2),
     *  SAD(2),ZE(7Ø,2),SSS(7Ø,2),FSS(7Ø,2),AKI(7Ø,2)
        DO 1 I=1,NSP
 1      AKI(I,IES)=STARK(I,IES)
        DO 2 I=1,NSP
        SUM=Ø.
        IF(I.EQ.1) GO TO 3
        K=I-NHBW+1
        IF(K.LT.1) K=1
        III=I-1
        DO 4 J=K,III
        KK=NHBW-I+J
 4      SUM=SUM+AA(I,KK,IES)*AKI(J,IES)
 3      AKI(I,IES)=(AKI(I,IES)-SUM)/AA(I,NHBW,IES)
 2      CONTINUE
        DO 5 I=1,NSP
        II=NSP-I+1
        SUM=Ø.
        IF(II.EQ.NSP) GO TO 6
        K=II+NHBW-1
        IF(K.GT.NSP) K=NSP
        KKKK=II+1
        DO 7 J=KKKK,K
        KK=NHBW-J+II
 7      SUM=SUM+AA(J,KK,IES)*AKI(J,IES)
 6      AKI(II,IES)=(AKI(II,IES)-SUM)/AA(II,NHBW,IES)
 5      CONTINUE
        RETURN
        END
```

```
A>TYPE OUTPUT.FOR
      SUBROUTINE OUTPUT(NGT,NBP)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON D(2,10),DEKES(2,10),FEKES(2,10),SEKES(2,2,10),SFIS(2)
     * ,SKAY(2,10),RM(10),RMEL(10),REL(70,5),R(70),AA(70,5,2),
     * SAD(2),ZE(70,2),SSS(70,2),FSS(70,2),AKI(70,2)
      PAUSE 1
      WRITE(2,1)
 1    FORMAT(12(/),20X,'NOTRON AKI DAGILIMI',//)
      DO 2 IEG=1,NGT
      PAUSE 1
      WRITE(2,3) IEG
 3    FORMAT(//,10X,'ENERJI GURUP NUMARASI=',I3)
      WRITE(2,4)
 4    FORMAT(/,4X,'NOD',8X,'AKI')
      WRITE(2,5) (I,AKI(I,IEG),I=1,NBP)
 5    FORMAT(I7,E14.6)
 2    CONTINUE
      RETURN
      END
```

Ek.2

ÖRNEK BİLGİSAYAR ÇIKIŞI

GIRDI LISTESI

GURUP SAYISI= 2

MALZEME SAYISI= 2

SERBEST NOTRON KAYNAGI= 0

FISIL MALZEME= 1

ELEMAN MERTEBESI= 4

ELEMAN SAYISI= 6

BOSLUK SINIR KOSULU= 1

MAXIMUM ITERASYON SAYISI= 100

YAKINSAMA PARAMETRESI= .000010

KEFF ONGORUSU= 1.000000

FISYON NORMALISASYON PARAMETRESI= 1.000000

TESIR KESITLERI LISTESI

MALZEME NOSU 1

DIFUZYON SABITLERI

.096185 .172917

CIKARTMA TESIR KESITLERI

.066210 .116022

SACILMA TESIR KESITLERI

ENERJI GRUBU 2
.062567

FISYON TESIR KESITLERI

.008120 .178400

MALZEME NOSU 2

DIFUZYON SABITLERI

.782400 .612670

CIKARTMA TESIR KESITLERI

.015520 .006042

SACILMA TESIR KESITLERI

ENERJI GRUBU 2
.010050

FISYON TESIR KESITLERI

0.000000 0.000000

FISYON SPEKTRUMU

1.000000 0.000000

MALZEME YARICAPI VE MALZEMEDEKI ELEMAN SAYILARI

MALZEME NOSU	MALZEME DIS YARICAPI	MALZEMEDEKI ELEMAN SAYISI
1	18.509998	3
2	54.609997	3

ELEMANLARA AIT BILGILER

1 INCI NOD	2 INCI NOD	3 UNCU NOD	4 UNCU NOD	5 INCI NOD
1.	2.	3.	4.	5.
5.	6.	7.	8.	9.
9.	10.	11.	12.	13.
13.	14.	15.	16.	17.
17.	18.	19.	20.	21.
21.	22.	23.	24.	25.

NOD KOORDINATLARI

NOD NOSU	
1	0.00000
2	1.54250
3	3.08500
4	4.62750
5	6.17000
6	7.71250
7	9.25500
8	10.79750
9	12.34000
10	13.88250
11	15.42500
12	16.96750
13	18.51000
14	21.51833
15	24.52666
16	27.53500
17	30.54333
18	33.55166
19	36.56000
20	39.56833
21	42.57666
22	45.58500
23	48.59333
24	51.60166
25	54.61000

1 INDI GURUP ELEMAN MATRISI

0.0000	0.0000	0.0000	0.0000	.2427
0.0000	0.0000	0.0000	-.4611	3.2566
0.0000	0.0000	.2812	-4.4252	11.1246
0.0000	-.4109	2.8658	-9.2702	13.5559
-.1561	-.9445	2.4084	-5.8614	9.6786
0.0000	0.0000	0.0000	-6.7826	20.6724
0.0000	0.0000	3.1709	-18.1207	33.3727
0.0000	-1.7824	8.5975	-22.9656	30.2765
-.4624	-2.3000	5.1081	-12.1029	19.3572
0.0000	0.0000	0.0000	-13.1061	36.8000
0.0000	0.0000	3.9605	-31.8161	55.6229
0.0000	-3.1456	14.3291	-36.6611	47.1931
-.7807	-3.6714	7.9877	-18.5064	20.9235
0.0000	0.0000	0.0000	-6.9454	25.0878
0.0000	0.0000	4.1086	-22.1830	30.1802
0.0000	-2.2946	10.1435	-26.3023	33.9723
-.5622	-2.6731	5.8976	-13.6892	21.2884
0.0000	0.0000	0.0000	-14.4079	39.5759
0.0000	0.0000	6.5633	-34.0991	58.4029
0.0000	-3.4012	15.1201	-38.3084	48.4624
-.8382	-3.8657	8.3522	-19.2418	29.6755
0.0000	0.0000	0.0000	-22.0535	54.0640
0.0000	0.0000	9.0179	-46.0151	77.6256
0.0000	-4.5977	20.0967	-52.2245	62.9465
1.1130	-5.0662	10.8069	-34.7943	46.0207

8 INCI GURUP ELEMAN MATRISI

0.0000	0.0000	0.0000	0.0000	.0762
0.0000	0.0000	0.0000	-.0708	.0881
0.0000	0.0000	.0740	-.9228	2.7820
0.0000	-.0920	.6415	-1.9911	3.5703
.0206	-.1252	.3514	-.9509	2.8677
0.0000	0.0000	0.0000	-1.1085	5.3464
0.0000	0.0000	.4994	-3.8268	6.3490
0.0000	-.3192	1.9246	-4.9051	8.0287
.0619	-.2626	.7760	-1.9206	4.5254
0.0000	0.0000	0.0000	-2.1282	9.8048
0.0000	0.0000	.9248	-6.7500	13.9152
0.0000	-.5465	3.2076	-7.8191	12.4670
.1032	-.5899	1.2022	-3.0103	8.5645
0.0000	0.0000	0.0000	-7.0002	19.2466
0.0000	0.0000	3.2589	-17.2048	20.2094
0.0000	-1.7367	7.8845	-20.5001	26.0772
.4400	-2.1161	4.6658	-10.8259	16.4047
0.0000	0.0000	0.0000	-11.4755	30.3675
0.0000	0.0000	5.2079	-26.5984	45.0306
0.0000	-2.6848	11.7528	-29.0727	27.1961
.6692	-3.0642	6.6347	-15.2222	22.9936
0.0000	0.0000	0.0000	-15.8706	41.4664
0.0000	0.0000	7.1566	-35.8920	59.0522
0.0000	-3.6309	15.6212	-39.1672	46.3151
.8895	-4.0102	8.5836	-19.6275	41.8194

ITERASYON SAYISI= 1

KEFF= .10000E+01 YAKINSAMA PARAMETRESI= .80864E-01
ITERASYON SAYISI= 2

KEFF= .11012E+01 YAKINSAMA PARAMETRESI= .10501E-01
ITERASYON SAYISI= 3

KEFF= .11071E+01 YAKINSAMA PARAMETRESI= .52727E-02
ITERASYON SAYISI= 4

KEFF= .11105E+01 YAKINSAMA PARAMETRESI= .30559E-02
ITERASYON SAYISI= 5

KEFF= .11126E+01 YAKINSAMA PARAMETRESI= .18769E-02
ITERASYON SAYISI= 6

KEFF= .11139E+01 YAKINSAMA PARAMETRESI= .11858E-02
ITERASYON SAYISI= 7

KEFF= .11147E+01 YAKINSAMA PARAMETRESI= .75891E-03
ITERASYON SAYISI= 8

KEFF= .11153E+01 YAKINSAMA PARAMETRESI= .46082E-03
ITERASYON SAYISI= 9

KEFF= .11157E+01 YAKINSAMA PARAMETRESI= .31786E-03
ITERASYON SAYISI= 10

KEFF= .11159E+01 YAKINSAMA PARAMETRESI= .20621E-03
ITERASYON SAYISI= 11

KEFF= .11160E+01 YAKINSAMA PARAMETRESI= .13405E-03
ITERASYON SAYISI= 12

KEFF= .11161E+01 YAKINSAMA PARAMETRESI= .87141E-04
ITERASYON SAYISI= 13

KEFF= .11162E+01 YAKINSAMA PARAMETRESI= .56658E-04
ITERASYON SAYISI= 14

KEFF= .11162E+01 YAKINSAMA PARAMETRESI= .36843E-04
ITERASYON SAYISI= 15

KEFF= .11162E+01 YAKINSAMA PARAMETRESI= .23958E-04
ITERASYON SAYISI= 16

KEFF= .11162E+01 YAKINSAMA PARAMETRESI= .15581E-04
ITERASYON SAYISI= 17

KEFF= .11163E+01 YAKINSAMA PARAMETRESI= .10132E-04
ITERASYON SAYISI= 18

KEFF= .11163E+01 YAKINSAMA PARAMETRESI= .65889E-05

NOTRON AKI DAGILIMI

ENERJI GURUP NUMARASI= 1

NOD	AKI
1	.815480E-01
2	.813256E-01
3	.799520E-01
4	.779590E-01
5	.751836E-01
6	.717172E-01
7	.674715E-01
8	.627502E-01
9	.574325E-01
10	.516786E-01
11	.454677E-01
12	.389906E-01
13	.321092E-01
14	.195822E-01
15	.120578E-01
16	.747907E-02
17	.462326E-02
18	.290852E-02
19	.182348E-02
20	.114377E-02
21	.713224E-03
22	.430342E-03
23	.256697E-03
24	.135607E-03
25	.432887E-04

ENERJI GURUP NUMARASI= 2

NOD	AKI
1	.436802E-01
2	.441490E-01
3	.431535E-01
4	.421586E-01
5	.406024E-01
6	.388530E-01
7	.364055E-01
8	.339666E-01
9	.310405E-01
10	.280314E-01
11	.246433E-01
12	.215543E-01
13	.188853E-01
14	.178567E-01
15	.142585E-01
16	.113762E-01
17	.884352E-02
18	.672799E-02
19	.502314E-02
20	.368099E-02
21	.264432E-02
22	.182472E-02
23	.117484E-02
24	.639336E-03
25	.178219E-03

Ek.3

KAREKÖKLÜ CHOLOSKY YÖNTEMİYLE LİNEER DENKLEM SİSTEMİ ÇÖZÜMÜ

A matrisi n boyutunda simetrik ve pozitif kesin bir matris olsun. A matrisi bütün $x \in R^n$ ve $x \neq 0$ koşullarıyle

$$(Ax,x) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j > 0$$

şartını sağlıyorsa pozitif kesindir.

Matris indirgenemeyen köşegen hakim, köşegen elemanları poz - itif ve gerçel simetrik ise pozitif kesin bir matristir.

$A.x = f$ lineer denklem sisteminde

A matrisi pozitif kesin bir matris ise,

L üst üçgen ve L^T alt üçgen matrisler olmak üzere $A.x = f$ lineer denklem sistemi $L.L^T.x = f$ şeklinde yazılabilir.

$L.y = f$ denirse

$y = L^T.x$ bulunur.

$L.y = f$ denklem sisiteminden y bilinmeyenleri bulunur ve diğer denklemde yerine yazılarak x bilinmeyen vektörü elde edilir.

L ve L^t matrislerinin elemanları aşağıdaki şekilde belirlenir.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1i} & \cdots & a_{1n} \\ a_{12} & a_{22} & a_{23} & \cdots & a_{2i} & \cdots & a_{2n} \\ \vdots & & & & & & \vdots \\ a_{1i} & a_{2i} & a_{3i} & & a_{ii} & & a_{in} \\ \vdots & & & & & & \vdots \\ a_{1n} & a_{2n} & a_{3n} & \cdots & a_{in} & \cdots & a_{nn} \end{bmatrix}$$

$$L = \begin{bmatrix} l_{11} & & & & & & \\ l_{12} & l_{22} & & & & & \\ l_{13} & l_{23} & l_{33} & & & & \\ \vdots & & & & & & \\ l_{1i} & l_{2i} & l_{3i} & \cdots & l_{ii} & & \\ \vdots & & & & & & \\ l_{1n} & l_{2n} & l_{3n} & \cdots & & & l_{nn} \end{bmatrix}$$

$A = L.L^t$ matrisyel eşitliğinden, L ve L^t matrislerinin elemanları A matrisinin elemanları cinsinden bulunabilir.

$$l_{ii} = \sqrt{a_{ii} - \sum_{j=1}^{j-1} l_{ji}^2}$$

$$l_{ji} = \frac{1}{l_{jj}} \left[a_{ji} - \sum_{k=1}^{j-1} l_{ki} l_{kj} \right] \qquad j < i$$

$L.y = f$ denklem sisteminin çözümü ile y bilinmeyenleri belirlenir.

$$y_i = \frac{1}{l_{ii}} \left[f_i - \sum_{j=1}^{i-1} l_{ij}\, y_j \right] \qquad i = 1,2 \;\ldots\; n$$

$L^t.x = y$ denklem sisteminden x bilinmeyenleri bulunur.

$$x_i = \frac{1}{l_{ii}} \left[y_i - \sum_{j=i+1}^{n} l_{ij}\, x_j \right] \qquad i = n, n-1, \;\ldots\; 1$$

KAYNAKLAR

/1/ B.Özgener , Difüzyon teörisi hesaplarında sonlu elemanlar yöntemi İTÜ NEE 1983

/2/ A.Kavas , Sonlu elemanlar yöntemiyle silindirsel geometride çok guruplu difüzyon denklemleri sayısal çözümleri İTÜ NEE M.M.L.S tezi 1984

/3/ A.Özgener , Reaktör fiziğinde sayısal analiz ders notları 1984

/4/ J.Duderstadt,L.Hamilton , Nuclear reactor analysis John Wiley Sons Inc 1976

/5/ A.Anacan, A.Yücel,T.Yarman , İTÜ TRIGA MARK-II reaktörü fizik hesapları

/6/ Ü.Yarımağan , FORTRAN IV

T E Ş E K K Ü R

Yüksek lisans tezimi yöneten , çalışmalarımda büyük yardımlarını gördüğüm Sayın Yrd.Doç.Dr Atilla Özgener'e teşekkür ederim.