

**FİZ433 FİZİKTE BİLGİSAYAR UYGULAMALARI**  
**(DERS NOTLARI)**

**Hazırlayan:**

**Prof.Dr. Orhan ÇAKIR**

**Ankara Üniversitesi, Fen Fakültesi, Fizik Bölümü**

**Ankara, 2017**

## İÇİNDEKİLER

1. LİNEER OLMAYAN DENKLEMLERİN KÖKLERİNİN BULUNMASI I/II
2. LİNEER DENKLEM SİSTEMLERİNİN ÇÖZÜLMESİ I/II
3. UYGUN EĞRİNİN BULUNMASI VE INTERPOLASYON I/II
4. SAYISAL İNTEGRAL HESAPLARI I/II
- 5. DİFERENSİYEL DENKLEMLERİN ÇÖZÜMLERİ I/II**
6. BENZETİM I/II
7. FİZİKTE SEMBOLİK HESAPLAMA I/II

EKLER

KAYNAKLAR

# KONU 10

## DİFERENSİYEL DENKLEMLERİN SAYISAL ÇÖZÜMLERİ II

### Runge-Kutta Yöntemleri

Runge-Kutta yöntemi, Euler yöntemindeki ortalama eğim hesaplama işlemlerini bir adım daha ileri götürür. Lineer interpolasyonda kullanılan eğim, aralığın sol ve sağ sınırındaki eğimler ile aralığın içinde bulunan birkaç nokta alınarak ağırlıklı ortalama olarak hesaplanır.

Eğimler için kullanılan bölmeleme ve bağıl ağırlıklar için birçok alternatif bulunmaktadır, böylece Runge-Kutta yöntemi birinci mertebeden adi diferensiyel denklemlerin çözüm yöntemi olarak bilinir.

Bütün bu yöntemler, fonksiyonun Taylor serisi açılımında  $h^k$  ( $k$  yöntemin mertebesi olmak üzere) merteye terimlerine kadar uyumludur. Bundan dolayı ikinci, üçüncü, dördüncü vs. mertebeden Runge-Kutta yöntemleri vardır. İkinci mertebeden yöntemler  $h^2$ 'li terime kadar bütün terimlere sahiptir. Dördüncü mertebeden Runge-Kutta yöntemi ise  $h^4$ 'lü terime kadar bütün terimlere sahiptir. Ayrıca, geliştirilmiş Euler yöntemi ve değiştirilmiş Euler yöntemi ikinci merteye Runge-Kutta yönteminin özel durumları olduğu gösterilebilir.

### Dördüncü mertebeden Runge-Kutta yöntemi

Bir adi diferensiyel denklemi çözerken en çok kullanılan yöntem dördüncü mertebeden Runge-Kutta yöntemidir. Dördüncü merteye yöntemi için  $y_{i+1}$  aşağıda verilen ifadeden elde edilir.

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Bu denklemde  $k_i$  değerleri elde edilen  $y$  değerindeki tahmini artışı ifade etmektedir.

$$k_1 = hf(x_i, y_i)$$

$$k_2 = hf(x_i + h/2, y_i + k_1/2)$$

$$k_3 = hf(x_i + h/2, y_i + k_2/2)$$

$$k_4 = hf(x_i + h, y_i + k_3)$$

Burada  $k_1$  sol sınırın eğim tahminini,  $k_4$  sağ sınırın eğim tahminini,  $k_2$  ve  $k_3$  ise aradaki iki noktanın eğim tahminlerini göstermektedir.  $k_i$  değerlerinin ağırlıklı ortalaması  $y'$ deki artışı belirlemek için kullanılır ve  $y_{i+1}$  değerinin bulunması için  $y_i$  değerine eklenir.

**Örnek:**  $dy/dx = x + y$  diferensiyel denklemini  $y(0) = 1$  başlangıç koşulu olmak üzere  $x=0.2, 0.4$  değerleri için dördüncü mertebeden Runge-Kutta yöntemiyle çözünüz.

**Çözüm:** Runge-Kutta yönteminde verilen denklemler kullanılır:

$$i=0 \text{ için } y_1 = y_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \text{ hesaplanmalıdır. Bunun için}$$

$$k_1 = hf(x_0, y_0) = 0.2[f(0.0, 1.0)] = 0.2(1.0) = 0.2,$$

$$k_2 = hf(x_0 + h/2, y_0 + k_1/2) = 0.2[f(0.1, 1.0 + 0.2/2)] = 0.2f(0.1, 1.1) = 0.2(1.2) = 0.24,$$

$$k_3 = hf(x_0 + h/2, y_0 + k_2/2) = 0.2[f(0.1, 1.0 + 0.24/2)] = 0.2f(0.1, 1.12) = 0.2(1.22) = 0.244,$$

$$k_4 = hf(x_0 + h, y_0 + k_3) = 0.2[f(0.2, 1.0 + 0.244)] = 0.2f(0.2, 1.244) = 0.2(1.444) = 0.2888.$$

Sonuç olarak,  $y_1 = 2.0 + [0.2 + 2(0.24) + 2(0.244) + 0.2888]/6 = 1.2428$  elde edilir.

Benzer şekilde  $i=1$  için,  $y_2 = y_1 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$  hesaplanmalıdır.

$$k_1 = hf(x_1, y_1) = 0.2[f(0.2, 1.2428)] = 0.2(1.4428) = 0.28856,$$

$$k_2 = hf(x_1 + h/2, y_1 + k_1/2) = 0.2[f(0.2 + 0.1, 1.2428 + 0.28856/2)] \\ = 0.2f(0.3, 1.38708) = 0.2(1.68708) = 0.337416,$$

$$k_3 = hf(x_1 + h/2, y_1 + k_2/2) = 0.2[f(0.3, 1.2428 + 0.337416/2)] \\ = 0.2f(0.3, 1.41151) = 0.2(1.71151) = 0.342302,$$

$$k_4 = hf(x_1 + h, y_1 + k_3) = 0.2[f(0.4, 1.2428 + 0.342302)] \\ = 0.2f(0.4, 1.5851) = 0.2(1.9851) = 0.39702,$$

Sonuç olarak,  $y_2 = 1.2428 + [0.28856 + 2(0.337416) + 2(0.342302) + 0.39702]/6 = 1.58364$  elde edilir.

Bu örnekteki problemin çözümü için hazırlanmış program aşağıda verilmiştir.

- **FORTRAN programı**  
program runge\_kutta  
implicit real\*8 (a-h,o-z)  
parameter (n=5)  
dimension x(0:n),y(0:n)  
x0=0.d0  
xn=1.d0  
y0=1.d0  
write(\*,\*)"x(i) ", " y(i)", " y(x)"  
call RK4(x0,xn,y0,n,x,y)  
do i=0,n

```

open(1,file="runge_kutta.txt")
write(1,*)x(i),y(i),fonk(x(i))
write(*,*)x(i),y(i),fonk(x(i))
enddo
end

```

```

subroutine RK4(x0,xn,y0,n,x,y)
implicit real*8 (a-h,o-z)
dimension x(0:n),y(0:n)
h=(xn-x0)/n
x(0)=x0
y(0)=y0
do i=0,n-1
x(i+1)=x(i)+h
xk1=h*f(x(i),y(i))
xk2=h*f(x(i)+h/2.d0,y(i)+xk1/2.d0)
xk3=h*f(x(i)+h/2.d0,y(i)+xk2/2.d0)
xk4=h*f(x(i)+h,y(i)+xk3)
y(i+1)=y(i)+(xk1+2.d0*(xk2+xk3)+xk4)/6.d0
enddo
return
end

```

```

function f(x,y)
implicit real*8 (a-h,o-z)
f=x+y
return

```

```

end

function fonk(x)
implicit real*8 (a-h,o-z)
fonk=-1.d0-x+2.d0*exp(x)
return
end

```

Program çalıştırıldığında  $x_i$  değerlerine karşı gelen Runge-Kutta yönteminden hesaplanan  $y_i$  değerleri elde edilmektedir. Çizelge 5.2’de sonuçları karşılaştırmak için analitik çözümden elde edilen  $y(x_i)$  değerleri de verilmiştir. Runge-Kutta yönteminde Çizelge 5.2’den görüldüğü gibi, Euler yöntemine göre oldukça duyarlı sonuçlar elde edilmektedir. Bu yöntemde, bu örnek için hatanın mertebesi  $h^5=0.2^5=0.00032=3.2 \times 10^{-4}$  ile verilir.

Çizelge: Runge-Kutta yöntemi ile  $y'=x+y$  denkleminin çözümünden elde edilen değerler

| $x_i$ | $y_i$      | $y(x_i)$   |
|-------|------------|------------|
| 0.    | 1.         | 1.         |
| 0.2   | 1.2428     | 1.24280552 |
| 0.4   | 1.58363592 | 1.5836494  |
| 0.6   | 2.04421291 | 2.0442376  |
| 0.8   | 2.65104165 | 2.65108186 |
| 1.    | 3.43650227 | 3.43656366 |

## Runge-Kutta yöntemlerinin bazı özellikleri

- i. Kendiliğinden başlar,  $y_{i+1}$  değerini belirlemek için sadece bir önceki  $(x_i, y_i)$  noktasından yararlanır.

ii. Fonksiyonun hiçbir türevini hesaplamaya gerek duymaz.

iii. Fonksiyonun  $h^n$  'inci ( $n$  yöntemin derecesidir) terim dahil olmak üzere Taylor serisi açılımına uyumludur.

iv. Kolay programı yazılabilir, çok kararlı ve doğrudur. Daha önce de bahsedildiği gibi dördüncü mertebeden Runge-Kutta yöntemi için kesme hatası  $O(h^5)$  mertebesindedir.

v. Runge-Kutta yöntemi kendi kendine başlayabilme özelliğine sahip olduğundan  $h$  adım aralığını değiştirmek kolaydır.

Dördüncü mertebeden Runge-Kutta yöntemi için bir dezavantaj,  $y$ 'nin yeni bir değerini hesaplamak için  $f(x, y)$  fonksiyonunu dört kez hesaplamak gereklidir. Fonksiyon karmaşık ise bu zaman alıcı olabilir. Bununla birlikte artan doğruluk nedeniyle dördüncü mertebeden Runge-Kutta yöntemi tercih edilir. Yüksek duyarlılığı nedeniyle, büyük adım aralıklarında bile kullanmak mümkündür. Kendi kendine başlama özelliği sayesinde Runge-Kutta yöntemi kullanılarak başlangıç çözümü elde edilebilir ve bundan sonra da tahmin-düzeltilme yöntemi kullanılarak çözümün doğruluğu artırılabilir.

Bazı problemlerin sayısal çözümü için adım aralığının otomatik olarak değiştirildiği yöntemleri kullanmak faydalı olacaktır. Çünkü çözüm bazı bölgelerde hızlı değişim gösterebilir, burada iyi duyarlılık elde edebilmek için küçük adım aralığı gerekir. Sabit ve çok küçük adım aralığı da hesaplamalarda uzun zaman alacağından, adım aralığının kontrolü önem taşımaktadır. Böyle bir yöntemin uygulanması için yerel kesme hatası her bir adımda tahmin edilir ve buna göre adım aralığı ayarlanır. Bir adımlı yöntemlere adım aralığı kontrolünü vermek için iki temel yaklaşım vardır. Bunlardan birincisi, aynı mertebeli fakat farklı adım aralıklı Runge-Kutta yöntemini kullanarak iki tahmin arasındaki fark olarak hata hesabıdır. İkincisi, farklı mertebeli Runge-Kutta yöntemi kullanarak iki tahmin arasındaki fark olarak yerel kesme hatasının hesabıdır. Adım yarılama veya adaptive RK yönteminde her bir adım iki defa alınır. Önce tam bir adım alınır ( $y_1$  tahmini), sonra bağımsız iki yarı adım alınır ( $y_2$  tahmini). İki sonucun farkı yerel kesme hatasının tahminini verir ( $\Delta$ ). Adım aralığı kontrolü kriterini eklemek için, dördüncü mertebeli RK yönteminde, düzeltme terimi  $y_2 \leftarrow y_2 + \Delta/15$  şeklinde verilir. Bu tahmin ise beşinci-mertebeli duyarlılığındadır.



## Çok Adımlı Yöntemler

Bütün Runge-Kutta formüllerinin seçkin özelliklerinden biri, tek adımlı yöntemler olmalarıdır. Bir sonraki  $y_{i+1}$  değerini elde ederken sadece bir önceki  $(x_i, y_i)$  noktasındaki bilgidен yararlanılır. Bu işlem Runge-Kutta yöntemine kendi kendine başlayan bir özellik vermesine rağmen, integralleme süreci birkaç adım ilerlediğinde elde edilen ve önceki adımlarda hesaplanan  $y$  değerleri ile ilgili ilave bilgileri kullanamaz.

Çok-adımlı yöntemler birkaç  $y(x)$  değerini kullanarak (örneğin  $y(x_i), y(x_{i-1}), y(x_{i-2}), \dots$ )  $y(x_{i+1})$ 'i tahmin etmeye çalışır. Bir sonraki değeri hesaplamak için  $y(x)$ 'in  $n$  tahmini değerini kullanan yöntem  $n$ -adımlı yöntem denir. Örnek olarak iki-adımlı yöntem iki tane  $y(x)$  değerini gerektirir, yani  $y(x_{i+1})$ 'i belirlemek için  $y(x_i)$  ve  $y(x_{i-1})$  gereklidir. Çok-adımlı yöntem birkaç noktanın bilgisine gerek duyduğundan kendi kendine başlayamaz. Bundan dolayı bütün çok adımlı yöntemlerin başlaması için Runge-Kutta yöntemi gibi tek adımlı yöntemlerle başlangıç değerlerinin hesaplanması gerekmektedir.

Çok-adımlı yöntemlerde temel prensip, önceki  $y$  değerlerini kullanarak  $f(x, y)$  türev fonksiyonunu tanımlayan bir polinom kurmak ve bunu sonraki aralığa extrapolasyon yapmaktır. Polinomun derecesi dikkate alınan noktaların sayısına bağlıdır. İki nokta kullanılırsa tahmini polinom birinci dereceden olacaktır. Üç nokta kullanılırsa tahmini polinom kuadratik olacaktır, dört nokta kullanılırsa tahmini polinom kübik olacaktır ( $f(x, y) = a_0 + a_1x + a_2x^2 + a_3x^3$ ). Kullanılan nokta sayısı arttıkça alınan polinomun mertebesi ve doğruluğu artacaktır.

### Adams Yöntemi

Çok-adımlı yöntemlere bir örnek Adams yöntemidir. İkinci mertebeli Adams yöntemi üç noktadan geçen ikinci dereceden bir polinom ile  $f(x, y)$  fonksiyonunu tahmin eder, Şekil 5.6.  $dy/dx = f(x, y)$  diferensiyel denklemini sınırları  $x_i$ 'den  $x_{i+1}$ 'e olan bir integral eşitliği şeklinde yazarız.

$$\int_{y_i}^{y_{i+1}} dy = \int_{x_i}^{x_{i+1}} f(x, y) dx \approx \int_{x_i}^{x_{i+1}} P_2(x) dx$$

Burada  $P_2(x) = a_0 + a_1x + a_2x^2$  yazılırsa;

$$\int_{y_i}^{y_{i+1}} dy = \int_{x_i}^{x_{i+1}} (a_0 + a_1x + a_2x^2) dx \Rightarrow y_{i+1} - y_i = a_0x + a_1 \frac{x^2}{2} + a_2 \frac{x^3}{3} \Big|_{x_i}^{x_{i+1}}$$

Yukarıdaki integral, her aralıkta yeniden hesaplandığı için sınır noktalarının konumundan etkilenmez ve integralin sınırları  $x_i = 0$  ve  $x_{i+1} = h$  alınabilir (burada aralıklar eşit kabul edilmiştir).

$$y_{i+1} = y_i + a_0h + a_1 \frac{h^2}{2} + a_2 \frac{h^3}{3}$$

Bilinmeyen  $a_0$ ,  $a_1$  ve  $a_2$  katsayılarını aşağıdaki denklemlerden hesaplayabiliriz.

$$x = 0 \text{ için } f(x, y) = f_i = a_0$$

$$x = -h \text{ için } f(x, y) = f_{i-1} = a_0 - a_1h + a_2h^2$$

$$x = -2h \text{ için } f(x, y) = f_{i-2} = a_0 - 2a_1h + 4a_2h^2$$

Yukarıdaki denklem sistemi çözülürse,

$$a_0 = f_i$$

$$a_1 = \frac{1}{2h}(3f_i - 4f_{i-1} + f_{i-2})$$

$$a_2 = \frac{1}{2h^2}(f_i - 2f_{i-1} + f_{i-2})$$

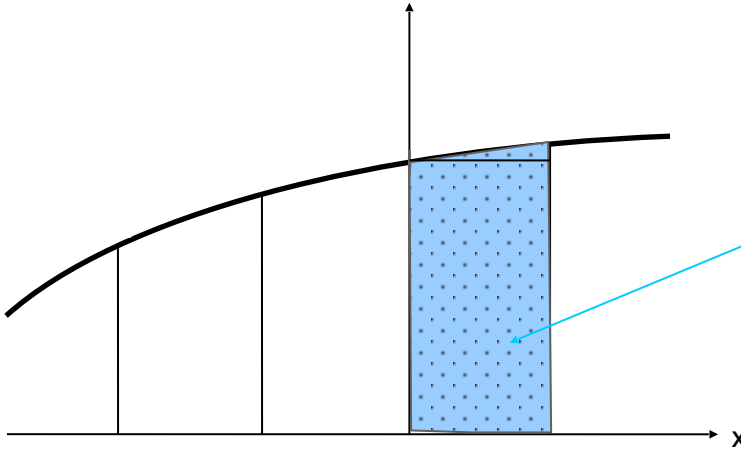
bulunur. Yukarıdaki katsayılar yerine yazılıp hesaplanırsa, Adams yöntemi elde edilir:

$$y_{i+1} = y_i + \frac{h}{12}(5f_{i-2} - 16f_{i-1} + 23f_i)$$

Eğer  $f(x, y)$  fonksiyonu  $x_{i-3}$ ,  $x_{i-2}$ ,  $x_{i-1}$  ve  $x_i$  noktaları gibi dört noktadan geçerse, o zaman ikinci mertebeden Adams yöntemine benzer bir yaklaşım kullanılarak aşağıda verilen denkleme göre  $y_{i+1}$  değerini belirleyebiliriz.

$$y_{i+1} = y_i + \frac{h}{24}(-9f_{i-3} + 37f_{i-2} - 59f_{i-1} + 55f_i) \quad (45)$$

Bu denklem aynı zamanda Adams-Bashforth formülü olarak da bilinir. Bu denkleme karşı gelen hatanın mertebesi  $O(h^5)$ 'dir.



Şekil 5.6 İkinci mertebeden Adams yönteminin grafiksel gösterimi

## Tahmin-Düzeltilme Yöntemleri

Tahmin-düzeltilme yöntemleri çok adımlı yöntemlerdir ve birkaç noktaya ait bilgilerden faydalanırlar. Bu yöntemde  $y_{i+1}$  değeri verilen bir formülle tahmin edilir ve bu değer başka bir formülle düzeltilir. Düzeltilme formülü yeterli duyarlık elde edilinceye kadar birkaç kez uygulanır. Bu nedenle tahmin-düzeltilme yöntemleri tekrarlama (iterasyon) işlemleri gerçekleştirir. Önceki noktalar hakkında bilgi gerektirir ve kendi kendine başlayan bir yöntem değildir. Bundan dolayı tahmin-düzeltilme yöntemleri diğer yöntemlerle (örneğin Runge-Kutta yöntemi) birleştirilerek kullanılır.

En basit tahmin-düzeltilme yöntemi geliştirilmiş Euler yöntemidir. Bu yöntemdeki tahmini  $y_{i+1}$  değeri, temel Euler yöntemiyle hesaplanır. Tahmin denklemi,

$$y_{i+1}^* = y_i + f(x_i, y_i)h$$

ile verilir ve düzeltilme denklemi

$$y_{i+1} = y_i + \frac{1}{2}h[f(x_i, y_i) + f(x_{i+1}, y_{i+1}^*)]$$

ile verilir.

## Milne-Simpson Yöntemi

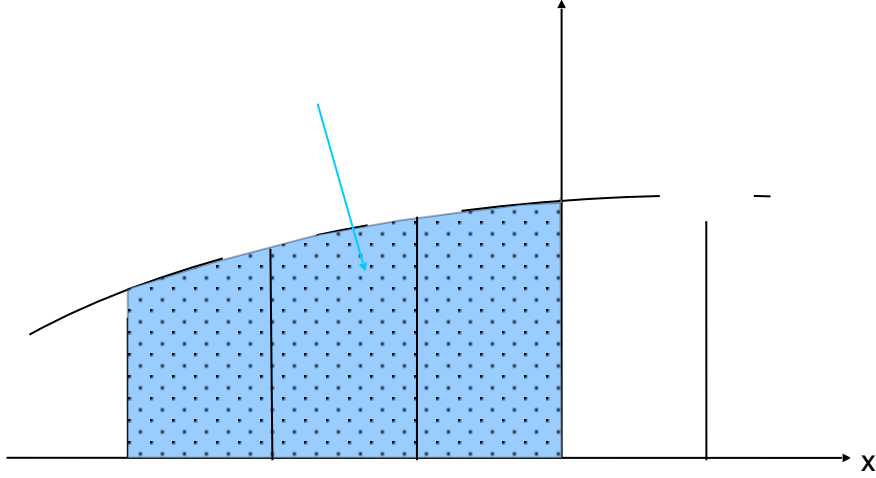
En iyi bilinen tahmin-düzeltilme yöntemlerinden biri Milne-Simpson yöntemidir, Şekil 5.7 ve 5.8.

Bu yöntem ilk önce türev değerlerini extrapole ederek  $y_{i+1}$  değerini tahmin eder. Bir sonraki adıma geçmeden önce tahmin edilen bu değer düzeltilir. Bu yöntem  $x_{i-3}$ ,  $x_{i-2}$ ,  $x_{i-1}$  ve  $x_i$

noktalarında  $y$ 'nin dört adet başlangıç değerini kullanır. Bu noktalarda  $x$  ve  $y$  değerleri bilindiğinden  $f(x, y)$  fonksiyonunu hesaplayabiliriz. Burada quadrature formülü,  $dy/dx = f(x, y)$  diferensiyel denklemini  $x_{i-3}$  ile  $x_{i+1}$  sınırları arasında integre etmek için kullanılır.

$$\int_{y_{i-3}}^{y_{i+1}} dy = \int_{x_{i-3}}^{x_{i+1}} f(x, y) dx = \int_{x_{i-3}}^{x_{i+1}} P_2(x) dx$$

quadratic interpolasyon fonksiyonu  $P_2(x) = a_0 + a_1x + a_2x^2$  yerine yazılırsa,  $x_{i-3} = -3h$  ve  $x_{i+1} = h$  alınır,  $x = x_{i+1}$  için tahmin edilen (extrapole edilen)  $y$  değeri;



Şekil 5.7 Milne tahmin yönteminin grafiksel gösterimi

$$y_{i+1} = y_{i-3} + \int_{-3h}^h (a_0 + a_1x + a_2x^2) dx$$

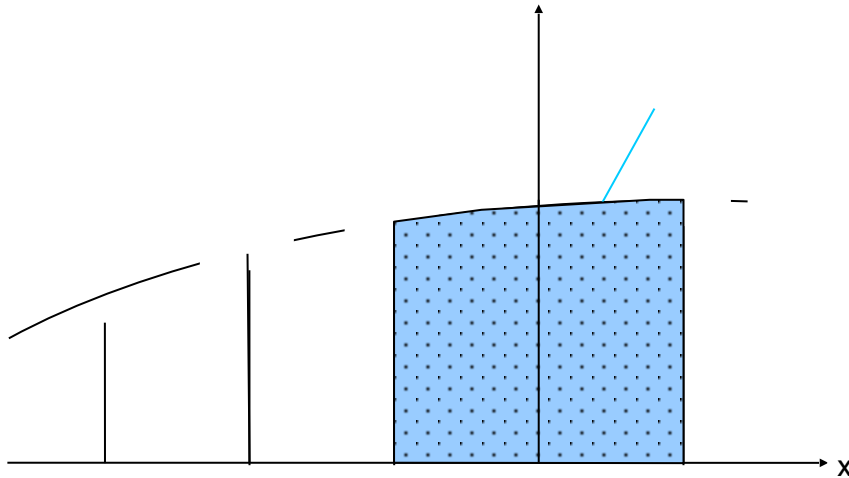
ile verilir. Bilinmeyen  $a_0$ ,  $a_1$  ve  $a_2$  katsayıları, interpolasyon fonksiyonun  $x_{i-2}$ ,  $x_{i-1}$  ve  $x_i$  noktalarına uyumuyla elde edilir. Bu işlem daha önceden ikinci mertebeden Adams yöntemi için yapılmıştı. Bu değerler yerine yazılıp integral hesaplanırsa  $y_{i+1}$  tahmini için Milne formülü elde edilir.

$$y_{i+1}^p = y_{i-3} + \frac{4}{3}h(2f_{i-2} - f_{i-1} + 2f_i) + \frac{28}{90}h^5 y^5$$

Milne düzeltme formülü ise, ikinci dereceden polinom  $x_{i-1}$ ,  $x_i$  ve  $x_{i+1}$  noktalarından geçecek şekilde elde edilir. Burada integral aşağıdaki gibi hesaplanır.

$$\int_{y_{i-1}}^{y_{i+1}} f(x, y) dx = \int_{x_{i-1}}^{x_{i+1}} P_2(x) dx = \int_{x_{i-1}}^{x_{i+1}} (a_0 + a_1x + a_2x^2) dx$$

Burada  $f(x, y)$  fonksiyonu yerine yine integral sınırları  $x_i = -h$  ve  $x_{i+1} = h$  olan quadratic fonksiyon kullanılmıştır.



Şekil 5.8 Milne düzeltme yönteminin grafiksel gösterimi

Yukarıdaki integralin Simpson'un bir-bölü-üç kuralına eşit olduğuna dikkat ediniz. Bilinmeyen  $a_0$ ,  $a_1$  ve  $a_2$  katsayıları çözülerek aşağıda verilen düzeltme formülü elde edilir;

$$y_{i+1}^C = y_{i-1} + \frac{h}{3}(f_{i-1} + 4f_i + f_{i+1}) - \frac{1}{90}h^5 y^5$$

Bu denklemde tahmin formülünden elde edilen  $y_{i+1}$  kullanılarak  $f_{i+1}$  değeri hesaplanır. Tahmin (predictor) denklemi ile düzeltme (corrector) denkleminin aynı olmadığına dikkat ediniz (aynı üç noktada fonksiyonu fit etmiyor). Ayrıca tahmin ve düzeltme formülleri için integral sınırları da farklıdır.

Tahmin ve düzeltme denklemlerinin son terimleri  $y_{i+1}$ 'in hesaplanmasında kullanılmaz. Bu terimler hesaplanan  $y_{i+1}$  değeri ile ilişkili hata terimleridir. Düzeltme formülündeki hatanın katsayısı (1/90) tahmin formülündekinden (28/90) çok daha küçüktür.

Genel olarak, tahmin ve düzeltme formüllerinden hesaplanan  $y_{i+1}$  değeri farklı olacaktır. Gerçek değer bu iki değer arasında bulunur ve düzeltme formülü ile elde edilen değere yakındır. Tahmin ve düzeltme değerleri arasındaki fark duyarlılık için basit bir tahmin verir ve  $h$  adım aralığını ayarlamak için kullanılabilir. Yeni  $y_{i+1}$  değerlerini düzeltme formülünü birkaç kez uygulanarak hesaplayabilir ve yakınsaklığı test edebiliriz. Buna rağmen, çoğu mühendislik problemlerinde yakınsaklık bir veya iki iterasyonda elde edilir, böylece bu yöntemleri uygulayan programların çoğu genellikle sadece bir çift iterasyon gerçekleştirir.

Milne yöntemi basit oluşu ve iyi duyarlılık (yerel hata  $O(h^5)$ ) vermesine rağmen, bazı durumlarda kararsızlık problemiyle karşılaşır. Özellikle çözümlerin üstel olduğu durumlarda yayılan hatalar da üstel olarak büyür. Çoğu problemler için diferensiyel denklemin çözümü önceden bilinmediğinden Milne yönteminin kullanılması pek fazla önerilmez.

## Adams-Moulton Yöntemi

Milne yöntemindeki kararsızlık probleminden etkilenmeyen tahmin-düzeltilme yöntemi Adams-Moulton yöntemidir. Bu yöntem  $y_{i+1}$  değerini tahmin ederken Adams-Bansforth formülünü kullanır.  $f_{i-3}$ ,  $f_{i-2}$ ,  $f_{i-1}$  ve  $f_i$  cinsinden tahmin eden formül;

$$y_{i+1}^p = y_i + \frac{h}{24}(-9f_{i-3} + 37f_{i-2} - 59f_{i-1} + 55f_i)$$

Tahmin edilen değer aşağıda verilen Adams-Moulton düzeltme formülü ile düzeltilir.

$$y_{i+1}^c = y_i + \frac{h}{24}(f_{i-2} - 5f_{i-1} + 19f_i + 9f_{i+1})$$

Bu yöntemi başlatmak için dört değere ihtiyaç duyulduğuna dikkat ediniz. Burada yerel hata  $O(h^5)$  mertebesindedir. Bu yöntem Milne-Simpson yöntemi gibi kararsızlık probleminden etkilenmediğinden kullanım alanı daha geniştir.

Hem Adams-Moulton yöntemi hem de Milne-Simpson yöntemi Runge-Kutta yönteminden iki kat daha verimlidir, çünkü bunlar adım başına fonksiyonun iki defa hesaplanmasını gerektirir, halbuki Runge-Kutta yönteminde dört defa hesap yapmak gerekmektedir. Tahmin-düzeltilme yöntemlerinin diğer bir üstünlüğü ise içinde duyarlılık kriterleri bulunmasıdır. Tahmin edilen ve düzeltilen değerler arasındaki fark tahmini duyarlılığı verir ve bu adım aralığını ayarlamak için kullanılabilir.

Tek adımlı yöntemlerle çok adımlı yöntemler birbirlerini tamamladığından diferensiyel denklemleri çözen pratik bir program her iki tekniği de kullanmalıdır.

- $y_1$ ,  $y_2$  ve  $y_3$ 'ü bulmak için Runge-Kutta yöntemiyle çözüme başlanır.
- Ardışık  $y_4$ ,  $y_5$ ,  $\dots$ ,  $y_n$  değerlerini hesaplamak için Tahmin-Düzeltilme yöntemleri kullanılır.
- İstenilen duyarlık için düzeltme formülünün üç veya daha fazla iterasyonu gerekliyse  $h$  adım aralığı azaltılmalıdır. Adım aralığını değiştirmek için  $y_i$ 'nin son değeri başlangıç



noktası olarak düşünülür ve çözüm bu noktadan Runge-Kutta yöntemi vasıtasıyla yeniden başlatılır.

Diferensiyel denklemleri çözmek için Adams-Moulton tahmin-düzeltilme yöntemini uygulayan program aşağıda verilmiştir.

- FORTRAN program  
program ada\_mou  
implicit real\*8 (a-h,o-z)  
parameter (n=10)  
dimension x(0:n),y(0:n)  
write(\*,\*)"x0, xn, y0"  
read(\*,\*)x0,xn,y0  
call adams\_moulton(x0,xn,y0,n,x,y)  
write(\*,\*)"x(i) ", " y(i)"  
do i=0,n  
open(1,file="ada\_mou.txt")  
write(1,\*)x(i),y(i)  
write(\*,\*)x(i),y(i)  
enddo  
end  
  
subroutine adams\_moulton(x0,xn,y0,n,x,y)  
implicit real\*8 (a-h,o-z)  
dimension x(0:n),y(0:n)  
h=(xn-x0)/n  
xn=x0+3.d0\*h  
call RK4(x0,xn,y0,3,x,y)  
do i=3,n-1

$$x(i+1)=x(i)+h$$

C....Tahmin etme

$$\begin{aligned} y(i+1)=y(i)+h/24.d0*( \\ . -9.d0*f(x(i-3),y(i-3)) \\ . +37.d0*f(x(i-2),y(i-2)) \\ . -59.d0*f(x(i-1),y(i-1)) \\ . +55.d0*f(x(i),y(i)) \end{aligned}$$

C.....Duzeltme

$$\begin{aligned} y(i+1)=y(i)+h/24.d0*( \\ . f(x(i-2),y(i-2)) \\ . -5.d0*f(x(i-1),y(i-1)) \\ . +19.d0*f(x(i),y(i)) \\ . +9.d0*f(x(i+1),y(i+1))) \end{aligned}$$

enddo

return

end

subroutine RK4(x0,xn,y0,n,x,y)

implicit real\*8 (a-h,o-z)

dimension x(0:n),y(0:n)

h=(xn-x0)/n

x(0)=x0

y(0)=y0

do i=0,n-1

x(i+1)=x(i)+h

xk1=h\*f(x(i),y(i))

xk2=h\*f(x(i)+h/2.d0,y(i)+xk1/2.d0)

```

xk3=h*f(x(i)+h/2.d0,y(i)+xk2/2.d0)
xk4=h*f(x(i)+h,y(i)+xk3)
y(i+1)=y(i)+(xk1+2.d0*(xk2+xk3)+xk4)/6.d0
enddo
return
end

function f(x,y)
implicit real*8 (a-h,o-z)
f=x+y
return
end

```

Program çalıştırıldığında aşağıdaki sonuçlar elde edilmektedir, Çizelge 5.3. Burada  $y'=x+y$  diferensiyel denklemini,  $x=0$  da  $y=0$  olacak şekilde başlangıç koşulu altında  $x[0:1]$  aralığında, sabit  $h=0.1$  adımlı, çözülmüştür. Elde edilen  $y$  değerleri ile analitik sonuç  $O(h^5)$  mertebesinde duyarlılık göstermektedir.

Çizelge 5.3 Adams-Moulton yöntemi ile  $y'=x+y$  denkleminin çözümünün sonuçları

| $x_i$ | $y_i$      |
|-------|------------|
| 0.0   | 2.         |
| 0.1   | 2.2155125  |
| 0.2   | 2.46420771 |
| 0.3   | 2.74957549 |
| 0.4   | 3.07547362 |
| 0.5   | 3.44616393 |
| 0.6   | 3.86635722 |
| 0.7   | 4.3412598  |

|     |            |
|-----|------------|
| 0.8 | 4.87662548 |
| 0.9 | 5.47881325 |
| 1.0 | 6.15485086 |

## İkinci Mertebeden Diferensiyel Denklemler

İkinci mertebeden veya daha yüksek mertebeden bir diferensiyel denklem, yardımcı bağımlı değişkenler tanımlanarak birinci mertebeden diferensiyel denklem sistemine indirgenebilir. İndirgenmiş birinci mertebeden diferensiyel denklemler aşağıdaki gibi yazılabilir:

$$y' = f_1(x, y, v)$$

$$v' = f_2(x, y, v)$$

Burada  $v = dy/dx$  olarak tanımlanmıştır. Çözüm için Runge-Kutta yöntemi kullanılırsa;

$$k_1 = hf_1(x_n, y_n, v_n)$$

$$l_1 = hf_2(x_n, y_n, v_n)$$

$$k_2 = hf_1\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, v_n + \frac{l_1}{2}\right)$$

$$l_2 = hf_2\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, v_n + \frac{l_1}{2}\right)$$

$$k_3 = hf_1\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, v_n + \frac{l_2}{2}\right)$$

$$l_3 = hf_2\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, v_n + \frac{l_2}{2}\right)$$

$$k_4 = hf_1(x_n + h, y_n + k_3, v_n + l_3)$$

$$l_4 = hf_1(x_n + h, y_n + k_3, v_n + l_3)$$

ve

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$v_{n+1} = v_n + \frac{1}{6}(l_1 + 2l_2 + 2l_3 + l_4)$$

ifadeleri elde edilir.

**Örnek:** Bir fiziksel örnek olarak Newton'un ikinci hareket yasasını temsil eden diferensiyel denklemi ele alalım:

$$F = ma = m \frac{d^2x}{dt^2}$$

Bu denklemde  $F$  kuvvet,  $m$  kütle ve  $a$  ivmedir. Burada  $x$  yer değiştirmeyi ve  $t$  zamanı göstermek üzere ivme ifadesi  $a = d^2x/dt^2$  şeklinde yazılabilir. Yukarıdaki denklemde  $v = dx/dt$  bağımlı değişken değiştirmesiyle birinci dereceden diferensiyel denkleme indirgenebilir (burada  $v$  hızı göstermektedir).

$$v = \frac{dx}{dt}$$

$$m \frac{dv}{dt} = F$$

Buradaki denklemler, birinci mertebeden denklemler için geliştirilen herhangi bir yöntemle çözülebilir.  $x$  ve  $v$  bağımlı değişkenleri her iki denklemde de bulunduğundan, her iki denklem eş zamanlı çözülmelidir. Böylece her  $t_{i+1} - t_i$  zaman aralığı için  $x_{i+1}$  ve  $v_{i+1}$  değerleri hesaplanmalıdır.

$x$  ve  $v$ 'nin hesaplanmasında başlangıç değerleri gerekmektedir. Eğer  $t$  bağımsız değişkenin bazı değerleri için  $x$  ve  $v$  bağımlı değişken değerleri açıkça belirtilirse, sistem başlangıç değer problemi olarak sınıflandırılır. Eğer  $t$  bağımsız değişkenin bazı değerleri için  $x$  ve  $v$  bağımlı değişken değerleri verilmemişse, bu sistem sınır değer problemi olarak isimlendirilir. Sınır değer problemlerinin sayısal çözümü başlangıç değer problemlerinden daha zordur.

İkinci mertebeden diferensiyel denklemlerle gösterilebilecek pratik önemi olan başlangıç değer problemleri vardır. Kütle-yay sistemi sürtünmeli bir ortamda, bir dış  $F(t)$  kuvvetinin etkisinde olduğunda hareket denklemi

$$m \frac{d^2x}{dt^2} + b \frac{dx}{dt} + kx(t) = F(t)$$

şeklinde yazılabilir. Bu denklemde  $m$  kütle,  $b$  sönüm katsayısı,  $k$  yay sabiti,  $t$  zaman ve  $x(t)$  zamanın fonksiyonu olarak yer değiştirir. Eğer  $v = dx/dt$  dersek ikinci mertebeden diferensiyel denklemi iki tane birinci mertebeden diferensiyel denklemle yer değiştirebiliriz.

$$\frac{dx}{dt} = v$$

$$\frac{dv}{dt} = \frac{1}{m} [F(t) - bv(t) - kx(t)]$$

Bu denklemlerin çözümleri eşzamanlı olarak yukarıda anlatılan yöntemlerle bulunabilir. Fiziksel örnekler aşağıda verilmiştir.

**Örnek Problem 1:** Sönümlü harmonik salıncı,  $-kx$  geri çağırıcı kuvvetinin ve  $-b\dot{x}$  sürtünme kuvvetinin etkisinde hareket etmektedir. Problemi Runge-Kutta yöntemi ile çözünüz. Burada  $k=1$  N/m,  $b=1$  N.s/m olarak

**Çözüm 1:** Burada birinci mertebeden diferensiyel denklemler aşağıdaki gibi yazılabilir.

$$\frac{dx}{dt} = v$$

$$\frac{dv}{dt} = \frac{1}{m}[-bv - kx]$$

biçiminde yazılabilir. Bu denklemin çözümü  $x = e^{st}$  olarak alınır,  $\dot{x} = se^{st}$  ve  $\ddot{x} = s^2 e^{st}$  elde edilir. Burada cebirsel denklem  $s^2 + sa + b = 0$  yazılabilir. Bu denklem çözüldürse  $s$  için iki kök bulunur. Bunlar

$$s = -\frac{a}{2} \pm \sqrt{\frac{a^2 - 4b}{4}}$$

elde edilir. İki kök değeri aynı değilse, bu durumda çözüm  $x = c_1 e^{s_1 t} + c_2 e^{s_2 t}$  olarak elde edilir. Genel çözüm aşağıdaki gibi yazılabilir.

$$x(t) = e^{-bt/2m} \left( A_1 e^{\sqrt{\beta^2 - w_2^2} t} + A_2 e^{-\sqrt{\beta^2 - w_2^2} t} \right)$$

Problemin sayısal çözümü için gerekli program aşağıda verilmiştir.

- FORTRAN program  
program iki\_der

```
implicit real*8 (a-h,o-z)
```

```
parameter (n=10)
```

```
dimension x(0:n),y(0:n),v(0:n)
```

```
write(*,*)"x0, xn, y0, v0"
```

```
read(*,*)x0,xn,y0,v0
```

```
call RK4(x0,xn,y0,v0,n,x,y,v)
```

```
write(*,*)"x(i) ", " y(i)", " v(i)"
```

```
do i=0,n
```

```
open(1,file="iki_der.txt")
```

```
write(1,*)x(i),y(i),v(i)
```

```
write(*,*)x(i),y(i),v(i)
```

```
enddo
```

```
end
```

```
subroutine RK4(x0,xn,y0,v0,n,x,y,v)
```

```
implicit real*8 (a-h,o-z)
```

```
dimension x(0:n),y(0:n),v(0:n)
```

```
h=(xn-x0)/n
```

```
x(0)=x0
```

```
y(0)=y0
```

```
v(0)=v0
```

```
do i=0,n-1
```

```
x(i+1)=x(i)+h
```

```
xk1=h*f1(x(i),y(i),v(i))
```

```
xl1=h*f2(x(i),y(i),v(i))
```



```

xk2=h*f1(x(i)+h/2.d0,y(i)+xk1/2.d0,v(i)+xl1/2.d0)
xl2= h*f2(x(i)+h/2.d0,y(i)+xk1/2.d0,v(i)+xl1/2.d0)
xk3=h*f1(x(i)+h/2.d0,y(i)+xk2/2.d0,v(i)+xl2/2.d0)
xl3=h*f2(x(i)+h/2.d0,y(i)+xk2/2.d0,v(i)+xl2/2.d0)
xk4=h*f1(x(i)+h,y(i)+xk3,v(i)+xl3)
xl4=h*f2(x(i)+h,y(i)+xk3,v(i)+xl3)
y(i+1)=y(i)+(xk1+2.d0*(xk2+xk3)+xk4)/6.d0
v(i+1)=v(i)+(xl1+2.d0*(xl2+xl3)+xl4)/6.d0
enddo
return
end

```

```

function f1(x,y,v)
implicit real*8 (a-h,o-z)
f1=x*0.d0+y*0.d0+v
return
end

```

```

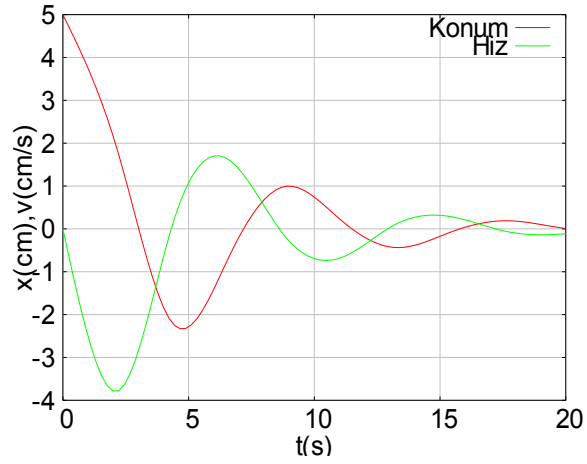
function f2(x,y,v)
implicit real*8 (a-h,o-z)
FF=0.d0
b=1.d0
xk=1.d0
f2=x*0.d0+FF-b*v-xk*y
return
end

```

Program çalıştırıldığında aşağıdaki Çizelge 5.4'deki değerler elde edilmektedir. Bu verilerin grafiği çizilirse Şekil 5.9 elde edilir.

Çizelge 5.9 Örnek problem 1 için Runge-Kutta yönteminden elde edilen sonuçlar

| t(s) | x(cm)  | v(cm/s) |
|------|--------|---------|
| 0.0  | 5      | 0       |
| 2.2  | 1.77   | -3.78   |
| 4.4  | -2.22  | 0.17    |
| 6.6  | -0.65  | 1.61    |
| 8.8  | 0.98   | -0.15   |
| 11.0 | 0.23   | -0.68   |
| 13.2 | -0.43  | 0.09    |
| 15.4 | -0.080 | 0.28    |
| 17.6 | 0.19   | -0.05   |
| 19.8 | 0.02   | -0.12   |
| 22.0 | -0.08  | 0.03    |



Şekil 5.9 İkinci mertebeden diferensiyel denklemin Runge-Kutta ile çözümü, zamana bağlı olarak konum ve hız grafiği, sönüm altı durum.

**Örnek Problem 2:** Bir RLC elektrik devresinin matematik modeli  $Q''(t)+20Q'(t)+125Q(t)=9\sin(5t)$  ile veriliyor. Burada  $Q(0)=0$  ve  $Q'(0)=0$  dır. Diferensiyel denklemi Runge-Kutta yöntemi ile  $[0 :2]$  aralığında çözünüz.  $M=40$  adım kullanınız ve adım aralığını  $h=0.05$  alınız.

**Çözüm 2 :**  $I(t)=Q'(t)$  devreden geçen akımı göstermektedir. Buna göre denklem sistemi aşağıdaki gibi yazılabilir.

$$Q'(t) = I(t)$$
$$I'(t) + 20I(t) + 125Q(t) = 9\sin(5t)$$

Bu denklem sisteminin çözümü önceki probleme benzer şekilde yapılabilir. Değişiklik yapılması gereken fonksiyon altprogramlarıdır.

- FORTRAN altprogramı  
function f1(t,q,xi)

implicit real\*8 (a-h,o-z)

f1=t\*0.d0+q\*0.d0+xi

return

end

function f2(t,q,xi)

implicit real\*8 (a-h,o-z)

FF=9.d0\*sin(5.d0\*t)

a=20.d0

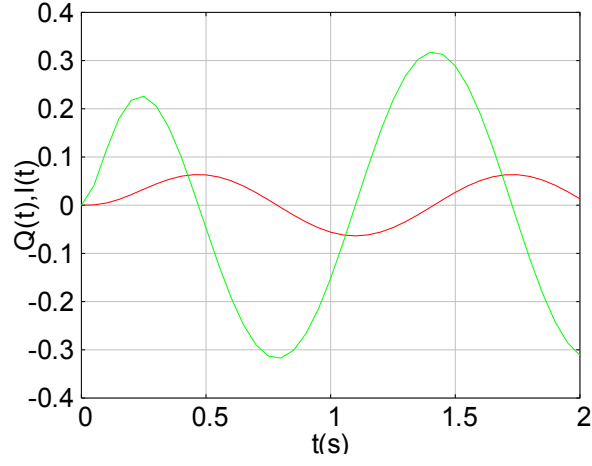
b=125.d0

f2=t\*0.d0+FF-a\*xi-b\*q

return

end

Program çalıştırıldığında elde edilen verilerden Şekil 5.10'de  $Q(t)$  ve  $I(t)$  geçen zamana bağlı olarak grafiği gösterilmiştir.



Şekil 5.10: Dış elektromotor kuvvetinin etkisindeki RLC devresinde yükün ve akımın zamana göre değişimi.

## ÖZET

Diferensiyel denklemler:  $dy/dt \approx \Delta y / \Delta t = f(t, y)$  diferensiyel denklemi verildiğinde  $y'$ 'yi zamanın fonksiyonu olarak bulmak gerekmektedir,  $y_{i+1} = y_i + f(t_i, y_i) \Delta t$ .