ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

DIFERENÇAS FINITAS

Ardson dos S. Vianna Jr.
Departamento de Eng. Quimica
Escola Politecnica - USP
ardson@usp.br

Sumário

- Introdução
- Fórmulas de diferenças finitas
- Aplicação a EDO
- Conclusão

Estudo de caso

Balanço de massa para pellet retangular

Taxa de difusão = taxa de reação

$$\frac{d^2y}{dx^2} - \phi^2 . y = 0 \qquad 1C.C. \; x = 0 \quad \frac{dy}{dx} = 0$$

$$2C.C. \quad x = 1 \quad y = 1$$

Diferenças finitas - Método

- Gerar a malha: pontos onde se resgata a solução
- Substituir os operadores diferenciais por operadores diferença
- Construir o sistema de equações
- Resolver o sistema
- Representar a solução

Fórmulas

Ponto de partida*: Série de Taylor*

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \frac{f'''(x_0)}{3!}(x - x_0)^3 + \ldots$$

$$R_{n+1} = \frac{f^{(n+1)}(x_0)}{(n+1)!}(\xi_i)^{n+1}$$

Fórmulas

Aproximação da 1ª derivada: truncamento da Série de Taylor

$$f(x) = f(x_0) + f'(x_0)(x - x_0)$$

Aproximar a derivada no ponto x_i, atribuir:

$x \leftarrow x_{i+1} \quad x_0 \leftarrow x_i$

Fórmulas

Fórmula *forward* ou adiantada

$$f(x_{i+1}) = f(x_i) + f'(x_i)(x_{i+1} - x_i) + \frac{f''(x_i)}{2!}(x_{i+1} - x_i)^2 + \frac{f'''(x_i)}{3!}(x_{i+1} - x_i)^3 + \ldots$$

$$u_{i+1} = u_i + h.u'_i$$

Fórmulas

Fórmula *backward* ou atrasada ou retrograda

$$f(x_{i-1}) = f(x_i) + f´(x_i)(x_{i-1} - x_i) + \frac{f´´(x_i)}{2!}(x_{i-1} - x_i)^2 + \frac{f´´´(x_i)}{3!}(x_{i-1} - x_i)^3 + \ldots$$

$$u_i = u_{i-1} + h.u´_i$$

Fórmulas

Fórmula centrada, somando

$$u_{i+1} = u_i + h.u'_i$$

$$u_i = u_{i-1} + h.u'_i$$

$$u'_i \cong \frac{u_{i+1} - u_{i-1}}{2.h}$$

ganho?

Fórmulas

$$f(x_{i+1}) = f(x_i) + f'(x_i)(x_{i+1} - x_i) + \frac{f''(x_i)}{2!}(x_{i+1} - x_i)^2 + \frac{f'''(x_i)}{3!}(x_{i+1} - x_i)^3 + \ldots$$

$$f(x_{i-1}) = f(x_i) + f'(x_i)(x_{i-1} - x_i) + \frac{f''(x_i)}{2!}(x_{i-1} - x_i)^2 + \frac{f'''(x_i)}{3!}(x_{i-1} - x_i)^3 + \ldots$$

Somando

$$u''_i \cong \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

Diferenças finitas - Método

- Gerar a malha: pontos onde se resgata a solução
- Substituir os operadores diferenciais por operadores diferença
- Construir o sistema de equações
- Resolver o sistema
- Representar a solução

Método - malha

- Gerar a malha: pontos onde se resgata a solução

$x_i = x_0 + i.h$

xo	u1
x1	u1
x2	u2
....	
xn	un

Método - redução do operador diferencial

$$\frac{d^2 y}{dx^2} - \phi^2 . y = 0$$

$$u``_i \cong \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

$$\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} - \phi^2 . u_i = 0$$

$$x = 0 \quad \frac{u_1 - u_0}{h} = 0$$

$$x = 1 \quad u_n = 1$$

Método – gerando o sistema de equações

- Montando o sistema

$$u_1 = u_0$$

$$u_{i+1} - (2 + h^2)\phi^2 u_i + u_{i-1} = 0$$

$$u_{n+1} = 1$$

- Sistema tridiagonal

Método – resolvendo o sistema

- Resolvendo o sistema – algoritmo de Thomas = TMDA

$$\begin{pmatrix} b_1 & c_1 & 0 & \dots & 0 \\ a_2 & b_2 & c_2 & \dots & 0 \\ 0 & a_3 & b_3 & & 0 \\ \dots & \dots & & & \\ \dots & \dots & \dots & a_n & b_n \end{pmatrix} . \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ \dots \\ u_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \dots \\ \dots \\ d_n \end{pmatrix}$$

Método – resolvendo o sistema

- Resolvendo o sistema – identificando os vetores a,b,c e d

- $b_1=1$, $c_1=-1$, $r_1=0$

- $b_i=-(2+h^2\Phi^2)$, $a_i=1$, $c_i=1$, $2 \leq i \leq n$

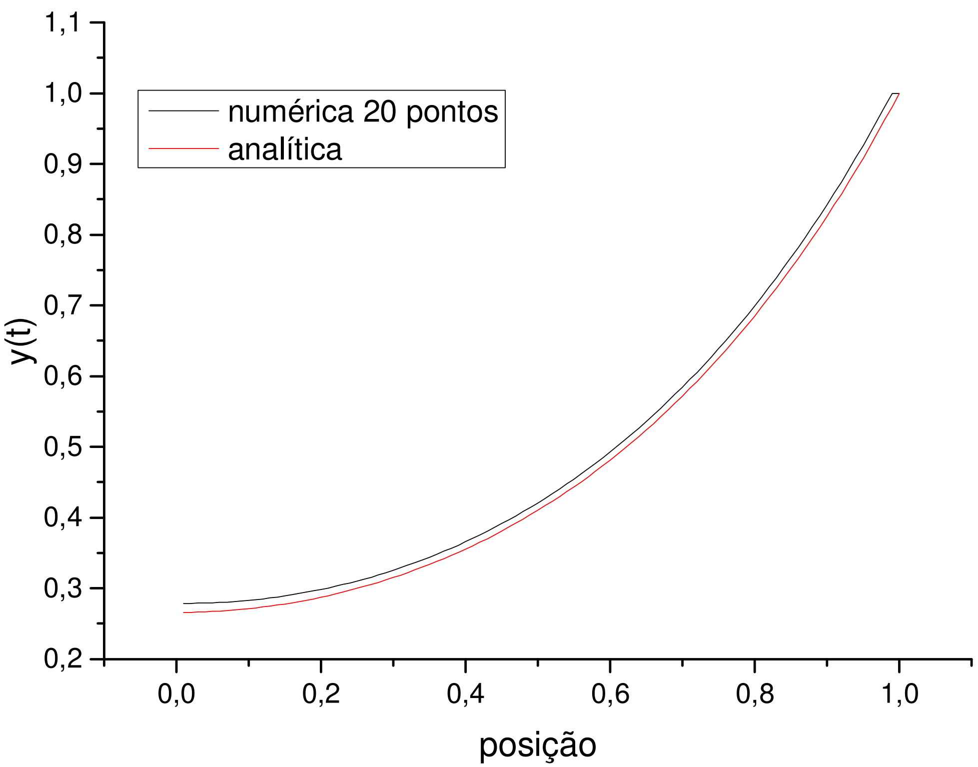
- $B_{n+1}=1$ $r_{n+1}=1$

- Programa difin.for

Programa em fortran

```
Program dif_finitas

      IMPLICIT INTEGER (I-N)
      IMPLICIT REAL(A-H,O-Z)
      DIMENSION A(1:10000), B(1:10000),C(1:10000)
      DIMENSION R(1:10000),U(1:10000)
      DIMENSION x(1:10000)
      OPEN(6,FILE='C:\difin.DAT',   STATUS='old')

     WRITE(*,*)'ENTRE COM O NUMERO DE PONTOS INTERNOS DA MALHA'
                          READ (*,*) ZN
                          H=(1.0-0.0)/(ZN)
                          HH=2.0
                          N=INT(ZN)
c     Construindo o sistema de equacoes segundo as diferecnas finitas

      do 20 i=1,n
                            a(i)=0.0
                            b(i)=0.0
                            c(i)=0.0
                            x(i)=0.0+I*h
 20   continue
      X0=0.0
                          U0=1.0
c     Primeira equação
      B(1)=-1.0
                          C(1)=1.0
                          R(1)=-1.0

c     Pontos internos
                           DO 25 I=2,N-1
         B(I)=-(2.0+H*H*HH*HH)
         C(I)=1.0
                             A(I)=1.0
                             R(I)=0.0
 25    CONTINUE
c     Última equação

                          B(N)=-(2.0+H*H*HH*HH)
      A(N)=2.0
      R(N)=0.0

                          CALL TRIDAG(A,B,C,R,U,N)

 anal=(exp(HH*0.0)/(1.0+exp(2.0*HH)))+
    *                        (exp(-HH*0.0)/(1.0+exp(-2.0*HH)))
                             WRITE (6,*) X0,U0,anal,ABS((U(I)-ANAL)/ANAL)

     do 45 i=1,n-1
                             anal=(exp(HH*x(i))/(1.0+exp(2.0*HH)))+
    *                        (exp(-HH*x(i))/(1.0+exp(-2.0*HH)))
                             WRITE (6,*) x(i),U(i),anal,ABS((U(I)-ANAL)/ANAL)
 45   continue


                            STOP
                            END
SUBROUTINE tridag(a,b,c,r,u,n)
     INTEGER n,NMAX
     REAL a(n),b(n),c(n),r(n),u(n)
     PARAMETER (NMAX=10000)
     INTEGER j
     REAL bet,gam(NMAX)
     if(b(1).eq.0.)pause 'tridag: rewrite equations'
     bet=b(1)
     u(1)=r(1)/bet
     do 11 j=2,n
       gam(j)=c(j-1)/bet
       bet=b(j)-a(j)*gam(j)
       if(bet.eq.0.)pause 'tridag failed'
       u(j)=(r(j)-a(j)*u(j-1))/bet
11   continue
     do 12 j=n-1,1,-1
       u(j)=u(j)-gam(j+1)*u(j+1)
12   continue
     return
     END
```

Método – representação dos resultados



Conclusão

- Implementação simples
- Conjunto finitos de pontos - mesh
- Complicação na resolução do sistema de equações

Bibliografia

- M.E. Davis, Numerical Methods and Modeling for Chemical Engineers, John Wiley & Sons, (1984).
- O.T Hanna, O.C. Sandall,Computational methods in Chemical Engineering, Prentice Hall, (1995).