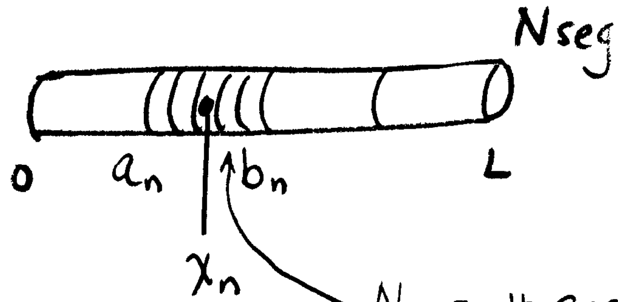
Method of Moments Example

1D Charge Distribution on a rod



N_n = # segments for basis $\int$
N_m = # segments for weight $\int$

$$\underbrace{V(x)}_{f(x)} = \frac{1}{4\pi\varepsilon_0} \int_0^L \rho_L(x') \frac{1}{|x-x'|}\, dx'$$

Thin Rod Approximation

$$\underbrace{\rho_L(x')}_{u(x')} \quad \underbrace{\frac{1}{|x-x'|}}_{g(x,x')}$$

Write unknown as Σ of weighted basis f^s

$$\rho_L(x') = \sum_{n=1}^{Nseg} A_n\, u_n(x')$$

Substitute it in, and bring Σ out:

$$4\pi\varepsilon_0 V(x) = \sum_{n=1}^{Nseg} A_n \underbrace{\int_0^L u_n(x') \frac{1}{|x-x'|}\, dx'}_{g_n(x)}$$

Numerical Integration (trapezoidal)

$$g_n(x) = \frac{u_n(a_n) g(x,a_n)}{2} + \frac{u_n(b_n) g(x,b_n)}{2} + \sum_{i=1}^{N_n-1} u_n(a_n+ih) * g(x, a_n+ih$$

But we don't know "x"!

Use inner product to weight residual

$$\langle W_m(x), 4\pi\varepsilon_0 V \rangle = \langle W_m(x), \sum_{n=1}^{Nseg} A_n g_n(x) \rangle$$

$$\int_0^L W_m(x)\, 4\pi\varepsilon_0 V \, dx = \int_0^L W_m(x) \sum_{n=1}^{Nseg} A_n g_n(x)\, dx$$

$$4\pi\varepsilon_0 V \int_0^L W_m(x)\, dx = \sum_{n=1}^{Nseg} \int_0^L W_m(x)\, g_n(x)\, dx$$

Using trapezoidal integration:

$$\int_0^L W_m(x)\, dx = \frac{W_m(a_m)}{2} + \frac{W_m(b_m)}{2} + \sum_{i=1}^{N_m - 1} W_m(a_m + ih_m$$

WW_m

$$\int_0^L W_m(x)\, g_n(x)\, dx = \frac{W_m(a_m)\, g_n(a_m)}{2} + \frac{W_m(b_m)\, g_n(b_m)}{2} + \sum_{i=1}^{N_m - 1} W_m(a_m + ih_m)\, g_n(a_m + ih_m)$$

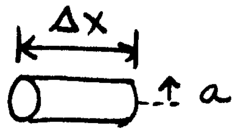
Wg_{nm}

Method of Moments:

Self-Term Evaluation

For $x_n = x_m$ (the "self-term") $\qquad \dfrac{1}{|x_n - x_m|} = \infty$

Let wire be a metal tube:



$$V(\text{tube center}) = \frac{1}{4\pi\varepsilon_0}\int_{\phi=0}^{2\pi}\int_{-\frac{\Delta x}{2}}^{\frac{\Delta x}{2}} \frac{(\rho_s)\,1\,\overbrace{(a\,d\phi\,dx')}^{ds'}}{\sqrt{a^2+(x')^2}}$$

$$= \frac{1}{4\pi\varepsilon_0}\,2\,(2\pi a\,\rho_s)\,\ln\left(\frac{\Delta x}{a}\right)$$

$$= \frac{1}{4\pi\varepsilon_0}\,2\,\rho_L\,\ln\left(\frac{\Delta x}{a}\right)$$

$$4\pi\varepsilon_0 V = \left[2\ln\left(\frac{\Delta x}{a}\right)\right]\underset{A_n}{\underset{\uparrow}{\rho_L}}$$

Matrix

$$\begin{bmatrix} 2\ln\frac{\Delta x}{a} & wg_{12} & wg_{13} & & wg_{1Nseg} \\ & & & & \\ & & & & \\ \underbrace{wg_{Nseg,1} \qquad\qquad}_{wg_{mn}} & & & & \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_{Nseg} \end{bmatrix} = \begin{bmatrix} 4\pi\varepsilon_0 V \\ \vdots \\ \vdots \\ 4\pi\varepsilon_0 V \end{bmatrix}$$

Self-terms

$$2\ln\left(\frac{\Delta x}{a}\right)$$

```
c*******************************
c  Solve for charge distribution on a rod using point matching
c   (rectangular basis function, delta weighting function)
c C.Furse 5-9-95
c file: /grad/furse/ee553.dir/mom2.f
c solve matrix with: /grad/furse/ee553.dir/mom2_gauss.f
c solution will be in: gauss.out
c*******************************
      parameter(Nseg = 20 ) ! # of segments on rod
      real amatrix(Nseg,Nseg),b(Nseg)
      real L ! length of rod (meters)
      common dx

c     --Define problem parameters--
      L = 1.0 ! length of rod (meters)
      dx = L/Nseg ! size of each segment (dx)
      radius = 0.001 ! radius of wire (meters)

c     --Set up matrix--
      do 10 m=1,Nseg
      do 10 n=1,Nseg
         if (m.eq.n) then ! self-term
            amatrix(m,n) = 2.*log(dx/radius)
         else
           amatrix(m,n) = wg(m,n)
         endif
10    continue

c     --Set up b vector--
      pi = acos(-1.)
      eo = 8.854e-12
      V = 1. ! volts
      do 20 m=1,Nseg
         b(m) = 4.*pi*eo*V*ww(m)
20    continue

c     --print out for matrix solver--
      open(60,file='matrix.in')
      write(60,*) Nseg,Nseg
      do i=1,Nseg
      do j=1,Nseg
         write(60,*) amatrix(i,j)
      enddo
      enddo
      do i=1,Nseg
         write(60,*) b(i)
      enddo
      close(60)

      end
c*******************************
c Define basis function: un(x')
c*******************************
      function u(n,xp)
      common dx

      xn = (float(n)-0.5)*dx

c     --Rectangular pulse--
      if ((xp.ge.(xn-dx/2.)).and.(xp.le.(xn+dx/2.))) then
         u = 1.
      else
         u = 0.
      endif

      return
      end
c*******************************
```



```
c Define g(x,x')
c*******************************
      function g(x,xp)

      if (x.ne.xp) then
         g = 1./abs(x-xp)
      else  ! take care of occasional overlaps
         g = 100.
      endif

      return
      end
c*******************************
c Use trapezcidal integration to find gn(x)
c*******************************
      function gn(n,x)
      common dx

      xn = (float(n)-0.5)*dx

      Nn = 10 ! # of integration points
      an = xn - dx/2. ! limits of n integration
      bn = xn + dx/2.
      hn = (bn-an)/Nn ! resolution of integration

      gnx =  u(n,an)*g(x,an)/2. + u(n,bn)*g(x,bn)/2.
      do i=1,Nn-1
         gnx = gnx + u(n,an+i*hn)*g(x,an+i*hn)
      enddo
      gnx = gnx*hn

      gn = gnx
      return
      end
c*******************************
c Use  integration to find int wm(x)gn(x)
c*******************************
      function wg(m,n)
      common dx

      xm = (float(m)-0.5)*dx

c     --Using delta pulse weighting function: int wm*gn = gn(xm)
      wmgn = gn(n,xm)

      wg = wmgn
      return
      end
c*******************************
c Use  integration to find int wm(x)
c*******************************
      function ww(m)
      common dx

      xm = (float(m)-0.5)*dx

c     --Assuming delta weight function--
      ww = 1.

      return
      end
```



```
x vector:
1.02483E-11
8.93698E-12
8.55336E-12
8.33986E-12
8.20308E-12
8.10978E-12
8.04500E-12
8.00103E-12
7.97360E-12
7.96039E-12
7.96039E-12
7.97360E-12
8.00103E-12
8.04501E-12
8.10978E-12
8.20308E-12
8.33986E-12
8.55336E-12
8.93698E-12
1.02483E-11
```