9/22/99

The Deformation of Simply Supported Plates 2.0

The eigenfunctions for simply supported rectangular plates are just the product sin(mπx/a)sin(nπy/b). This simple situation makes it pretty easy to find the deformation in terms of series of eigenfunctions. The expression for the deformation converges pretty rapidly and it is practical to use the eigenfunction technique to compute surface shapes. The point of this note is to set up the general procedure and to run through some examples.

The first thing to do is to select the number of eigenfunctions to use

```
STUDENT > N:=5:M:=5:
```

Allocate storage for the eigenfunctions and the coefficients. (These eigenfunctions don't have zero components, but I'd like to keep this general for systems that do.) I had been skeeping m = 0 and n = 0 terms for generality, but there are no such terms for the double sine expansion, so there is little point in retaining them.

```
STUDENT > f:=array(1..M,1..N):A:=array(1..M,1..N):
```

Define the eigenfunctions

```
STUDENT > for m from 1 to M do for n from 1 to N do
          f[m,n]:=sin(m*Pi*x/a)*sin(n*Pi*y/b)
          od:od:
```

Add up the eigenfunctions multiplied by the unknown coefficients to form the truncated representation of the displacement w

```
STUDENT > w:=0:
          for m from 1 to M do for n from 1 to N do
          w:=w+A[m,n]*f[m,n] od:od:
```

form the Laplacian and the biharmonic of w

```
STUDENT > Lw:=diff(diff(w,x),x)+diff(diff(w,y),y):
STUDENT > LLw:=diff(diff(Lw,x),x)+diff(diff(Lw,y),y):
```

Suppose the surface loading to be representable by a series of the same eigenfunctions as the displacement and find the general solution in terms of the undefined expansion. Individual problems may then be worked by simply finding the appropriate coefficients. Note that the surface loading is actually the surface loading divided by the flexural rigidity.

```
STUDENT > Q:=array(1..M,1..N):
          stress:=0:
          for m from 1 to M do for n from 1 to N do
          stress:=stress+Q[m,n]*f[m,n] od:od:
          stress;
```

Once this has been done, the coefficients of w are easily related to the coefficients Q[m,n] by working out the terms in LLw. The terms a/2 and b/2 will show up on both sides of the equation, so I do not need to carry them here.

Calculate the divisors

```
STUDENT > c:=array(1..M,1..N):
          for m from 1 to M do for n from 1 to N do
          factor(coeff(LLw,A[m,n])):
          c[m,n]:=subs(sin(m*Pi*x/a)=1,sin(n*Pi*y/b)=1,"):
          od:od:
```

Now I can invert the two series representations by equating the coefficients of each term. In the generic case they are all nonzero, and I am printing none of them.

```
STUDENT > for m from 1 to M do for n from  1 to N do
          A[m,n]:=Q[m,n]/c[m,n] od:od:
```

We now have a generic representation for the displacement w.

If we are to work complete problems, we need expressions for the moments, shear forces and stresses in terms of w.
The moment and force notation is obvious. The stress notation is sigma1 = sigmax, sigma2 = sigmay, sigma3 = sigmaz, tau = tauxy, tau1 = tauxz and tau2 = tauyz.

```
STUDENT > Mx:=-EI*(diff(diff(w,x),x)+nu*diff(diff(w,y),y)):
          My:=-EI*(nu*diff(diff(w,x),x)+diff(diff(w,y),y)):
          Qx:=-EI*diff(Lw,x):
          Qy:=-EI*diff(Lw,y):
          sigma1:=-Y/(1-nu^2)*(diff(diff(w,x),x)+nu*diff(diff(w,y),y
          ))*z:
          sigma2:=-Y/(1-nu^2)*(nu*diff(diff(w,x),x)+diff(diff(w,y),y
          ))*z:
          tau:=-Y/(1+nu)*diff(diff(w,x),y)*z:
          tau1:=-Y/2/(1-nu^2)*diff(Lw,x)*((h/2)^2-z^2):
          tau2:=-Y/2/(1-nu^2)*diff(Lw,y)*((h/2)^2-z^2):
          sigma3:=Y/6/(1-nu^2)*LLw*(2*(h/2)^3+3*(h/2)^2*z-z^3):
```

```
STUDENT > save `simply supported 5x5.m`:
STUDENT >
```