

The background of the slide is a photograph of the Iowa State University campus, featuring the Old Capitol building with its large dome on the left and various other university buildings and trees in the background. The entire image is covered with a semi-transparent red overlay.

Final C++ and FEM Project

12/19/2022

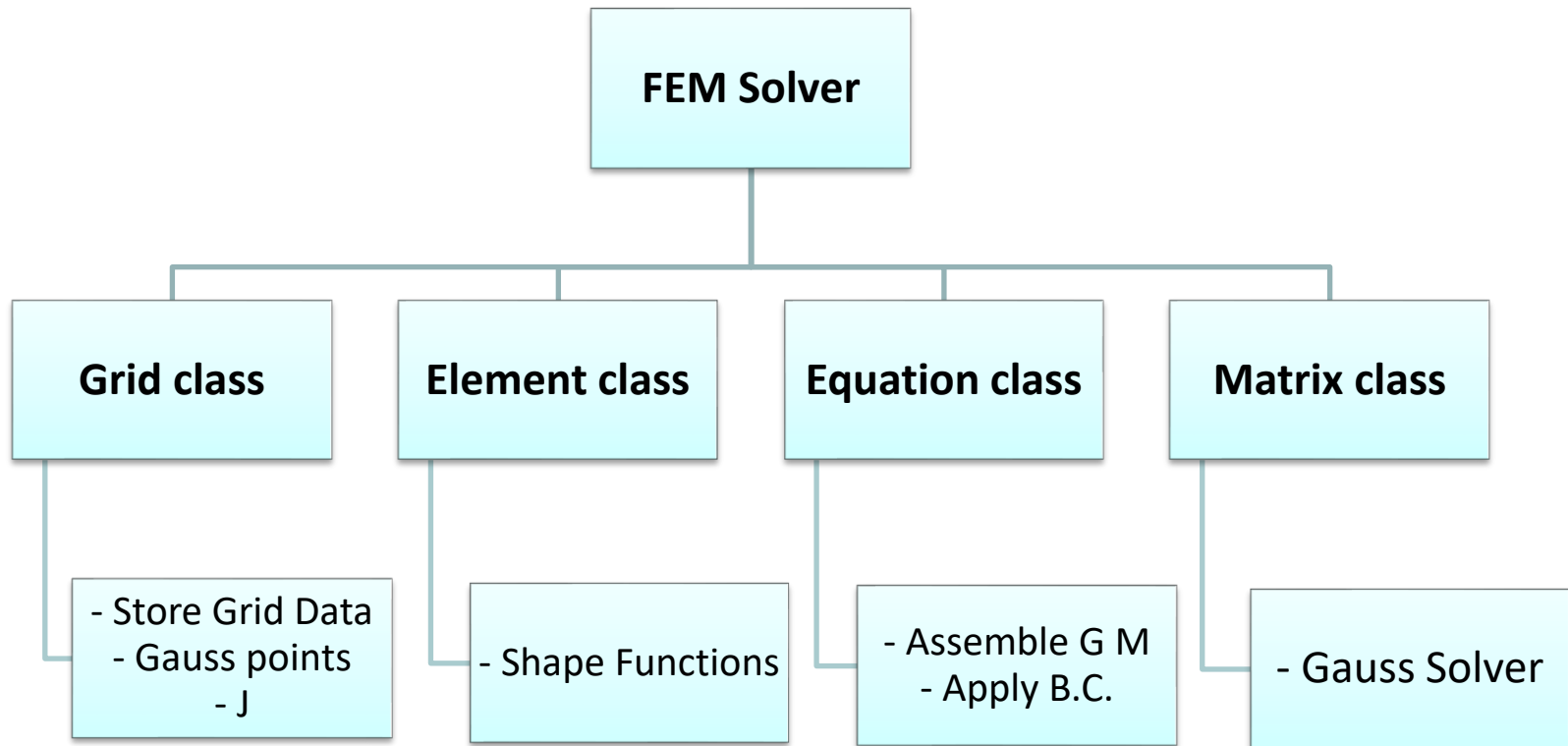
Heat Transfer Equation (1D):

$$\begin{aligned}\nabla^2 u &= -\pi^2 \sin(\pi x) \\ u &= 0; \quad \text{on } \partial\Omega\end{aligned}$$

Galerkin Form:

$$\int [N_{i,x} N_{j,x}] u_i dx = \int N_i f dx$$

Implementation of SSHT



Implementation of SSHT

Calculating Local Stiffness Matrices:

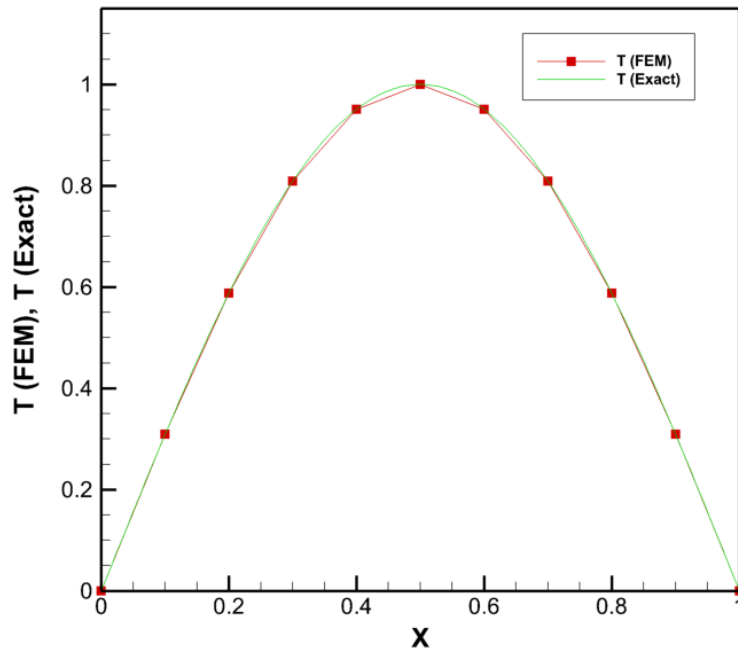
```
for (int elem_ID = 0; elem_ID < p_grid.elem_num; elem_ID++) {  
    for (int k = 0; k < p_grid.gauss_pts; k++) {  
  
        double detJxW = p_grid.detJxW(elem_ID);  
        double force = -1 * M_PI * M_PI * sin(M_PI * p_grid.x_gs_pts[elem_ID][k]);  
  
        for (int i = 0; i <= 1; i++) {  
            for (int j = 0; j <= 1; j++) {  
  
                e.Ae[i][j] += (-1) * fe.dN(i, elem_ID) * fe.dN(j, elem_ID) * detJxW;  
            }  
            e.be[i] += fe.N(i, k) * force * detJxW;  
        }  
    }  
}
```

Implementation of SSHT

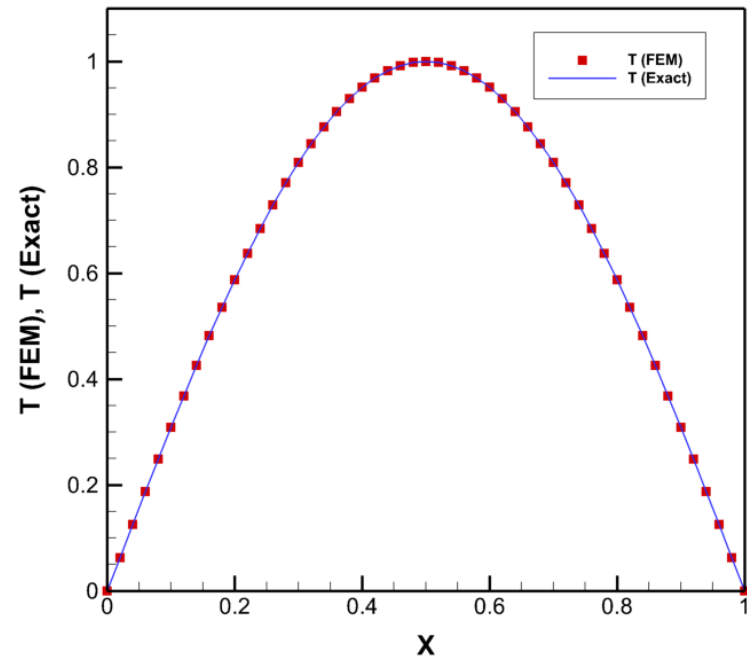
Assembling Global Stiffness Matrix:

```
void equation::assemble(Matrix &A, Matrix &b, grid &g, element &e ) {  
  
    for (int i = 0; i < e.nbf; i++){  
        for (int j = 0; j < e.nbf; j++){  
            A.mat[g.conn[elem_ID][i]][g.conn[elem_ID][j]] += Ae[i][j];  
            Ae[i][j] = 0;  
        }  
        b.vec[g.conn[elem_ID][i]] += be[i];  
        be[i] = 0;  
    }  
}
```

Results

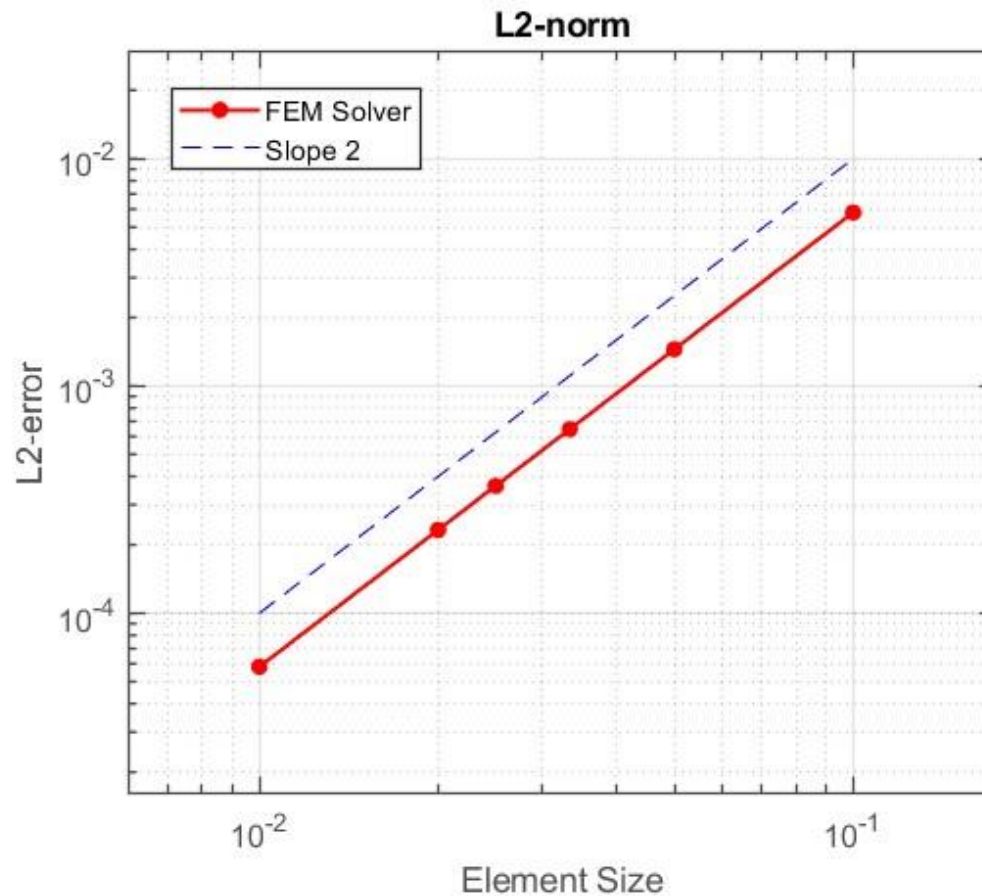


No. of Elements = 10



No. of Elements = 50

Convergence study of SSHT



Element Size = [0.1, 0.05, 0.033, 0.025, 0.02, 0.01]