

Finite Element Analysis of 2 D Heat Equation

Written by Dr Kewei Li, TU Graz

For Computation Biomechanics Course

Last edit: April 29, 2019

```
In[1]:= ClearAll["Global`*"];

(* Basic FE model data *)
nnp = 9;
numel = 4;
ndm = 2;
ndf = 1;
nel = 4;

(* Shape functions for Q4 element *)
N1 = (1 -  $\xi$ ) (1 -  $\eta$ ) / 4;
N2 = (1 +  $\xi$ ) (1 -  $\eta$ ) / 4;
N3 = (1 +  $\xi$ ) (1 +  $\eta$ ) / 4;
N4 = (1 -  $\xi$ ) (1 +  $\eta$ ) / 4;

(* Material parameters *)
kappa = 2.0;

(* Geometrical data *)
e1xy = {{0, 0}, {1, 0}, {2, 0}, {0, 1}, {1, 1}, {2, 1}, {0, 2}, {1, 2}, {2, 2}};
ld = {{1, 2, 5, 4}, {2, 3, 6, 5}, {4, 5, 8, 7}, {5, 6, 9, 8}};
elh = 1.0;
elh2 = elh / 2.0;

(* Boundary conditions *)
ubce = {1, 3};
ubcn = {1, 4, 7};
fbce = {2, 4};
fbcn = {3, 6, 9};
qhat = 5;
uhat = 20;
uhatn = {uhat, uhat, uhat};

(* Initialization of element and global stiffness matrices *)
ke = ConstantArray[0.0, {nel * ndf, nel * ndf, numel}];
Ke = ConstantArray[0.0, {nnp * ndf, nnp * ndf, numel}];
Kmat = ConstantArray[0.0, {nnp * ndf, nnp * ndf}];
Kmr = ConstantArray[0.0, {nnp * ndf - Length@ubcn * ndf, nnp * ndf - Length@ubcn * ndf}];

(* Initialization of matrices for load vector *)
fed = ConstantArray[0.0, {nel * ndf, numel}];
feb = ConstantArray[0.0, {nel * ndf, numel}];
Fe = ConstantArray[0.0, {nnp * ndf, numel}];
Fmat = ConstantArray[0.0, {nnp * ndf}];
Fmr = ConstantArray[0.0, {nnp * ndf - Length@ubcn * ndf}];

(* Initialization of solution matrices *)
Usol = ConstantArray[0.0, {nnp * ndf, ndm + 1}];
```

```

noubcn = Complement[Range[Length[Usol]], ubcn] ;

Print["*****"] ;
Print["          Program Output          "] ;
Print["*****"] ;

(* Loop through all the elements to compute the element stiffness matrix *)
Do[
  e = ii;

  x1 = e1xy[[ld[[e, 1]], 1]]; y1 = e1xy[[ld[[e, 1]], 2]];
  x2 = e1xy[[ld[[e, 2]], 1]]; y2 = e1xy[[ld[[e, 2]], 2]];
  x3 = e1xy[[ld[[e, 3]], 1]]; y3 = e1xy[[ld[[e, 3]], 2]];
  x4 = e1xy[[ld[[e, 4]], 1]]; y4 = e1xy[[ld[[e, 4]], 2]];

  Print["-----"] ;
  Print["Processing element ", e, ":"] ;
  Print["-----"] ;

  Print@ListLinePlot[{{x1, y1}, {x2, y2}, {x3, y3}, {x4, y4}, {x1, y1}},
    Filling -> Bottom, AspectRatio -> 1, PlotMarkers -> Automatic,
    PlotRange -> {{0, 2.1}, {0, 2.1}}, ImageSize -> Small] ;

  x = Simplify[N1 * x1 + N2 * x2 + N3 * x3 + N4 * x4] ;
  y = Simplify[N1 * y1 + N2 * y2 + N3 * y3 + N4 * y4] ;

  Jmat = {{D[x,  $\xi$ ], D[x,  $\eta$ ]}, {D[y,  $\xi$ ], D[y,  $\eta$ ]}} ;

  Jac = Det[Jmat] ;
  Jinv = Inverse[Jmat] ;

  d $\xi$ dx = Jinv[[1, 1]];
  d $\xi$ dy = Jinv[[1, 2]];
  d $\eta$ dx = Jinv[[2, 1]];
  d $\eta$ dy = Jinv[[2, 2]];

  dN1dx = D[N1,  $\xi$ ] d $\xi$ dx + D[N1,  $\eta$ ] d $\eta$ dx;
  dN1dy = D[N1,  $\xi$ ] d $\xi$ dy + D[N1,  $\eta$ ] d $\eta$ dy;

  dN2dx = D[N2,  $\xi$ ] d $\xi$ dx + D[N2,  $\eta$ ] d $\eta$ dx;
  dN2dy = D[N2,  $\xi$ ] d $\xi$ dy + D[N2,  $\eta$ ] d $\eta$ dy;

  dN3dx = D[N3,  $\xi$ ] d $\xi$ dx + D[N3,  $\eta$ ] d $\eta$ dx;
  dN3dy = D[N3,  $\xi$ ] d $\xi$ dy + D[N3,  $\eta$ ] d $\eta$ dy;

  dN4dx = D[N4,  $\xi$ ] d $\xi$ dx + D[N4,  $\eta$ ] d $\eta$ dx;
  dN4dy = D[N4,  $\xi$ ] d $\xi$ dy + D[N4,  $\eta$ ] d $\eta$ dy;

  Bmat = {{dN1dx, dN2dx, dN3dx, dN4dx}, {dN1dy, dN2dy, dN3dy, dN4dy}} ;

  Print["Creating the element B matrix... \n", (Bmat) // MatrixForm] ;

  intg = (Transpose[Bmat].Bmat) * Jac ;

```

```

gp1 =  $\frac{-1}{\sqrt{3}}$ ;
gp2 =  $\frac{1}{\sqrt{3}}$ ;
w1 = 1.0;
w2 = 1.0;
w3 = 1.0;
w4 = 1.0;

(* Computing the element stiffness matrix ke(4x4) *)

ke[[All, All, e]] =
  kappa * (intg * w1 /. {ξ → gp1, η → gp1}) + kappa * (intg * w2 /. {ξ → gp2, η → gp1}) +
  kappa * (intg * w3 /. {ξ → gp2, η → gp2}) + kappa * (intg * w4 /. {ξ → gp1, η → gp2});

Print["Creating the element stiffness matrix ke(4x4)... \n",
  (ke[[All, All, e]] ) // MatrixForm] ;

(* Computing matrices for load vector *)
intfed = (2.0 * x + y^2) * {N1, N2, N3, N4} * Jac ;

fed[[All, e]] = (intfed * w1 /. {ξ → gp1, η → gp1}) + (intfed * w2 /. {ξ → gp2, η → gp1}) +
  (intfed * w3 /. {ξ → gp2, η → gp2}) + (intfed * w4 /. {ξ → gp1, η → gp2});

intfeb = {N2, N3};
feb[[2 ;; 3, e]] =
  If[fbc[[1]] == e || fbc[[2]] == e, qhat * elh2 * (intfeb * w1 /. {ξ → 1.0, η → gp1}) +
    qhat * elh2 * (intfeb * w2 /. {ξ → 1.0, η → gp2}), 0.0];

Do[
  Do[
    Ke[[ld[[e, i]], ld[[e, j]], e]] = Ke[[ld[[e, i]], ld[[e, j]], e]] + ke[[i, j, e]],
    {j, Length@ke}];
  Fe[[ld[[e, i]], e]] = Fe[[ld[[e, i]], e]] + fed[[i, e]] - feb[[i, e]],
  {i, Length@ke}];

Print["Convert to the element stiffness matrix Ke(9x9) \n",
  (Ke[[All, All, e]] ) // MatrixForm] ;
Print["Element load vector Fe(9x1) \n", (Fe[[All, e]] ) // MatrixForm] ,

{ii, 1, numel}];

Print["-----"] ;
Print["Assemble the Global stiffness matrix K(9x9): "] ;
Print["-----"] ;

Do[
  Kmat[[All, All]] = Kmat[[All, All]] + Ke[[All, All, i]],
  {i, 1, numel}];

Print[(Kmat) // MatrixForm]

Print["-----"] ;

```

```

Print["Assemble the load vector F(9×1): "];
Print["-----"];

Do[
  Fmat[[All]] = Fmat[[All]] + Fe[[All, i]],
  {i, 1, numel}];

Print[(Fmat) // MatrixForm]

(*adding the inhomogenous Dirichlet BC : *)
Do[
  Fmat[[noubcn[[i]]]] = Fmat[[noubcn[[i]]]] - {Kmat[[noubcn[[i]], ubcn[[1]]]],
    Kmat[[noubcn[[i]], ubcn[[2]]]], Kmat[[noubcn[[i]], ubcn[[3]]]]}.uhatn,
  {i, 1, Length@noubcn}];

Print["-----"];
Print["Reduction of stiffness matrix: "];
Print["-----"];

removev[a_?MatrixQ, pos_List] := Module[{tmp, length = Length[a]},
  tmp = Complement[Range[length], pos];
  a[[tmp, tmp]]];

Kmr = removev[Kmat, ubcn];
Print[(Kmr) // MatrixForm]

Print["-----"];
Print["Reduction of the load vector: "];
Print["-----"];

removev[a_?VectorQ, pos_List] := Module[{tmp, length = Length[a]},
  tmp = Complement[Range[length], pos];
  a[[tmp]]];

Fmr = removev[Fmat, ubcn];

Print[(Fmr) // MatrixForm]

Print["-----"];
Print["Solving the linear system of equations to find u: "];
Print["-----"];

Umat = LinearSolve[Kmr, Fmr];

Do[
  Usol[[i, 1 ;; 2]] = Usol[[i, 1 ;; 2]] + e1xy[[i, All]],
  {i, 1, nnp * ndf}];

(* add the computed u to the full-length solution array Usol *)
noubcn = Complement[Range[Length[Usol]], ubcn];

Do[
  Usol[[noubcn[[i]], 3]] = Usol[[noubcn[[i]], 3]] + Umat[[i]],
  {i, 1, Length@noubcn}];

(* add the BC to the solution array *)
Do[

```

```

Usol[[ubcn[[i]], 3]] = Usol[[ubcn[[i]], 3]] + uhatn[[i]],
{i, 1, Length@ubcn}];

Print[(Usol) // MatrixForm]

fig2 = ListDensityPlot[Usol, InterpolationOrder → 1, ImageSize → Medium,
  ColorFunction → "Rainbow", Mesh → 11, MeshFunctions -> {#3 &},
  BaseStyle → {FontWeight → "Normal", FontSize → 18}, PlotLegends →
    BarLegend[Automatic, LabelStyle → {Black, FontFamily → "STIX", FontSize → 18}],
  LabelStyle → Directive[Black, FontSize → 18, FontFamily → "STIX"],
  PlotRangePadding → Scaled[.0]]];

Show[fig2, PlotRange → {{0, 2}, {0, 2}}, PlotRangePadding → Scaled[.00]]

Print["----- END OF THE PROGRAM OUTPUT -----\\n"] ;

*****

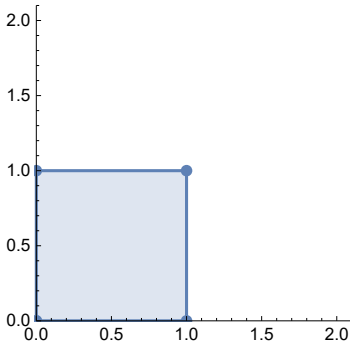
                                Program Output

*****

-----

Processing element 1:

-----



Creating the element B matrix...


$$\begin{pmatrix} \frac{1}{2}(-1+\eta) & \frac{1-\eta}{2} & \frac{1+\eta}{2} & \frac{1}{2}(-1-\eta) \\ \frac{1}{2}(-1+\xi) & \frac{1}{2}(-1-\xi) & \frac{1+\xi}{2} & \frac{1-\xi}{2} \end{pmatrix}$$


Creating the element stiffness matrix ke(4x4)...


$$\begin{pmatrix} 1.33333 & -0.333333 & -0.666667 & -0.333333 \\ -0.333333 & 1.33333 & -0.333333 & -0.666667 \\ -0.666667 & -0.333333 & 1.33333 & -0.333333 \\ -0.333333 & -0.666667 & -0.333333 & 1.33333 \end{pmatrix}$$


```

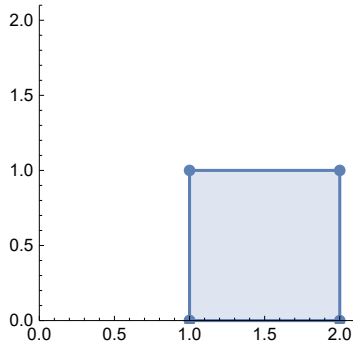
Convert to the element stiffness matrix $K_e(9 \times 9)$

$$\begin{pmatrix} 1.33333 & -0.333333 & 0. & -0.333333 & -0.666667 & 0. & 0. & 0. & 0. \\ -0.333333 & 1.33333 & 0. & -0.666667 & -0.333333 & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -0.333333 & -0.666667 & 0. & 1.33333 & -0.333333 & 0. & 0. & 0. & 0. \\ -0.666667 & -0.333333 & 0. & -0.333333 & 1.33333 & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

Element load vector $F_e(9 \times 1)$

$$\begin{pmatrix} 0.208333 \\ 0.375 \\ 0. \\ 0.291667 \\ 0.458333 \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix}$$

Processing element 2:



Creating the element B matrix...

$$\begin{pmatrix} \frac{1}{2}(-1+\eta) & \frac{1-\eta}{2} & \frac{1+\eta}{2} & \frac{1}{2}(-1-\eta) \\ \frac{1}{2}(-1+\xi) & \frac{1}{2}(-1-\xi) & \frac{1+\xi}{2} & \frac{1-\xi}{2} \end{pmatrix}$$

Creating the element stiffness matrix $k_e(4 \times 4)$...

$$\begin{pmatrix} 1.33333 & -0.333333 & -0.666667 & -0.333333 \\ -0.333333 & 1.33333 & -0.333333 & -0.666667 \\ -0.666667 & -0.333333 & 1.33333 & -0.333333 \\ -0.333333 & -0.666667 & -0.333333 & 1.33333 \end{pmatrix}$$

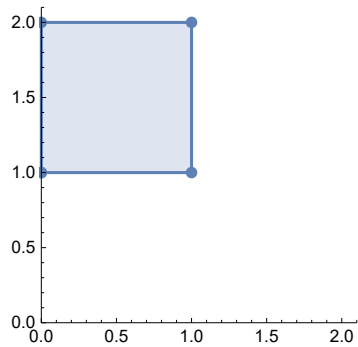
Convert to the element stiffness matrix $K_e(9 \times 9)$

$$\begin{pmatrix} 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.33333 & -0.333333 & 0. & -0.333333 & -0.666667 & 0. & 0. & 0. \\ 0. & -0.333333 & 1.33333 & 0. & -0.666667 & -0.333333 & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & -0.333333 & -0.666667 & 0. & 1.33333 & -0.333333 & 0. & 0. & 0. \\ 0. & -0.666667 & -0.333333 & 0. & -0.333333 & 1.33333 & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

Element load vector $F_e(9 \times 1)$

$$\begin{pmatrix} 0. \\ 0.708333 \\ -1.625 \\ 0. \\ 0.791667 \\ -1.54167 \\ 0. \\ 0. \\ 0. \end{pmatrix}$$

Processing element 3:



Creating the element B matrix...

$$\begin{pmatrix} \frac{1}{2}(-1+\eta) & \frac{1-\eta}{2} & \frac{1+\eta}{2} & \frac{1}{2}(-1-\eta) \\ \frac{1}{2}(-1+\xi) & \frac{1}{2}(-1-\xi) & \frac{1+\xi}{2} & \frac{1-\xi}{2} \end{pmatrix}$$

Creating the element stiffness matrix $ke(4 \times 4)$...

$$\begin{pmatrix} 1.33333 & -0.333333 & -0.666667 & -0.333333 \\ -0.333333 & 1.33333 & -0.333333 & -0.666667 \\ -0.666667 & -0.333333 & 1.33333 & -0.333333 \\ -0.333333 & -0.666667 & -0.333333 & 1.33333 \end{pmatrix}$$

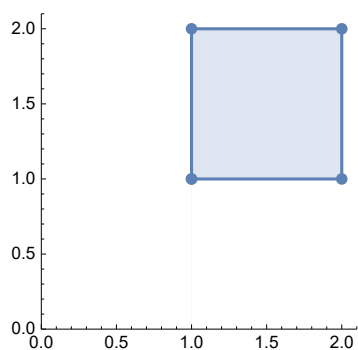
Convert to the element stiffness matrix $Ke(9 \times 9)$

$$\begin{pmatrix} 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 1.33333 & -0.333333 & 0. & -0.333333 & -0.666667 & 0. \\ 0. & 0. & 0. & -0.333333 & 1.33333 & 0. & -0.666667 & -0.333333 & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & -0.333333 & -0.666667 & 0. & 1.33333 & -0.333333 & 0. \\ 0. & 0. & 0. & -0.666667 & -0.333333 & 0. & -0.333333 & 1.33333 & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

Element load vector $Fe(9 \times 1)$

$$\begin{pmatrix} 0. \\ 0. \\ 0. \\ 0.625 \\ 0.791667 \\ 0. \\ 0.875 \\ 1.04167 \\ 0. \end{pmatrix}$$

Processing element 4:



Creating the element B matrix...

$$\begin{pmatrix} \frac{1}{2}(-1+\eta) & \frac{1-\eta}{2} & \frac{1+\eta}{2} & \frac{1}{2}(-1-\eta) \\ \frac{1}{2}(-1+\xi) & \frac{1}{2}(-1-\xi) & \frac{1+\xi}{2} & \frac{1-\xi}{2} \end{pmatrix}$$

Creating the element stiffness matrix $k_e(4 \times 4)$...

$$\begin{pmatrix} 1.33333 & -0.333333 & -0.666667 & -0.333333 \\ -0.333333 & 1.33333 & -0.333333 & -0.666667 \\ -0.666667 & -0.333333 & 1.33333 & -0.333333 \\ -0.333333 & -0.666667 & -0.333333 & 1.33333 \end{pmatrix}$$

Convert to the element stiffness matrix $K_e(9 \times 9)$

$$\begin{pmatrix} 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 1.33333 & -0.333333 & 0. & -0.333333 & -0.666667 \\ 0. & 0. & 0. & 0. & -0.333333 & 1.33333 & 0. & -0.666667 & -0.333333 \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & -0.333333 & -0.666667 & 0. & 1.33333 & -0.333333 \\ 0. & 0. & 0. & 0. & -0.666667 & -0.333333 & 0. & -0.333333 & 1.33333 \end{pmatrix}$$

Element load vector $F_e(9 \times 1)$

$$\begin{pmatrix} 0. \\ 0. \\ 0. \\ 0. \\ 1.125 \\ -1.20833 \\ 0. \\ 1.375 \\ -0.958333 \end{pmatrix}$$

Assemble the Global stiffness matrix $K(9 \times 9)$:

$$\begin{pmatrix} 1.33333 & -0.333333 & 0. & -0.333333 & -0.666667 & 0. & 0. & 0. & 0. \\ -0.333333 & 2.66667 & -0.333333 & -0.666667 & -0.666667 & -0.666667 & 0. & 0. & 0. \\ 0. & -0.333333 & 1.33333 & 0. & -0.666667 & -0.333333 & 0. & 0. & 0. \\ -0.333333 & -0.666667 & 0. & 2.66667 & -0.666667 & 0. & -0.333333 & -0.666667 & 0. \\ -0.666667 & -0.666667 & -0.666667 & -0.666667 & 5.33333 & -0.666667 & -0.666667 & -0.666667 & -0.666667 \\ 0. & -0.666667 & -0.333333 & 0. & -0.666667 & 2.66667 & 0. & -0.666667 & -0.333333 \\ 0. & 0. & 0. & -0.333333 & -0.666667 & 0. & 1.33333 & -0.333333 & 0. \\ 0. & 0. & 0. & -0.666667 & -0.666667 & -0.666667 & -0.333333 & 2.66667 & -0.333333 \\ 0. & 0. & 0. & 0. & -0.666667 & -0.333333 & 0. & -0.333333 & 1.33333 \end{pmatrix}$$

Assemble the load vector $F(9 \times 1)$:

$$\begin{pmatrix} 0.208333 \\ 1.08333 \\ -1.625 \\ 0.916667 \\ 3.16667 \\ -2.75 \\ 0.875 \\ 2.41667 \\ -0.958333 \end{pmatrix}$$

Reduction of stiffness matrix:

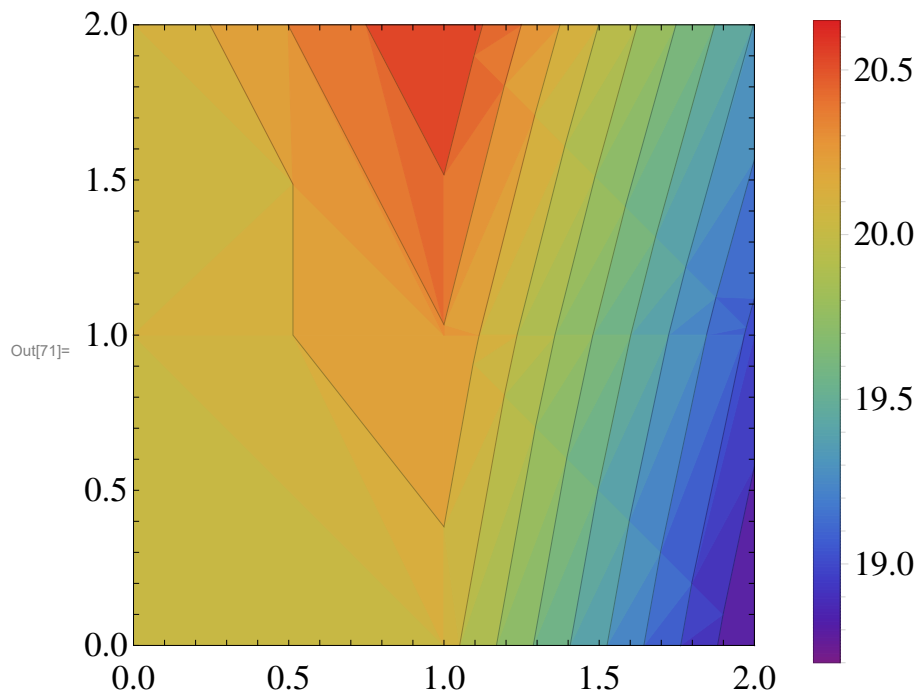
$$\begin{pmatrix} 2.66667 & -0.333333 & -0.666667 & -0.666667 & 0. & 0. \\ -0.333333 & 1.33333 & -0.666667 & -0.333333 & 0. & 0. \\ -0.666667 & -0.666667 & 5.33333 & -0.666667 & -0.666667 & -0.666667 \\ -0.666667 & -0.333333 & -0.666667 & 2.66667 & -0.666667 & -0.333333 \\ 0. & 0. & -0.666667 & -0.666667 & 2.66667 & -0.333333 \\ 0. & 0. & -0.666667 & -0.333333 & -0.333333 & 1.33333 \end{pmatrix}$$

Reduction of the load vector:

$$\begin{pmatrix} 21.0833 \\ -1.625 \\ 43.1667 \\ -2.75 \\ 22.4167 \\ -0.958333 \end{pmatrix}$$

Solving the linear system of equations to find u:

$$\begin{pmatrix} 0. & 0. & 20. \\ 1. & 0. & 20.066 \\ 2. & 0. & 18.6978 \\ 0. & 1. & 20. \\ 1. & 1. & 20.3104 \\ 2. & 1. & 18.9796 \\ 0. & 2. & 20. \\ 1. & 2. & 20.6466 \\ 2. & 2. & 19.343 \end{pmatrix}$$



END OF THE PROGRAM OUTPUT