1 Mathemtica Scipts

1.1 Mass Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*Mass matrix entry for RZ BLD Gauss-Legendre on quadrilaterals*)
\
Clear[r0, r1, r2, r3, z0, z1, z2, z3, mr0, mr1, mr2, mr3, mz0, mz1, \
mz2, mz3]
(*Define physical integration point coordinates to compare to MFEM \
output:r0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
r1=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);
z0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
z3=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);*)
(*These make the mesh \
rectangular:*)
z2 := z3
z1 := z0
r2 := r1
r3 := r0
(*Quadrilateral transformation back to reference square-more general \
than rectangle transofrmation-but NOT curved mesh*)

r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0
z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0

(*Mesh nodes are needed to conserve cell volume.Cannot define mr0=0 \
nor mz0=0 here.Perform integration and then set the values.*)
\
(*Comment out to get the general mesh element mr1=0.25;
mz3=0.25;*)
mr2 := mr1
```

```
mr3 := mr0

mz1 := mz0

mz2 := mz3

mr[rho_, kappa_] := (mr1 - mr0) rho + (mr3 - mr0) kappa + (mr2 + mr0 -
      mr1 - mr3) rho kappa + mr0

mz[rho_, kappa_] := (mz1 - mz0) rho + (mz3 - mz0) kappa + (mz2 + mz0 -
      mz1 - mz3) rho kappa + mz0

a = {mr[rho, kappa], mz[rho, kappa]};

b = {rho, kappa};

(*Determinant of the Jacobian-for determining area of the physical \
element.*)

j = Det[D[a, {b}]];

(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)


BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)

BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 0.5

BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)

BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 0.5

(*Check for orthogonality.*)

BLD1[r0, z0]*BLD2[r0, z0];

(*The r variable gets transformed back to the reference element*)

M =
 Integrate[
  mr[rho, kappa]*BLD1[rho, kappa]*BLD1[rho, kappa]*j, {rho, 0,
   1}, {kappa, 0, 1}]
```

```
mr0 = 0.25;

mz0 = 0.25;

mr1 = 0.5;

mz3 = 0.5;


M
```

1.2 Angular Redistribution Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*Angular Redistribution matrix for RZ BLD Gauss-Legendre on \
quadrilaterals*)Clear[r0, r1, r2, r3, z0, z1, z2, z3, mr0, mr1, mr2, \
mr3, mz0, mz1, mz2, mz3]
(*Define physical integration point coordinates to compare to MFEM \
output:r0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
r1=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);
z0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
z3=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);*)
(*These make the mesh \
rectangular:*)
z2 := z3
z1 := z0
r2 := r1
r3 := r0
(*Quadrilateral transformation back to reference square-more general \
than rectangle transofrmation-but NOT curved mesh*)

r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0
z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0
```

```
(*Mesh nodes are needed to conserve cell volume.Cannot define mr0=0 \
nor mz0=0 here.Perform integration and then set the values.*)
\
(*Comment out to get the general mesh element mr1=0.25;
mz3=0.25;*)
mr2 := mr1
mr3 := mr0
mz1 := mz0
mz2 := mz3
mr[rho_, kappa_] := (mr1 - mr0) rho + (mr3 - mr0) kappa + (mr2 + mr0 -
      mr1 - mr3) rho kappa + mr0
mz[rho_, kappa_] := (mz1 - mz0) rho + (mz3 - mz0) kappa + (mz2 + mz0 -
      mz1 - mz3) rho kappa + mz0
a = {mr[rho, kappa], mz[rho, kappa]};
b = {rho, kappa};
(*Determinant of the Jacobian-for determining area of the physical \
element.*)
j = Det[D[a, {b}]];
(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)

BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
```

```
M = Integrate[
  BLD1[rho, kappa]*BLD1[rho, kappa]*j, {rho, 0, 1}, {kappa, 0, 1}]

mr0 = 0;
mz0 = 0;
mr1 = 0.25;
mz3 = 0.25;

thisalpha = 0.350021175;
w = 0.523598775598298;
M*thisalpha
```

1.3 r-Leakage Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*r-Leakage matrix for RZ BLD Gauss-Legendre on \
quadrilaterals*)Clear[r0, r1, r2, r3, z0, z1, z2, z3, mr0, mr1, mr2, \
mr3, mz0, mz1, mz2, mz3, mu, rho, kappa]
(*Define physical integration point coordinates to compare to MFEM \
output: *)
r0 = (1 + (-Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
r1 = (1 + (Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
z0 = (1 + (-Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
z3 = (1 + (Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
(*Use to define rectangular mesh:*)
r2 = r1;
r3 = r0;
z1 = z0;
z2 = z3;
(*Transform arbitrary quadrilateral to reference square*)

r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
```

```
     r3) rho kappa + r0
z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0
(*Mesh nodes are needed to conserve cell volume.Cannot define mr0=0 \
nor mz0=0 here.Perform integration and then set the values.*)
\
(*Comment out to get the general mesh element mr1=0.25;
mz3=0.25;*)
mr2 := mr1
mr3 := mr0
mz1 := mz0
mz2 := mz3
mr[rho_, kappa_] := (mr1 - mr0) rho + (mr3 - mr0) kappa + (mr2 + mr0 -
      mr1 - mr3) rho kappa + mr0
mz[rho_, kappa_] := (mz1 - mz0) rho + (mz3 - mz0) kappa + (mz2 + mz0 -
      mz1 - mz3) rho kappa + mz0
a = {mr[rho, kappa], mz[rho, kappa]};
b = {rho, kappa};
(*Determinant of the Jacobian-for determining area of the physical \
element.*)
j = Det[D[a, {b}]];
(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)

BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
```

```
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
(*The r value gets transformed back to the reference element*)
Kr = \
-mu*Integrate[
   BLD1[rho, kappa]*D[BLD1[rho, kappa], rho]*j, {rho, 0, 1}, {kappa,
    0, 1}]
mr0 = 0.0;
mr1 = 0.25;
mz0 = 0.0;
mz3 = 0.25;
mu = -0.495004692;
BLD1[r0, z0]
rho = r1;
D[BLD1[rho, kappa], kappa]
Kr
```

1.4 z-Leakage Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*z-Leakage matrix for RZ BLD Gauss-Legendre on \
quadrilaterals*)Clear[r0, r1, r2, r3, z0, z1, z2, z3, mr0, mr1, mr2, \
mr3, mz0, mz1, mz2, mz3, xi, rho, kappa]
(*Define physical integration point coordinates to compare to MFEM \
output:*)
r0 = (1 + (-Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
r1 = (1 + (Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
z0 = (1 + (-Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
z3 = (1 + (Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);

(*Use to define rectangular mesh:*)
r2 = r1;
```

```
r3 = r0;

z1 = z0;

z2 = z3;

(*Transform arbitrary quadrilateral to reference square*)


r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0

z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0

(*Mesh nodes are needed to conserve cell volume.Cannot define mr0=0 \
nor mz0=0 here.Perform integration and then set the values.*)
\
(*Comment out to get the general mesh element mr1=0.25;
mz3=0.25;*)

mr2 := mr1

mr3 := mr0

mz1 := mz0

mz2 := mz3

mr[rho_, kappa_] := (mr1 - mr0) rho + (mr3 - mr0) kappa + (mr2 + mr0 -
      mr1 - mr3) rho kappa + mr0

mz[rho_, kappa_] := (mz1 - mz0) rho + (mz3 - mz0) kappa + (mz2 + mz0 -
      mz1 - mz3) rho kappa + mz0

a = {mr[rho, kappa], mz[rho, kappa]};

b = {rho, kappa};

(*Determinant of the Jacobian-for determining area of the physical \
element.*)

j = Det[D[a, {b}]];

(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)


BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
```

```
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
(*The r value gets transformed back to the reference element*)
Kz = \
-xi*Integrate[
   BLD1[rho, kappa]*D[BLD1[rho, kappa], kappa]*j, {rho, 0, 1}, {kappa,
     0, 1}]
mr0 = 0.0;
mr1 = 0.25;
mz0 = 0.0;
mz3 = 0.25;
xi = -0.868890301;
D[BLD1[rho, kappa], kappa]
xi*j
Kz
Kr + Kz
```

1.5 z-Leakage Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*z-Leakage matrix for RZ BLD Gauss-Legendre on \
quadrilaterals*)Clear[r0, r1, r2, r3, z0, z1, z2, z3, mr0, mr1, mr2, \
mr3, mz0, mz1, mz2, mz3, xi, rho, kappa]
(*Define physical integration point coordinates to compare to MFEM \
output:*)
r0 = (1 + (-Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);
```

```
r1 = (1 + (Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);

z0 = (1 + (-Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);

z3 = (1 + (Sqrt[1/3]))/(1 - (-1))*(0.25 - 0);


(*Use to define rectangular mesh:*)

r2 = r1;

r3 = r0;

z1 = z0;

z2 = z3;

(*Transform arbitrary quadrilateral to reference square*)


r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0

z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0

(*Mesh nodes are needed to conserve cell volume.Cannot define mr0=0 \
nor mz0=0 here.Perform integration and then set the values.*)
\
(*Comment out to get the general mesh element mr1=0.25;
mz3=0.25;*)

mr2 := mr1

mr3 := mr0

mz1 := mz0

mz2 := mz3

mr[rho_, kappa_] := (mr1 - mr0) rho + (mr3 - mr0) kappa + (mr2 + mr0 -
      mr1 - mr3) rho kappa + mr0

mz[rho_, kappa_] := (mz1 - mz0) rho + (mz3 - mz0) kappa + (mz2 + mz0 -
      mz1 - mz3) rho kappa + mz0

a = {mr[rho, kappa], mz[rho, kappa]};

b = {rho, kappa};

(*Determinant of the Jacobian-for determining area of the physical \
element.*)
```

```
j = Det[D[a, {b}]];
(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)

BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
(*The r value gets transformed back to the reference element*)
Kz = \
-xi*Integrate[
   BLD1[rho, kappa]*D[BLD1[rho, kappa], kappa]*j, {rho, 0, 1}, {kappa,
     0, 1}]
mr0 = 0.0;
mr1 = 0.25;
mz0 = 0.0;
mz3 = 0.25;
xi = -0.868890301;
D[BLD1[rho, kappa], kappa]
xi*j
Kz
Kr + Kz
```

1.6 r Surface-Integral Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*r surface-integral matrix for RZ BLD Gauss-Legendre on \
quadrilaterals*)
Clear[r0, r1, r2, r3, z0, z1, z2, z3]
(*Define physical integration point coordinates to compare to MFEM \
output:
r0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
r1=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);
z0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
z3=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);*)
(*Use to define rectangular \
mesh:*)
r2 = r1;
r3 = r0;
z1 = z0;
z2 = z3;
(*Transform arbitrary quadrilateral to reference square*)

r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0
z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0
a = {r[rho, kappa], z[rho, kappa]};
b = {rho, kappa};
(*Determinant of the Jacobian-for determining area of the physical \
element.*)
j = Det[D[a, {b}]];
(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)
```

```
BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
(*The r value gets transformed back to the reference element*)
Lr =
 Integrate[r1*BLD1[1, kappa]*BLD1[1, kappa]*j, {kappa, 0, 1}] -
  Integrate[r0*BLD1[0, kappa]*BLD1[0, kappa]*j, {kappa, 0, 1}]
r0 = 0.0;
r1 = 0.25;
z0 = 0.0;
z3 = 0.25;
```

1.7 z Surface-Integral Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*z surface-integral matrix for RZ BLD Gauss-Legendre on \
quadrilaterals*)
Clear[r0, r1, r2, r3, z0, z1, z2, z3]
(*Define physical integration point coordinates to compare to MFEM \
output:
r0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
r1=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);
z0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
z3=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);*)
(*Use to define rectangular \
```

```
mesh:*)
r2 = r1;
r3 = r0;
z1 = z0;
z2 = z3;
(*Transform arbitrary quadrilateral to reference square*)

r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0
z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0
a = {r[rho, kappa], z[rho, kappa]};
b = {rho, kappa};
(*Determinant of the Jacobian-for determining area of the physical \
element.*)
j = Det[D[a, {b}]];
(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)

BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
(*The r value gets transformed back to the reference element*)
Lz =
 Integrate[r[rho, 1]*BLD1[rho, 1]*BLD1[rho, 1]*j, {rho, 0, 1}] -
```

```
  Integrate[r[rho, 0]*BLD1[rho, 0]*BLD1[rho, 0]*j, {rho, 0, 1}]
r0 = 0.0;
r1 = 0.25;
z0 = 0.0;
z3 = 0.25;
Lz
Lr + Lz
```

1.8 LinearForm Angular Redistribution Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*LinearForm Angular Redistribution matrix for RZ BLD Gauss-Legendre \
on quadrilaterals*)Clear[r0, r1, r2, r3, z0, z1, z2, z3, mr0, mr1, \
mr2, mr3, mz0, mz1, mz2, mz3]
(*Define physical integration point coordinates to compare to MFEM \
output:r0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
r1=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);
z0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
z3=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);*)
(*These make the mesh \
rectangular:*)
z2 := z3
z1 := z0
r2 := r1
r3 := r0
(*Quadrilateral transformation back to reference square-more general \
than rectangle transofrmation-but NOT curved mesh*)

r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0
z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0
```

```
(*Mesh nodes are needed to conserve cell volume.Cannot define mr0=0 \
nor mz0=0 here.Perform integration and then set the values.*)
\
(*Comment out to get the general mesh element mr1=0.25;
mz3=0.25;*)
mr2 := mr1
mr3 := mr0
mz1 := mz0
mz2 := mz3
mr[rho_, kappa_] := (mr1 - mr0) rho + (mr3 - mr0) kappa + (mr2 + mr0 -
      mr1 - mr3) rho kappa + mr0
mz[rho_, kappa_] := (mz1 - mz0) rho + (mz3 - mz0) kappa + (mz2 + mz0 -
      mz1 - mz3) rho kappa + mz0
a = {mr[rho, kappa], mz[rho, kappa]};
b = {rho, kappa};
(*Determinant of the Jacobian-for determining area of the physical \
element.*)
j = Det[D[a, {b}]];
(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)

BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
```

```
M = Integrate[
  BLD2[rho, kappa]*BLD2[rho, kappa]*j, {rho, 0, 1}, {kappa, 0, 1}]

mr0 = 0;
mz0 = 0;
mr1 = 0.25;
mz3 = 0.25;

alphatau = 0.845025867;
psiMinusHalf = 0.1591549;
M*alphatau*psiMinusHalf
```

1.9 LinearForm Scattering/Fixed Source Matrix Entries for RZ BLD Gauss-Legendre on Quadrilaterals

```
(*LinearForm scattering/fixed sources for RZ BLD Gauss-Legendre on \
quadrilaterals*)Clear[r0, r1, r2, r3, z0, z1, z2, z3, mr0, mr1, mr2, \
mr3, mz0, mz1, mz2, mz3, Source]
(*Define physical integration point coordinates to compare to MFEM \
output:r0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
r1=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);
z0=(1+(-Sqrt[1/3]))/(1-(-1))*(0.25-0);
z3=(1+(Sqrt[1/3]))/(1-(-1))*(0.25-0);*)
(*These make the mesh \
rectangular:*)
z2 := z3
z1 := z0
r2 := r1
r3 := r0
(*Quadrilateral transformation back to reference square-more general \
than rectangle transofrmation-but NOT curved mesh*)
```

```
r[rho_, kappa_] := (r1 - r0) rho + (r3 - r0) kappa + (r2 + r0 - r1 -
     r3) rho kappa + r0
z[rho_, kappa_] := (z1 - z0) rho + (z3 - z0) kappa + (z2 + z0 - z1 -
     z3) rho kappa + z0

(*Mesh nodes are needed to conserve cell volume.Cannot define mr0=0 \
nor mz0=0 here.Perform integration and then set the values.*)
\
(*Comment out to get the general mesh element mr1=0.25;
mz3=0.25;*)
mr2 := mr1
mr3 := mr0
mz1 := mz0
mz2 := mz3
mr[rho_, kappa_] := (mr1 - mr0) rho + (mr3 - mr0) kappa + (mr2 + mr0 -
      mr1 - mr3) rho kappa + mr0
mz[rho_, kappa_] := (mz1 - mz0) rho + (mz3 - mz0) kappa + (mz2 + mz0 -
      mz1 - mz3) rho kappa + mz0
a = {mr[rho, kappa], mz[rho, kappa]};
b = {rho, kappa};
(*Determinant of the Jacobian-for determining area of the physical \
element.*)
j = Det[D[a, {b}]];
(*Basis functions.The coefficients are from MATLAB script \
"GaussLegendreQuadrilateral.m".*)

BLD1[rho_, kappa_] :=
 3*rho*kappa - (3 + Sqrt[3])/2*rho - (3 + Sqrt[3])/2*
   kappa + (1 + Sqrt[3]/2)
BLD2[rho_, kappa_] := -3*rho*kappa + (3 + Sqrt[3])/2*rho +
  3/(3 + Sqrt[3])*kappa - 1/2
BLD3[rho_, kappa_] :=
```

```
 3*rho*kappa - 3/(3 + Sqrt[3])*rho -
  3/(3 + Sqrt[3])*kappa + (1 - Sqrt[3]/2)
BLD4[rho_, kappa_] := -3*rho*kappa +
  3/(3 + Sqrt[3])*rho + (3 + Sqrt[3])/2*kappa - 1/2
M = Integrate[
  Source*mr[rho, kappa]*BLD2[rho, kappa]*j, {rho, 0, 1}, {kappa, 0, 1}]
(*Source=0.7/(2*Pi);*)
Source = (0.3*1)/(2*Pi);
mr0 = 0;
mz0 = 0;
mr1 = 0.25;
mz3 = 0.25;
M
```