by Nasser M. Abbasi

```
Manipulate[poisson2DDirichletUnitSquareDirect[n,
  {west, south, east, north}, force, plotPoints, showGrid, zscale, autozScale],
 {{n, 10, "grid points n"}, 5, 31, 1, Appearance → "Labeled", ImageSize → Small},
 Delimiter,
 Style[Text["Dirichlet boundary conditions"], Bold],
 {{west, 0, "west"}, 0, .1, .01, Appearance → "Labeled", ImageSize → Small},
 {{south, 0, "south"}, 0, .1, .01, Appearance → "Labeled", ImageSize → Small},
 {{east, 0, "east"}, 0, .1, .01, Appearance → "Labeled", ImageSize → Small},
 {{north, 0, "north"}, 0, .1, .01, Appearance → "Labeled", ImageSize → Small},
 Delimiter,
 Style[Text["Plot options"], Bold],
 {{plotPoints, 30, "Plot Points"}, 5, 100, 1, Appearance → "Labeled", ImageSize → Small},
 {{zscale, 0.05, "max z scale"}, 0.001, .1, 0.001, Appearance → "Labeled",
  Enabled → Dynamic[autozScale == False], ImageSize → Small},
 {{showGrid, True, "show mesh"}, {True, False}, ImageSize → Small},
 {{autozScale, True, "automatic zscale"}, {True, False}, ImageSize → Small},

 ContinuousAction → False,
 SynchronousUpdating → True,
 AutorunSequencing → {2, 3, 4, 5},
 FrameMargins → 0,
 ImageMargins → 0,

 Initialization :→
  (

   $MinPrecision = $MachinePrecision; $MaxPrecision = $MachinePrecision;
   force[{x_, y_}] := -Exp[-(x - 0.25)^2 - (y - 0.6)^2];
   (*force[{x_,y_}]:=Cos[40 x y];*)


   (*--------------------------*)
   (*                          *)
   (*--------------------------*)
   getSparseALaplace2DfastMethod[
     nn_?(IntegerQ[#] && Positive[#] &)] := Module[{ii, tmp, mat},
     mat = SparseArray[
       {
        Band[{1, 1}] → -4.,
        Band[{2, 1}] → 1.,
        Band[{1, 2}] → 1.,
        Band[{1, nn + 1}] → 1.,
        Band[{nn + 1, 1}] → 1.
       }, {nn^2, nn^2}, 0.
      ];

     tmp = Table[ii * nn, {ii, nn - 1}];
     mat[[tmp, tmp + 1]] = 0.;
     mat[[tmp + 1, tmp]] = 0.;
     mat
```

```
 ];

(*--------------------------*)
(*                          *)
(*--------------------------*)
poisson2DDirichletUnitSquareDirect[n_?(IntegerQ[#] && Positive[#] &),
  bc_List, force_, plotPoints_?(IntegerQ[#] && Positive[#] &), showGrid_, zscale_,
  autozScale_] := Module[{h, xcoord, ycoord, f, grid, A, i, j, sol, b, westBC,
   southBC, eastBC, northBC, ff, sol2, pError},

  westBC = bc[[1]];
  southBC = bc[[2]];
  eastBC = bc[[3]];
  northBC = bc[[4]];
  h = 1/(n - 1);
  xcoord = Table[(j - 1) * h, {i, 1, n}, {j, 1, n}];
  ycoord = Table[(n - i) * h, {i, 1, n}, {j, 1, n}];
  f = Table[force[{xcoord[[i, j]], ycoord[[i, j]]}], {i, 1, n}, {j, 1, n}];

  ff = f;
  ff[[2, 2 ;; -2]] += northBC/h^2;
  ff[[-2, 2 ;; -2]] += southBC/h^2;
  ff[[2 ;; -2, 2]] += westBC/h^2;
  ff[[2 ;; -2, -2]] += eastBC/h^2;
  A = getSparseALaplace2DfastMethod[n - 2];
  sol = LinearSolve[A/h^2, -Reverse@Flatten@ff[[2 ;; -2, 2 ;; -2]]];

  b = Reverse@Partition[sol, n - 2];
  sol2 = Table[0, {i, 1, n}, {j, 1, n}];
  sol2[[2 ;; -2, 2 ;; -2]] = b;
  sol2[[1, All]] = northBC;
  sol2[[-1, All]] = southBC;
  sol2[[All, 1]] = westBC;
  sol2[[All, -1]] = eastBC;
  sol =
   Flatten[Table[{xcoord[[i, j]], ycoord[[i, j]], sol2[[i, j]]}, {i, 1, n}, {j, 1, n}], 1];

  pSol = ListPlot3D[sol,
    PlotLabel → Style[
      Row[{"Numerical solution to Δ^2u= -Exp[-(x-0.25)^2-(y-0.6)^2] on unit square"}],
      Bold],
    ImagePadding → {{40, 20}, {30, 30}},
    AxesLabel → {"x", "y", None},
    PlotRange → {All, All, If[autozScale, All, {-zscale, zscale}]},
```

```
        ColorFunction → "SouthwestColors",
        MaxPlotPoints → plotPoints,
        ImageSize → 450,
        Mesh → If[showGrid, n, None],
        BoundaryStyle → Directive[Red, Thick]];

     pSol

    ];

  }
]
```

grid points n 7

Dirichlet boundary conditions

west 0.

south 0.03

east 0.08

north 0.05

Plot options

Plot Points 18

max z scale 0.05

show mesh ☑

automatic zscale ☑