In[1655]:=
```
Clear[sortPointsCC];
sortPointsCC[polyinds_, indTopts_, ptsToInds_] := Block[{cent, ordering, polyPoints},
  polyPoints = Lookup[indTopts, polyinds];
  cent = Mean@polyPoints;
  ordering = Ordering[ArcTan[#[[1]], #[[2]]] &@(# - cent) & /@ polyPoints];
  Lookup[ptsToInds, Part[polyPoints, ordering]]
 ]
```

In[1657]:=
```
T3candidates[vertexToCell_, indTopts_, cellToVertexG_] :=
  Block[{outervertindices, outercellsinds, outercells, regmem, outerverticespts},
   {outervertindices, outercellsinds} = Through[{Keys, Union@*Flatten@*Values}[#]] &@
     Select[vertexToCell, Length[#] < 3 &];
   outercells = Lookup[indTopts, cellToVertexG@#] & /@ outercellsinds;
   regmem = SignedRegionDistance@*Polygon /@ outercells;
   outerverticespts = Lookup[indTopts, outervertindices];
   {Position[(Thread[(#[outerverticespts] < 0)] & /@ regmem), True],
    outercellsinds, outercells, outervertindices}
  ];
```

In[1658]:=
```
T3Transition[markers_, outercellsinds_,
   outercells_, outervertindices_, vertToCell_, pToI_, ItoP_, CVG_] :=
  Block[{ci, vi, minorcellind, vert, vertexToCell = vertToCell, numcells,
    majorcellind, intersectcell, ptsToInd = pToI, majorcell, cellToVertexG = CVG,
    commonvertexQ, edgespartof, indTopts = ItoP, edgecoords, edgesminorcell,
    lines, intersects, fpos, intersectpts, newptsindices, ls},
   If[markers ≠ {},
    Do[
     (*take the marker and handle cases *)
     {ci, vi} = marker;
     minorcellind = outercellsinds[[ci]];
     vert = outervertindices[[vi]];
     majorcellind = If[Head[#] === Integer, numcells = 1;
         #, numcells = Length[#];
         #] &@Replace[Lookup[vertexToCell, vert], {z_Integer} :> z];
     intersectcell = Lookup[ptsToInd, outercells[[ci]]];
     majorcell = Lookup[cellToVertexG, majorcellind];
     Print[Flatten@{majorcellind, minorcellind}];
     commonvertexQ = If[numcells == 1,
        (Union[Flatten@Cases[Partition[majorcell, 2, 1, 1],
             {OrderlessPatternSequence[vert, _]}]] ⋂ intersectcell),
        Function[(Union[Flatten@Cases[Partition[#, 2, 1, 1],
              {OrderlessPatternSequence[vert, _]}]] ⋂ intersectcell)] /@ majorcell
       ] // (If[# ≠ {}, First@#, {}] &)@*Flatten;
     Which[
      (*Case A*)
      numcells == 1 && commonvertexQ === {},
      edgespartof = Cases[Partition[cellToVertexG[majorcellind], 2, 1, 1],
        {OrderlessPatternSequence[vert, _]}];
```

```
edgecoords = Lookup[indTopts, #] & /@ edgespartof;
edgesminorcell = Partition[cellToVertexG[minorcellind], 2, 1, 1];
lines = Map[Line@Lookup[indTopts, #] &, edgesminorcell];
intersects =
 Map[Function[x, Map[RegionIntersection[Line[x], #] &, lines]], edgecoords];
fpos = Last@FirstPosition[intersects, _Point, {2}];
intersectpts = Cases[intersects, {__?NumberQ}, {-2}];
newptsindices = Range[Max[ptsToInd] + 1, Max[ptsToInd] + 2];
AppendTo[indTopts, Thread[newptsindices → intersectpts]];
KeyDropFrom[indTopts, vert];
ptsToInd = AssociationMap[Reverse, indTopts];
cellToVertexG = MapAt[sortPointsCC[
    Flatten[# /. Thread[vert → {newptsindices}]], indTopts, ptsToInd] &,
  cellToVertexG, Key[majorcellind]];
cellToVertexG = MapAt[
  Block[{y},
    y = Partition[#, 2, 1, 1];
    sortPointsCC[DeleteDuplicates@
      Flatten@Replace[y, {x : OrderlessPatternSequence @@ y[[fpos]]} :→
         Flatten@Insert[{x}, newptsindices, 2], {1}], indTopts, ptsToInd]
   ] &, cellToVertexG, Key[minorcellind]];
,
(*Case B*)
numcells == 2 && commonvertexQ === {},
edgesminorcell = Partition[cellToVertexG[minorcellind], 2, 1, 1];
lines = Map[Line@Lookup[indTopts, #] &, edgesminorcell];
Do[
 edgespartof = Cases[Partition[cellToVertexG[majcelliter], 2, 1, 1],
   {OrderlessPatternSequence[vert, _]}];
 edgecoords = Lookup[indTopts, #] & /@ edgespartof;
 intersects =
  Map[Function[x, Map[RegionIntersection[Line[x], #] &, lines]], edgecoords];
 fpos = Last@FirstPosition[intersects, _Point, {2}];
 intersectpts = Cases[intersects, {__?NumberQ}, {-2}];
 Scan[(If[KeyFreeQ[ptsToInd, #],
     newptsindices = Max[ptsToInd] + 1;
     ptsToInd[#] = newptsindices;
     indTopts[newptsindices] = #]) &, intersectpts];
 newptsindices = Lookup[ptsToInd, intersectpts];
 cellToVertexG = MapAt[sortPointsCC[
     Flatten[# /. Thread[vert → {newptsindices}]], indTopts, ptsToInd] &,
   cellToVertexG, Key[majcelliter]]
 , {majcelliter, majorcellind}
];
KeyDropFrom[indTopts, vert];
ptsToInd = AssociationMap[Reverse, indTopts];
cellToVertexG = MapAt[
  Block[{y},
    y = Partition[#, 2, 1, 1];
    sortPointsCC[DeleteDuplicates@
      Flatten@Replace[y, {x : OrderlessPatternSequence @@ y[[fpos]]} :→
```

```
          Flatten@Insert[{x}, newptsindices, 2], {1}], indTopts, ptsToInd]
    ] &, cellToVertexG, Key[minorcellind]],
(*Case C*)
numcells == 1 && commonvertexQ =!= {},
edgespartof = Cases[Partition[cellToVertexG[majorcellind], 2, 1, 1],
  {OrderlessPatternSequence[vert, _]}];
edgecoords = Lookup[indTopts, #] & /@ edgespartof;
edgesminorcell = Partition[cellToVertexG[minorcellind], 2, 1, 1];
lines = Map[Line@Lookup[indTopts, #] &, edgesminorcell];
intersects =
 Map[Function[x, Map[RegionIntersection[Line[x], #] &, lines]], edgecoords];
fpos = Last@FirstPosition[intersects, _Point, {2}];
intersectpts = First@
  DeleteCases[Cases[intersects, {__?NumberQ}, {-2}], indTopts[commonvertexQ]];
newptsindices = Max[ptsToInd] + 1;
AppendTo[indTopts, newptsindices → intersectpts];
KeyDropFrom[indTopts, {vert, commonvertexQ}];
ptsToInd = AssociationMap[Reverse, indTopts];
cellToVertexG =
 MapAt[DeleteDuplicates[# /. (commonvertexQ | vert) → newptsindices] &,
  cellToVertexG, Key[majorcellind]];
cellToVertexG = MapAt[# /. commonvertexQ → newptsindices &,
  cellToVertexG, Key[minorcellind]],
(*Case D*)
numcells == 2 && commonvertexQ =!= {},
ls = {};
edgesminorcell = Partition[cellToVertexG[minorcellind], 2, 1, 1];
lines = Map[Line@Lookup[indTopts, #] &, edgesminorcell];
Do[
 edgespartof = Cases[Partition[cellToVertexG[majcelliter], 2, 1, 1],
   {OrderlessPatternSequence[vert, _]}];
 edgecoords = Lookup[indTopts, #] & /@ edgespartof;
 intersects =
  Map[Function[x, Map[RegionIntersection[Line[x], #] &, lines]], edgecoords];
 fpos = Last@FirstPosition[intersects, _Point, {2}];
 intersectpts =
  DeleteCases[Cases[intersects, {__?NumberQ}, {-2}], indTopts[commonvertexQ]];
 If[Length[intersectpts] == 2, AppendTo[ls, intersectpts]];
 Scan[(If[KeyFreeQ[ptsToInd, #],
     newptsindices = Max[ptsToInd] + 1;
     ptsToInd[#] = newptsindices;
     indTopts[newptsindices] = #]) &, intersectpts];
 newptsindices = Lookup[ptsToInd, intersectpts];
 cellToVertexG = MapAt[
   sortPointsCC[DeleteDuplicates@
      Flatten[# /. (commonvertexQ | vert) → newptsindices], indTopts, ptsToInd] &,
   cellToVertexG, Key[majcelliter]];
 ,
 {majcelliter, majorcellind}
];
```

```
    cellToVertexG = MapAt[
      sortPointsCC[DeleteDuplicates@Flatten[# /.
           commonvertexQ → Lookup[ptsToInd, intersectpts]], indTopts, ptsToInd] &,
      cellToVertexG, Key[minorcellind]];
    ];
    KeyDropFrom[indTopts, {vert, commonvertexQ}];
    ptsToInd = AssociationMap[Reverse, indTopts];
    vertexToCell = GroupBy[
      Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];
    , {marker, markers}]
   ];
   {indTopts, ptsToInd, vertexToCell, cellToVertexG}
  ];
```

Case A

In[1659]:=
```
SeedRandom[3];
mesh = VoronoiMesh[RandomReal[1, {200, 2}], {{0, 1}, {0, 1}}, ImageSize → Medium];
```

In[1661]:=
```
pts = MeshPrimitives[mesh, 0] /. Point → Sequence;
cornerpts = pts[[-4 ;;]];
pts = pts[[1 ;; -5]];
```

In[1664]:=
```
$ptsToInd = ptsToInd = AssociationThread[pts → Range@Length@pts];
$indTopts = indTopts = AssociationMap[Reverse][ptsToInd];
```

In[1666]:=
```
cellmeshprim = MeshPrimitives[mesh, 2];
cells = (MeshPrimitives[#, 0] & /@ cellmeshprim) /. Point → Sequence /.
    Thread[cornerpts → Nothing];
```

In[1668]:=
```
$cellToVertexG =
  cellToVertexG = AssociationThread[Range[Length@cells] → Map[ptsToInd, cells, {2}]];
$vertexToCell = vertexToCell =
    GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];
```

In[1670]:=
```
KeyDropFrom[cellToVertexG, {9, 169, 199}];
```

In[1671]:=
```
indTopts[280] = {0.8936, 0.7856};
indTopts[60] = {0.8936, 0.7856};
```

In[1673]:=
```
AppendTo[indTopts, 399 → {0.8936, 0.7856}];
```

In[1674]:=
```
KeyDropFrom[indTopts, {280, 60}];
```

In[1675]:=
```
KeyDropFrom[vertexToCell, {280, 60}];
```

In[1676]:=
```
AppendTo[vertexToCell, 399 → 127];
```

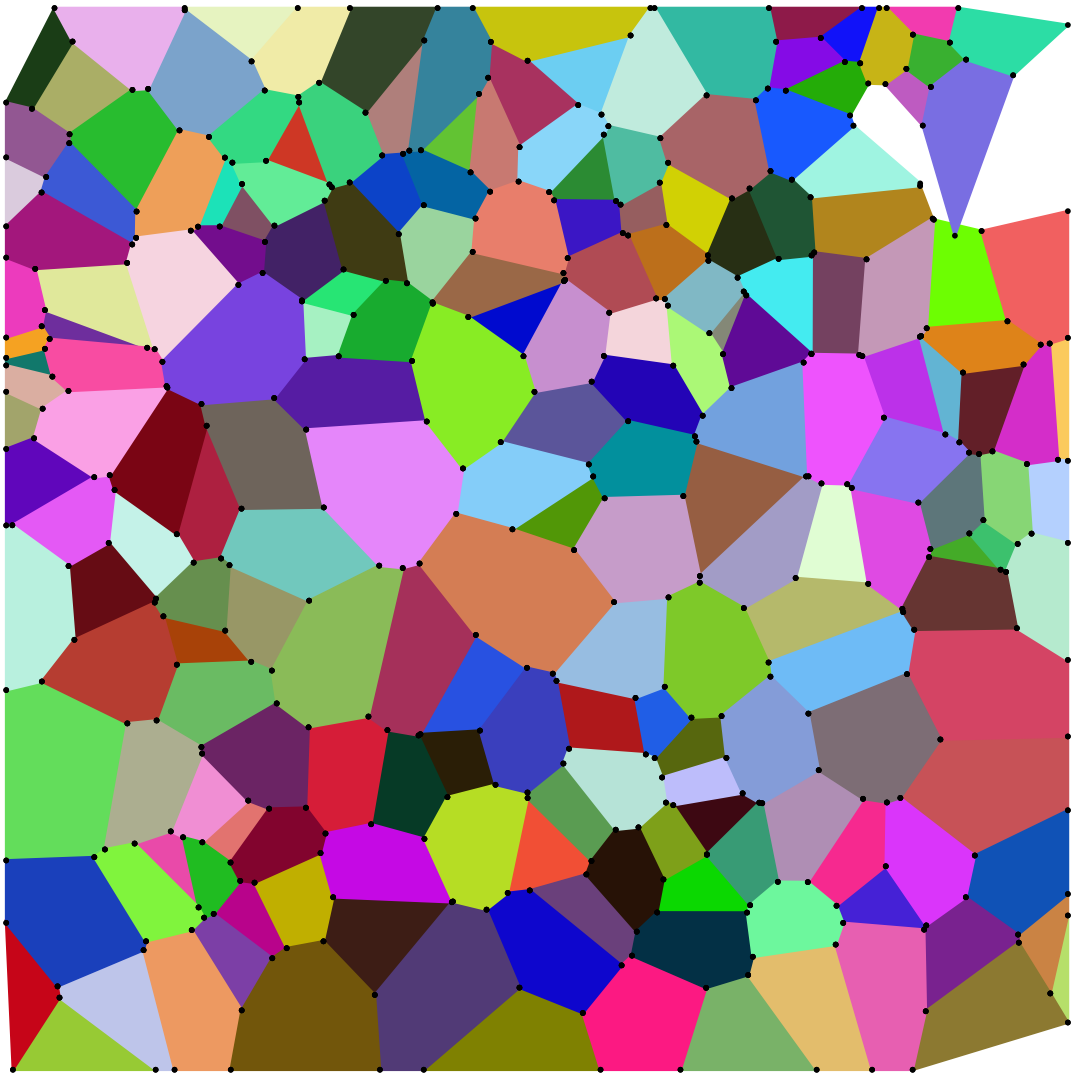
In[1677]:=
```
ptsToInd = AssociationMap[Reverse, indTopts];
```

In[1678]:=
```
cellToVertexG = Map[DeleteDuplicates, cellToVertexG /. {280 → 399, 60 → 399}];
```

In[1679]:=
```
vertexToCell = Association@
   DeleteCases[Normal[vertexToCell /. {9 | 169 | 199 → Sequence[]}], HoldPattern[_ → {}]];
```

In[1680]:=
```
Graphics[{Map[{RandomColor[], Polygon@Lookup[indTopts, #]} &, Values@cellToVertexG],
  Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1]}, ImageSize → Large]
```

Out[1680]=



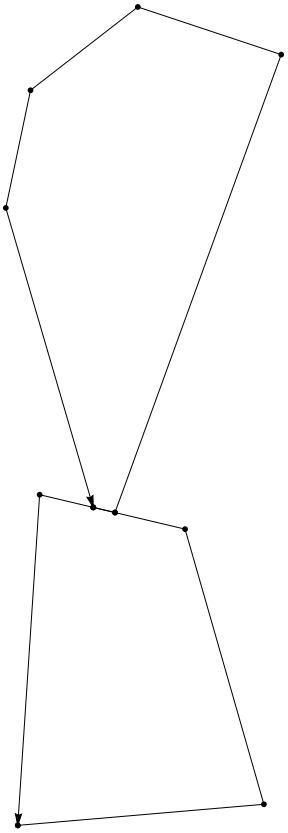
In[1681]:=
```
{markers, outercellsinds, outercells, outervertindices} =
  T3candidates[vertexToCell, indTopts, cellToVertexG];
```

In[1682]:=
```
{indTopts, ptsToInd, vertexToCell, cellToVertexG} = T3Transition[markers, outercellsinds,
   outercells, outervertindices, vertexToCell, ptsToInd, indTopts, cellToVertexG];
```

```
{127, 18}
```

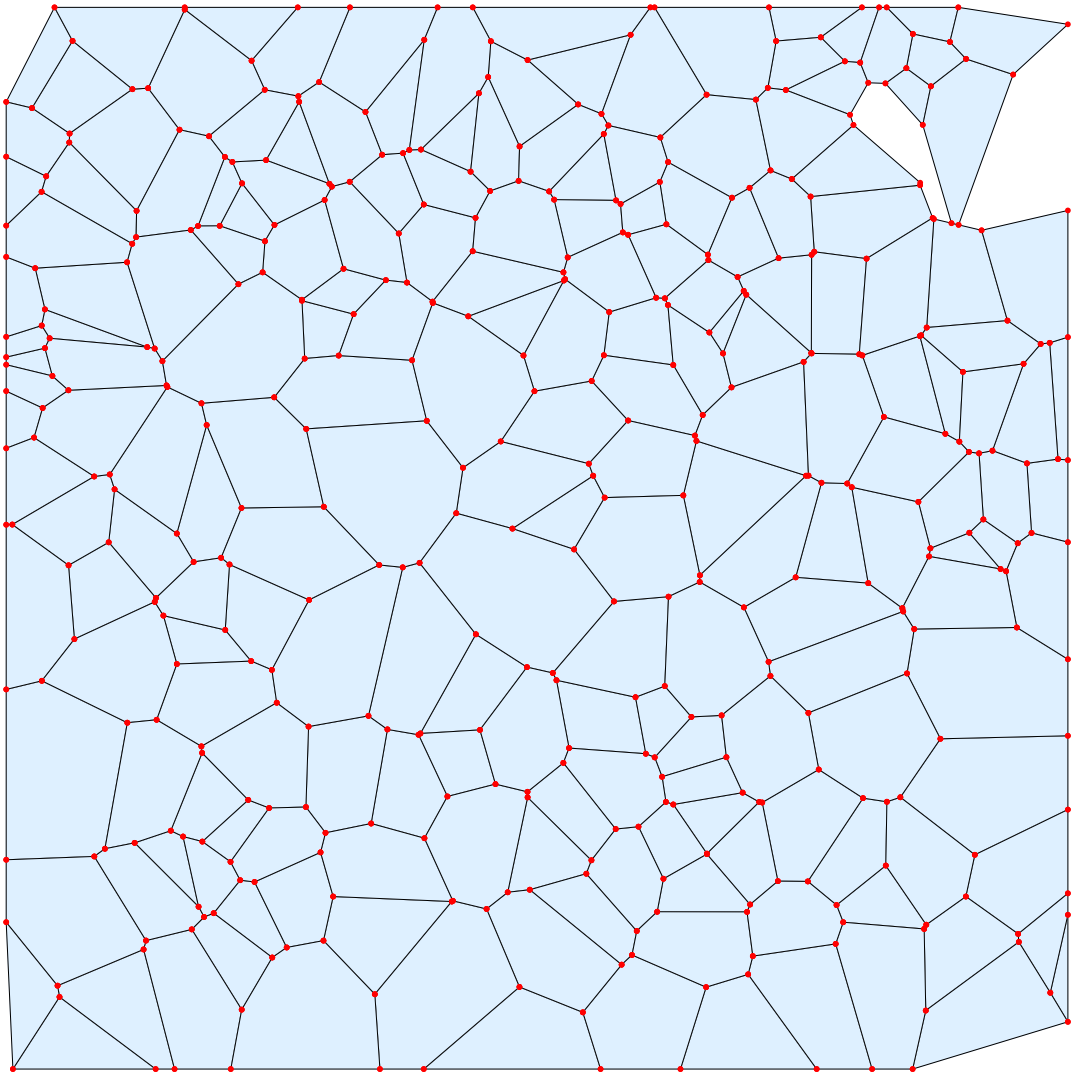
In[1683]:=
```
Graphics[Through[{Point, Arrow}[#]] &@Append[#, First@#] &@
    Lookup[indTopts, Lookup[cellToVertexG, #]] & /@ {127, 18}, ImageSize → Medium]
```

Out[1683]=



In[1684]:=
```
Graphics[{Map[{EdgeForm[Black], FaceForm[LightBlue], Polygon@Lookup[indTopts, #]} &,
    Values@cellToVertexG],
  Red, Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1] //
   DeleteDuplicates}, ImageSize → Large]
```

Out[1684]=



Case B

In[1685]:=
```
SeedRandom[3];
mesh = VoronoiMesh[RandomReal[1, {200, 2}], {{0, 1}, {0, 1}}, ImageSize → Medium];
```

In[1687]:=
```
pts = MeshPrimitives[mesh, 0] /. Point → Sequence;
cornerpts = pts[[-4 ;;]];
pts = pts[[1 ;; -5]];
```

In[1690]:=
```
$ptsToInd = ptsToInd = AssociationThread[pts → Range@Length@pts];
$indTopts = indTopts = AssociationMap[Reverse][ptsToInd];
```

In[1692]:=
```
cellmeshprim = MeshPrimitives[mesh, 2];
cells = (MeshPrimitives[#, 0] & /@ cellmeshprim) /. Point → Sequence /.
   Thread[cornerpts → Nothing];
```

In[1694]:=
```
$cellToVertexG =
  cellToVertexG = AssociationThread[Range[Length@cells] → Map[ptsToInd, cells, {2}]];
$vertexToCell = vertexToCell =
   GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];
```

In[1696]:=
```
KeyDropFrom[cellToVertexG, {9, 169, 199}];
```

In[1697]:=
```
indTopts[280] = {0.8936, 0.7856};
indTopts[60] = {0.8936, 0.7856};
```

In[1699]:=
```
AppendTo[indTopts, 399 → {0.8936, 0.7856}];
```

In[1700]:=
```
KeyDropFrom[indTopts, {280, 60}];
```

In[1701]:=
```
KeyDropFrom[vertexToCell, {280, 60}];
```

In[1702]:=
```
AppendTo[vertexToCell, 399 → 127];
```

In[1703]:=
```
ptsToInd = AssociationMap[Reverse, indTopts];
```

In[1704]:=
```
cellToVertexG = Map[DeleteDuplicates, cellToVertexG /. {280 → 399, 60 → 399}];
```

In[1705]:=
```
vertexToCell = Association@
   DeleteCases[Normal[vertexToCell /. {9 | 169 | 199 → Sequence[]}], HoldPattern[_ → {}]];
```

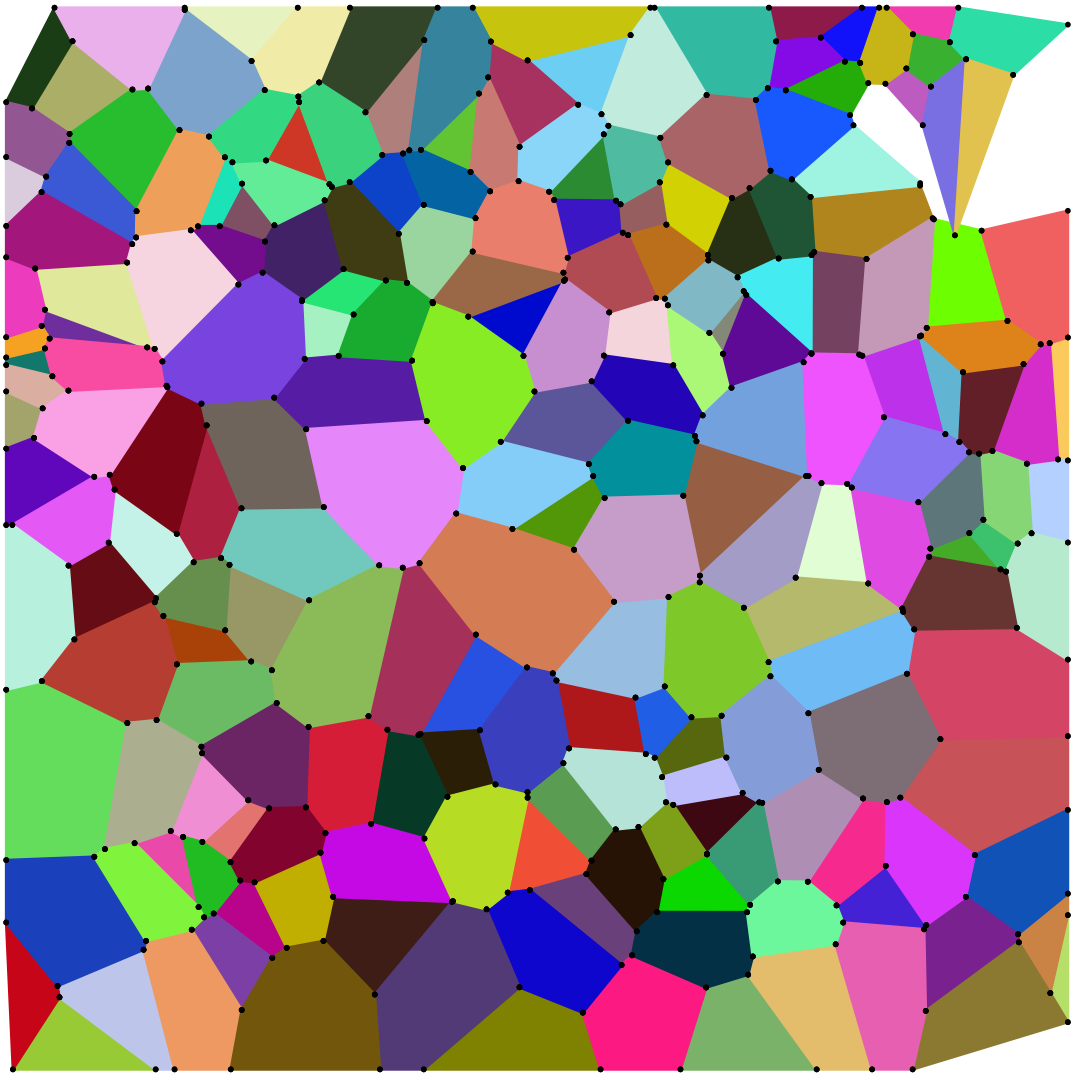
In[1706]:=
```
cellToVertexG[127] = {256, 399, 95, 61};
```

In[1707]:=
```
AppendTo[cellToVertexG, 201 → {279, 95, 399}];
```

In[1708]:=
```
vertexToCell =
  GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];
```

In[1709]:=
```
Graphics[{Map[{RandomColor[], Polygon@Lookup[indTopts, #]} &, Values@cellToVertexG],
  Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1]}, ImageSize → Large]
```

Out[1709]=



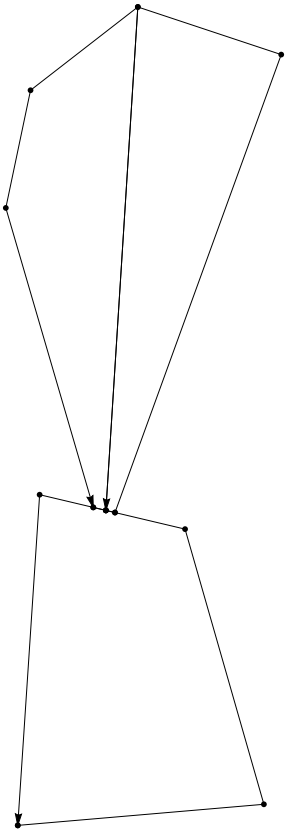
In[1710]:=
```
{markers, outercellsinds, outercells, outervertindices} =
  T3candidates[vertexToCell, indTopts, cellToVertexG];
```

In[1711]:=
```
{indTopts, ptsToInd, vertexToCell, cellToVertexG} = T3Transition[markers, outercellsinds,
   outercells, outervertindices, vertexToCell, ptsToInd, indTopts, cellToVertexG];
```

```
{127, 201, 18}
```

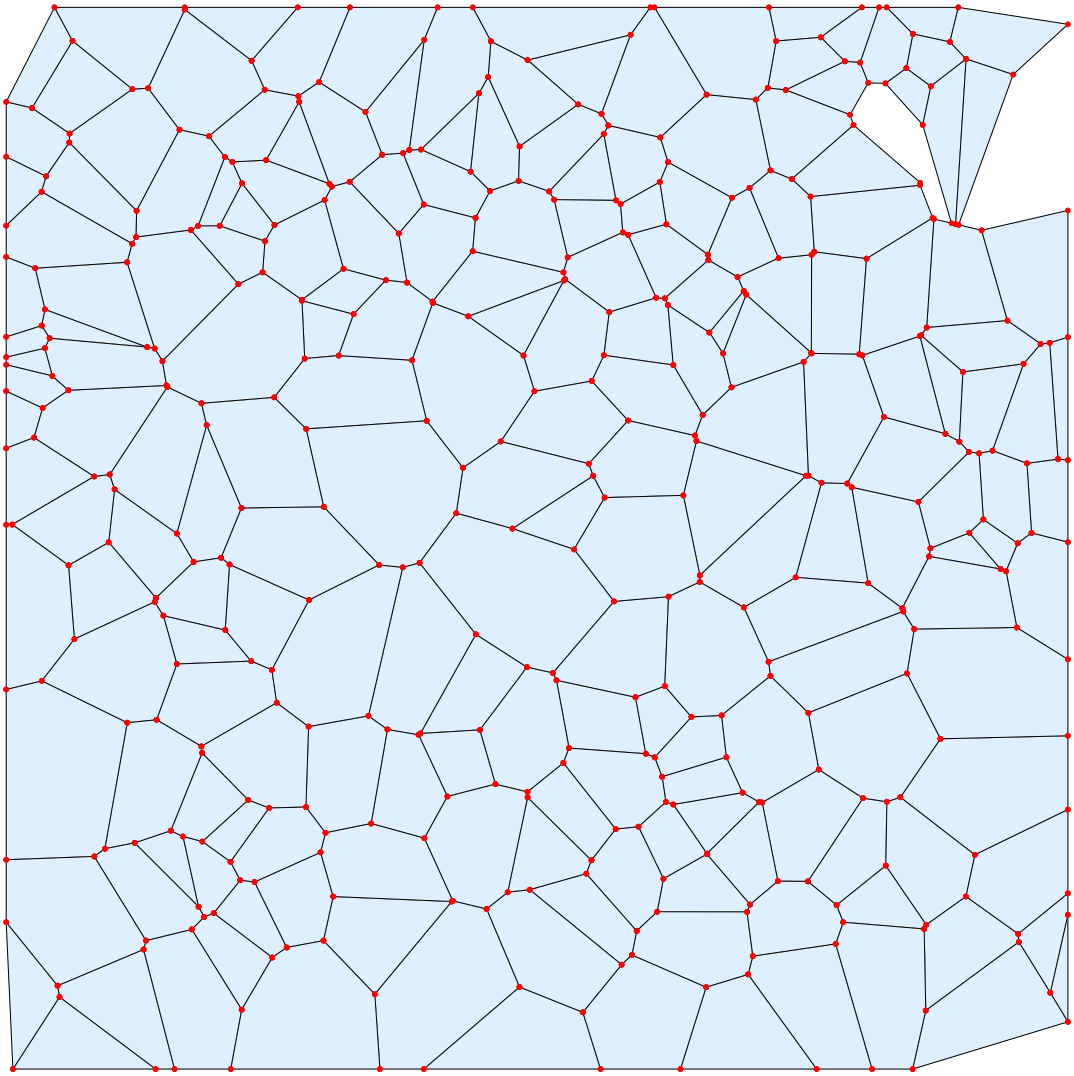
In[1712]:=
```
Graphics[Through[{Point, Arrow}[#]] &@Append[#, First@#] &@
    Lookup[indTopts, Lookup[cellToVertexG, #]] & /@ {127, 201, 18}, ImageSize → Medium]
```

Out[1712]=



In[1713]:=
```
Graphics[{Map[{EdgeForm[Black], FaceForm[LightBlue], Polygon@Lookup[indTopts, #]} &,
    Values@cellToVertexG],
  Red, Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1] //
   DeleteDuplicates}, ImageSize → Large]
```

Out[1713]=



Case C

In[1714]:=
```
SeedRandom[3];
mesh = VoronoiMesh[RandomReal[1, {200, 2}], {{0, 1}, {0, 1}}, ImageSize → Medium];
```

In[1716]:=
```
pts = MeshPrimitives[mesh, 0] /. Point → Sequence;
```

In[1717]:=
```
cornerpts = pts[[-4 ;;]];
pts = pts[[1 ;; -5]];
```

In[1719]:=
```
$ptsToInd = ptsToInd = AssociationThread[pts → Range@Length@pts];
$indTopts = indTopts = AssociationMap[Reverse][ptsToInd];
```

In[1721]:=
```
cellmeshprim = MeshPrimitives[mesh, 2];
cells = (MeshPrimitives[#, 0] & /@ cellmeshprim) /. Point → Sequence /.
   Thread[cornerpts → Nothing];
```

In[1723]:=
```
$cellToVertexG =
  cellToVertexG = AssociationThread[Range[Length@cells] → Map[ptsToInd, cells, {2}]];
$vertexToCell = vertexToCell =
   GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];
```

In[1725]:=
```
KeyDropFrom[cellToVertexG, {9, 169, 199}];
```

In[1726]:=
```
vertexToCell = Association@
   DeleteCases[Normal[vertexToCell /. {9 | 169 | 199 → Sequence[]}], HoldPattern[_ → {}]];
```

In[1727]:=
```
vertexToCell =
  GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];
```

In[1728]:=
```
AppendTo[indTopts, 400 → {1.05`, 0.6893896226716034`}];
```

In[1729]:=
```
cellToVertexG[13] = {72, 331, 369, 400, 351};
```

In[1730]:=
```
indTopts[72] = {0.9828574335519906`, 0.6740108914858205`};
```

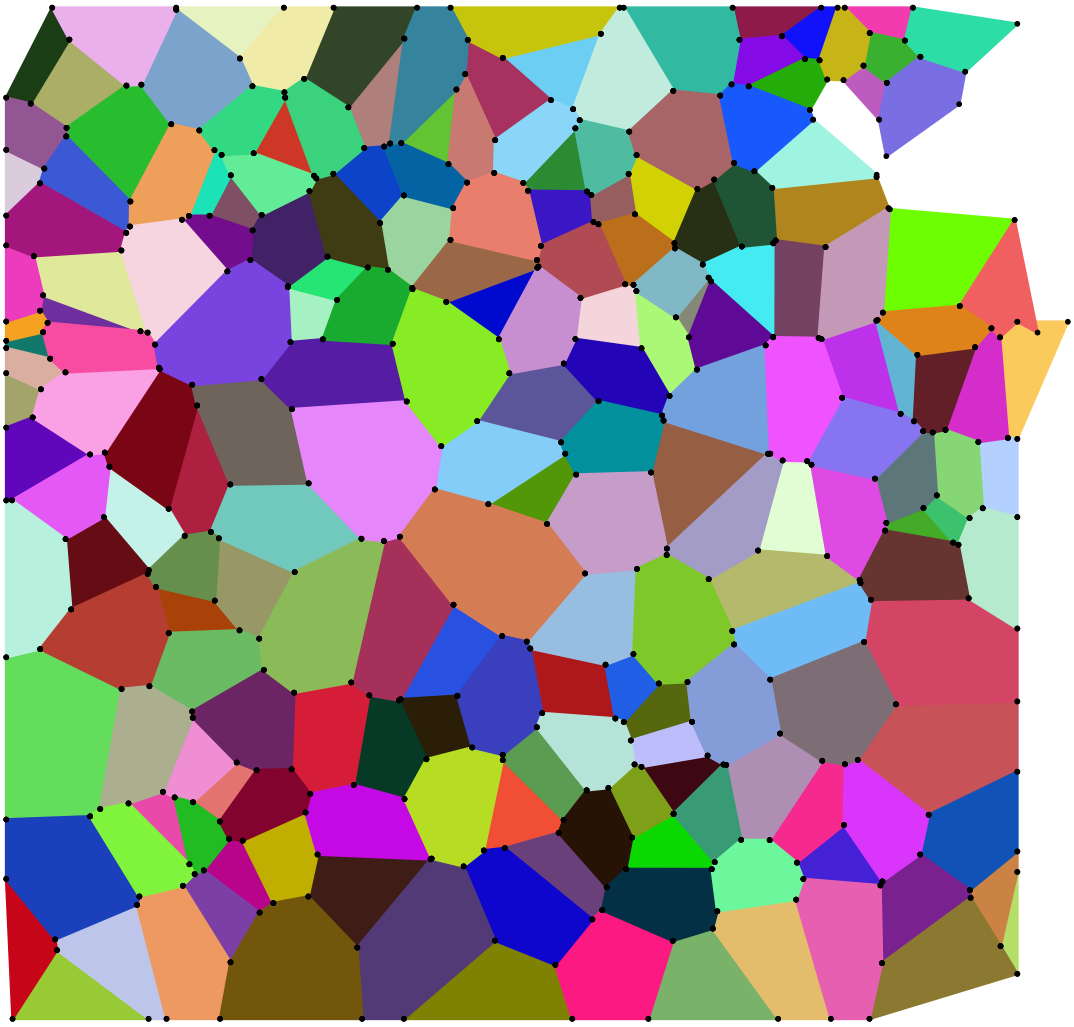
In[1731]:=
```
indTopts[355] = {1.02`, 0.6787049941899824`};
```

In[1732]:=
```
indTopts[140] = {0.997315022930967`, 0.7900129962646899`};
```

In[1733]:=
```
ptsToInd = AssociationMap[Reverse, indTopts];
```

In[1734]:=
```
Graphics[{Map[{RandomColor[], Polygon@Lookup[indTopts, #]} &, Values@cellToVertexG],
  Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1]}, ImageSize → Large]
```

Out[1734]=



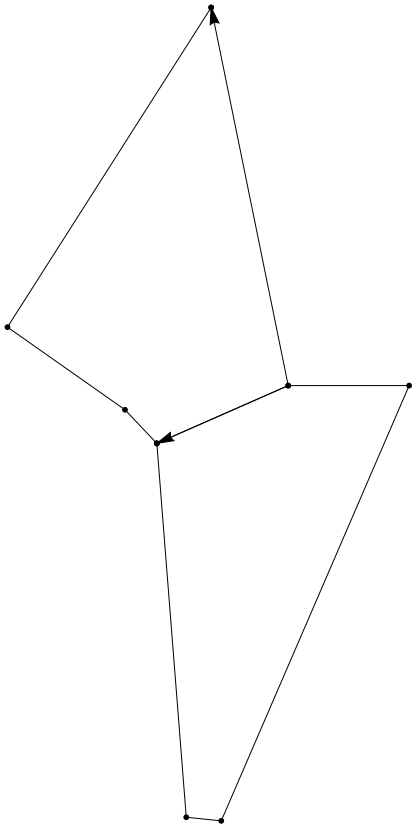
In[1735]:=
```
{markers, outercellsinds, outercells, outervertindices} =
  T3candidates[vertexToCell, indTopts, cellToVertexG];
```

In[1736]:=
```
{indTopts, ptsToInd, vertexToCell, cellToVertexG} = T3Transition[markers, outercellsinds,
   outercells, outervertindices, vertexToCell, ptsToInd, indTopts, cellToVertexG];
```

```
{97, 13}
```

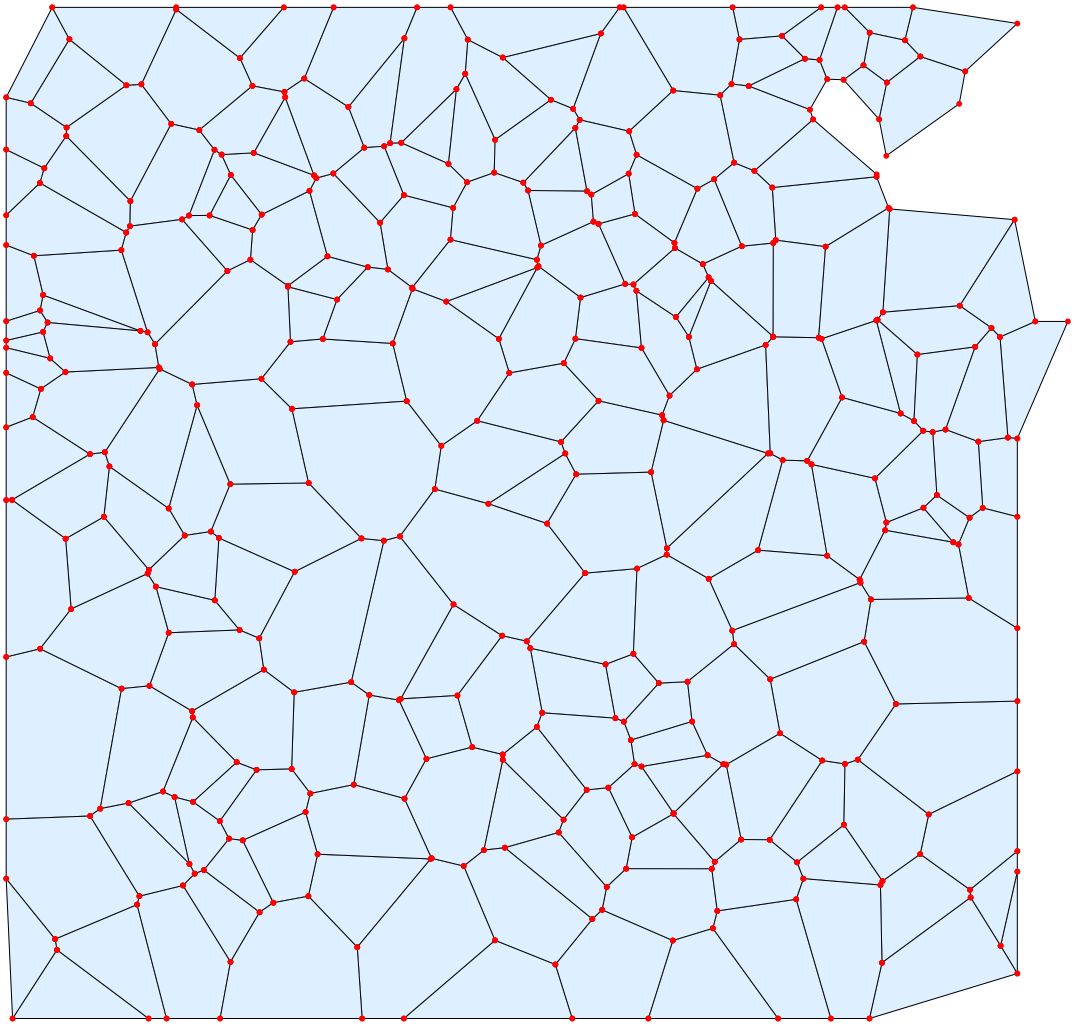
In[1737]:=
```
Graphics[Through[{Point, Arrow}[#]] &@Append[#, First@#] &@
    Lookup[indTopts, Lookup[cellToVertexG, #]] & /@ {97, 13}, ImageSize → Medium]
```

Out[1737]=



In[1738]:=
```
Graphics[{Map[{EdgeForm[Black], FaceForm[LightBlue], Polygon@Lookup[indTopts, #]} &,
   Values@cellToVertexG],
  Red, Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1] //
   DeleteDuplicates}, ImageSize → Large]
```

Out[1738]=



Case D

In[1739]:=
```
SeedRandom[3];
mesh = VoronoiMesh[RandomReal[1, {200, 2}], {{0, 1}, {0, 1}}, ImageSize → Medium];
```

In[1741]:=
```
pts = MeshPrimitives[mesh, 0] /. Point → Sequence;
```

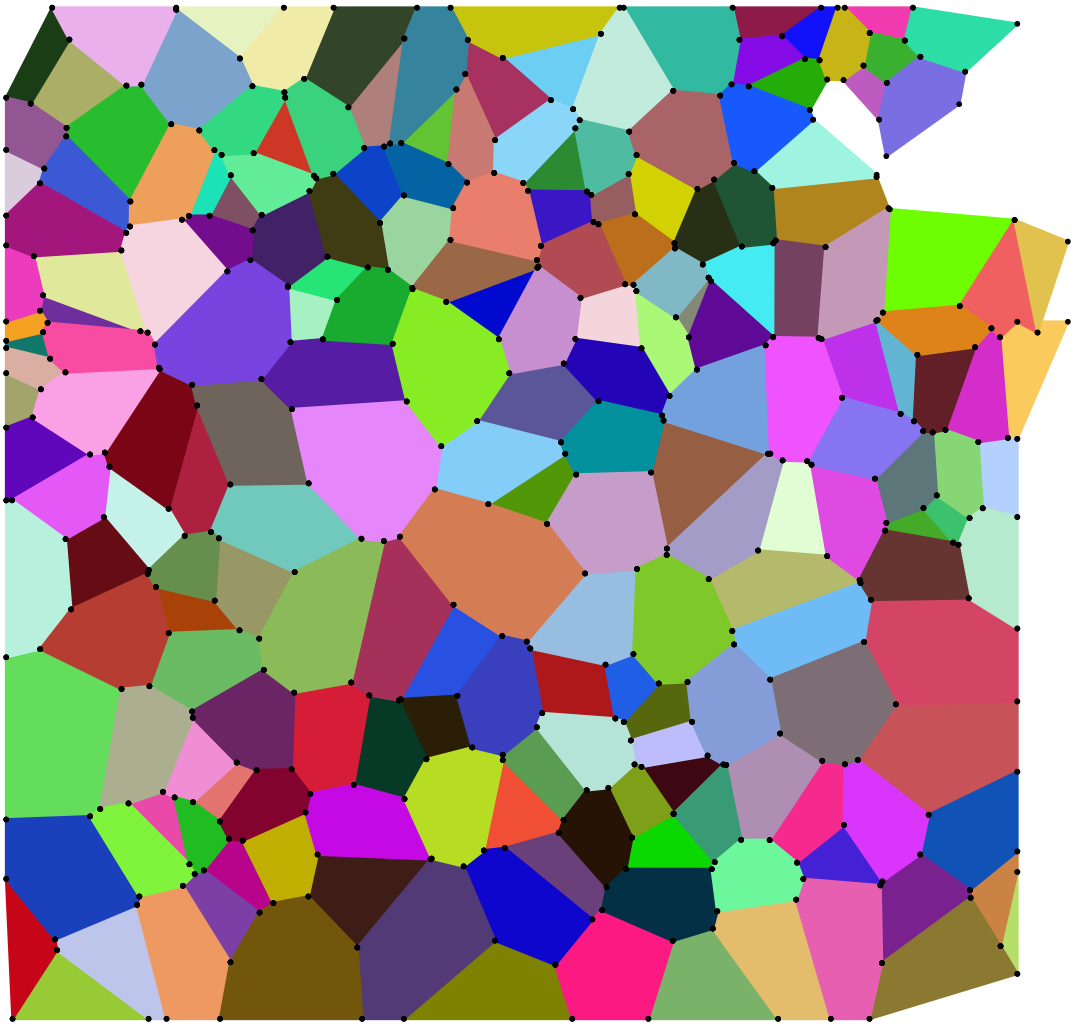
In[1742]:=
```
cornerpts = pts[[-4 ;;]];
pts = pts[[1 ;; -5]];
```

In[1744]:=
```
$ptsToInd = ptsToInd = AssociationThread[pts → Range@Length@pts];
$indTopts = indTopts = AssociationMap[Reverse][ptsToInd];
```

```
In[1746]:= cellmeshprim = MeshPrimitives[mesh, 2];
           cells = (MeshPrimitives[#, 0] & /@ cellmeshprim) /. Point → Sequence /.
              Thread[cornerpts → Nothing];

In[1748]:= $cellToVertexG =
             cellToVertexG = AssociationThread[Range[Length@cells] → Map[ptsToInd, cells, {2}]];
           $vertexToCell = vertexToCell =
              GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];

In[1750]:= KeyDropFrom[cellToVertexG, {9, 169, 199}];

In[1751]:= vertexToCell = Association@
              DeleteCases[Normal[vertexToCell /. {9 | 169 | 199 → Sequence[]}], HoldPattern[_ → {}]];

In[1752]:= vertexToCell =
             GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];

In[1753]:= AppendTo[indTopts, 400 → {1.05`, 0.6893896226716034`}];

In[1754]:= cellToVertexG[13] = {72, 331, 369, 400, 351};

In[1755]:= indTopts[72] = {0.9828574335519906`, 0.6740108914858205`};

In[1756]:= indTopts[355] = {1.02`, 0.6787049941899824`};

In[1757]:= indTopts[140] = {0.997315022930967`, 0.7900129962646899`};

In[1758]:= indTopts[401] = {1.05`, 0.7687049941899824`};

In[1759]:= ptsToInd = AssociationMap[Reverse, indTopts];

In[1760]:= cellToVertexG[201] = {140, 355, 401};

In[1761]:= vertexToCell =
             GroupBy[Flatten[(Reverse[#, 2] &) @* Thread /@ Normal@cellToVertexG], First → Last];
```

In[1762]:=
```
Graphics[{Map[{RandomColor[], Polygon@Lookup[indTopts, #]} &, Values@cellToVertexG],
  Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1]}, ImageSize → Large]
```

Out[1762]=



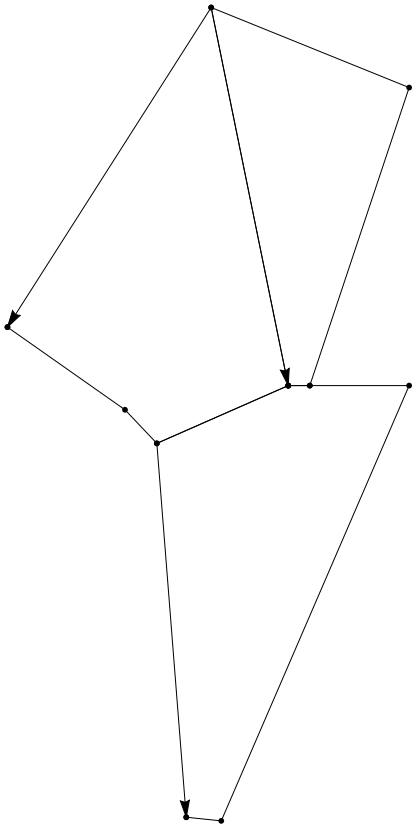
In[1763]:=
```
{markers, outercellsinds, outercells, outervertindices} =
  T3candidates[vertexToCell, indTopts, cellToVertexG];
```

In[1764]:=
```
{indTopts, ptsToInd, vertexToCell, cellToVertexG} = T3Transition[markers, outercellsinds,
   outercells, outervertindices, vertexToCell, ptsToInd, indTopts, cellToVertexG];
```

```
{97, 201, 13}
```

In[1765]:=
```
Graphics[Through[{Point, Arrow}[#]] &@Append[#, First@#] &@
    Lookup[indTopts, Lookup[cellToVertexG, #]] & /@ {97, 201, 13}, ImageSize → Medium]
```

Out[1765]=



In[1766]:=
```
Graphics[{Map[{EdgeForm[Black], FaceForm[LightBlue], Polygon@Lookup[indTopts, #]} &,
    Values@cellToVertexG],
  Red, Point@Flatten[Lookup[indTopts, #] & /@ Values@cellToVertexG, 1]}, ImageSize → Large]
```

Out[1766]=