In [44]:
```
import os
import json
from networkx.readwrite import json_graph

def read_graph_from_json(json_file):
    with open(json_file) as f:
        data = json.load(f)
    return json_graph.adjacency_graph(data)
```

In [45]:
```
from gerrychain import Graph

#read the iowa json county graph
# Define the file path and file name
filepath = '/Users/adenijibabalola/Desktop/DeterministicOperationsResearch/I
filename = 'IA_counties.json'

# Use os.path.join to correctly construct the file path
full_path = os.path.join(filepath,filename)
# Load the graph
G = Graph.from_json(full_path)

#Make sure the file is read correctly by printing the node #, and its popula
for node in G.nodes:
    print(node, G.nodes[node].get('POP100', 'Population not found'))
```

```
0 12943
1 10330
2 12138
3 10033
4 98537
5 10565
6 12329
7 20070
8 492401
9 9814
10 40105
11 17488
12 9110
13 15627
14 5674
15 20646
16 16878
17 26715
18 37813
19 25698
20 93667
21 5896
22 15211
23 11658
24 38910
25 174669
```

```
26 33555
27 6605
28 16662
29 14334
30 9388
31 105941
32 16384
33 16525
34 20565
35 6192
36 43127
37 8771
38 35437
39 20482
40 11934
41 3704
42 8751
43 9597
44 10837
45 17135
46 7577
47 43235
48 8634
49 15039
50 10019
51 24988
52 14182
53 10623
54 22565
55 35872
56 14061
57 10679
58 7005
59 13127
60 7645
61 18505
62 20760
63 7078
64 14828
65 14582
66 36999
67 15663
68 16548
69 9469
70 8996
71 4663
72 46460
73 9927
74 7443
75 131144
76 9748
77 19509
78 52403
```

```
79 7496
80 12012
81 10795
82 99266
83 18662
84 11746
85 25575
86 6497
87 19485
88 230299
89 17703
90 17043
91 20823
92 14484
93 99678
94 33414
95 22190
96 12317
97 7203
98 152854
```

In [47]:

```python
## Let's impose a 1% population deviation (+/-0.5%)
deviation = 0.01

import math
# No of districts
k = 4
#Summing up the total population in counties
total_population = sum(G.nodes[node]['POP100'] for node in G.nodes)

# Equations for bounds
L = math.ceil((1-deviation/2)*total_population/k)
U = math.floor((1+deviation/2)*total_population/k)

print("Using L =",L,"and U =",U,"and K =",k)
```

```
Using L = 793605 and U = 801580 and K = 4
```

In [48]:

```python
# Begin with model
import gurobipy as gp
from gurobipy import GRB

# create model
m = gp.Model()

# create variables
x = m.addVars(G.nodes, k, vtype=GRB.BINARY) # x[i,j] equals one when county 
y = m.addVars(G.edges, vtype=GRB.BINARY)    # y[u,v] equals one when edge {u

# objective is to minimize cut edges
m.setObjective( gp.quicksum( y[u,v] for u,v in G.edges ), GRB.MINIMIZE )
```

In [49]:
```
# add constraints saying that each county i is assigned to one district
m.addConstrs( gp.quicksum(x[i,j] for j in range(k)) == 1 for i in G.nodes)

# add constraints saying that each district has population at least L and at
m.addConstrs( gp.quicksum( G.nodes[i]['POP100'] * x[i,j] for i in G.nodes) >
m.addConstrs( gp.quicksum( G.nodes[i]['POP100'] * x[i,j] for i in G.nodes) <

# add constraints saying that edge {u,v} is cut if u is assigned to district
m.addConstrs( x[u,j] - x[v,j] <= y[u,v] for u,v in G.edges for j in range(k)

m.update()
```

In [50]:
```
# Now, let's add contiguity constraints and re-solve the model.
# We will use the contiguity constraints of Hojny et al. (MPC, 2021)
#   https://link.springer.com/article/10.1007/s12532-020-00186-3

# Add root variables: r[i,j] equals 1 if node i is the "root" of district j
r = m.addVars( G.nodes, k, vtype=GRB.BINARY )

# Add flow variables: f[u,v] = amount of flow sent across arc uv
#  Flows are sent across arcs of the directed version of G which we call DG

import networkx as nx
DG = nx.DiGraph(G)      # directed version of G

f = m.addVars( DG.edges )
```

In [51]:
```
# The big-M proposed by Hojny et al.
M = G.number_of_nodes() - k + 1

# Each district j should have one root
m.addConstrs( gp.quicksum( r[i,j] for i in G.nodes ) == 1 for j in range(k)

# If node i is not assigned to district j, then it cannot be its root
m.addConstrs( r[i,j] <= x[i,j] for i in G.nodes for j in range(k) )

# if not a root, consume some flow.
# if a root, only send out (so much) flow.
m.addConstrs( gp.quicksum( f[j,i] - f[i,j] for j in G.neighbors(i) )
             >= 1 - M * gp.quicksum( r[i,j] for j in range(k) ) for i in G.n

# do not send flow across cut edges
m.addConstrs( f[i,j] + f[j,i] <= M * ( 1 - y[i,j] ) for i,j in G.edges )

m.update()
```

In [52]:
```
# solve IP model
m.optimize()
```

```
Gurobi Optimizer version 10.0.3 build v10.0.3rc0 (mac64[arm])
```

```
CPU model: Apple M2
Thread count: 8 physical cores, 8 logical processors, using up to 8 threads

Optimize a model with 1716 rows, 1458 columns and 6990 nonzeros
Model fingerprint: 0xe75d93b4
Variable types: 444 continuous, 1014 integer (1014 binary)
Coefficient statistics:
  Matrix range     [1e+00, 5e+05]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 8e+05]
Presolve time: 0.02s
Presolved: 1716 rows, 1458 columns, 6990 nonzeros
Variable types: 444 continuous, 1014 integer (1014 binary)

Root relaxation: objective 0.000000e+00, 745 iterations, 0.02 seconds (0.02
work units)

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0    0.00000    0  412          -    0.00000      -     -    0s
     0     0    0.13229    0  420          -    0.13229      -     -    0s
     0     0    3.89100    0  415          -    3.89100      -     -    0s
     0     0    4.39046    0  425          -    4.39046      -     -    0s
     0     0    4.44167    0  429          -    4.44167      -     -    0s
     0     0    4.45279    0  427          -    4.45279      -     -    0s
     0     0    4.46568    0  430          -    4.46568      -     -    0s
     0     0    4.49784    0  426          -    4.49784      -     -    0s
     0     0    4.50690    0  424          -    4.50690      -     -    0s
     0     0    4.50918    0  423          -    4.50918      -     -    0s
     0     0    4.50918    0  421          -    4.50918      -     -    0s
     0     0    4.50918    0  421          -    4.50918      -     -    0s
     0     0    4.50918    0  421          -    4.50918      -     -    0s
     0     0    4.50918    0  421          -    4.50918      -     -    0s
     0     2    4.52615    0  421          -    4.52615      -     -    0s
H  785   663                      37.0000000    9.01575  75.6%   152    2s
H 1327   854                      35.0000000   10.14711  71.0%   164    4s
  1502   977   32.68564   28  435   35.00000   10.14711  71.0%   165    5s
  2360  1336   19.52204   19  418   35.00000   14.67644  58.1%   175   10s
  4335  2033   20.03843   22  414   35.00000   17.14163  51.0%   172   15s
  7097  3762   26.60019   25  436   35.00000   18.60925  46.8%   168   20s
  9960  5399   23.49082   21  439   35.00000   19.48936  44.3%   165   25s
 13963  7646   33.28237   41  306   35.00000   20.23112  42.2%   164   30s
 14504  9084   27.38567   30  311   35.00000   20.33970  41.9%   164   37s
 19191 10621   32.84419   44  330   35.00000   21.05841  39.8%   162   40s
 23175 12813   30.29846   32  380   35.00000   21.56821  38.4%   160   45s
 26498 14540   28.14321   34  315   35.00000   21.89176  37.5%   159   50s
 29071 15916   32.35565   36  429   35.00000   22.09372  36.9%   158   64s
 30122 16266     cutoff   38            35.00000   22.18979  36.6%   157   65s
 34888 18687   32.79401   34  341   35.00000   22.54188  35.6%   155   70s
 38969 20681   32.93731   30  315   35.00000   22.80566  34.8%   154   75s
 42100 22306   28.07201   30  426   35.00000   22.98507  34.3%   153   80s
```

```
46246 24537     cutoff   51         35.00000   23.22425  33.6%   153   85s
49482 26189   26.85835   28  419    35.00000   23.38478  33.2%   151   90s
54408 28418     cutoff   37         35.00000   23.62722  32.5%   150   95s
58474 30390   27.77101   34  373    35.00000   23.79131  32.0%   150  101s
60668 31456   29.05549   30  292    35.00000   23.87242  31.8%   149  105s
64400 33318   31.31166   36  379    35.00000   24.01657  31.4%   148  110s
68752 35315   25.80558   29  190    35.00000   24.13502  31.0%   148  115s
72945 37403   25.87921   30  336    35.00000   24.25644  30.7%   147  120s
76188 39009   33.33893   42  280    35.00000   24.34214  30.5%   146  125s
80382 41086   30.45402   34  315    35.00000   24.46442  30.1%   146  130s
84743 43145     cutoff   43         35.00000   24.55677  29.8%   145  135s
89031 45097     cutoff   29         35.00000   24.66427  29.5%   144  140s
93756 47391   31.24341   37  308    35.00000   24.78137  29.2%   143  145s
96840 48869   26.53343   32  286    35.00000   24.84809  29.0%   143  150s
101786 51074   33.69711   37  263    35.00000   24.96028  28.7%   143  155s
106057 52806   32.12102   32  397    35.00000   25.04343  28.4%   142  160s
110181 54551   28.80701   28  448    35.00000   25.11854  28.2%   142  165s
114393 56458   29.55163   31  300    35.00000   25.19519  28.0%   141  170s
119278 58480   29.41319   34  231    35.00000   25.27284  27.8%   141  175s
123210 60141     cutoff   54         35.00000   25.33620  27.6%   140  180s
127050 61915   27.17192   30  371    35.00000   25.39278  27.4%   140  185s
131403 63675   31.24369   38  204    35.00000   25.45028  27.3%   139  190s
135763 65508   33.58586   34  347    35.00000   25.52004  27.1%   139  195s
139555 67175   27.25432   31  386    35.00000   25.57117  26.9%   139  200s
144519 69464     cutoff   33         35.00000   25.64277  26.7%   138  205s
148895 71098     cutoff   39         35.00000   25.69861  26.6%   138  210s
152568 72718   28.51571   33  318    35.00000   25.74158  26.5%   137  215s
157662 74830     cutoff   42         35.00000   25.79969  26.3%   137  220s
162742 76965   32.70012   32  399    35.00000   25.85932  26.1%   136  225s
166891 78724   33.54547   39  217    35.00000   25.90103  26.0%   136  230s
170712 80275   32.62345   37  371    35.00000   25.93956  25.9%   136  235s
175345 82133   30.95576   41  236    35.00000   25.98855  25.7%   135  240s
179599 84023   31.78475   31  305    35.00000   26.03657  25.6%   135  245s
184214 85981   30.87069   33  392    35.00000   26.07887  25.5%   135  250s
189157 88077   28.94397   32  383    35.00000   26.13242  25.3%   135  255s
192194 89525   28.94644   28  388    35.00000   26.15756  25.3%   134  260s
197047 91438   27.95455   32  397    35.00000   26.19905  25.1%   134  265s
200485 92835     cutoff   34         35.00000   26.23501  25.0%   134  270s
205526 94914   29.32311   28  411    35.00000   26.28537  24.9%   133  275s
209886 96740     cutoff   51         35.00000   26.32722  24.8%   133  280s
214782 98525     cutoff   35         35.00000   26.37682  24.6%   133  285s
218788 100092     cutoff   41         35.00000   26.40910  24.5%   133  290s
223032 101818   33.84527   35  254    35.00000   26.44607  24.4%   132  295s
227429 103527   32.22844   43  243    35.00000   26.47672  24.4%   132  300s
231968 105259     cutoff   36         35.00000   26.51263  24.2%   132  305s
235969 106841   31.90975   34  393    35.00000   26.54838  24.1%   132  310s
240461 108458   30.70625   28  409    35.00000   26.58113  24.1%   131  315s
245234 110428   29.85883   35  309    35.00000   26.62007  23.9%   131  320s
248987 111753   31.21761   39  328    35.00000   26.64634  23.9%   131  325s
253234 113397   31.56452   40  369    35.00000   26.68020  23.8%   131  330s
257143 114892   32.37363   37  383    35.00000   26.70660  23.7%   131  335s
261369 116472   33.90007   35  340    35.00000   26.73480  23.6%   131  340s
265424 118042   32.22041   38  343    35.00000   26.76521  23.5%   130  345s
```

```
270312 119799   30.71576   29  345   35.00000   26.79471  23.4%   130  350s
274414 121400   29.90779   32  313   35.00000   26.82340  23.4%   130  355s
278422 122888     cutoff   43        35.00000   26.84936  23.3%   130  360s
282768 124672   32.43335   39  259   35.00000   26.87829  23.2%   130  365s
286925 126274   29.00347   32  358   35.00000   26.90545  23.1%   129  370s
290813 127708   30.75098   33  409   35.00000   26.93083  23.1%   129  375s
295140 129443     cutoff   32        35.00000   26.95543  23.0%   129  380s
299301 130780     cutoff   33        35.00000   26.97987  22.9%   129  385s
304078 132595     cutoff   32        35.00000   27.01039  22.8%   129  390s
308179 133890     cutoff   40        35.00000   27.03179  22.8%   128  395s
312292 135195     cutoff   32        35.00000   27.05415  22.7%   128  400s
316401 136562   29.33346   33  353   35.00000   27.08089  22.6%   128  405s
319824 138048     cutoff   40        35.00000   27.09853  22.6%   128  410s
322758 139201   28.33413   28  388   35.00000   27.11371  22.5%   128  415s
327045 140899   28.15586   41  290   35.00000   27.13822  22.5%   128  420s
331052 142220   33.63282   32  330   35.00000   27.16033  22.4%   128  425s
334899 143596     cutoff   29        35.00000   27.18076  22.3%   128  430s
339090 145069   28.97371   35  361   35.00000   27.20520  22.3%   127  435s
343581 146429   32.48662   34  358   35.00000   27.22739  22.2%   127  440s
347377 147831     cutoff   45        35.00000   27.25202  22.1%   127  445s
352465 149724   30.86873   29  428   35.00000   27.28155  22.1%   127  450s
356701 151131     cutoff   35        35.00000   27.30239  22.0%   127  455s
360667 152578   33.17712   39  285   35.00000   27.32506  21.9%   127  460s
364569 154051     cutoff   37        35.00000   27.34632  21.9%   127  465s
368819 155375   33.91904   36  333   35.00000   27.36679  21.8%   126  470s
373374 157025   30.69474   40  268   35.00000   27.38797  21.7%   126  475s
376941 157972   31.96930   38  344   35.00000   27.40422  21.7%   126  480s
380382 159516   33.18478   30  424   35.00000   27.42193  21.7%   126  485s
384884 160897     cutoff   42        35.00000   27.44218  21.6%   126  490s
389235 162377   32.91974   39  321   35.00000   27.45905  21.5%   126  495s
394010 163978   31.36474   35  426   35.00000   27.48212  21.5%   126  500s
397755 165187   30.91252   34  327   35.00000   27.49911  21.4%   126  505s
402025 166644   30.71045   31  362   35.00000   27.51894  21.4%   125  510s
406106 168104   29.46169   28  439   35.00000   27.53531  21.3%   125  515s
410111 169382   29.25569   32  290   35.00000   27.55311  21.3%   125  520s
414419 170669     cutoff   37        35.00000   27.57077  21.2%   125  525s
418131 171982     cutoff   29        35.00000   27.58494  21.2%   125  530s
423010 173637     cutoff   37        35.00000   27.60554  21.1%   125  535s
427248 174980   33.98781   36  337   35.00000   27.62363  21.1%   125  540s
431548 176285     cutoff   34        35.00000   27.64061  21.0%   125  545s
435463 177577   30.81218   35  312   35.00000   27.65583  21.0%   125  550s
439763 179102     cutoff   42        35.00000   27.67254  20.9%   124  555s
445182 180921   32.60452   38  375   35.00000   27.69168  20.9%   124  560s
449251 182122     cutoff   41        35.00000   27.70703  20.8%   124  565s
453368 183353   30.09552   34  331   35.00000   27.72575  20.8%   124  570s
457560 184767     cutoff   36        35.00000   27.73986  20.7%   124  575s
462756 186513     cutoff   45        35.00000   27.76272  20.7%   124  580s
466703 187820   29.86952   32  367   35.00000   27.77676  20.6%   124  585s
471836 189392   32.39715   38  437   35.00000   27.79473  20.6%   123  591s
475373 190379     cutoff   46        35.00000   27.80810  20.5%   123  595s
479329 191636   29.53675   35  373   35.00000   27.82523  20.5%   123  600s
483611 192818   29.76693   30  328   35.00000   27.83996  20.5%   123  605s
486813 193826   30.85290   34  260   35.00000   27.85110  20.4%   123  610s
```

491158	195004	33.60104	42	293	35.00000	27.86524	20.4%	123	615s
495869	196495	cutoff	34		35.00000	27.87984	20.3%	123	620s
500083	197834	29.83190	41	264	35.00000	27.89529	20.3%	123	625s
504511	199206	29.55898	37	313	35.00000	27.90976	20.3%	123	630s
508579	200271	cutoff	36		35.00000	27.92433	20.2%	123	635s
512195	201574	cutoff	37		35.00000	27.93612	20.2%	123	640s
517381	203128	30.33917	29	320	35.00000	27.95152	20.1%	122	645s
520824	204109	33.96907	34	297	35.00000	27.96264	20.1%	122	650s
526087	205584	28.91999	29	399	35.00000	27.97896	20.1%	122	655s
529504	206763	29.37531	35	322	35.00000	27.99011	20.0%	122	660s
534159	208105	30.71242	32	241	35.00000	28.00458	20.0%	122	665s
538172	209336	31.42570	32	372	35.00000	28.01619	20.0%	122	671s
542274	210431	33.74150	46	215	35.00000	28.03013	19.9%	122	675s
546113	211522	32.05245	35	313	35.00000	28.04409	19.9%	122	680s
550438	212843	cutoff	37		35.00000	28.05733	19.8%	122	685s
553258	213678	30.97821	27	345	35.00000	28.06657	19.8%	122	690s
557609	215028	cutoff	41		35.00000	28.08105	19.8%	122	695s
562132	216303	32.39627	37	408	35.00000	28.09380	19.7%	122	700s
566533	217585	31.98235	40	343	35.00000	28.10655	19.7%	121	705s
570816	218699	32.48763	33	433	35.00000	28.12020	19.7%	121	710s
575157	219830	33.25283	41	286	35.00000	28.13192	19.6%	121	715s
579487	221141	32.23223	32	399	35.00000	28.14552	19.6%	121	720s
583419	222336	cutoff	36		35.00000	28.15750	19.6%	121	725s
587767	223502	32.76089	32	338	35.00000	28.17146	19.5%	121	730s
592946	225033	infeasible	45		35.00000	28.18732	19.5%	121	736s
597209	225967	32.78623	36	299	35.00000	28.19910	19.4%	121	740s
600546	226915	cutoff	45		35.00000	28.20743	19.4%	121	745s
605200	228115	30.67344	34	368	35.00000	28.22286	19.4%	121	750s
609538	229370	cutoff	44		35.00000	28.23693	19.3%	121	755s
613511	230470	31.83382	39	225	35.00000	28.24924	19.3%	121	760s
618731	232142	cutoff	32		35.00000	28.26505	19.2%	120	765s
623279	233395	cutoff	35		35.00000	28.27565	19.2%	120	770s
626630	234368	32.40789	34	371	35.00000	28.28643	19.2%	120	775s
631198	235540	33.74528	36	227	35.00000	28.29707	19.2%	120	780s
636491	237129	31.79914	40	286	35.00000	28.31176	19.1%	120	785s
H637643	218540				34.0000000	28.31426	16.7%	120	788s
637654	219480	cutoff	44		34.00000	28.31464	16.7%	120	794s
642790	219504	31.95443	31	301	34.00000	28.33195	16.7%	120	796s
645935	220249	29.24911	38	242	34.00000	28.34214	16.6%	120	800s
649715	220958	29.92034	43	196	34.00000	28.35474	16.6%	120	805s
654263	221814	31.72929	36	310	34.00000	28.36898	16.6%	120	810s
658582	222560	cutoff	45		34.00000	28.38290	16.5%	120	815s
662748	223229	31.43541	32	248	34.00000	28.39622	16.5%	120	820s
666722	224002	31.01846	29	371	34.00000	28.40925	16.4%	120	825s
671144	224814	32.14599	27	389	34.00000	28.42244	16.4%	120	830s
676314	225667	32.22421	38	326	34.00000	28.43757	16.4%	119	836s
679634	226334	cutoff	33		34.00000	28.44890	16.3%	119	840s
684085	227079	30.88490	29	351	34.00000	28.46285	16.3%	119	845s
688516	227811	32.31138	35	272	34.00000	28.47479	16.3%	119	850s
692871	228576	31.01274	38	385	34.00000	28.48884	16.2%	119	855s
697083	229367	32.04092	32	402	34.00000	28.50169	16.2%	119	860s
701494	230131	cutoff	32		34.00000	28.51543	16.1%	119	865s
705759	230824	cutoff	37		34.00000	28.52964	16.1%	119	870s

```
709985 231507   32.42137   33  353   34.00000   28.54214  16.1%   119  875s
712174 231974     cutoff   47        34.00000   28.54826  16.0%   119  880s
716847 232664   32.79425   43  294   34.00000   28.56068  16.0%   119  885s
720646 233343   30.05282   36  443   34.00000   28.57151  16.0%   119  890s
725169 233982   30.93792   34  381   34.00000   28.58488  15.9%   119  896s
728798 234538   30.53164   41  347   34.00000   28.59450  15.9%   119  900s
733124 235142     cutoff   34        34.00000   28.60726  15.9%   119  905s
737481 235729   31.78585   36  195   34.00000   28.61908  15.8%   118  910s
741861 236443     cutoff   37        34.00000   28.63166  15.8%   118  915s
746497 237111     cutoff   38        34.00000   28.64430  15.8%   118  920s
750850 237728   28.94301   31  314   34.00000   28.65557  15.7%   118  925s
755511 238293     cutoff   37        34.00000   28.66865  15.7%   118  930s
760019 238847   32.80348   36  277   34.00000   28.68029  15.6%   118  935s
764264 239430   32.48031   34  412   34.00000   28.69279  15.6%   118  940s
768504 239996   32.39182   41  107   34.00000   28.70385  15.6%   118  945s
772210 240515     cutoff   37        34.00000   28.71411  15.5%   118  950s
776365 241056   31.94366   31  309   34.00000   28.72627  15.5%   118  955s
780631 241651     cutoff   31        34.00000   28.73771  15.5%   118  960s
785054 242370   32.64039   40  249   34.00000   28.75065  15.4%   118  965s
788024 242755   31.31489   32  306   34.00000   28.75796  15.4%   118  970s
792545 243385   30.70009   31  387   34.00000   28.77033  15.4%   118  975s
796364 243948   32.38027   29  402   34.00000   28.78113  15.3%   118  980s
800914 244570   32.43753   32  422   34.00000   28.79343  15.3%   118  985s
805265 245119   30.62013   33  363   34.00000   28.80494  15.3%   117  990s
809623 245706   31.82100   32  373   34.00000   28.81670  15.2%   117  995s
814238 246271     cutoff   37        34.00000   28.82722  15.2%   117 1000s
817621 246695   32.71006   36  293   34.00000   28.83586  15.2%   117 1005s
821525 247204   30.80031   38  291   34.00000   28.84612  15.2%   117 1010s
826143 247726   29.15866   35  372   34.00000   28.85963  15.1%   117 1015s
829985 248183   31.63050   33  335   34.00000   28.87063  15.1%   117 1020s
834688 248811   31.70993   35  373   34.00000   28.88349  15.0%   117 1025s
839048 249365   30.36886   42  319   34.00000   28.89455  15.0%   117 1030s
843589 249879     cutoff   31        34.00000   28.90599  15.0%   117 1035s
847018 250319   32.74612   31  341   34.00000   28.91444  15.0%   117 1040s
851792 250941   31.89926   44  320   34.00000   28.92631  14.9%   117 1045s
856349 251394     cutoff   35        34.00000   28.93728  14.9%   117 1050s
860853 252008   30.95189   37  316   34.00000   28.94824  14.9%   117 1055s
865461 252608   32.21538   32  395   34.00000   28.95898  14.8%   117 1060s
869962 253137     cutoff   33        34.00000   28.97112  14.8%   117 1065s
873717 253616   32.38655   31  461   34.00000   28.98012  14.8%   117 1070s
878227 254002   32.38304   45  347   34.00000   28.99048  14.7%   116 1075s
882802 254591   30.63113   32  406   34.00000   29.00126  14.7%   116 1080s
887167 255117   30.37983   37  300   34.00000   29.01166  14.7%   116 1085s
891918 255658   32.44275   38  318   34.00000   29.02188  14.6%   116 1090s
896200 256143   32.53225   44  166   34.00000   29.03236  14.6%   116 1095s
900634 256752   31.73896   33  425   34.00000   29.04292  14.6%   116 1100s
904526 257138   32.80578   38  371   34.00000   29.05242  14.6%   116 1105s
909073 257644     cutoff   34        34.00000   29.06346  14.5%   116 1110s
913312 258065     cutoff   34        34.00000   29.07354  14.5%   116 1115s
917924 258610   31.22891   30  427   34.00000   29.08380  14.5%   116 1120s
922640 259161   32.86844   44  254   34.00000   29.09465  14.4%   116 1125s
927058 259572   31.26740   45  283   34.00000   29.10434  14.4%   116 1130s
931456 260067   32.56347   36  267   34.00000   29.11315  14.4%   116 1135s
```

```
 936229 260567   32.20304   31  302   34.00000   29.12430  14.3%   116 1140s
H939064 202446                       33.0000000   29.13108  11.7%   116 1143s
 939860 202381     cutoff   32         33.00000   29.13393  11.7%   116 1145s
 941165 202300   31.43135   36  386   33.00000   29.13806  11.7%   116 1150s
 944972 202173     cutoff   36         33.00000   29.15300  11.7%   116 1155s
 949600 201937     cutoff   35         33.00000   29.16958  11.6%   115 1160s
 954460 201785     cutoff   30         33.00000   29.18653  11.6%   115 1165s
 959161 201602     cutoff   35         33.00000   29.20214  11.5%   115 1170s
 962769 201440   31.26730   39  255   33.00000   29.21458  11.5%   115 1175s
 968209 201261     cutoff   36         33.00000   29.23479  11.4%   115 1181s
 972531 201019     cutoff   42         33.00000   29.25063  11.4%   115 1185s
 976285 200845     cutoff   34         33.00000   29.26393  11.3%   115 1190s
 981204 200689   31.70201   34  334   33.00000   29.27973  11.3%   115 1195s
 984759 200499   30.97863   32  447   33.00000   29.29105  11.2%   115 1200s
 989386 200269   31.94406   32  375   33.00000   29.30630  11.2%   115 1205s
 993788 199954   31.77997   40  272   33.00000   29.32270  11.1%   115 1210s
 998306 199647   30.92026   31  268   33.00000   29.33811  11.1%   115 1215s
 1002831 199366     cutoff   37         33.00000   29.35283  11.1%   115 1220
s
 1007362 199073     cutoff   38         33.00000   29.36870  11.0%   115 1225
s
 1011808 198736   31.11063   33  394   33.00000   29.38358  11.0%   115 1230
s
 1016485 198437   31.87153   31  407   33.00000   29.39989  10.9%   115 1235
s
 1021087 198143   31.94314   29  420   33.00000   29.41476  10.9%   114 1240
s
 1025698 197835     cutoff   38         33.00000   29.43071  10.8%   114 1245
s
 1030157 197512   30.46166   46  286   33.00000   29.44588  10.8%   114 1250
s
 1034645 197169     cutoff   35         33.00000   29.46084  10.7%   114 1255
s
 1039361 196777     cutoff   35         33.00000   29.47640  10.7%   114 1260
s
 1043828 196395   30.61832   37  372   33.00000   29.49098  10.6%   114 1265
s
 1048567 195974     cutoff   33         33.00000   29.50765  10.6%   114 1270
s
 1053017 195614   31.74468   36  318   33.00000   29.52422  10.5%   114 1277
s
 1055002 195330   31.82572   35  363   33.00000   29.53062  10.5%   114 1280
s
 1059659 194932   30.60691   30  363   33.00000   29.54706  10.5%   114 1285
s
 1064014 194469     cutoff   36         33.00000   29.56214  10.4%   114 1290
s
 1067913 194085   31.76121   36  278   33.00000   29.57493  10.4%   114 1295
s
 1072708 193547   31.42854   37  427   33.00000   29.59176  10.3%   114 1300
s
 1077388 193121   31.44226   42  185   33.00000   29.60726  10.3%   114 1305
s
```

```
 1082136 192611     cutoff   34         33.00000   29.62272  10.2%   114 1310
s
 1086715 192122     cutoff   43         33.00000   29.63786  10.2%   114 1315
s
 1091505 191658   30.97463   39  363    33.00000   29.65275  10.1%   113 1320
s
 1096241 191045     cutoff   35         33.00000   29.66877  10.1%   113 1325
s
 1100838 190464     cutoff   32         33.00000   29.68382  10.0%   113 1330
s
 1105774 189901   31.85304   31  302    33.00000   29.69940  10.0%   113 1335
s
 1110533 189273   30.89193   43  312    33.00000   29.71395  10.0%   113 1340
s
 1115173 188657     cutoff   32         33.00000   29.72920  9.91%   113 1346
s
 1119036 188205   31.36064   29  387    33.00000   29.74089  9.88%   113 1350
s
 1123844 187521   31.91942   33  342    33.00000   29.75701  9.83%   113 1355
s
 1128305 186846   31.35067   31  325    33.00000   29.77221  9.78%   113 1360
s
 1133338 186179   31.42791   46  213    33.00000   29.78820  9.73%   113 1365
s
 1137911 185503   31.92634   46  191    33.00000   29.80406  9.68%   113 1370
s
 1142924 184879     cutoff   35         33.00000   29.81970  9.64%   113 1375
s
 1146615 184289   31.44727   31  342    33.00000   29.83057  9.60%   113 1380
s
 1151492 183497     cutoff   38         33.00000   29.84676  9.56%   113 1385
s
 1156227 182758     cutoff   32         33.00000   29.86298  9.51%   113 1390
s
 1160992 181945   31.83515   31  273    33.00000   29.87843  9.46%   112 1395
s
 1165872 181090     cutoff   38         33.00000   29.89474  9.41%   112 1400
s
 1169971 180486     cutoff   33         33.00000   29.90871  9.37%   112 1405
s
 1174247 179659     cutoff   40         33.00000   29.92181  9.33%   112 1410
s
 1179402 178799     cutoff   35         33.00000   29.93908  9.28%   112 1415
s
 1184064 177921     cutoff   40         33.00000   29.95556  9.23%   112 1420
s
 1188724 177136     cutoff   35         33.00000   29.97063  9.18%   112 1425
s
 1192898 176424   31.69949   31  343    33.00000   29.98419  9.14%   112 1430
s
 1197888 175555   30.42157   32  340    33.00000   30.00085  9.09%   112 1435
s
 1202993 174612   31.76865   44  199    33.00000   30.01722  9.04%   112 1440
```

```
s
 1207795 173705     cutoff   36        33.00000   30.03351  8.99%   112 1445
s
 1212671 172702     cutoff   33        33.00000   30.05018  8.94%   112 1450
s
 1217678 171709     cutoff   36        33.00000   30.06635  8.89%   112 1455
s
 1221518 170937     cutoff   42        33.00000   30.07913  8.85%   112 1460
s
 1226693 169888   31.06893   38  225   33.00000   30.09540  8.80%   112 1465
s
 1231357 168938     cutoff   33        33.00000   30.11073  8.76%   111 1470
s
 1235383 168151     cutoff   33        33.00000   30.12544  8.71%   111 1475
s
 1239776 167035     cutoff   31        33.00000   30.14126  8.66%   111 1480
s
 1244982 165970 infeasible   39        33.00000   30.15835  8.61%   111 1485
s
 1250194 164876   31.98217   38  370   33.00000   30.17434  8.56%   111 1490
s
 1254077 163982     cutoff   35        33.00000   30.18662  8.53%   111 1495
s
 1259108 162847   31.42824   36  155   33.00000   30.20360  8.47%   111 1500
s
 1263953 161621     cutoff   28        33.00000   30.21962  8.43%   111 1505
s
 1267545 160753     cutoff   32        33.00000   30.23247  8.39%   111 1510
s
 1272567 159448     cutoff   37        33.00000   30.25000  8.33%   111 1515
s
 1276686 158380     cutoff   33        33.00000   30.26411  8.29%   111 1520
s
 1281720 157063     cutoff   30        33.00000   30.28190  8.24%   111 1525
s
 1287102 155618   31.79839   35  269   33.00000   30.30084  8.18%   111 1530
s
 1291041 154480     cutoff   45        33.00000   30.31494  8.14%   111 1535
s
 1296414 153018   31.99021   27  352   33.00000   30.33395  8.08%   110 1540
s
 1301532 151588   31.74992   45  343   33.00000   30.35202  8.02%   110 1545
s
 1305585 150438   31.78531   33  362   33.00000   30.36553  7.98%   110 1550
s
 1310810 148935     cutoff   49        33.00000   30.38397  7.93%   110 1555
s
 1316130 147236   31.65126   30  335   33.00000   30.40444  7.87%   110 1560
s
 1321400 145640     cutoff   42        33.00000   30.42383  7.81%   110 1565
s
 1326542 144140     cutoff   35        33.00000   30.44182  7.75%   110 1570
s
```

```
 1331697 142760     cutoff   32          33.00000   30.46235  7.69%   110 1577
s
 1333869 141782   31.94244   38  361    33.00000   30.47051  7.67%   110 1580
s
 1340213 139688     cutoff   27          33.00000   30.49448  7.59%   110 1585
s
 1344182 138394     cutoff   34          33.00000   30.50941  7.55%   110 1590
s
 1349421 136645   30.87259   46  177    33.00000   30.52891  7.49%   110 1595
s
 1353845 135173     cutoff   37          33.00000   30.54499  7.44%   109 1600
s
 1357897 133772   31.63878   34  352    33.00000   30.56166  7.39%   109 1605
s
 1363460 131868     cutoff   41          33.00000   30.58187  7.33%   109 1610
s
 1367151 130473     cutoff   34          33.00000   30.59690  7.28%   109 1615
s
 1372565 128450   31.78089   42  265    33.00000   30.61781  7.22%   109 1620
s
 1377835 126443     cutoff   40          33.00000   30.63917  7.15%   109 1625
s
 1381945 124807     cutoff   41          33.00000   30.65656  7.10%   109 1630
s
 1386543 123005     cutoff   34          33.00000   30.67350  7.05%   109 1635
s
 1390643 121291   31.57359   44  184    33.00000   30.69254  6.99%   109 1640
s
 1395774 119139     cutoff   33          33.00000   30.71602  6.92%   109 1645
s
 1401215 116786   31.10054   39  163    33.00000   30.73900  6.85%   109 1650
s
 1406655 114447     cutoff   45          33.00000   30.76237  6.78%   108 1655
s
 1411936 112079     cutoff   42          33.00000   30.78479  6.71%   108 1660
s
 1418296 109137     cutoff   28          33.00000   30.81410  6.62%   108 1665
s
 1423534 106561     cutoff   35          33.00000   30.83935  6.55%   108 1670
s
 1429132 103854   31.17267   35  261    33.00000   30.86564  6.47%   108 1675
s
 1434371 101325     cutoff   37          33.00000   30.89064  6.39%   108 1680
s
 1439470 98695     cutoff   30          33.00000   30.91733  6.31%   108 1685s
 1444537 96023     cutoff   47          33.00000   30.94344  6.23%   108 1690s
 1450944 92668     cutoff   32          33.00000   30.97594  6.13%   108 1695s
 1456372 89687   31.32656   46  344    33.00000   31.00545  6.04%   107 1700s
 1462618 86152   31.73002   33  348    33.00000   31.03859  5.94%   107 1705s
 1468928 82539     cutoff   39          33.00000   31.07354  5.84%   107 1710s
 1475437 78614     cutoff   40          33.00000   31.11195  5.72%   107 1715s
 1481596 74813     cutoff   34          33.00000   31.14905  5.61%   107 1720s
 1488766 70302     cutoff   30          33.00000   31.19352  5.47%   107 1725s
```

```
 1494905 66249     cutoff   30         33.00000   31.23432  5.35%   106 1730s
 1501874 61297     cutoff   33         33.00000   31.28269  5.20%   106 1735s
 1509014 56126     cutoff   40         33.00000   31.33756  5.04%   106 1740s
 1516047 50790     cutoff   36         33.00000   31.39368  4.87%   106 1745s
 1524543 43996     cutoff   38         33.00000   31.46423  4.65%   105 1750s
 1531302 38470   31.81784   42  296    33.00000   31.52435  4.47%   105 1755s
 1541216 29917     cutoff   41         33.00000   31.62003  4.18%   105 1760s
 1550755 21267     cutoff   41         33.00000   31.72209  3.87%   104 1765s
 1564598  8198     cutoff   40         33.00000   31.88192  3.39%   104 1770s

Cutting planes:
  Gomory: 29
  Flow cover: 8
  Zero half: 1
  RLT: 22

Explored 1573757 nodes (162462278 simplex iterations) in 1772.22 seconds (33
25.92 work units)
Thread count was 8 (of 8 available processors)

Solution count 5: 33 33 34 ... 37

Optimal solution found (tolerance 1.00e-04)
Best objective 3.300000000000e+01, best bound 3.300000000000e+01, gap 0.000
0%
```

In [53]:

```
print("The number of cut edges is",m.objval)

# retrieve the districts and their populations
districts = [ [i for i in G.nodes if x[i,j].x > 0.5] for j in range(k)]
district_counties = [ [ G.nodes[i]["NAMELSAD20"] for i in districts[j] ] for
district_populations = [ sum(G.nodes[i]["POP100"] for i in districts[j]) for

# print district info
for j in range(k):
    print("District",j,"has population",district_populations[j],"and contain
    print("")
```

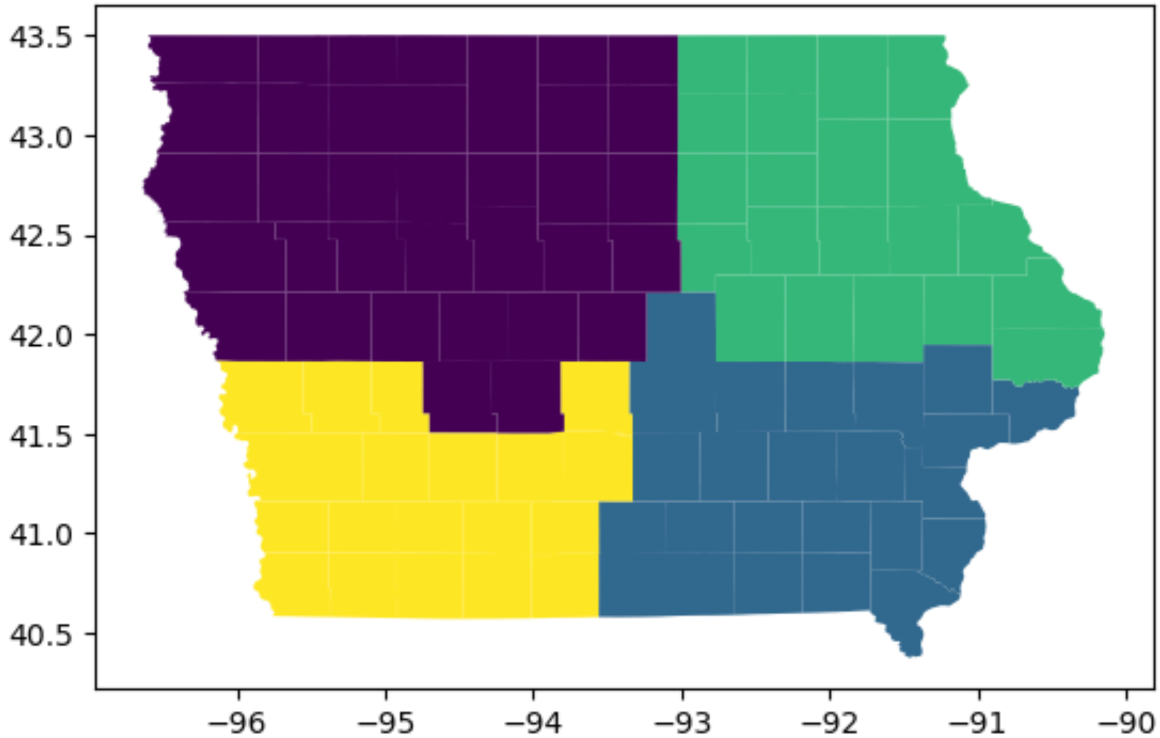
```
The number of cut edges is 33.0
District 0 has population 797302 and contains counties ['Wright County', 'St
ory County', 'Sac County', 'Hardin County', 'Boone County', 'Plymouth Count
y', 'Cherokee County', 'Emmet County', 'Woodbury County', 'Clay County', 'Cr
awford County', 'Osceola County', 'Cerro Gordo County', 'Greene County', 'Ly
on County', 'Monona County', 'Humboldt County', 'Hamilton County', 'Franklin
County', "O'Brien County", 'Guthrie County', 'Sioux County', 'Winnebago Coun
ty', 'Ida County', 'Carroll County', 'Pocahontas County', 'Kossuth County',
'Webster County', 'Palo Alto County', 'Calhoun County', 'Worth County', 'Han
cock County', 'Dickinson County', 'Buena Vista County', 'Dallas County']

District 1 has population 796929 and contains counties ['Keokuk County', 'Ma
rshall County', 'Davis County', 'Jasper County', 'Des Moines County', 'Scott
County', 'Lee County', 'Iowa County', 'Wapello County', 'Henry County', 'Lou
isa County', 'Monroe County', 'Muscatine County', 'Lucas County', 'Washingto
n County', 'Cedar County', 'Jefferson County', 'Poweshiek County', 'Wayne Co
unty', 'Marion County', 'Mahaska County', 'Appanoose County', 'Van Buren Cou
nty', 'Johnson County']

District 2 has population 798070 and contains counties ['Mitchell County', '
Grundy County', 'Winneshiek County', 'Delaware County', 'Floyd County', 'Jon
es County', 'Butler County', 'Buchanan County', 'Tama County', 'Bremer Count
y', 'Allamakee County', 'Howard County', 'Clinton County', 'Black Hawk Count
y', 'Fayette County', 'Chickasaw County', 'Dubuque County', 'Benton County',
'Jackson County', 'Linn County', 'Clayton County']

District 3 has population 798068 and contains counties ['Montgomery County',
'Union County', 'Polk County', 'Audubon County', 'Pottawattamie County', 'Ta
ylor County', 'Page County', 'Fremont County', 'Adams County', 'Cass Count
y', 'Decatur County', 'Harrison County', 'Madison County', 'Ringgold Count
y', 'Clarke County', 'Warren County', 'Adair County', 'Shelby County', 'Mill
s County']
```

In [54]:

```python
# To check if satisfies contiguity contraint
for district in districts:
    print("Is district =", district, "connected?", nx.is_connected( G.subgra
```

```
Is district = [0, 4, 9, 16, 17, 19, 23, 30, 31, 32, 33, 35, 36, 37, 40, 42,
43, 49, 50, 52, 53, 55, 57, 58, 62, 63, 64, 66, 70, 73, 74, 81, 89, 91, 93]
connected? True
Is district = [3, 10, 12, 18, 24, 25, 26, 28, 38, 39, 44, 46, 47, 48, 54, 6
1, 67, 83, 86, 94, 95, 96, 97, 98] connected? True
Is district = [5, 6, 7, 11, 13, 15, 29, 34, 45, 51, 56, 69, 72, 75, 77, 80,
82, 85, 87, 88, 90] connected? True
Is district = [1, 2, 8, 14, 20, 21, 22, 27, 41, 59, 60, 65, 68, 71, 76, 78,
79, 84, 92] connected? True
```

In [55]:
```
# Let's draw it on a map
import geopandas as gpd

# Read Iowa counties shapefile from "IA_counties.shp"
filepath = '/Users/adenijibabalola/Desktop/DeterministicOperationsResearch/I
filename = 'IA_counties.shp'

# Use os.path.join to correctly construct the file path
full_path = os.path.join(filepath,filename)

# Read geopandas dataframe from file
df = gpd.read_file(full_path)

# Which district is each county assigned to?
assignment = [ -1 for i in G.nodes ]

labeling = { i : j for i in G.nodes for j in range(k) if x[i,j].x > 0.5 }

# Now add the assignments to a column of the dataframe and map it
node_with_this_geoid = { G.nodes[i]['GEOID20'] : i for i in G.nodes }

# pick a position u in the dataframe
for u in range(G.number_of_nodes()):

    geoid = df['GEOID20'][u]

    # what node in G has this geoid?
    i = node_with_this_geoid[geoid]

    # position u in the dataframe should be given
    # the same district # that county i has in 'labeling'
    assignment[u] = labeling[i]

# now add the assignments to a column of our dataframe and then map it
df['assignment'] = assignment

my_fig = df.plot(column='assignment').get_figure()
```




```
In [ ]:
```