General observations

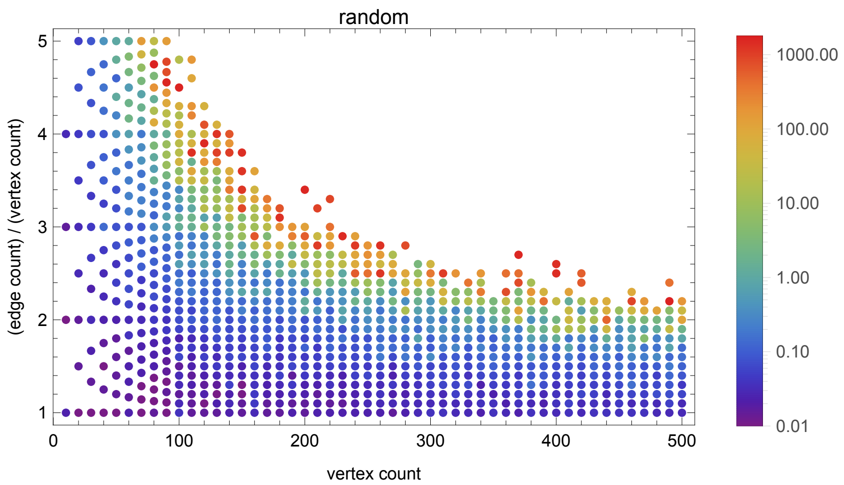
We can use the time taken by SageMath's ILP-based FAS solver (always run on the computer "Particulator") as a proxy for the "hardness" of problems (graphs).

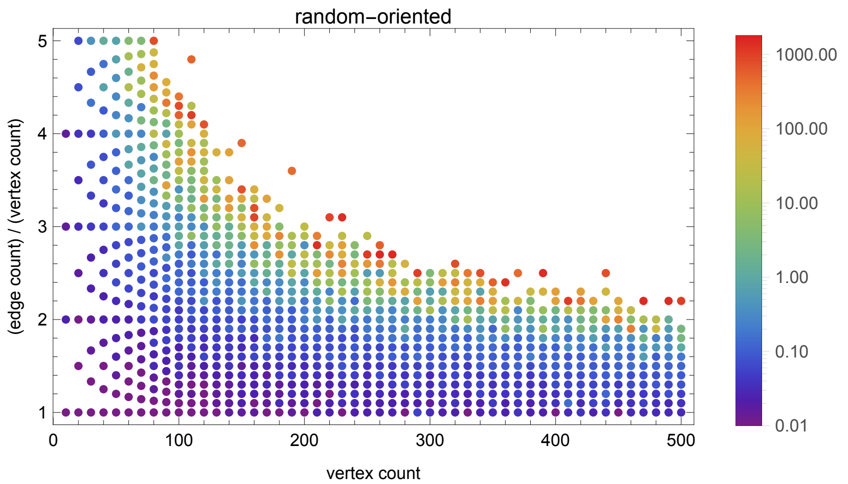
- Of the randomly generated graph sets (i.e. everything except Kautz and de Bruijn), *random* and *random-oriented* are the hardest to compute the exact minimum FAS for.
- Three-dimensional Delaunay graphs are easier. Strangely, Delaunay graph with edges oriented as ←, → or ↔ is easier than those oriented as ← or →, despite having more edges.
- Planar graphs are much easier than the rest . ILP-based solver handles them in sub-exponential (but slower than polynomial) time.

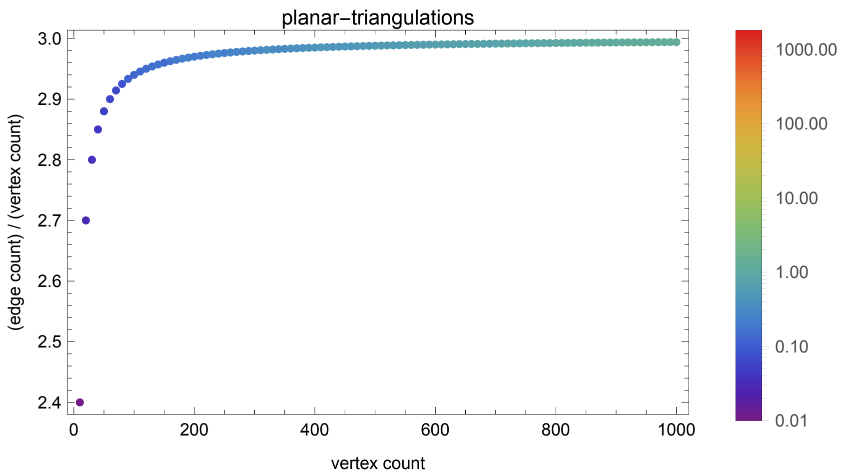
Edge and vertex counts

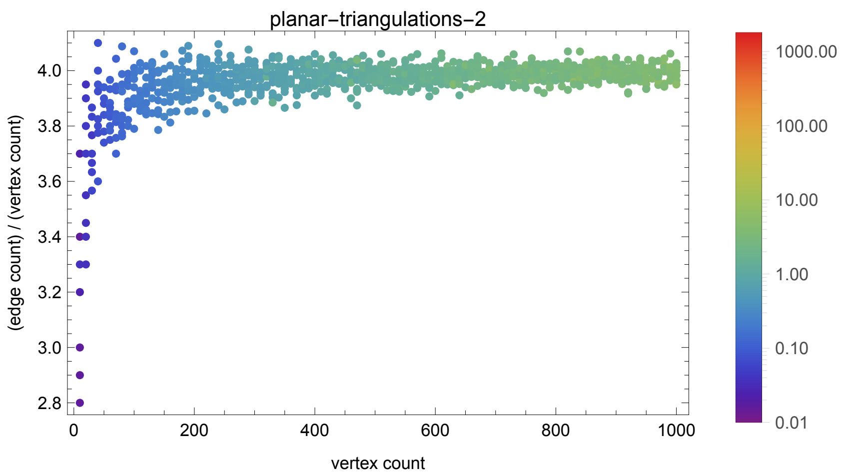
The following plots show what edge and vertex counts the graphs have. Each point represents a graph. The colour represents the time in seconds taken to compute the minimum FAS using SageMath's ILP-based solver. For random & random-oriented, the time limit was 3 minutes. For the geometric graphs it was 30 minutes. It is this time limit that defines the region where we have data points (for higher edge counts it is prohibitively expensive to compute the MFAS).

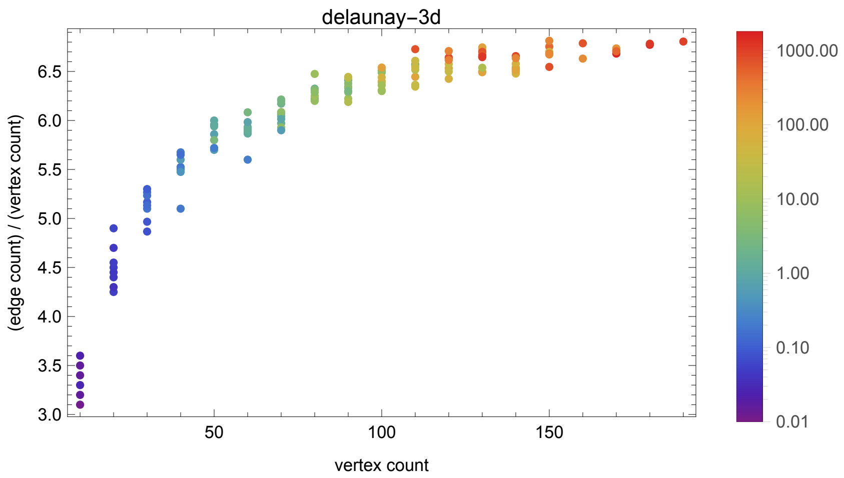
- Colour shows: time taken by SageMath's ILP-based solver (in seconds).
- Colour bar is the same for all plots.

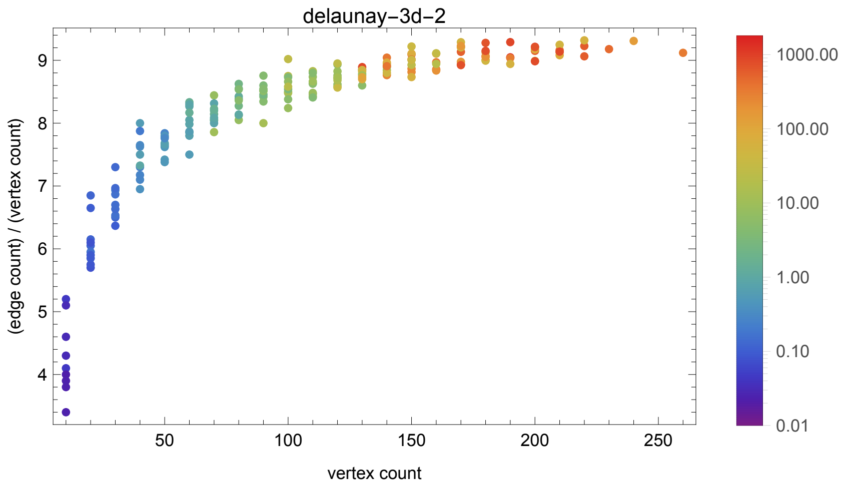


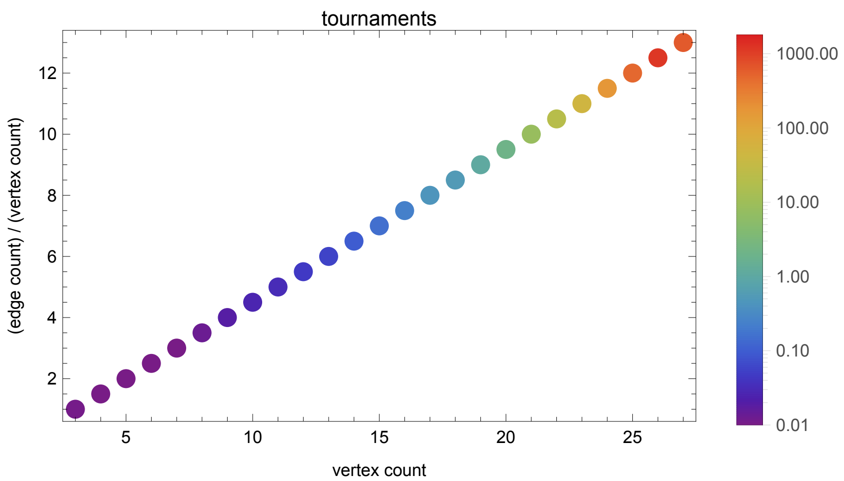

random–oriented
(edge count) / (vertex count)
1
2
3
4
5
0
100
200
300
400
500
vertex count
1000.00
100.00
10.00
1.00
0.10
0.01

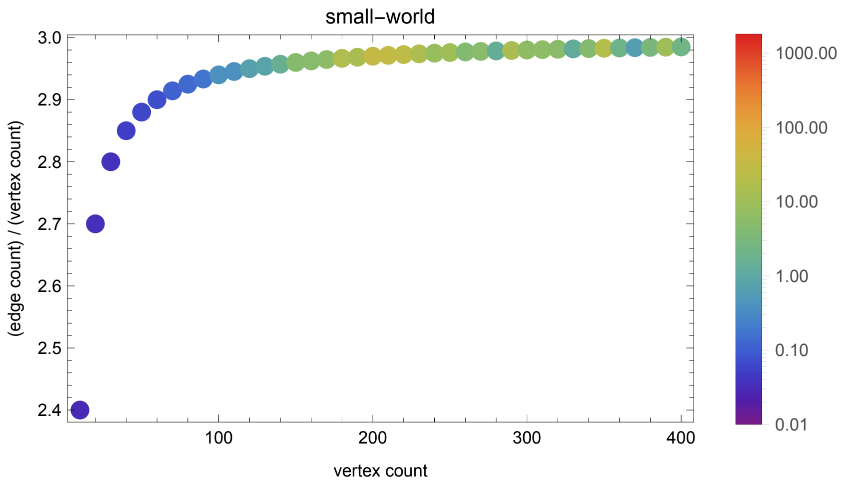

planar–triangulations
(edge count) / (vertex count)
2.4
2.5
2.6
2.7
2.8
2.9
3.0
0
200
400
600
800
1000
vertex count
1000.00
100.00
10.00
1.00
0.10
0.01

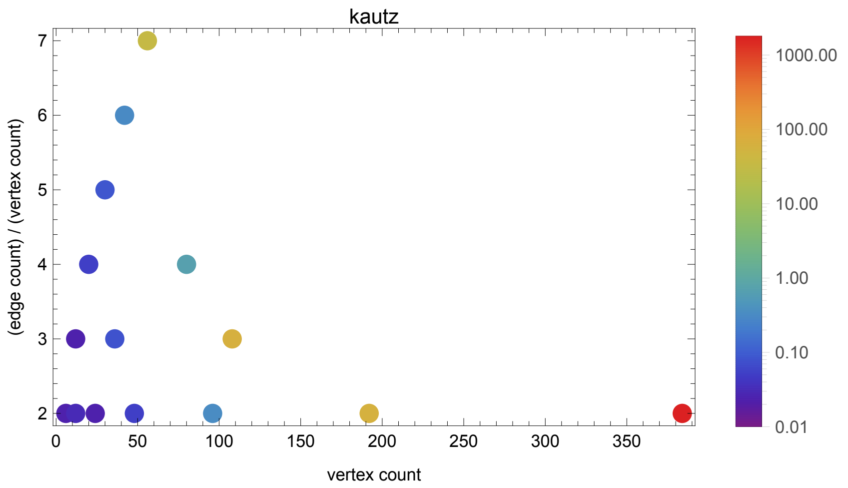

planar–triangulations–2
(edge count) / (vertex count)
2.8
3.0
3.2
3.4
3.6
3.8
4.0
0
200
400
600
800
1000
vertex count
1000.00
100.00
10.00
1.00
0.10
0.01

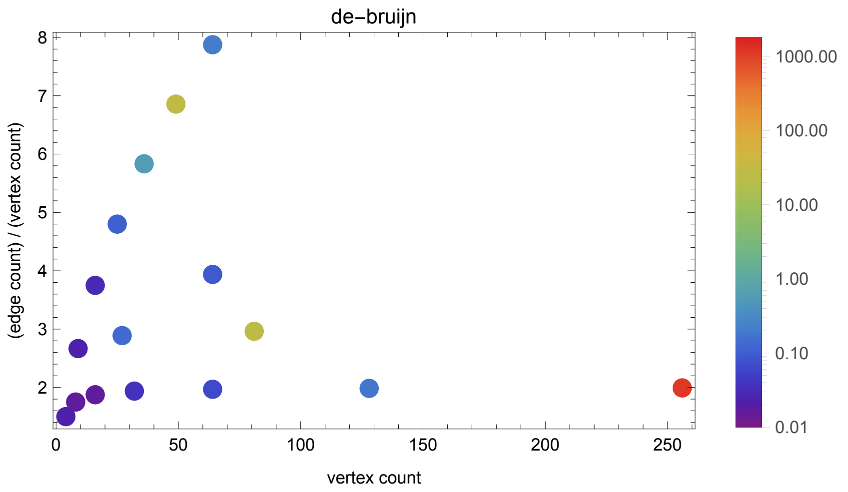

delaunay–3d
(edge count) / (vertex count)
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0
50
100
150
vertex count
1000.00
100.00
10.00
1.00
0.10
0.01


delaunay–3d–2
(edge count) / (vertex count)
9
8
7
6
5
4
50
100
150
200
250
vertex count
1000.00
100.00
10.00
1.00
0.10
0.01


tournaments
(edge count) / (vertex count)
12
10
8
6
4
2
5
10
15
20
25
vertex count
1000.00
100.00
10.00
1.00
0.10
0.01

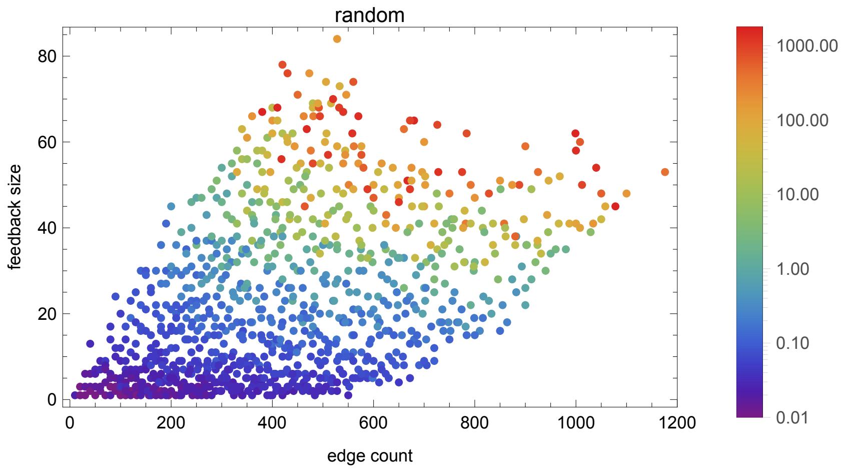

small–world
(edge count) / (vertex count)
vertex count
3.0
2.9
2.8
2.7
2.6
2.5
2.4
100
200
300
400
1000.00
100.00
10.00
1.00
0.10
0.01

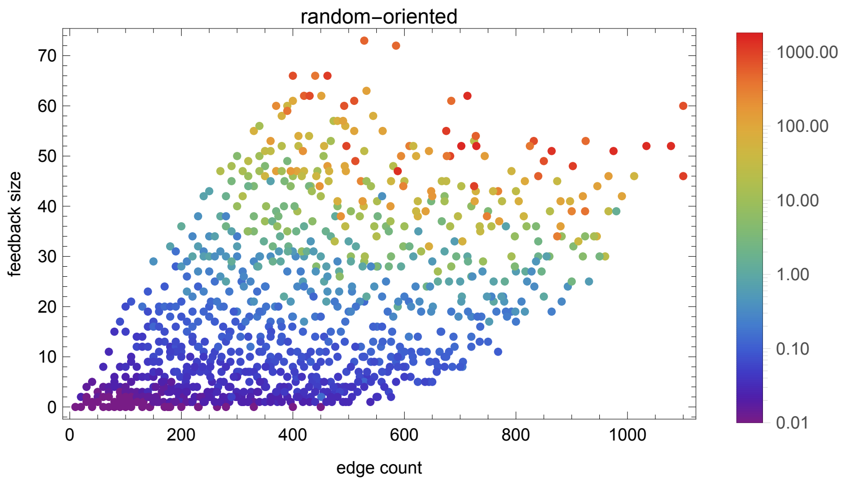

kautz
(edge count) / (vertex count)
vertex count
7
6
5
4
3
2
0
50
100
150
200
250
300
350
1000.00
100.00
10.00
1.00
0.10
0.01

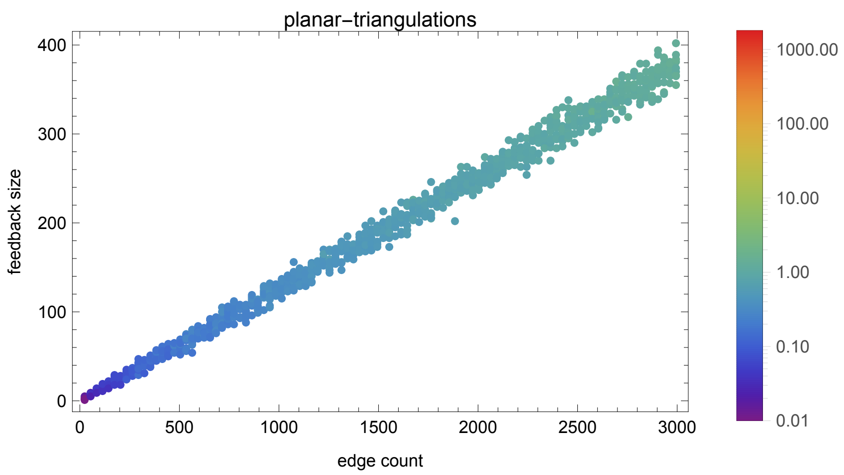

de–bruijn
(edge count) / (vertex count)
vertex count
8
7
6
5
4
3
2
0
50
100
150
200
250
1000.00
100.00
10.00
1.00
0.10
0.01

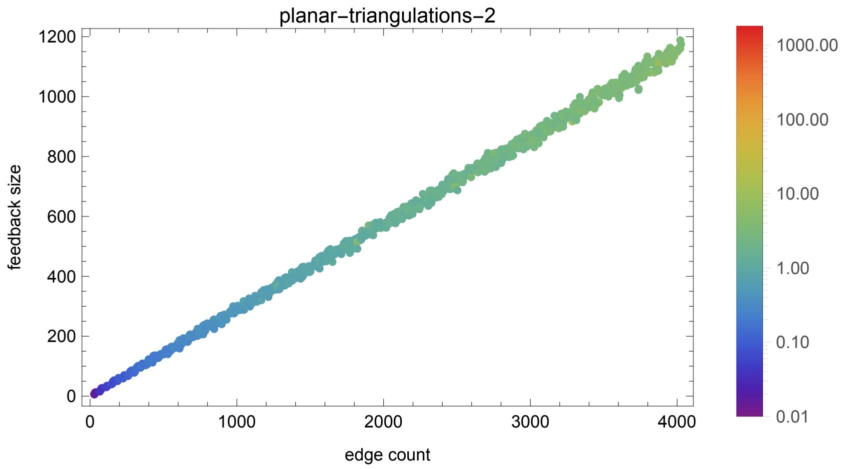
Feedback sizes

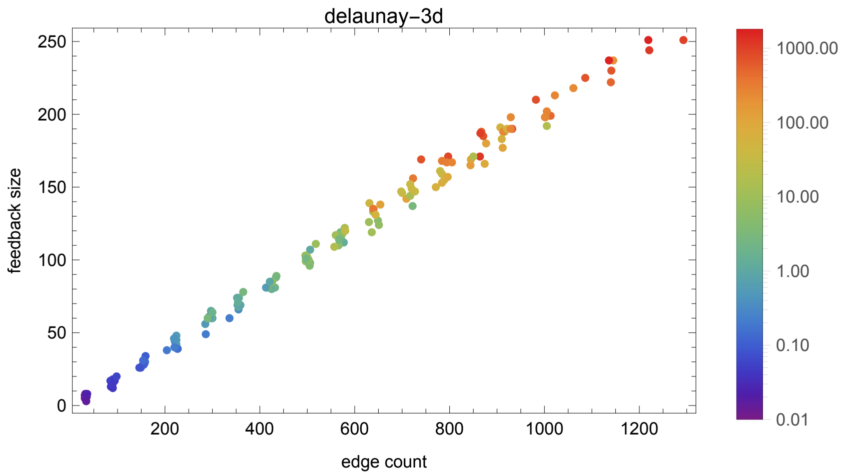
The following plots are similar to the ones above, but timings are shown against edge-counts and feedback sizes.

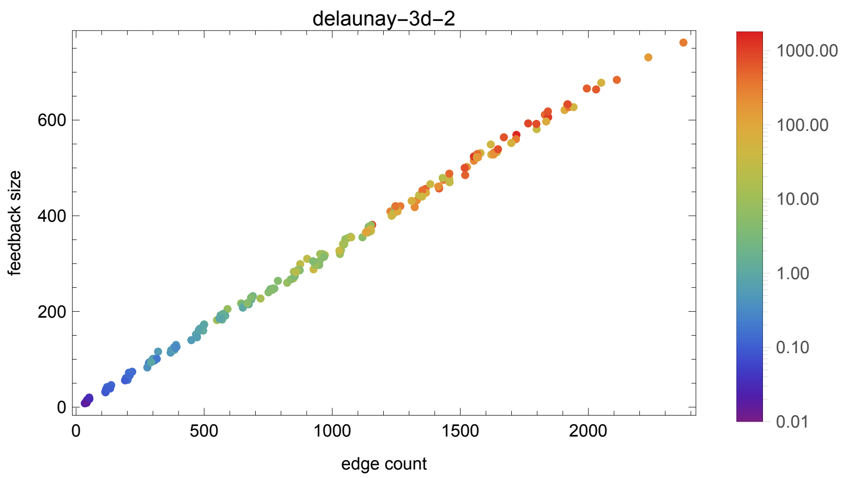


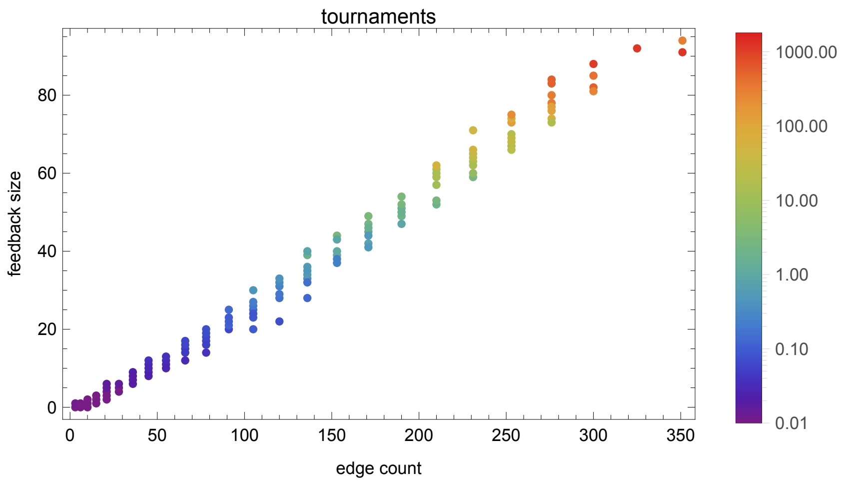


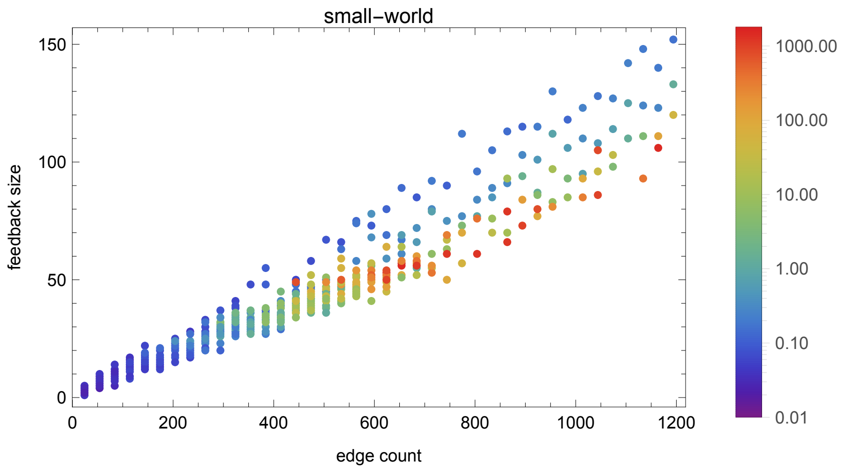


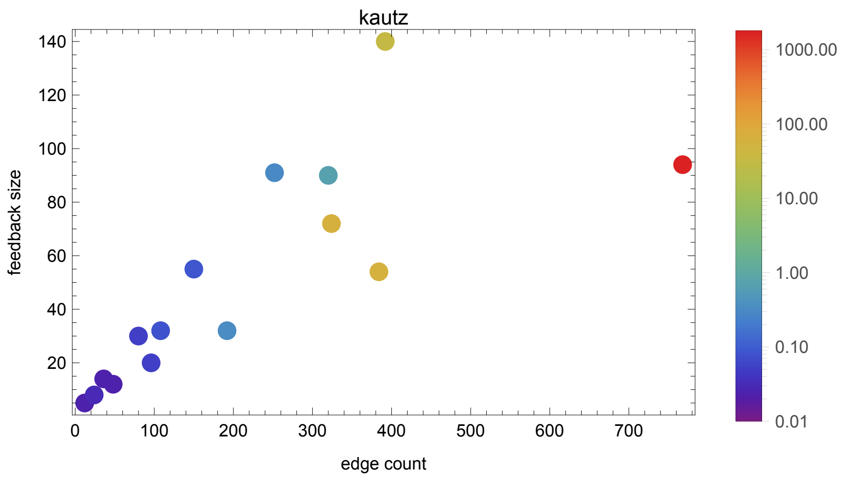

planar–triangulations–2
feedback size
edge count
0
200
400
600
800
1000
1200
0
1000
2000
3000
4000
1000.00
100.00
10.00
1.00
0.10
0.01

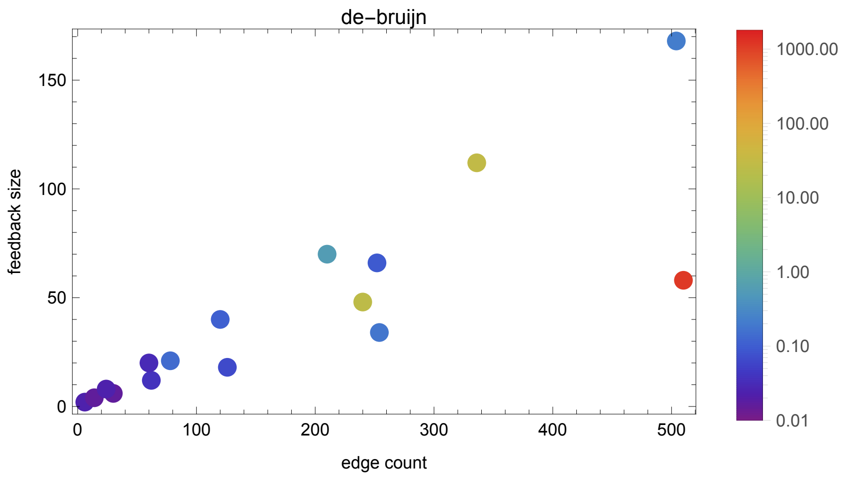

delaunay–3d
feedback size
edge count
0
50
100
150
200
250
200
400
600
800
1000
1200
1000.00
100.00
10.00
1.00
0.10
0.01


delaunay–3d–2
feedback size
edge count
0
200
400
600
0
500
1000
1500
2000
1000.00
100.00
10.00
1.00
0.10
0.01


tournaments
feedback size
edge count
0
20
40
60
80
0
50
100
150
200
250
300
350
1000.00
100.00
10.00
1.00
0.10
0.01


small–world
feedback size
edge count
0
50
100
150
0
200
400
600
800
1000
1200
1000.00
100.00
10.00
1.00
0.10
0.01


kautz
feedback size
edge count
20
40
60
80
100
120
140
0
100
200
300
400
500
600
700
1000.00
100.00
10.00
1.00
0.10
0.01

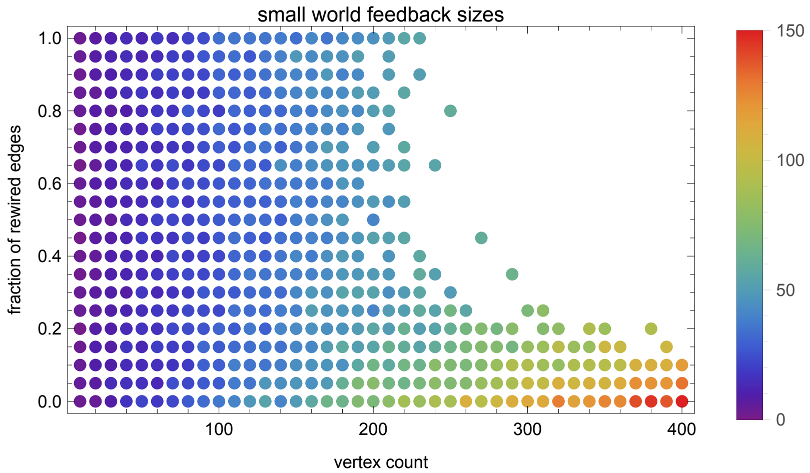


"Small-world" graphs

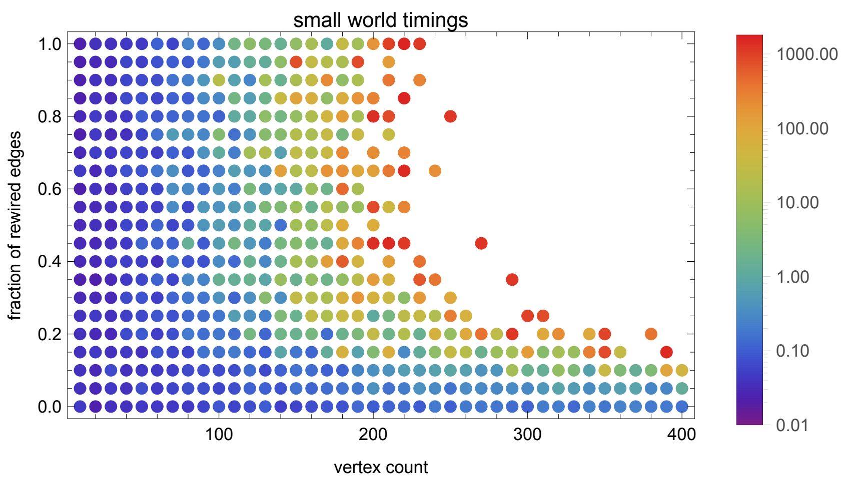
"Small-world" graphs are created by taking a planar triangulation and randomly rewiring a fraction p of its edges. Here we avoid reciprocal edges entirely.

The edge count is always $E = 3\,V - 6$, as for triangulations.

Feedback sizes of small world graph as a function of vertex count and fraction of rewired edges:



Timings of the ILP-based FAS solver as a function of vertex count and fraction of rewired edges:

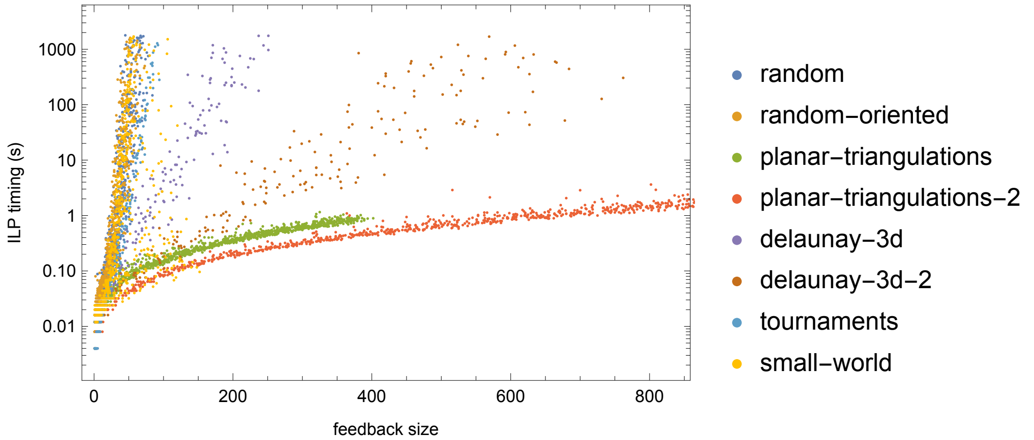


Minimum FAS size vs timing

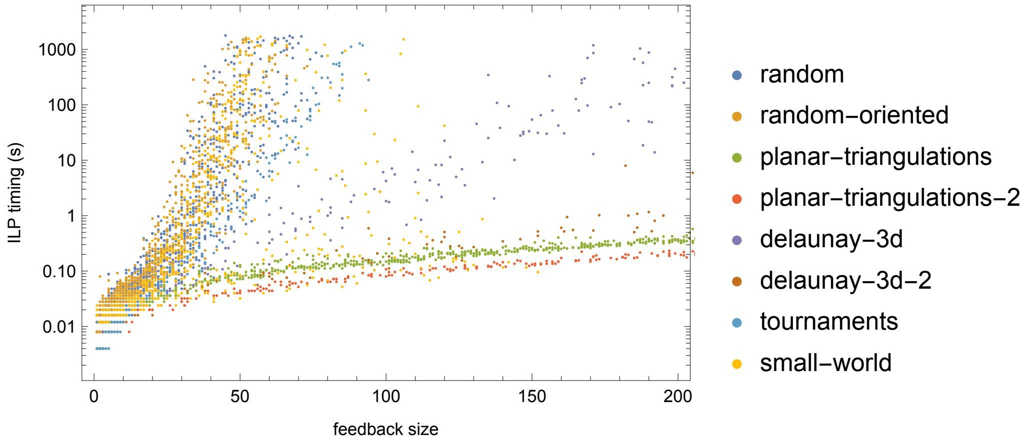
The time taken to compute a minimum FAS with SageMath's ILP-based solver vs. its size.

- Timing vs FAS size is exponential for *random* and *random-oriented*
- Timing is sub-exponental but slower than polynomial for the planar graphs.

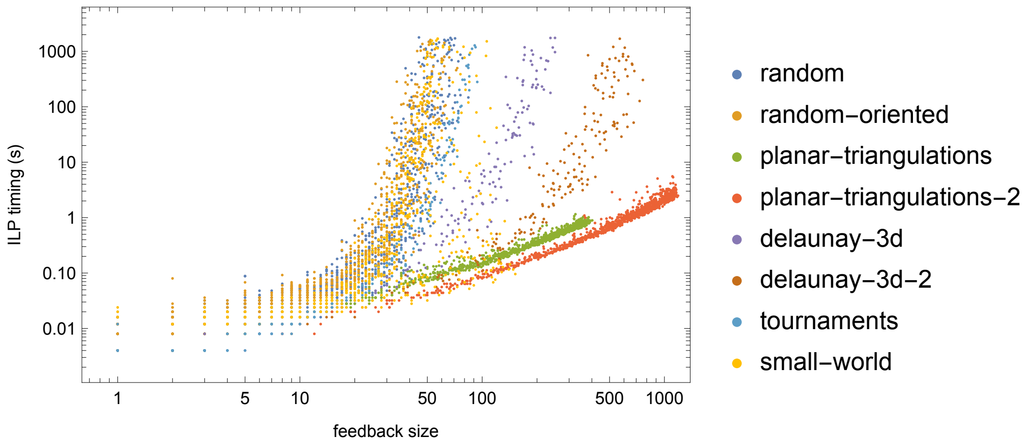
Log-linear plot:



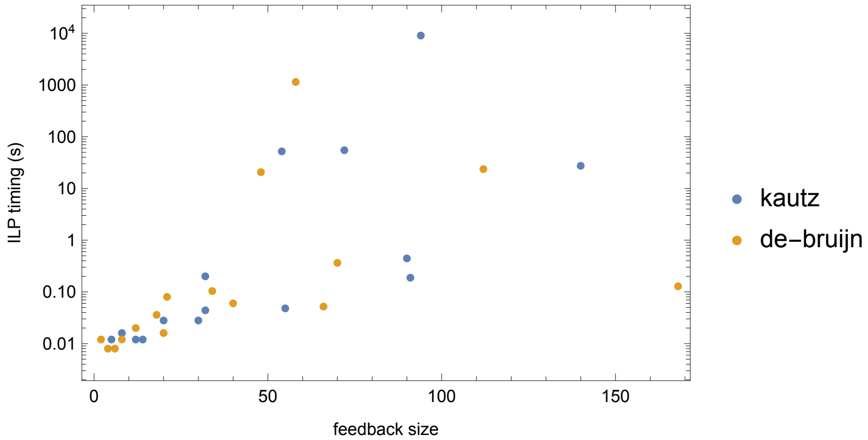
Log-linear plot truncated to a feedback size of 200:



Log-log plot:



Log-linear plot for Kautz and de Bruijn graphs:

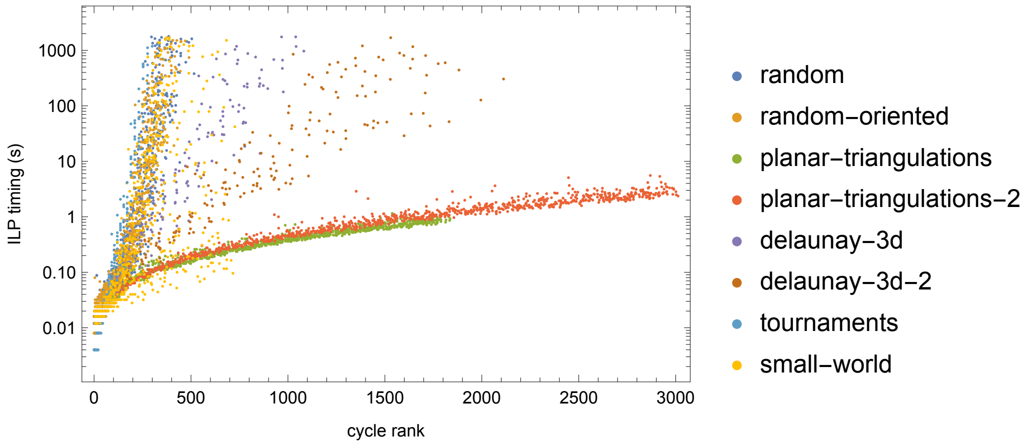


Cycle rank vs timing

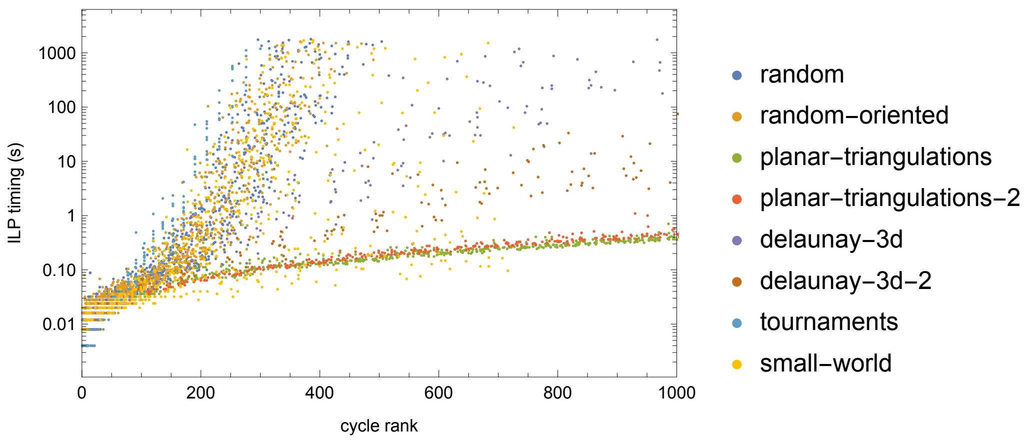
The time taken to compute a minimum FAS with SageMath's ILP-based solver vs. the graph's cycle rank.

Cycle rank: the dimension of the cycle space of the strongly connected part of the graph.

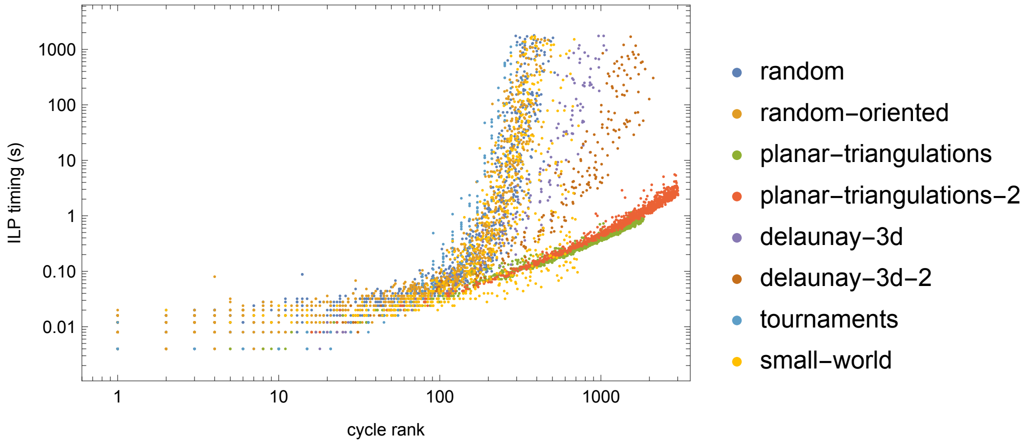
Log-linear plot:



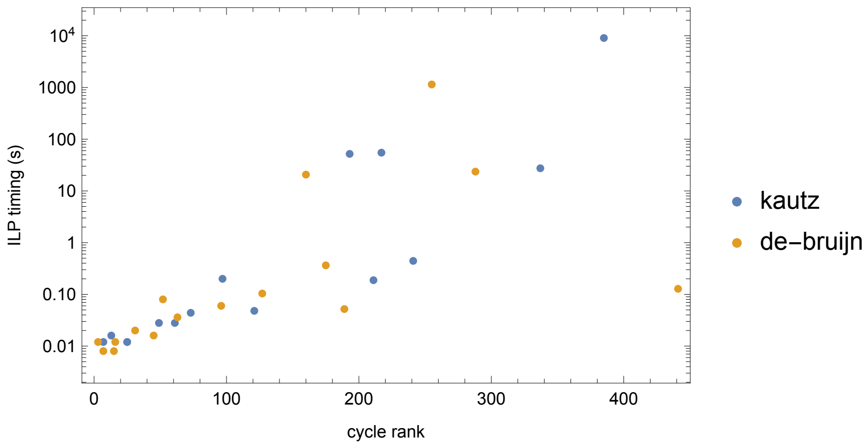
Log-linear plot truncated to a cycle rank of 1000:



Log-log plot:



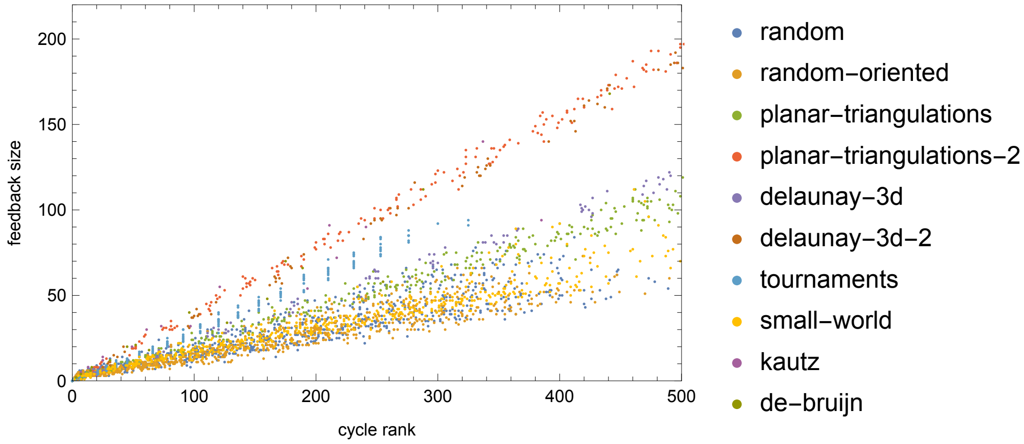
Log-linear plot for Kautz and de Bruijn graphs:



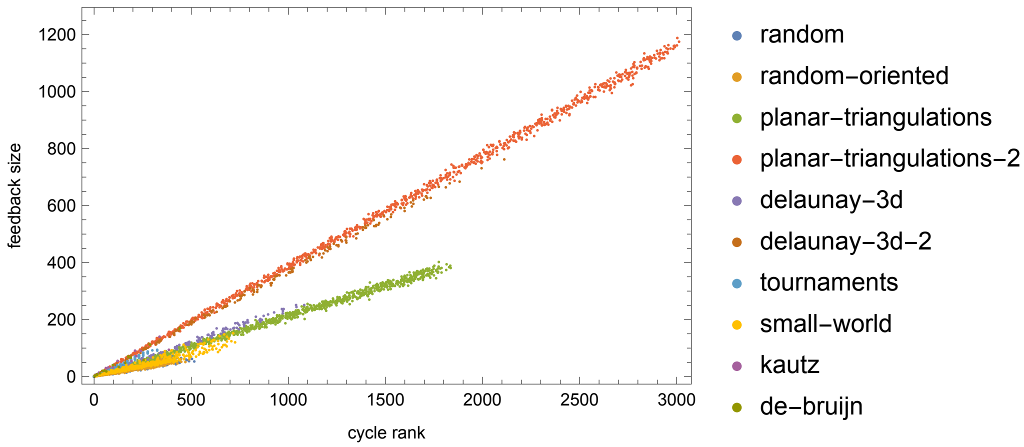
Cycle rank vs feedback size

The following plots show the size of the the directed cycle basis vs. the minimum feedback arc set size.

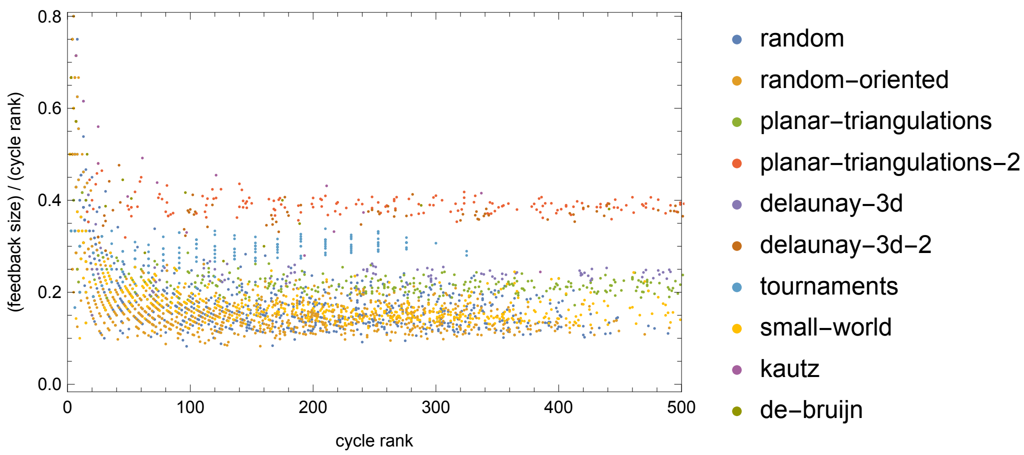
Feedback size vs cycle rank, truncated to max cycle rank of 500.



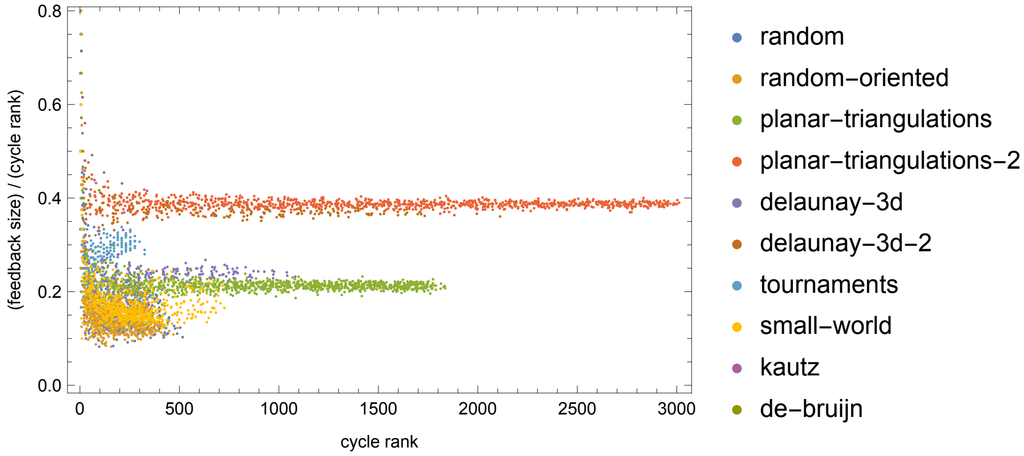
Same as above, without truncation.



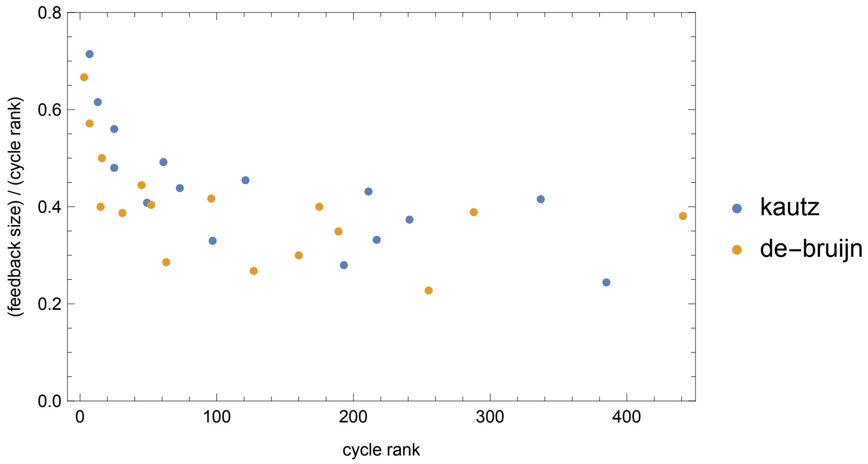
The ratio $\frac{\text{feedback size}}{\text{cycle rank}}$ vs. cycle rank, truncated to max cycle rank of 500.



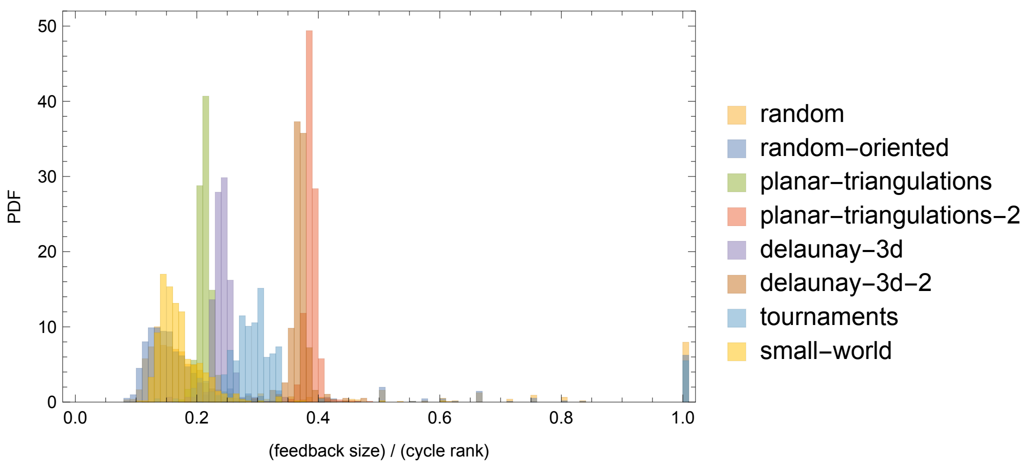
Same as above, without truncation.



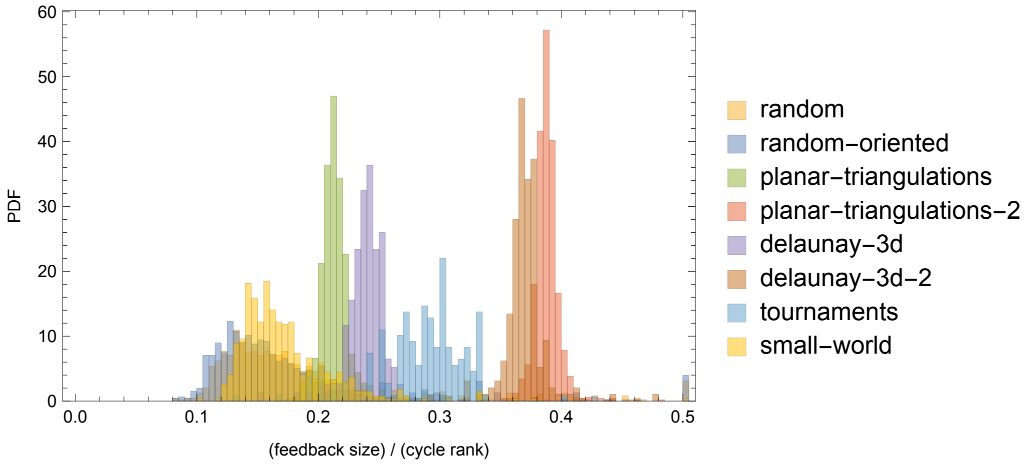
Same as above, showing only Kautz and de Bruijn graphs.



Histogram (probability density) of $\frac{\text{feedback size}}{\text{cycle rank}}$ values for all types of graphs, except Kautz and de Bruijn.



Same as above, truncated to [0, 0.5] and with half the bin width.



Click here to see an interactive version of this histogram.

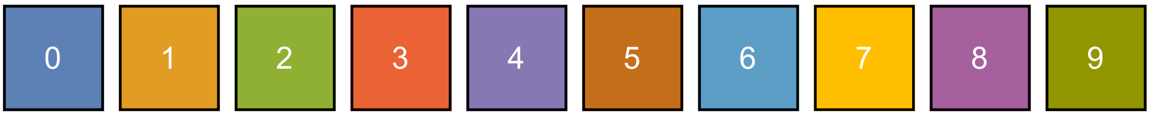
Edge connectivity

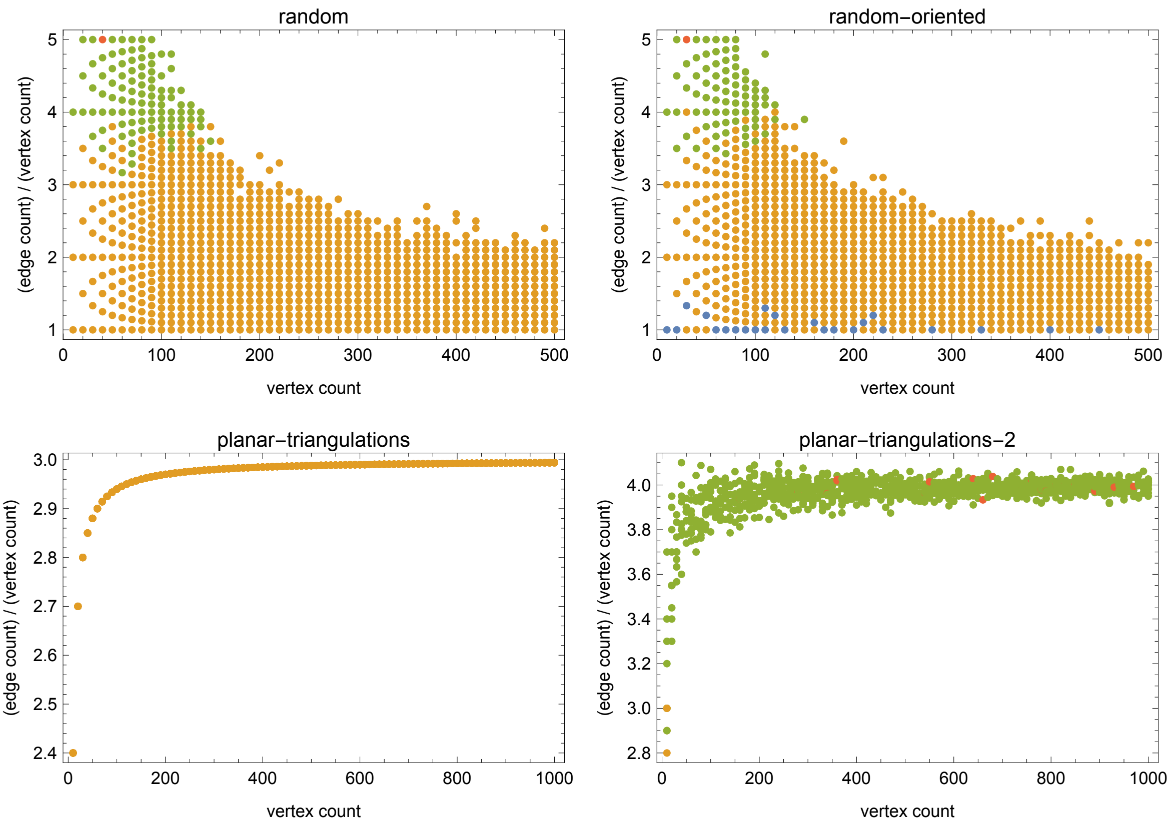
Definiton: The *edge connectivity* of a graph is the smallest number of edges that must be removed to disconnect it.

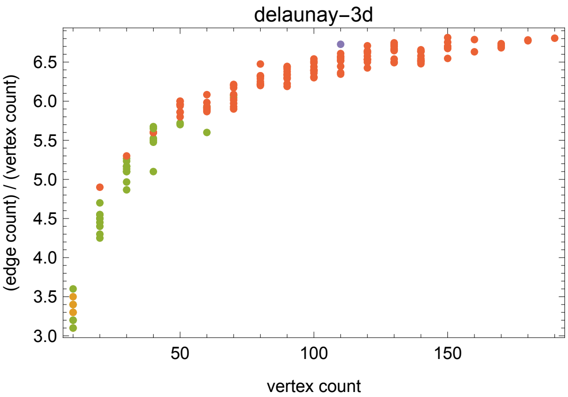
The super-algorithm cannot work on a graph that has edge connectivity larger than 1. Therefore, if there is a subgraph of higher-than-one edge connectivity, the super-algorithm will not be able to make the graph acyclic. *Typically*, it will not be able to remove any edges at all.

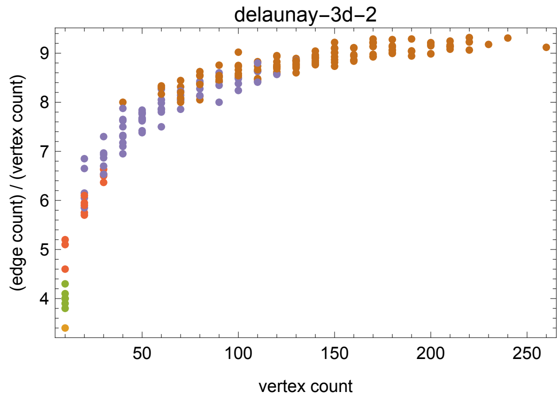
- The colour shows the maximal edge connectivity over all subgraphs (i.e. not the edge connectivity of the entire graph).
- *Typically*, the subgraph with the maximal edge connectivity contains most vertices (*in this particular dataset*).

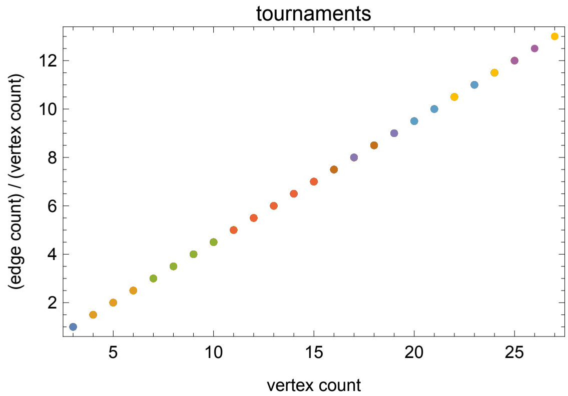
Legend:

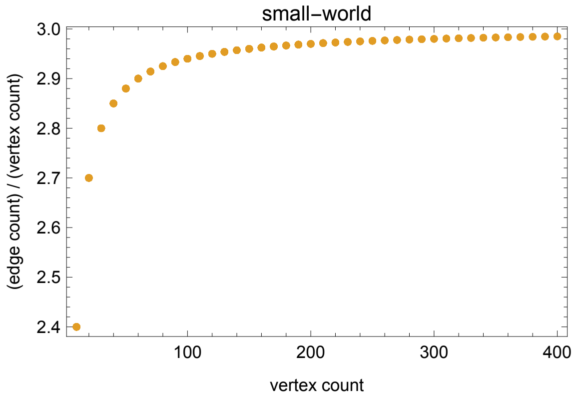


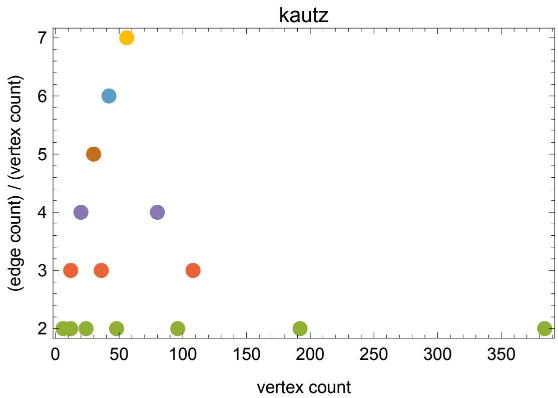


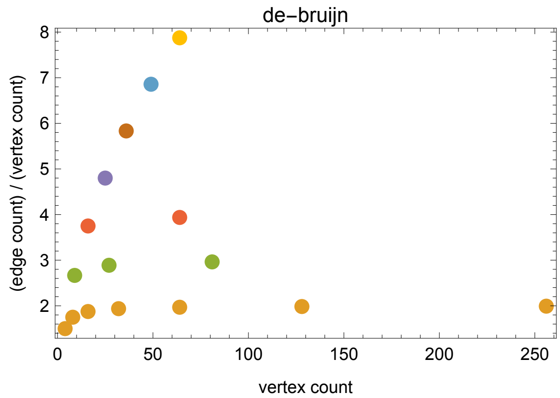




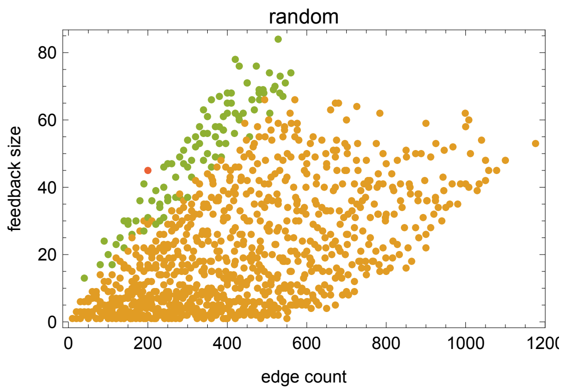


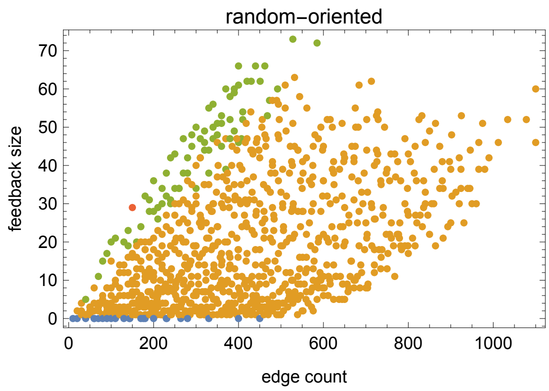


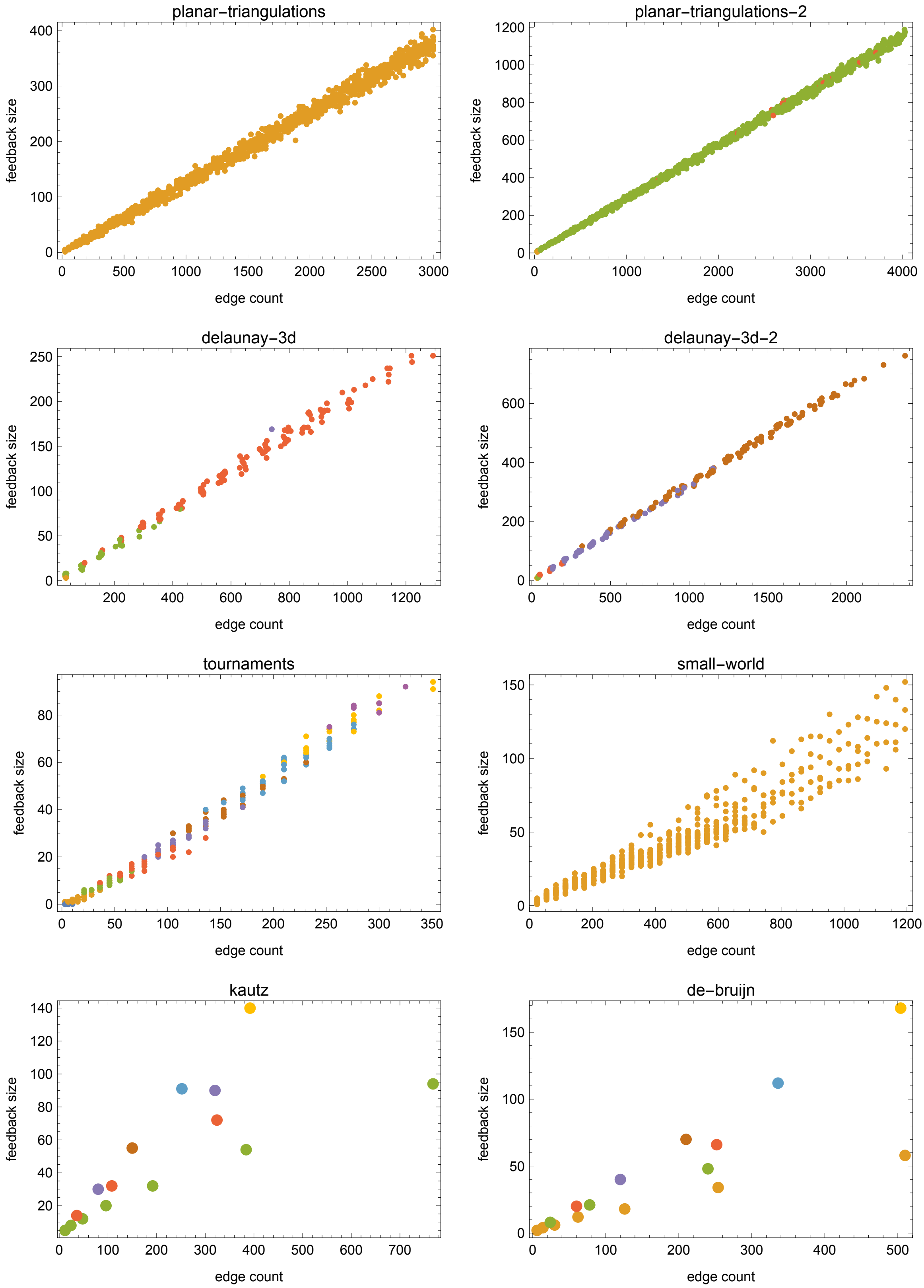




The same as above in (edge count, feedback size) space.





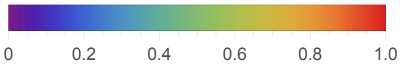


Super-algorithm

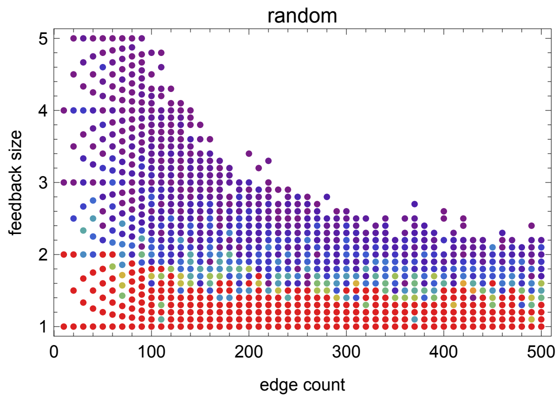
- Colour shows: (number of arcs the super-algorihm removed) / (minimum FAS size)

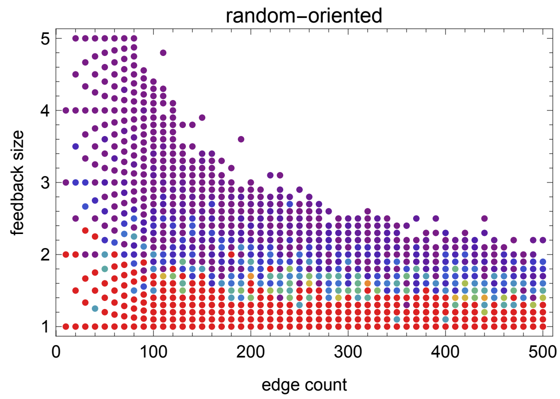
- This number is between 0 and 1.

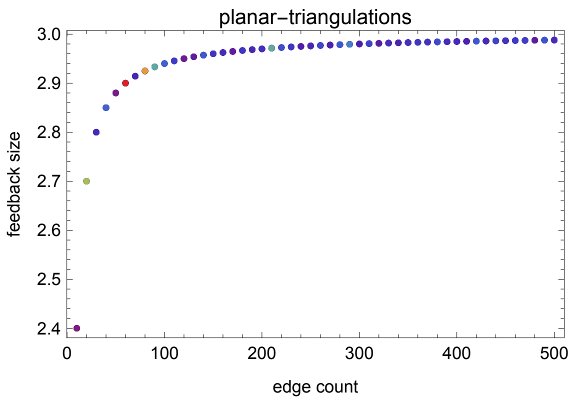
Legend (same for all plots):

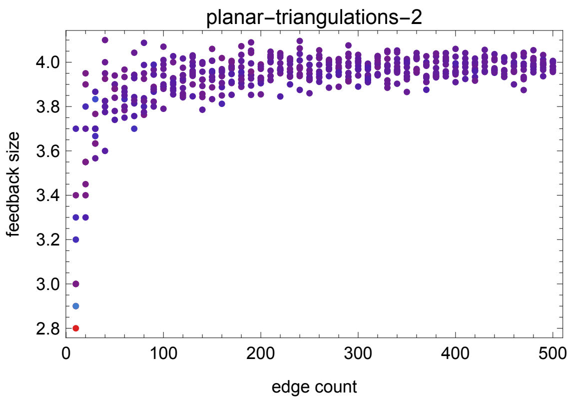


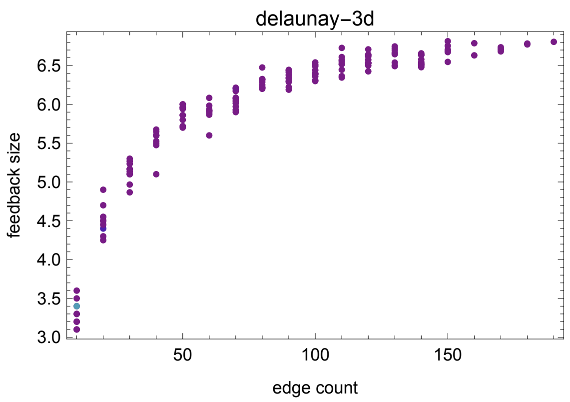
- = super-algorithm made the graph acyclic
- = super-algorithm could not remove any arcs
- For random graphs, there is a sharp boundary between the two just below $\frac{\text{edge count}}{\text{vertex count}} \approx 2$.
- All other types of graphs have $\frac{\text{edge count}}{\text{vertex count}} > 2$ and the super-algorithm cannot handle them.

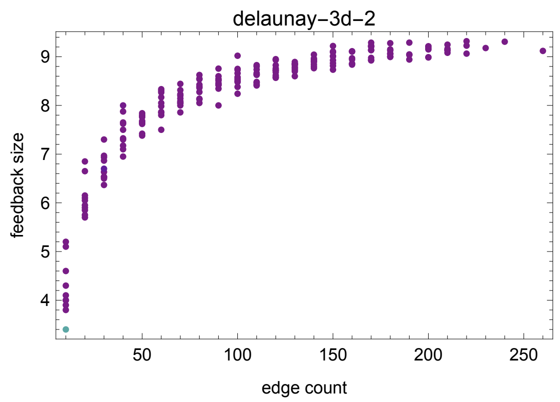


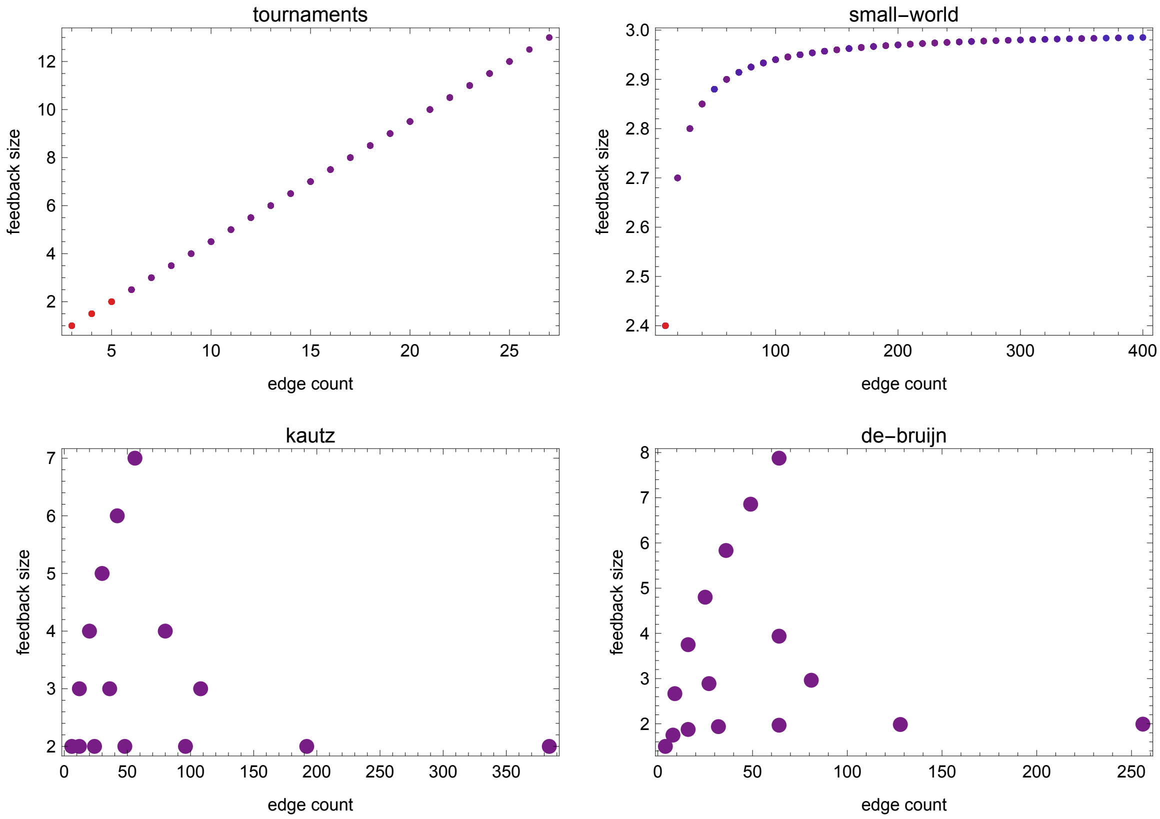




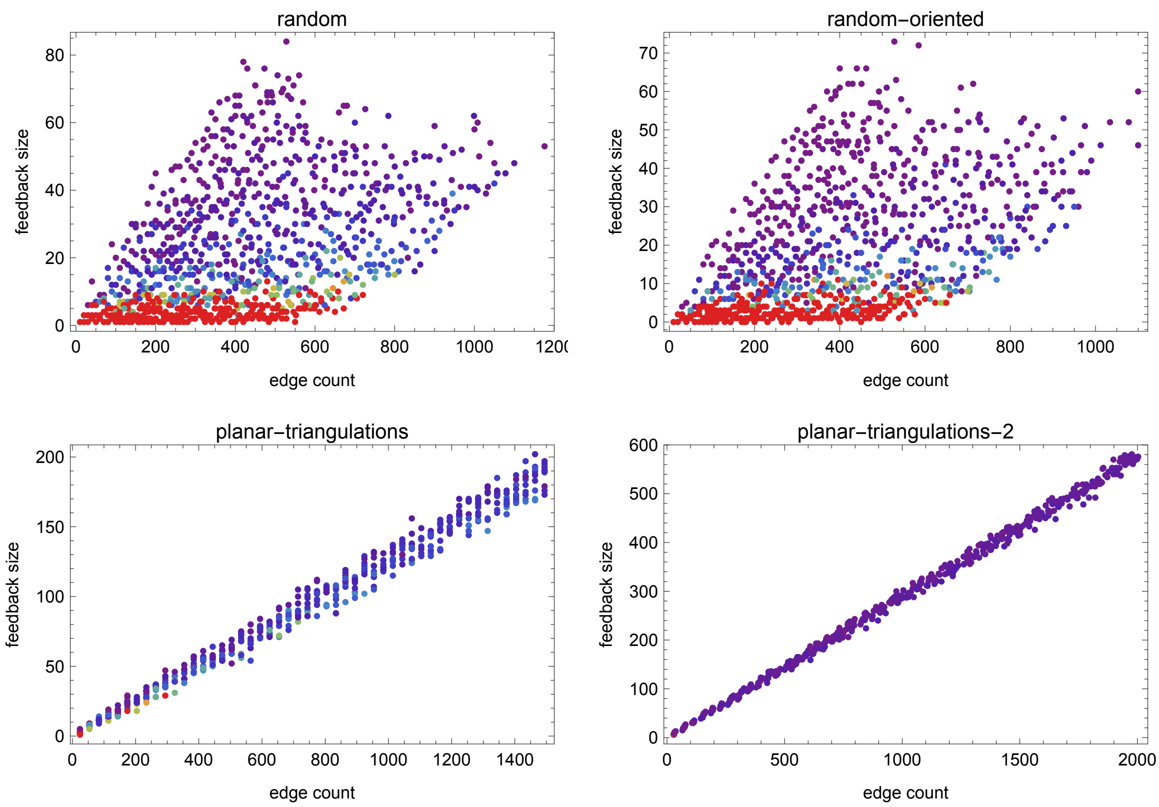


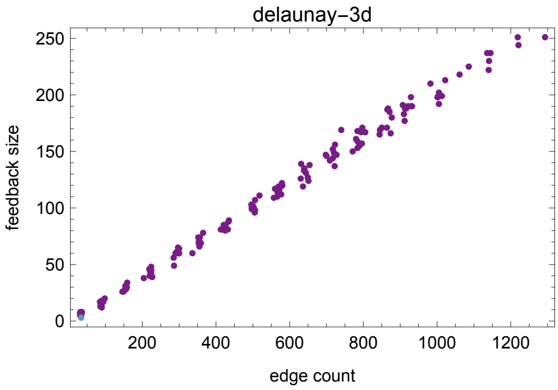


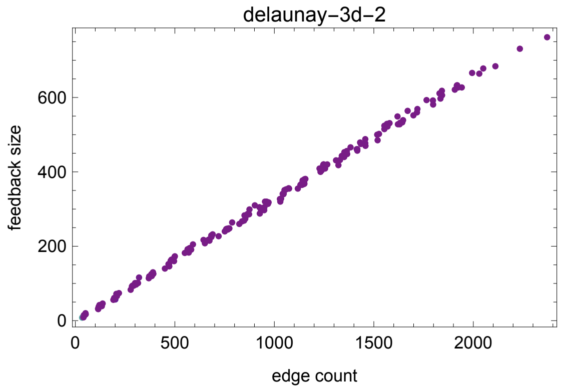


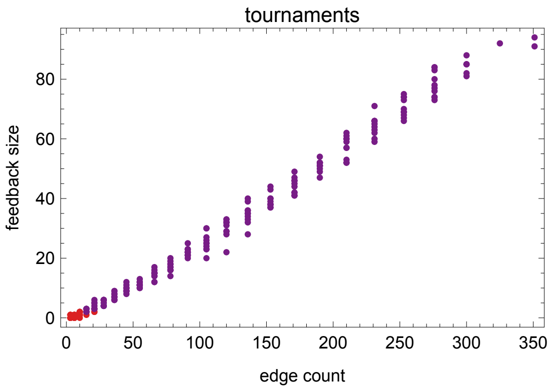


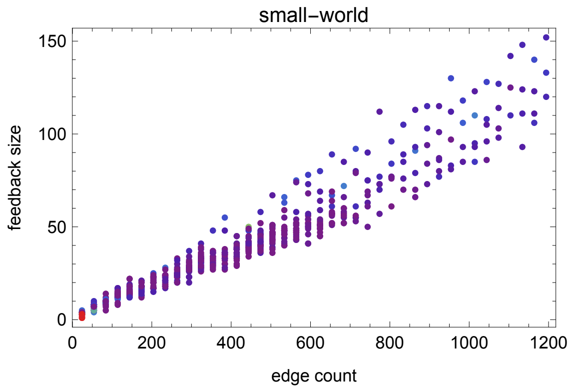
The same as above in (edge count, feedback size) space.

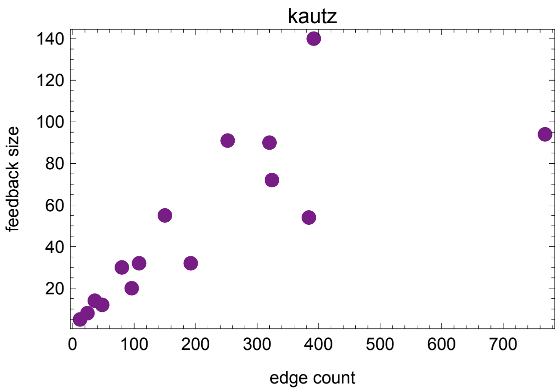


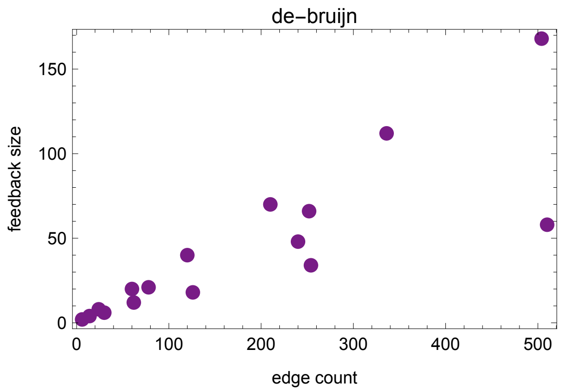

delaunay-3d
feedback size
edge count
0 50 100 150 200 250
200 400 600 800 1000 1200


delaunay-3d-2
feedback size
edge count
0 200 400 600
0 500 1000 1500 2000


tournaments
feedback size
edge count
0 20 40 60 80
0 50 100 150 200 250 300 350


small-world
feedback size
edge count
0 50 100 150
0 200 400 600 800 1000 1200


kautz
feedback size
edge count
20 40 60 80 100 120 140
0 100 200 300 400 500 600 700


de-bruijn
feedback size
edge count
0 50 100 150
0 100 200 300 400 500