# Metaheuristics

## Community Detection

Jorik De Waen
Jeroen Craps

# The problem

## Community

Finding a concrete definition of what a community is.

As well as quantifying how well a community structure is.

## Graph

Dealing with the information that is available in the graph.

Understanding what is represented.

## NP-Hard

Scaling to very large graphs.

Defining what a very large graph is and how we can tackle the problem scalably.

# Solution

## Reducing the search space

By reducing the amount of nodes and edges in the graph, we think that we can handle graphs, which were initially larger, in a comparable amount of time.

# Challenges

Challenge 1

**Wisely**

By reducing the graph we will remove certain options. To goal is to make smart decisions on which options are being pruned.

Challenge 2

**Lossless or lossy**

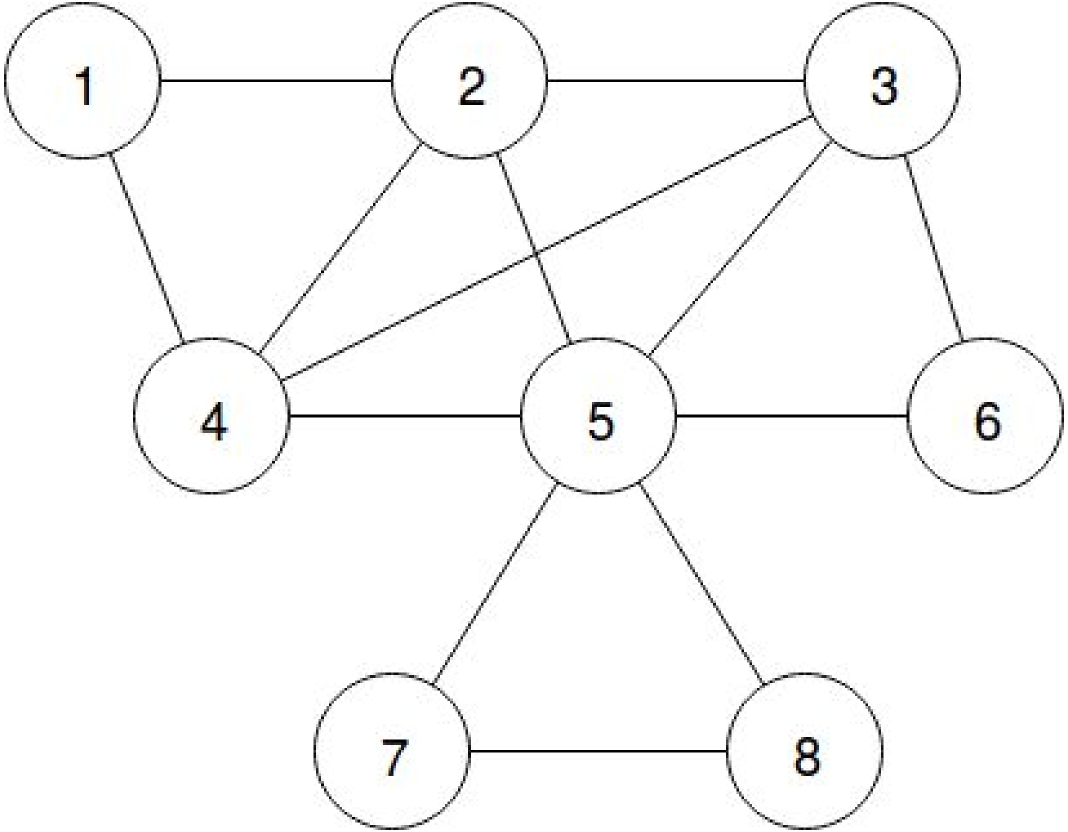
Preventing information loss. So that the important remains in the graph to do careful evaluation.

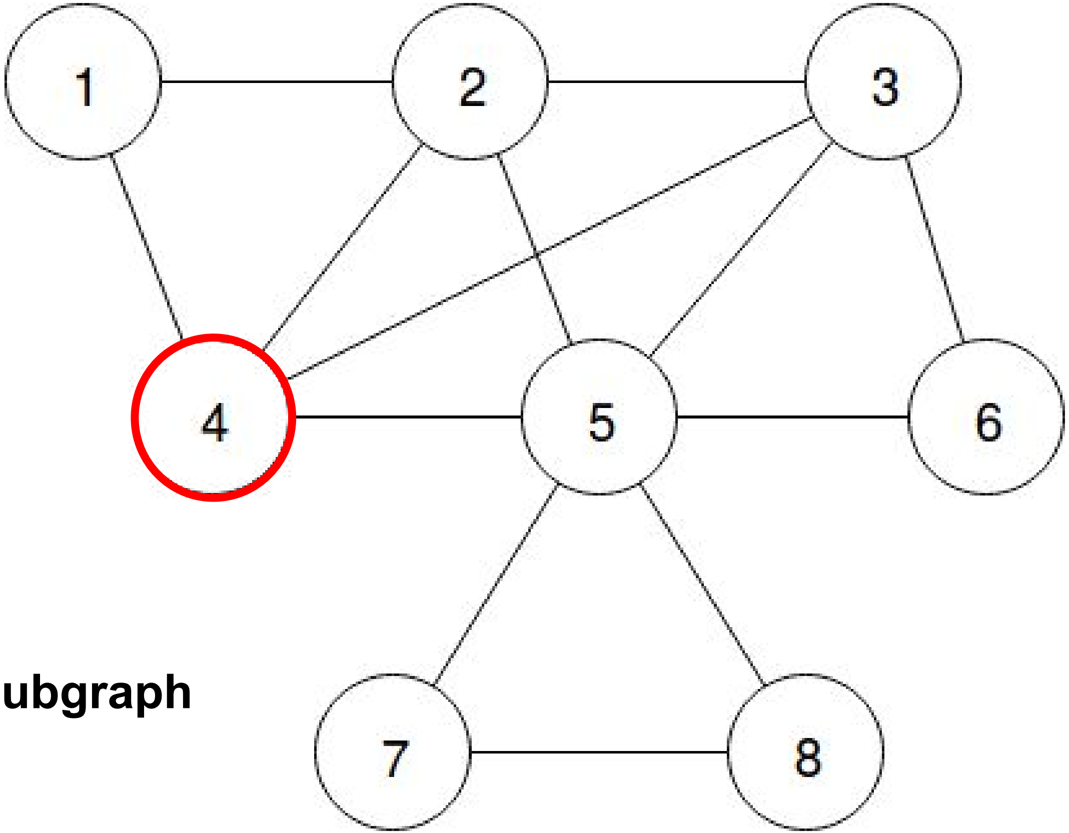
Challenge 3

**Efficiently**

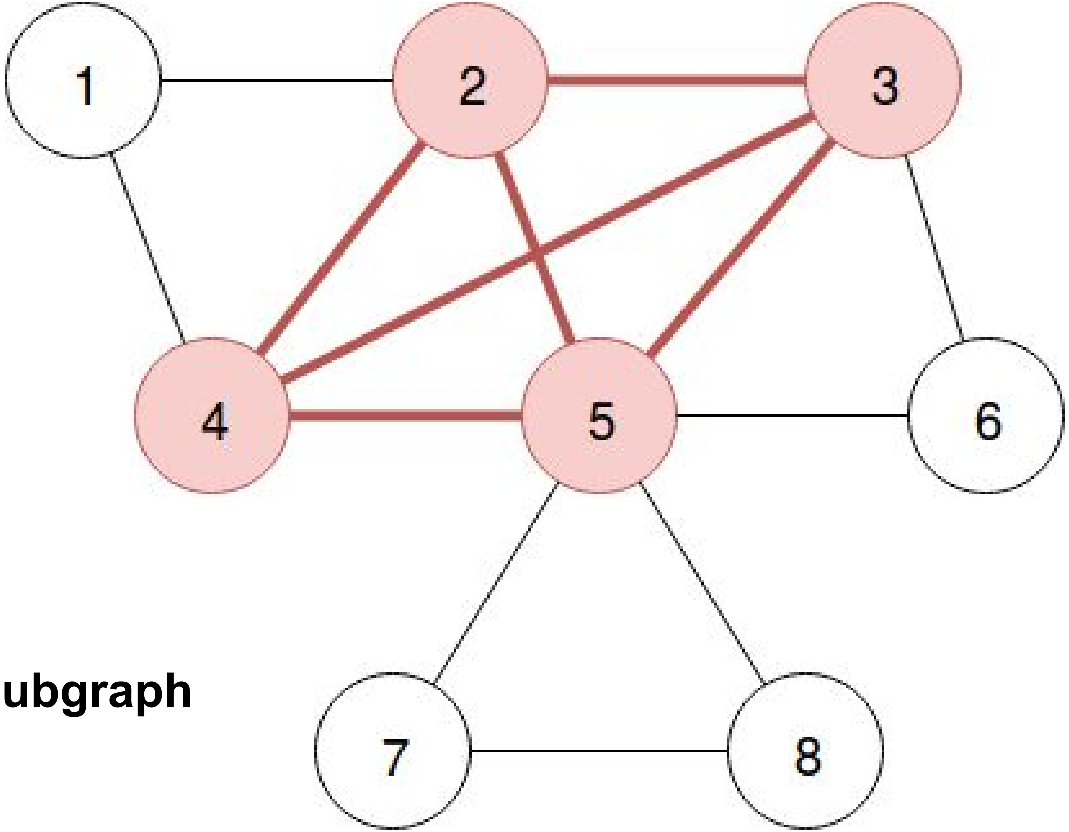
Low computational cost while achieving good results.

# Preprocessing

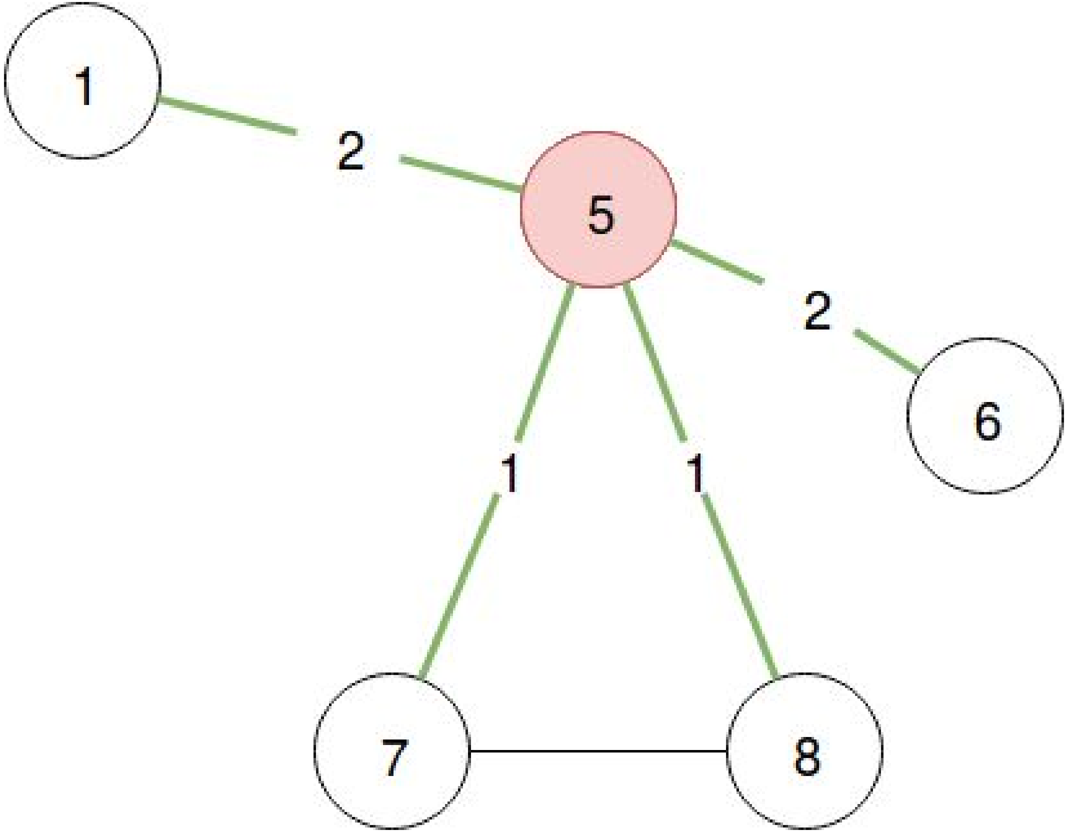

1
2
3
4
5
6
7
8

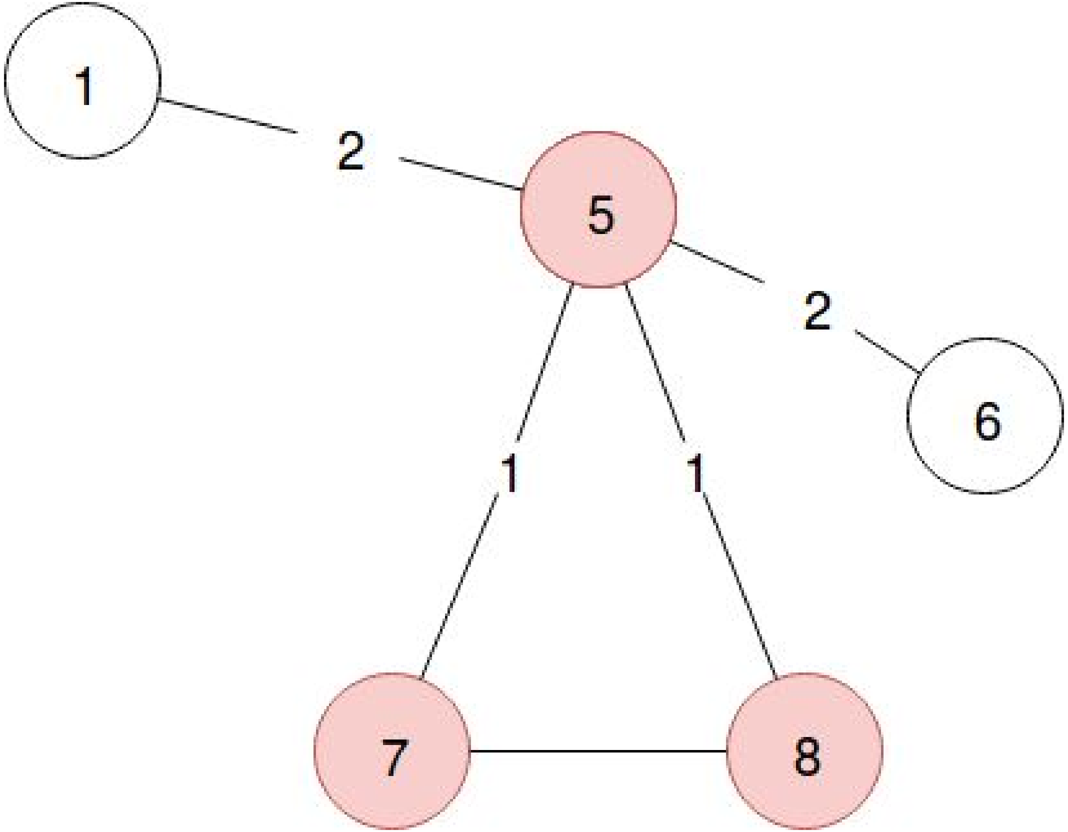


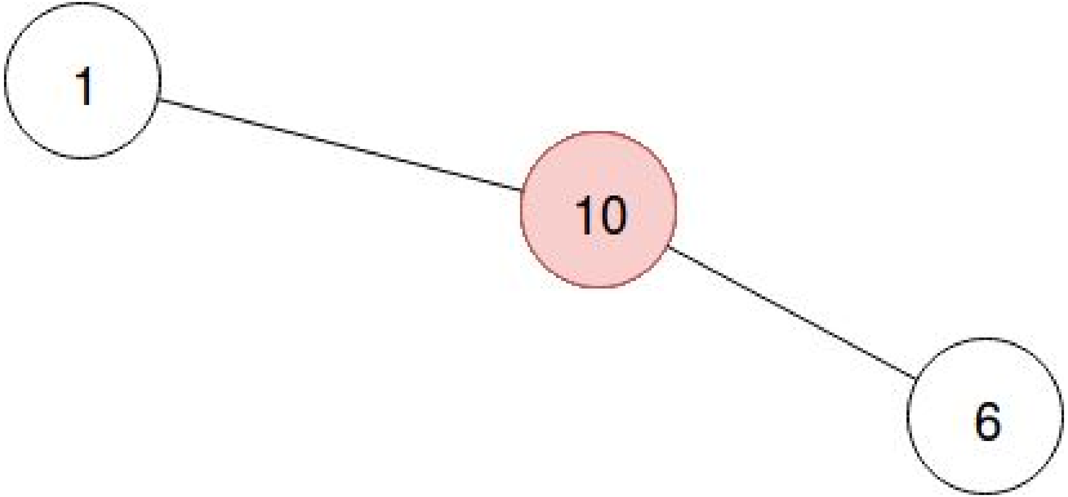
**Clique:**
**A complete subgraph**

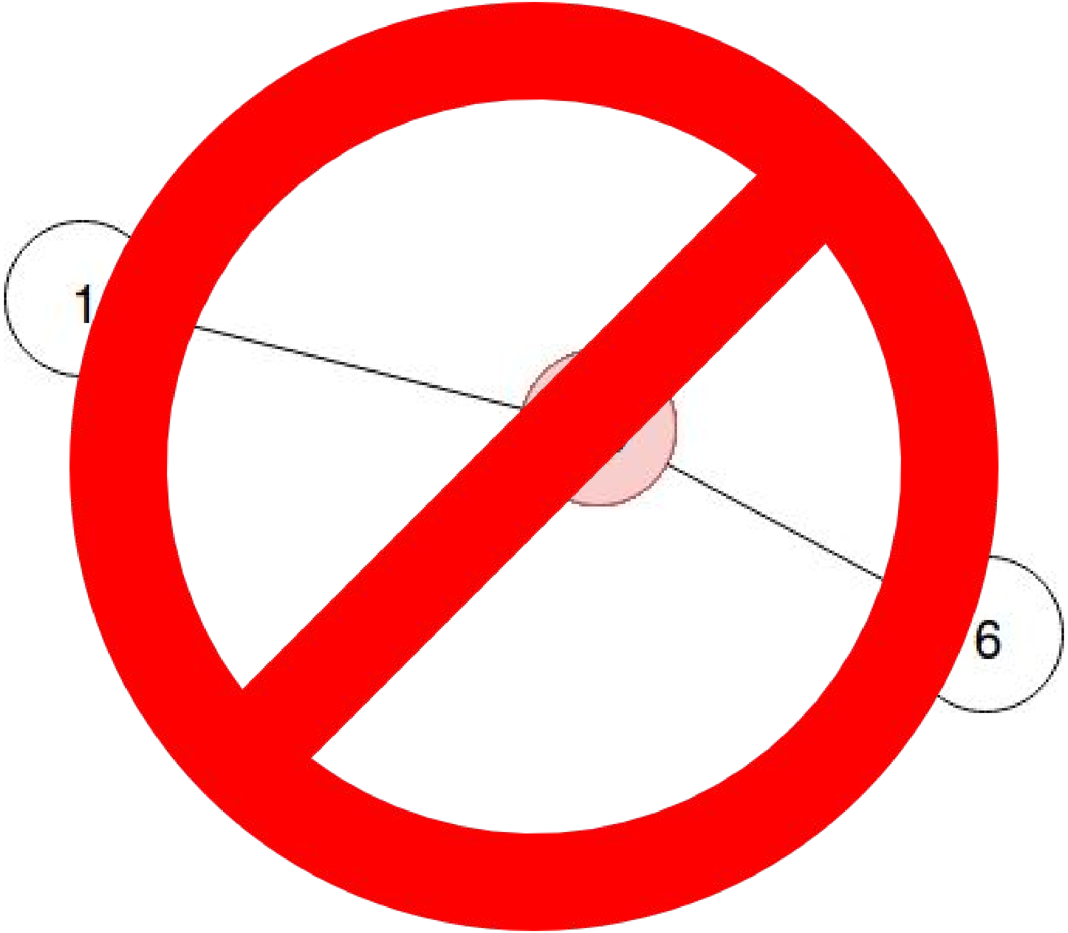


**Clique:**
**A complete subgraph**


1
2
5
2
6
1
1
7
8

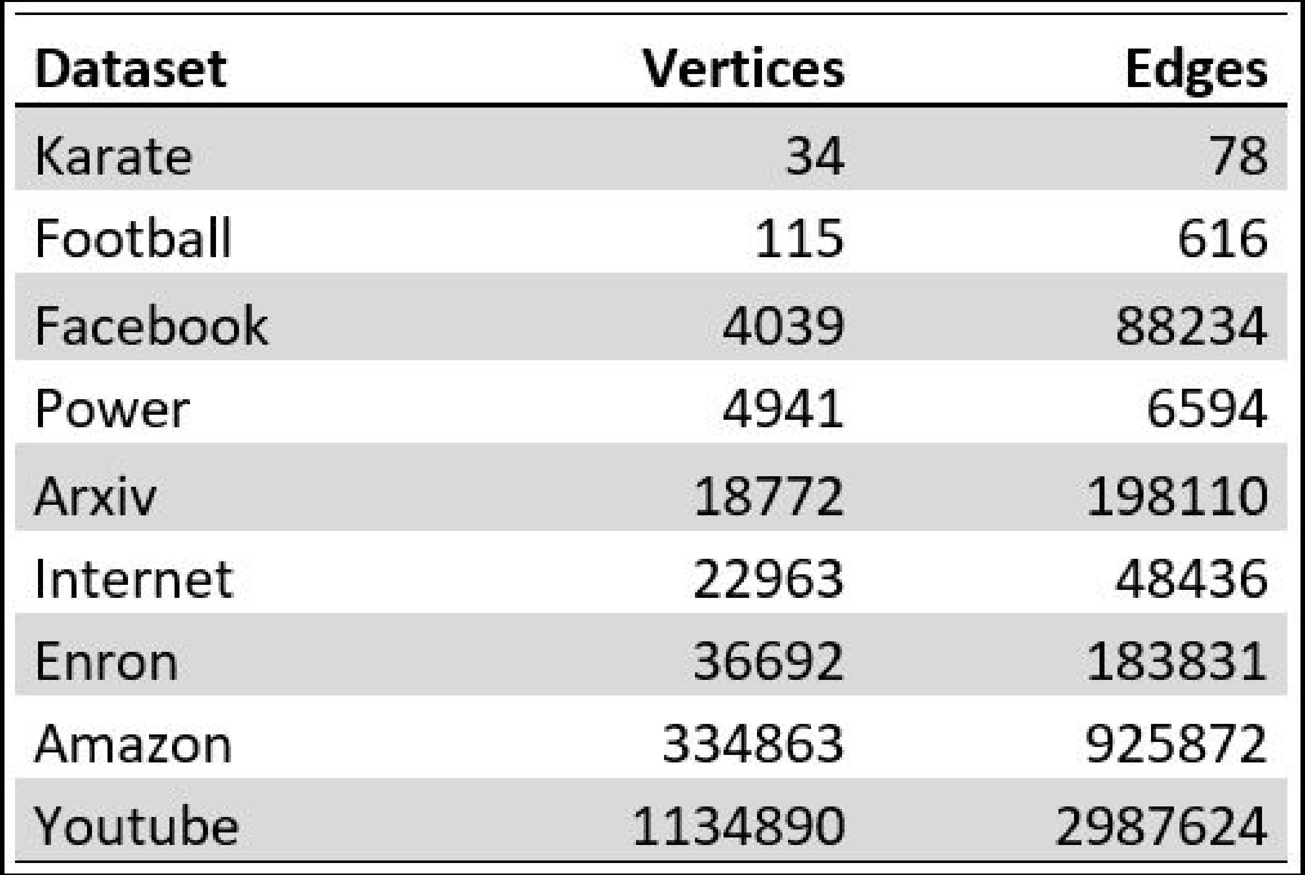

1
2
5
2
6
1
1
7
8


1
10
6

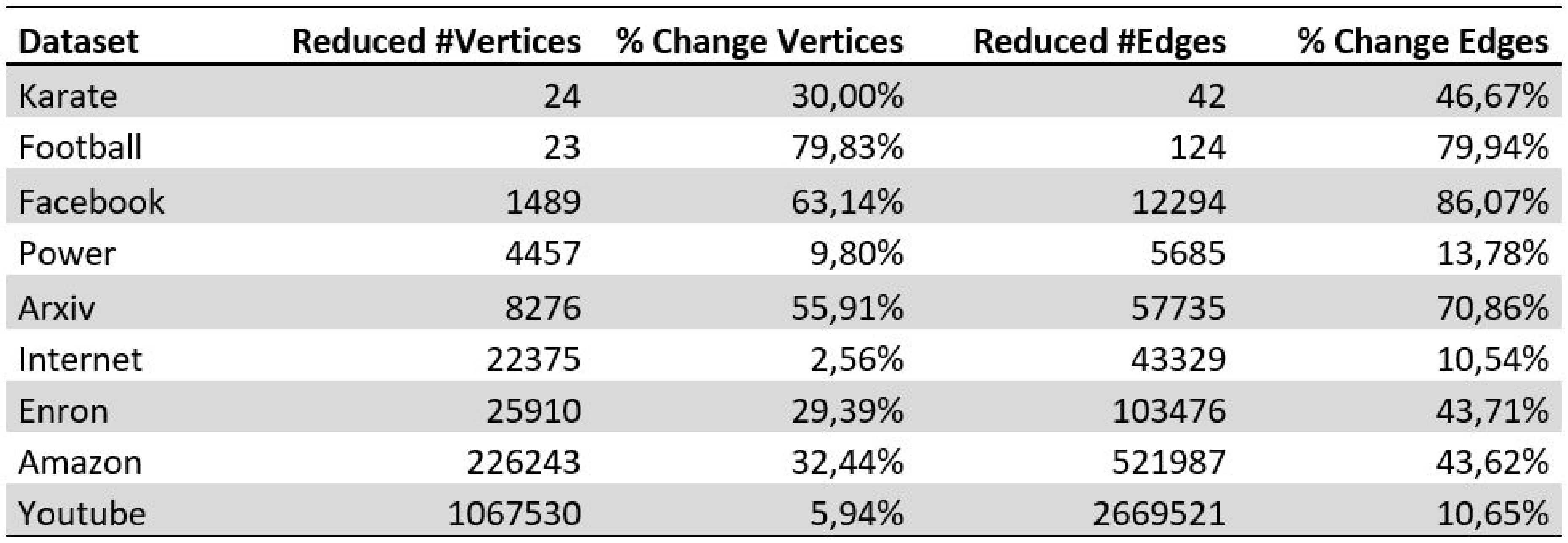

1
6

# Preprocessing Results

SNAP

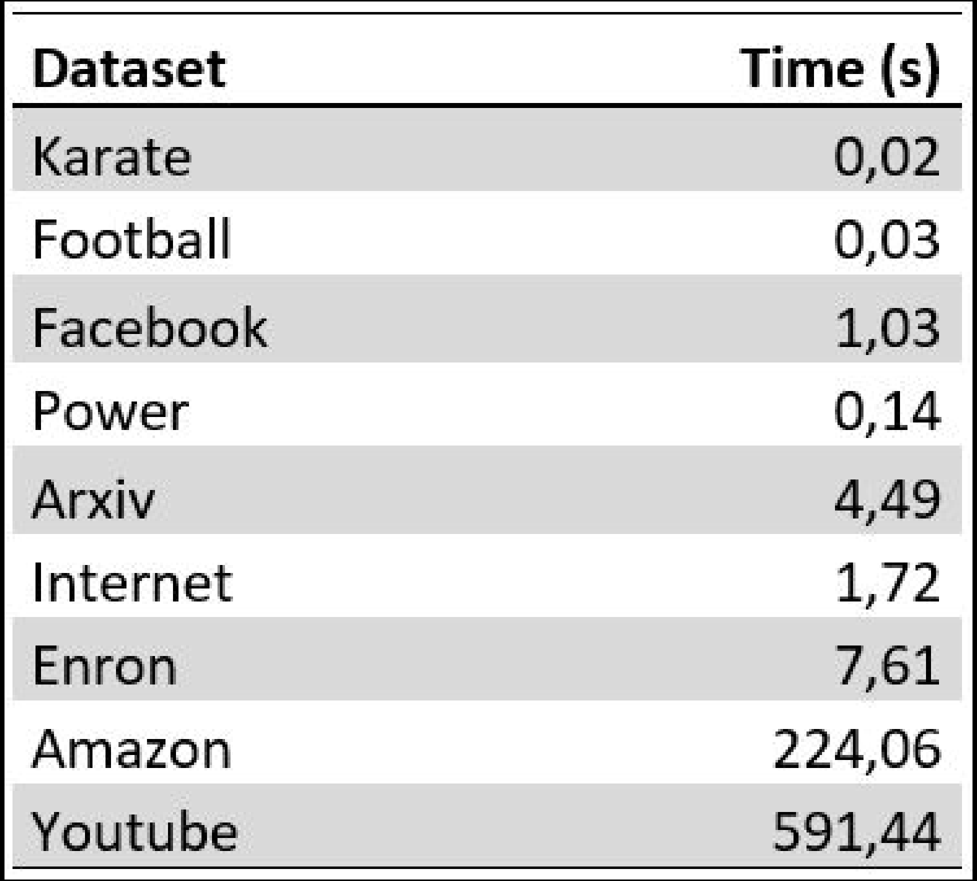


| Dataset | Vertices | Edges |
|---|---:|---:|
| Karate | 34 | 78 |
| Football | 115 | 616 |
| Facebook | 4039 | 88234 |
| Power | 4941 | 6594 |
| Arxiv | 18772 | 198110 |
| Internet | 22963 | 48436 |
| Enron | 36692 | 183831 |
| Amazon | 334863 | 925872 |
| Youtube | 1134890 | 2987624 |



| Dataset | Reduced #Vertices | % Change Vertices | Reduced #Edges | % Change Edges |
|---|---:|---:|---:|---:|
| Karate | 24 | 30,00% | 42 | 46,67% |
| Football | 23 | 79,83% | 124 | 79,94% |
| Facebook | 1489 | 63,14% | 12294 | 86,07% |
| Power | 4457 | 9,80% | 5685 | 13,78% |
| Arxiv | 8276 | 55,91% | 57735 | 70,86% |
| Internet | 22375 | 2,56% | 43329 | 10,54% |
| Enron | 25910 | 29,39% | 103476 | 43,71% |
| Amazon | 226243 | 32,44% | 521987 | 43,62% |
| Youtube | 1067530 | 5,94% | 2669521 | 10,65% |

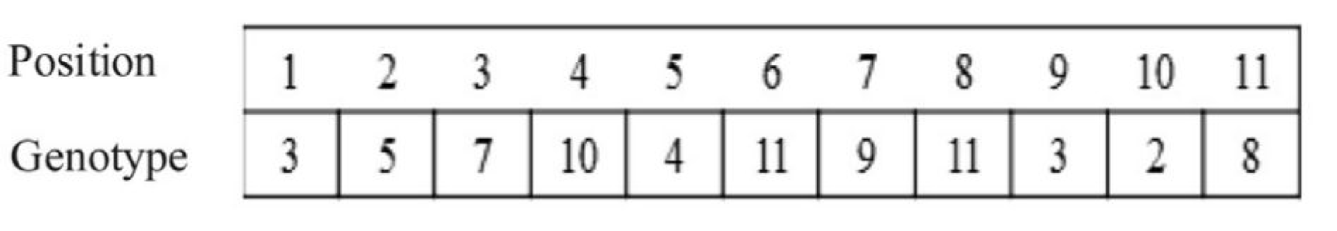
**Highly connected graphs are more easily reduced!**



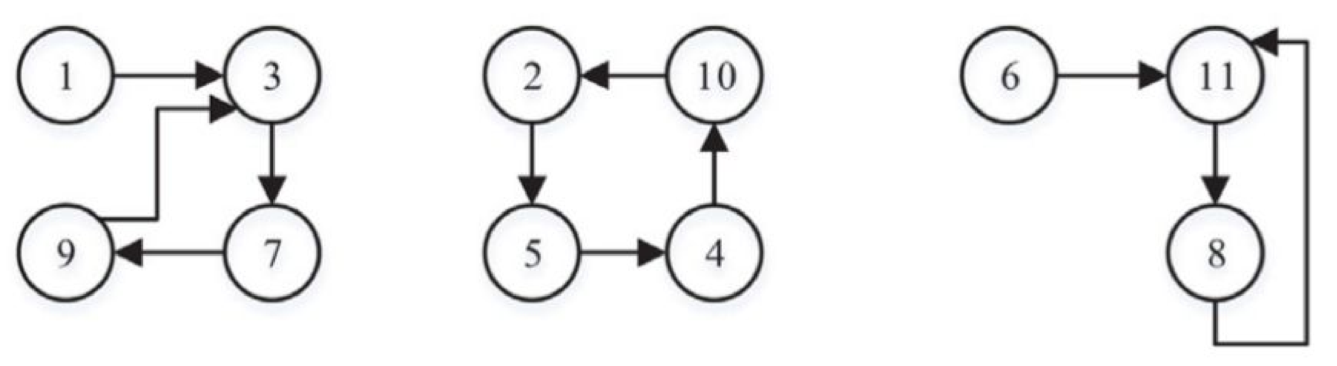
| Dataset | Time (s) |
|---|---:|
| Karate | 0,02 |
| Football | 0,03 |
| Facebook | 1,03 |
| Power | 0,14 |
| Arxiv | 4,49 |
| Internet | 1,72 |
| Enron | 7,61 |
| Amazon | 224,06 |
| Youtube | 591,44 |

# Genetic Algorithm

# Representation

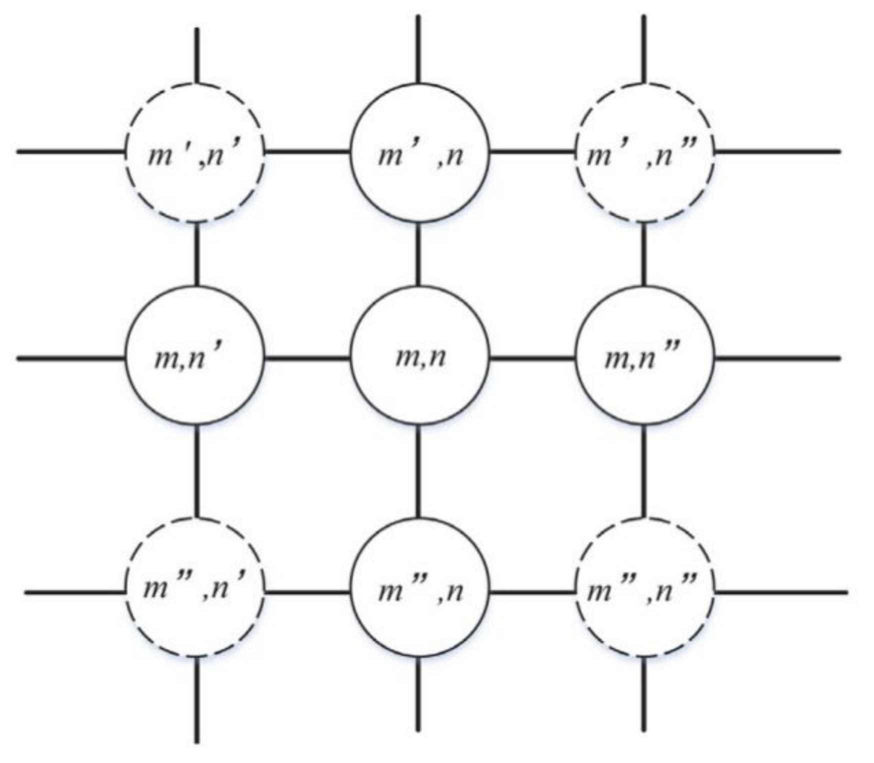


| Position | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Genotype | 3 | 5 | 7 | 10 | 4 | 11 | 9 | 11 | 3 | 2 | 8 |



# Genetic Algorithm: Overview

- Population lives on lattice:



# Genetic Algorithm Operators

1. Randomly split or merge communities based on best neighbor
2. Hybrid crossover with best neighbor: 50% Uniform, 50% 1-point crossover
3. Adaptive Mutation: Mutation chance increase if best fitness stays steady
4. Self Learning (SL) Operator: Only for best few agents
    - Nested genetic algorithm
    - Smaller graph, populated by mutating agent
    - Very small genetic diversity: Exploitation instead of Exploration

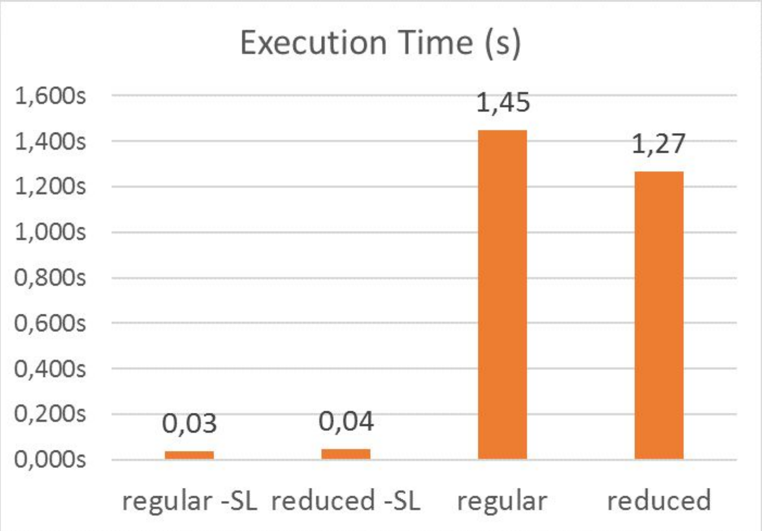
# Genetic Algorithm Results

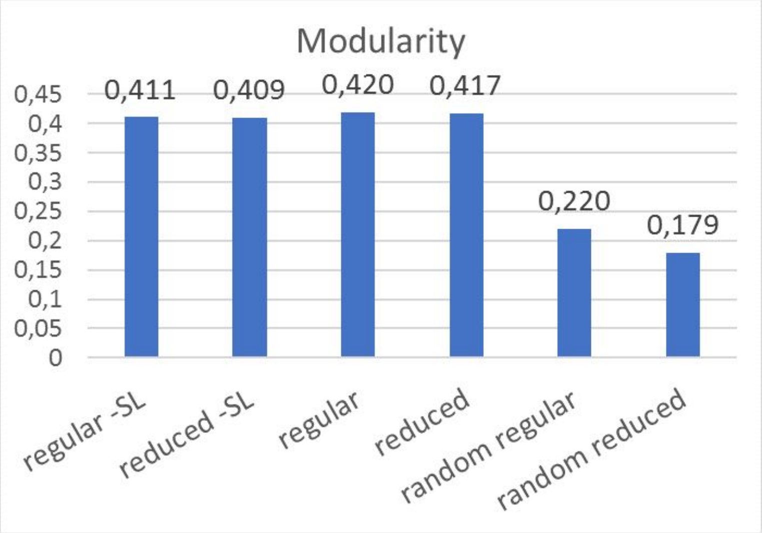
SNAP

# Karate

Nodes: 34 (-30%)
Edges: 78 (-46%)

- Self Learning (SL): long execution time
- Regular vs Reduces: Same execution time
- Baseline advantage to regular
- All good results, SL a bit higher

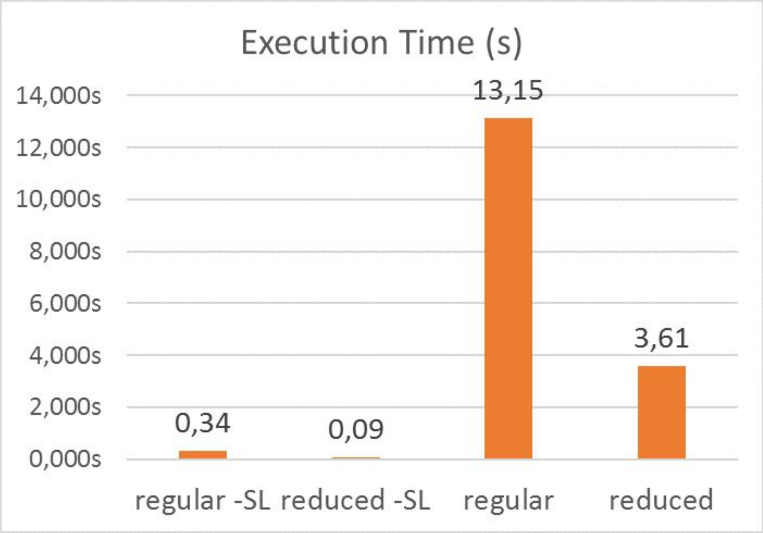


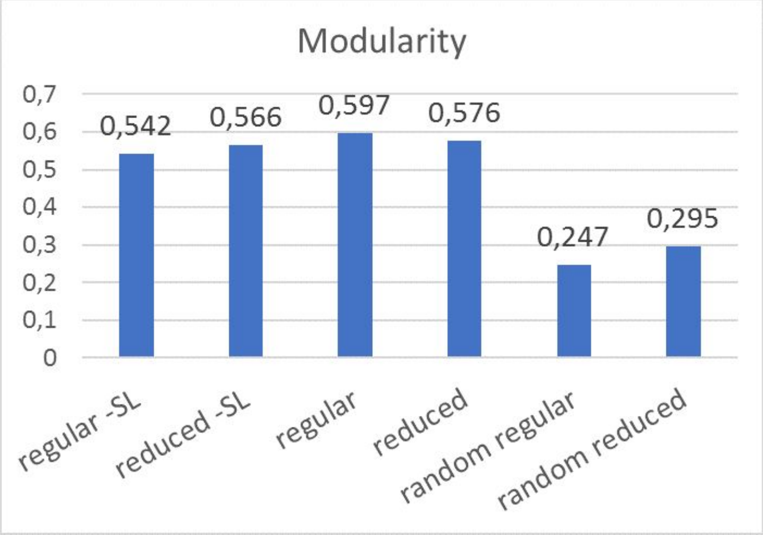


# Football

Nodes: 115 (-80%)
Edges: 616 (-80%)

- SL: long execution time
- In both cases: time advantage for reduced graph
- Reduced: slightly better without SL and baseline
- Regular: slightly better with SL

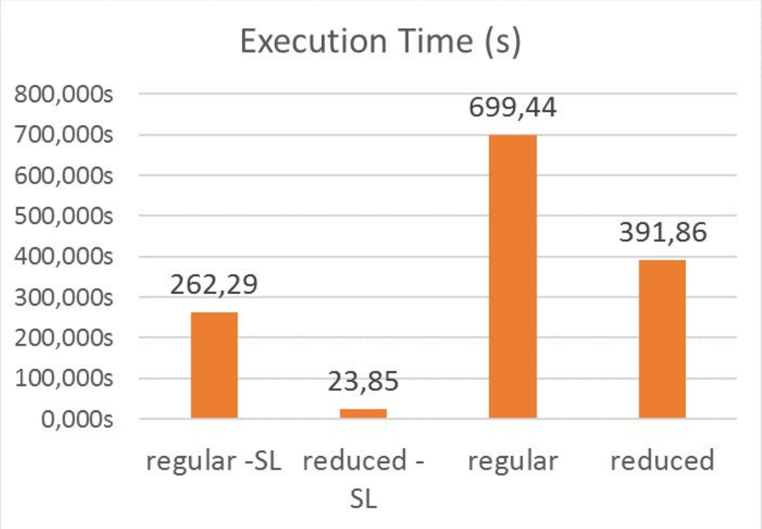


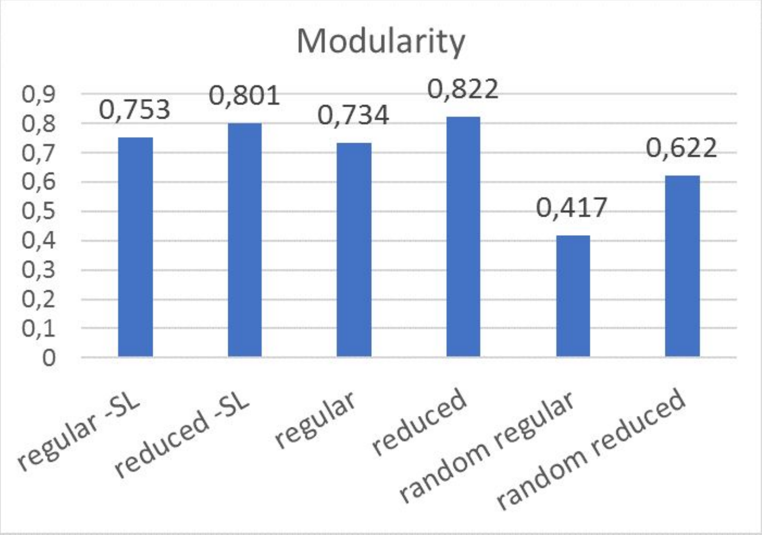


# Facebook

Nodes: 4039 (-63%)
Edges: 88 234 (-86%)

- SL becoming a problem
- Huge time advantage for reduced graph
- Better result for reduced graph (in shorter time!)
- Huge advantage for reduced graph in baseline

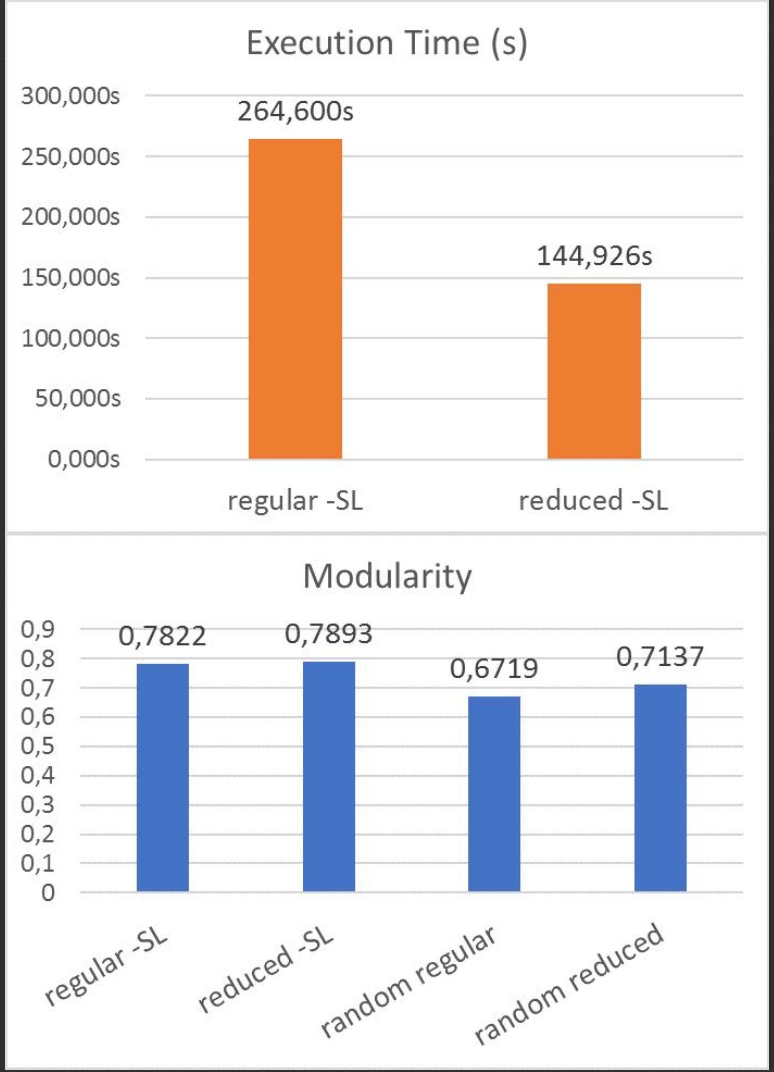




# Power

Nodes: 4941 (-10%)
Edges: 6594 (-14%)

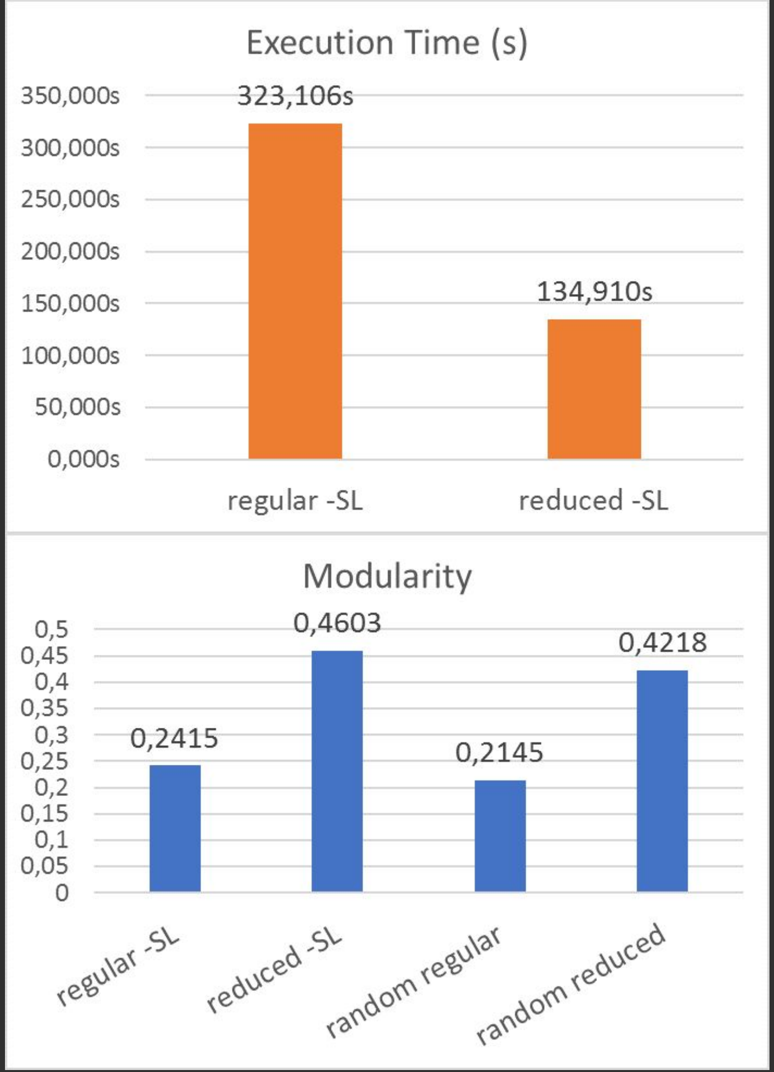
- Large time advantage for reduced graph
- Similar results
- Slight baseline advantage for reduced graph



# Arxiv

Nodes: 18 772 (-56%)
Edges: 57 735 (-71%)

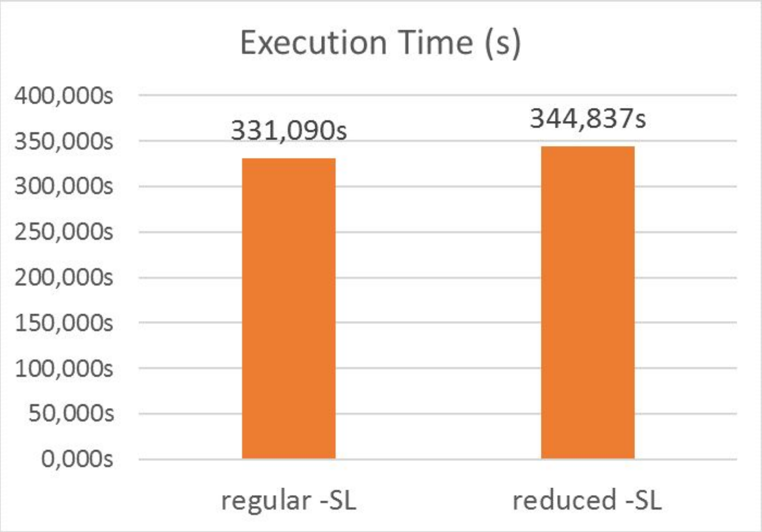
- Large time advantage for reduced graph
- Reduced graph scores much better, also in baseline
- Both cases: barely any improvements above baseline

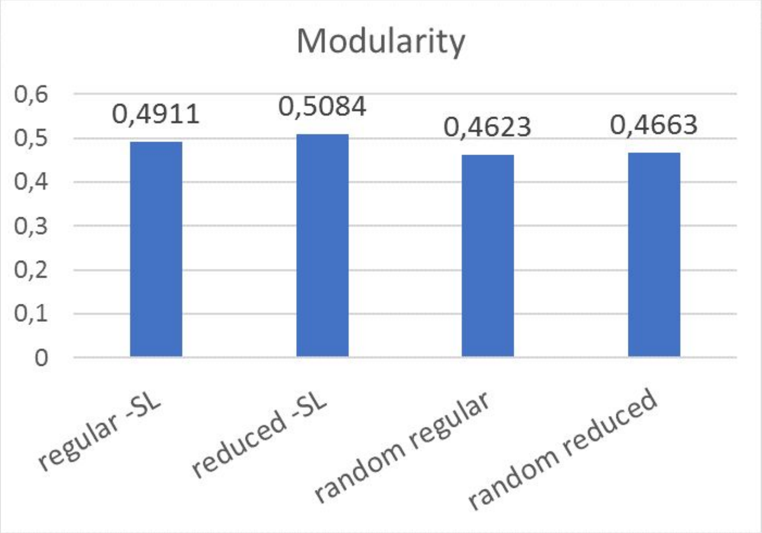


# Internet

Nodes: 22 963 (-3%)
Edges: 48 436 (-11%)

- Very little reduction: no real difference

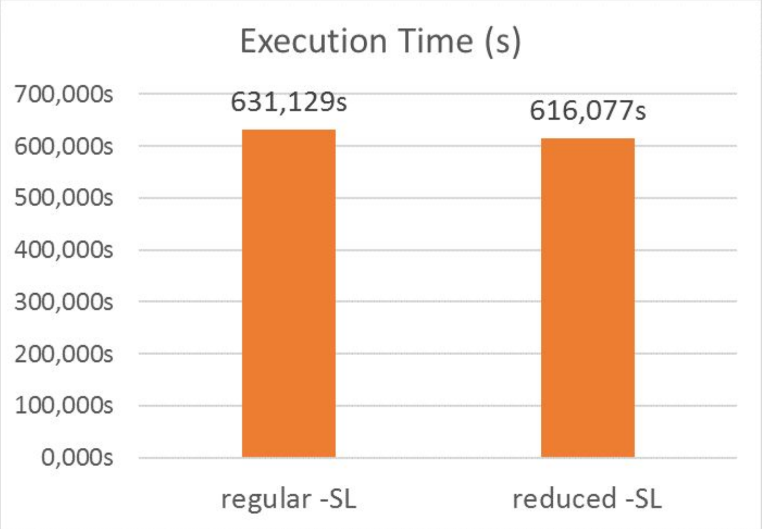


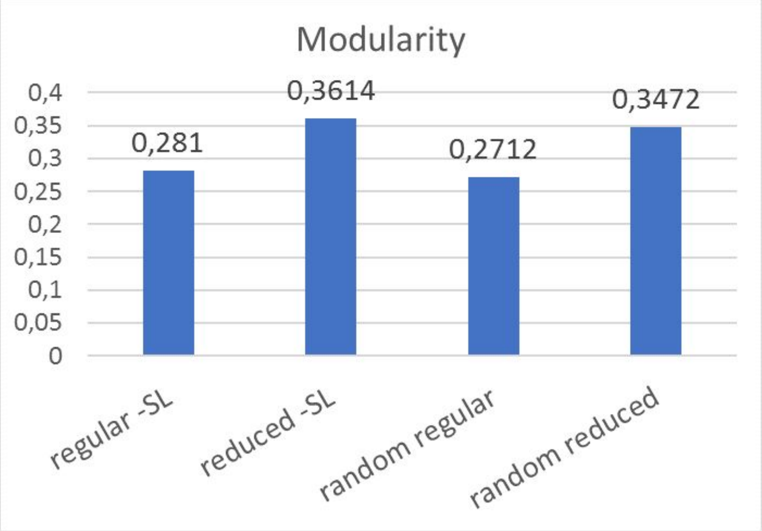


# Enron

Nodes: 25 910 (-29%)
Edges: 183 831 (-44%)

- Advantage for reduced graph
- Practically all from baseline
- Genetic algorithm has become ineffective





# Conclusion

# Conclusions

- Preprocessing was very effective!
  - But it depends on the structure of the graph
  - Performs better as graphs become more connected
- Straightforward idea that can achieve significant improvements
- Implementation issue with Self Learning Operator
- Looks like it can be applied to other algorithms as well
  - How does this impact search for overlapping communities?