Engineering Overlapping Community Detection based on the Ego-Splitting Framework

Armin Wiebigke

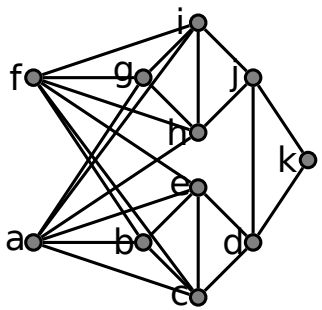
March 18, 2019

Idea

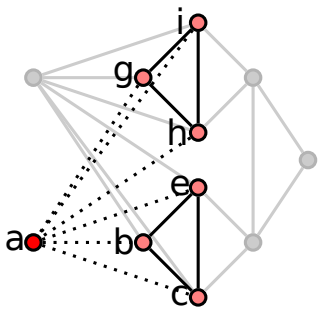
- Problem: Overlapping community detection
- Reduce problem to non-overlapping communities (partition)
  - Many established algorithms
- Two partition algorithms: local and global
- *Ego-net* of node u: Subgraph induced by the neighbors of u

The Framework

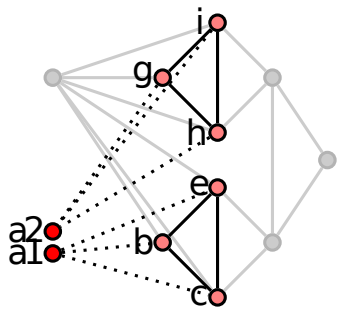
- For each node:
  - Create the ego-net graph (triangle search)
  - Use the local partition algorithm on the ego-net
- Create persona graph:
  - Create a persona for each partition
  - Add edges between personas



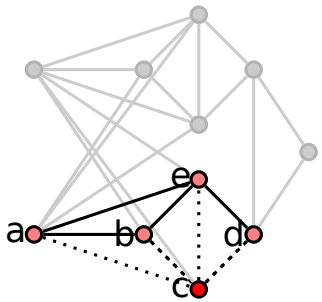
(a) original graph G



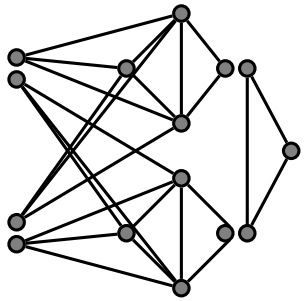
(b) ego-net of a



(c) split ing a in two personas

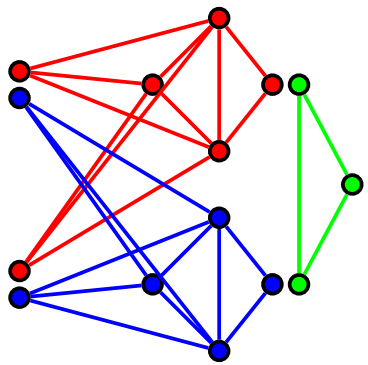


(d) ego-net of c (one persona)

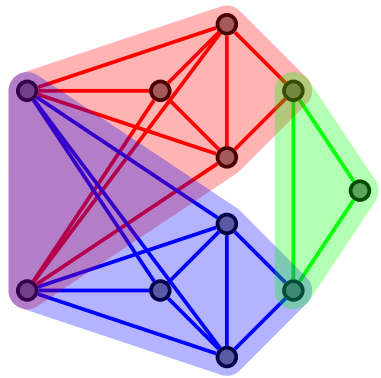

(e) persona graph

The Framework

- Use the global partition algorithm on the persona graph
  - Each partition is one community
- For each node: Collect communities from all personas



(a) non-overlapping partition of the persona graph



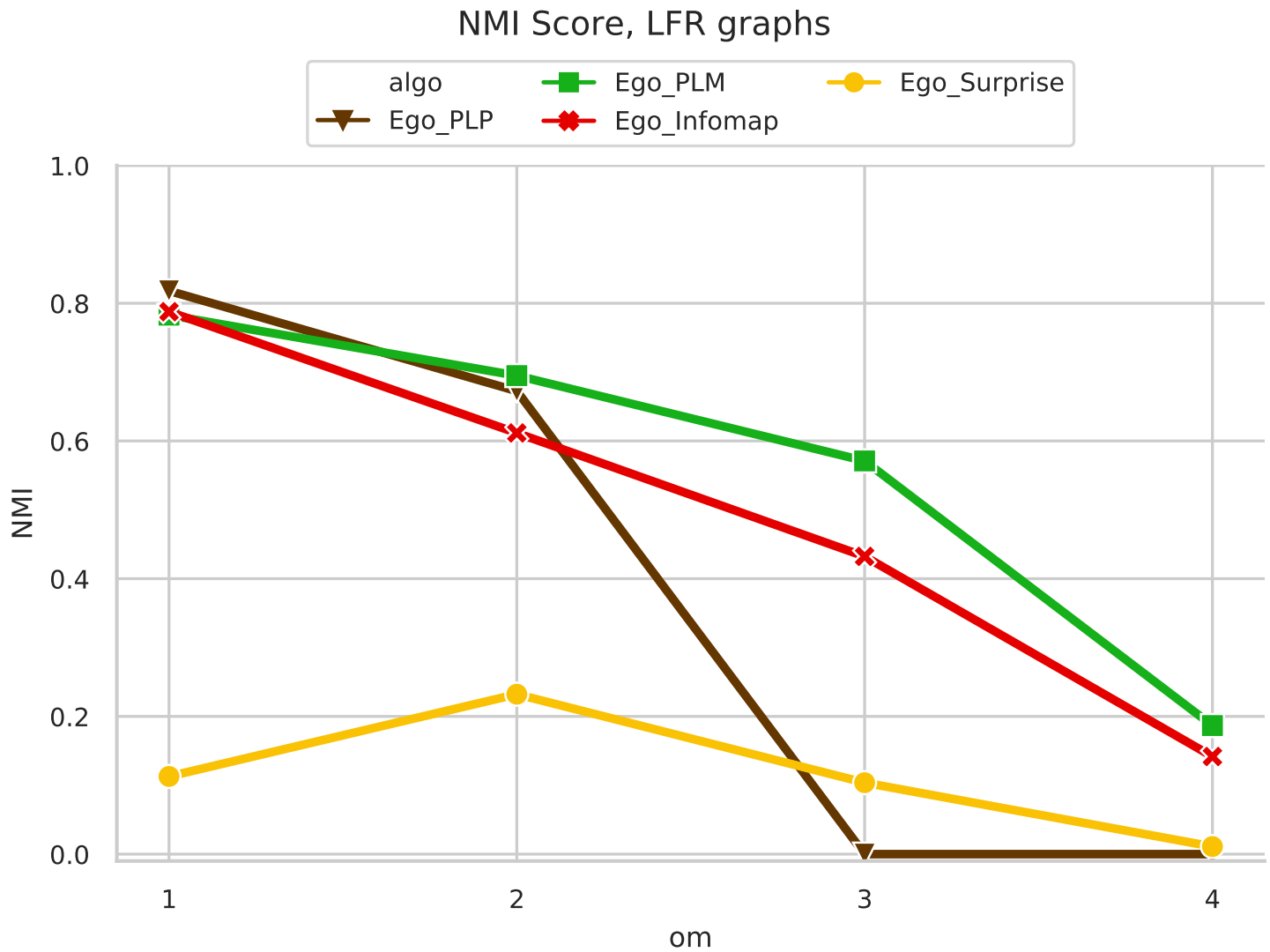
(b) overlapping clusters in original graph

Problems & Possible Solutions

- Local partitioning is critical
  - number of Communities $\leq$ number of Personas
  - Detecting communities in ego-net can be difficult/impossible
- Connect personas of one node with additional edges
- Increase size of ego-net
  - neighbors of neighbors
  - squares

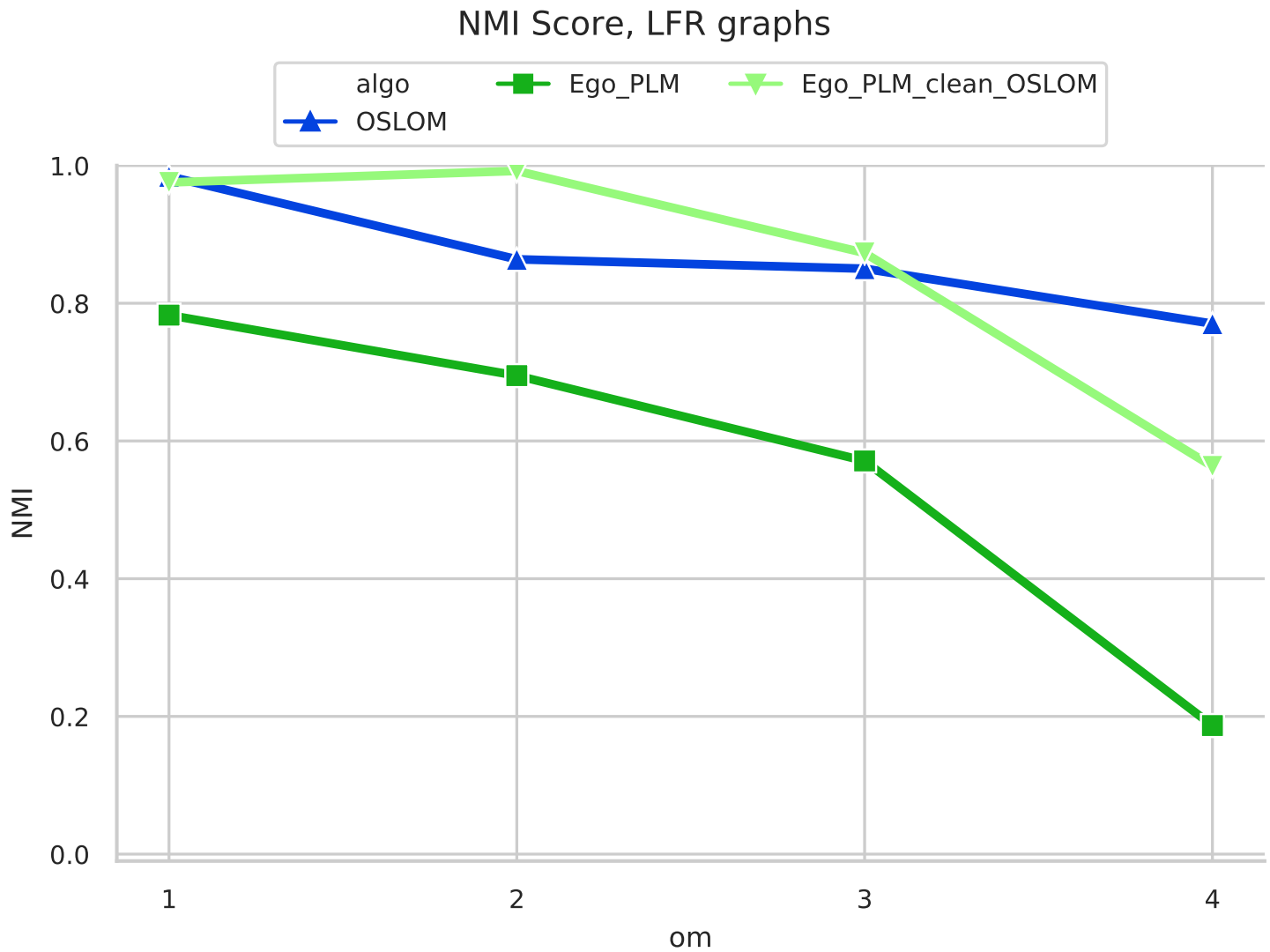
Problems & Possible Solutions

- Which partition algorithm(s)?

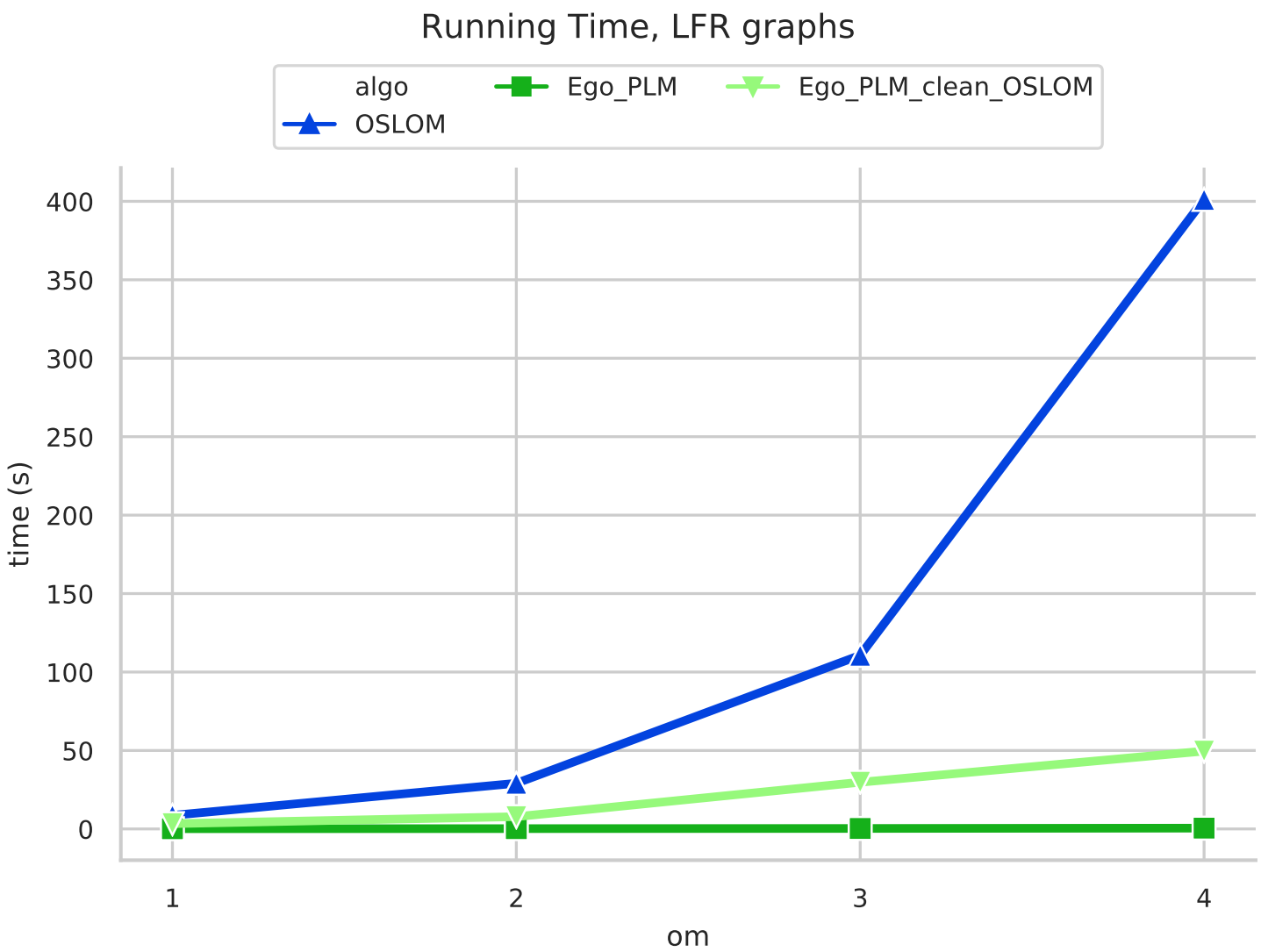


Possible Improvements

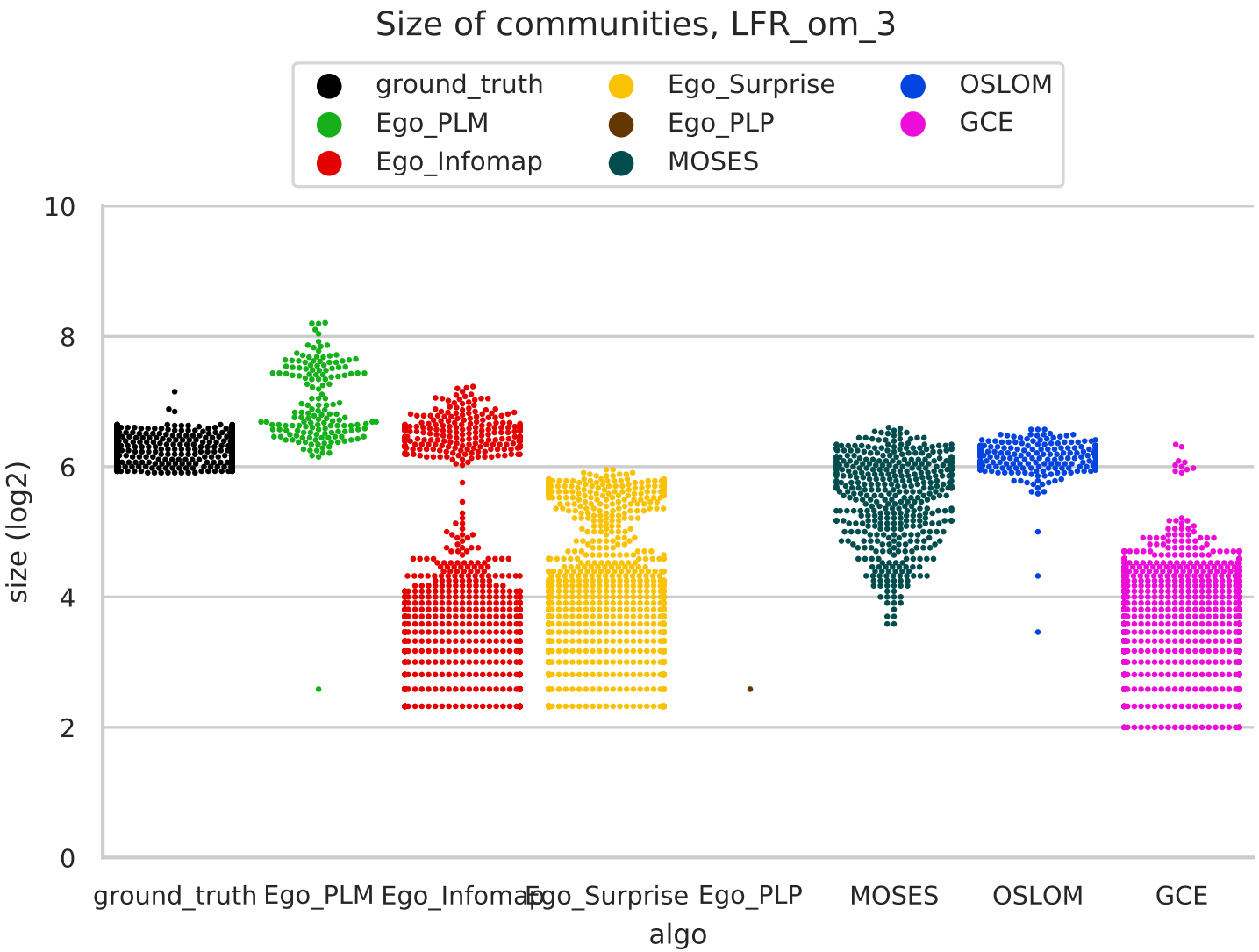
- Clean-up at the end (OSLOM)



Plots


Running Time, LFR graphs
algo
Ego_PLM
Ego_PLM_clean_OSLOM
OSLOM
400
350
300
250
200
150
100
50
0
time (s)
1
2
3
4
om

Plots


Size of communities, LFR_om_3
ground_truth
Ego_PLM
Ego_Infomap
Ego_Surprise
Ego_PLP
MOSES
OSLOM
GCE
size (log2)
10
8
6
4
2
0
ground_truth
Ego_PLM
Ego_Infomap
Ego_Surprise
Ego_PLP
MOSES
OSLOM
GCE
algo