Data Mining 10.12.2017 – Piotr Mrowczynski – Assignment 5

The goal of this assignment is to understand distributed graph partitioning using gossip-based peer-to-peer techniques, such as, JaBeJa described in [F. Rahimian, et al., JA-BE-JA: A Distributed Algorithm for Balanced Graph Partitioning, SASO2013]

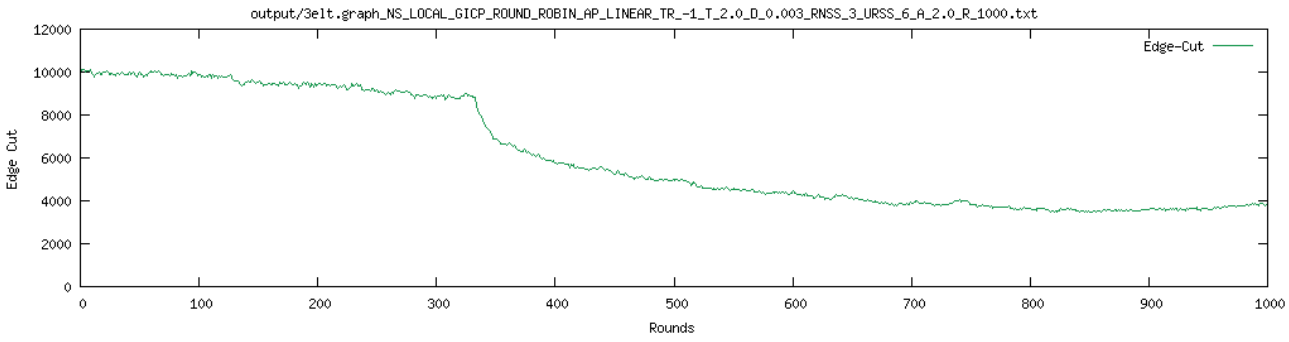
Project structure

```
k-way-graph-partitioning
├── src/main/java/se/kth/jabeja
│   └── Jabeja.java                : JaBeJa algorithm implementation
├── compile.sh                     : Building project
├── run.sh                         : Running project
├── plot.sh                        : Plotting results
└── FreqItemsAssoc.odt
```

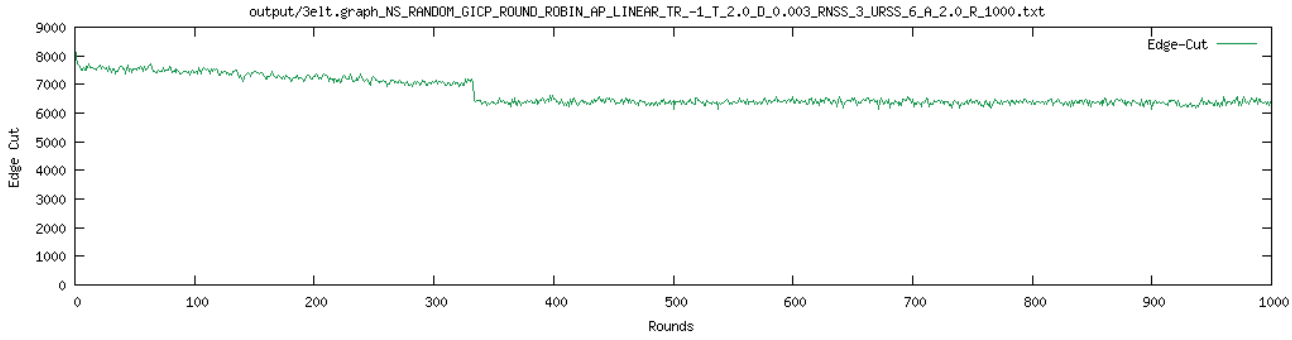
3elt.graph

1000 rounds, 0.003 delta, 2.0 temp, 2.0 alpha

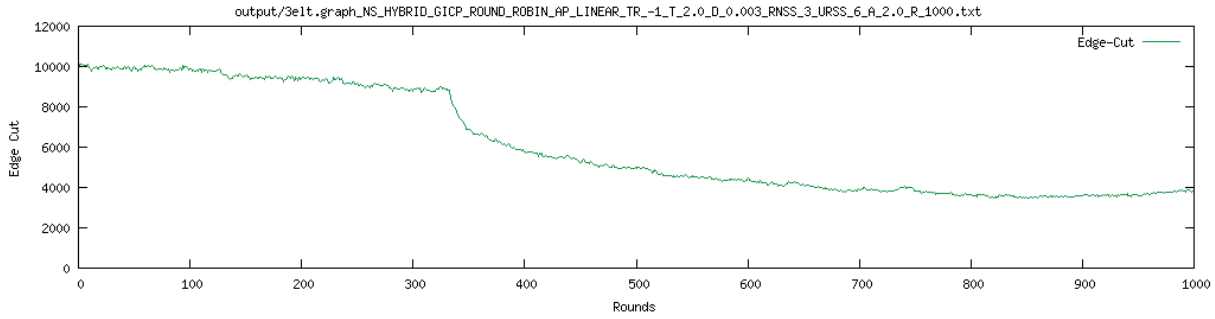
Linear anealing with local candidates – ~4000edge cut



Linear anealing with random candidates – ~6200edge cut



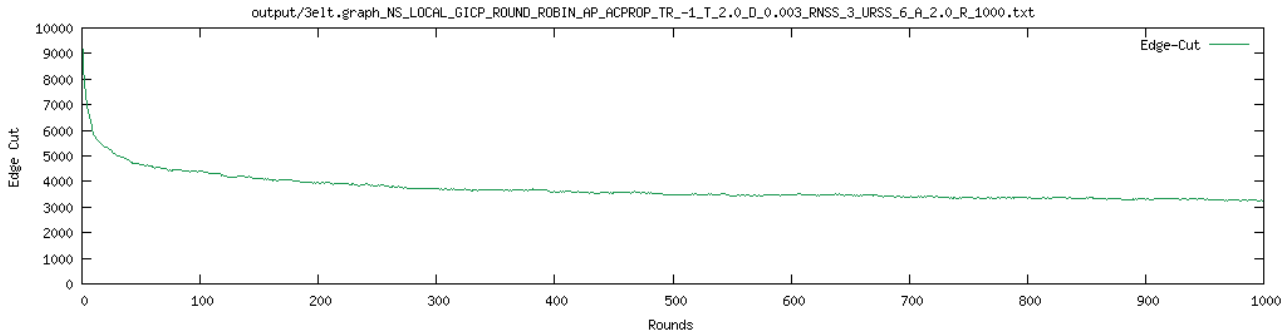
Linear anealing with hybrid candidates – ~3800edge cut



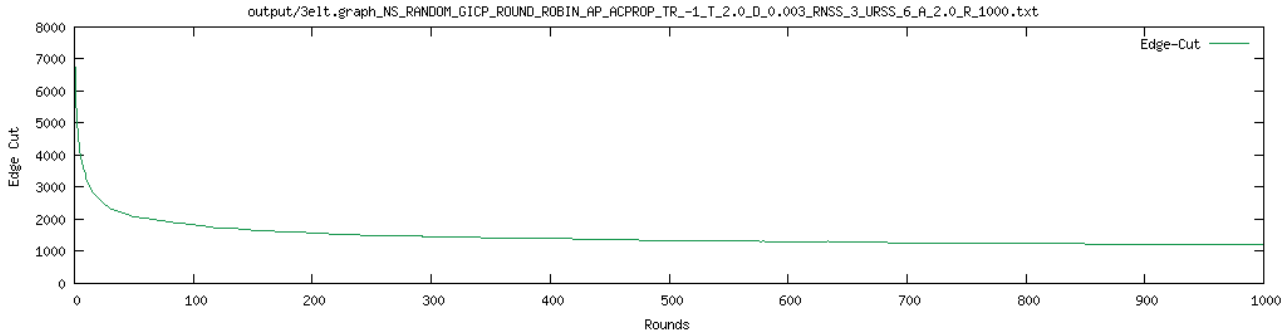
Linear anealing conclusions

For this graph using linear anealing, local and hybrid gave good results, while random gave bad results

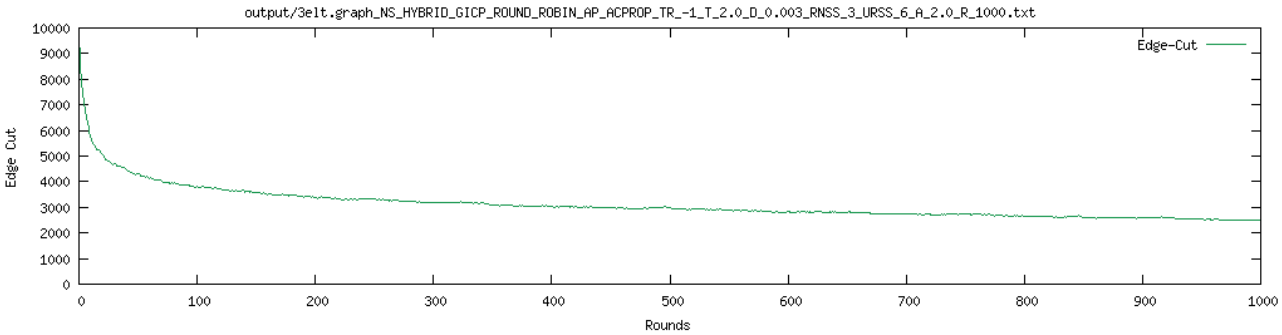
Acceptance Probability with local candidates – ~3200edge cut



Acceptance Probability with random candidates – ~1200edge cut



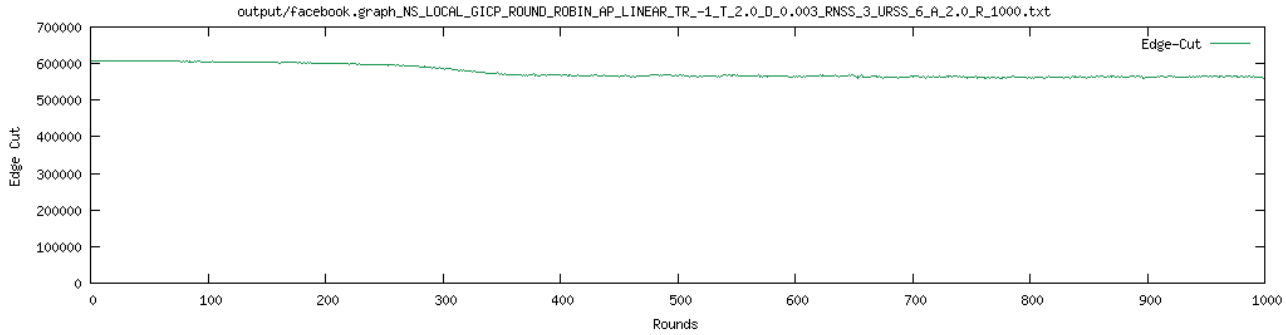
Acceptance Probability with hybrid candidates – ~2500edge cut



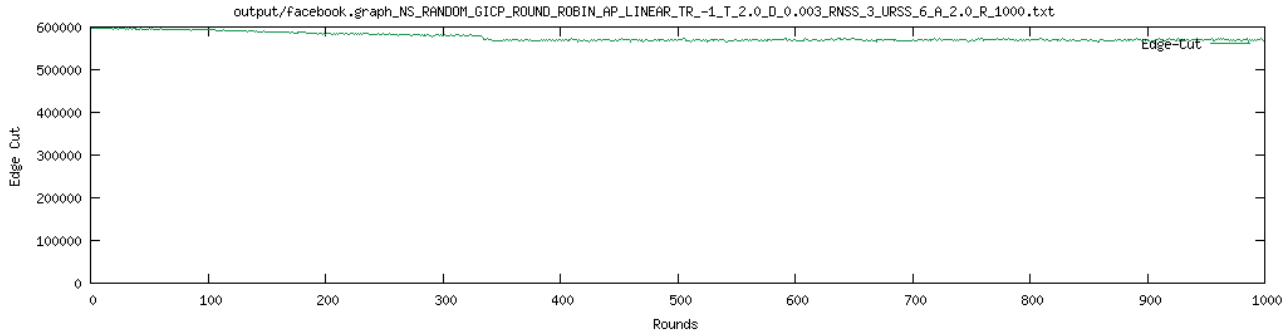
Facebook

1000 rounds, 0.003 delta, 2.0 temp, 2.0 alpha

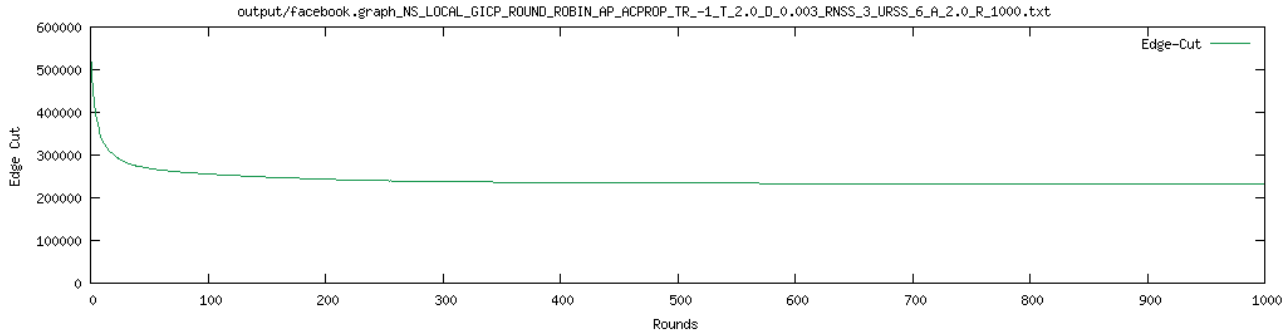
Linear anealing with local candidates – ~565000 edge cut



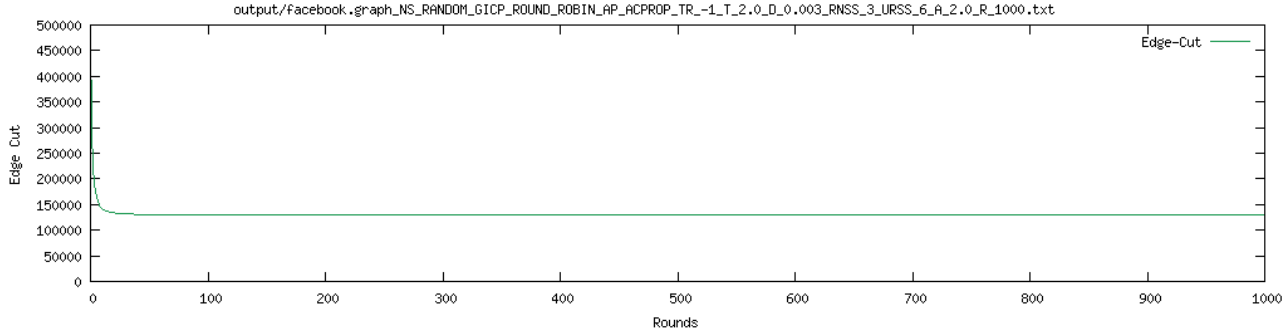
Linear anealing with random candidates – ~580000 edge cut



Acceptance Probability with local candidates – ~220000 edge cut



Acceptance Probability with random candidates – ~130000 edge cut

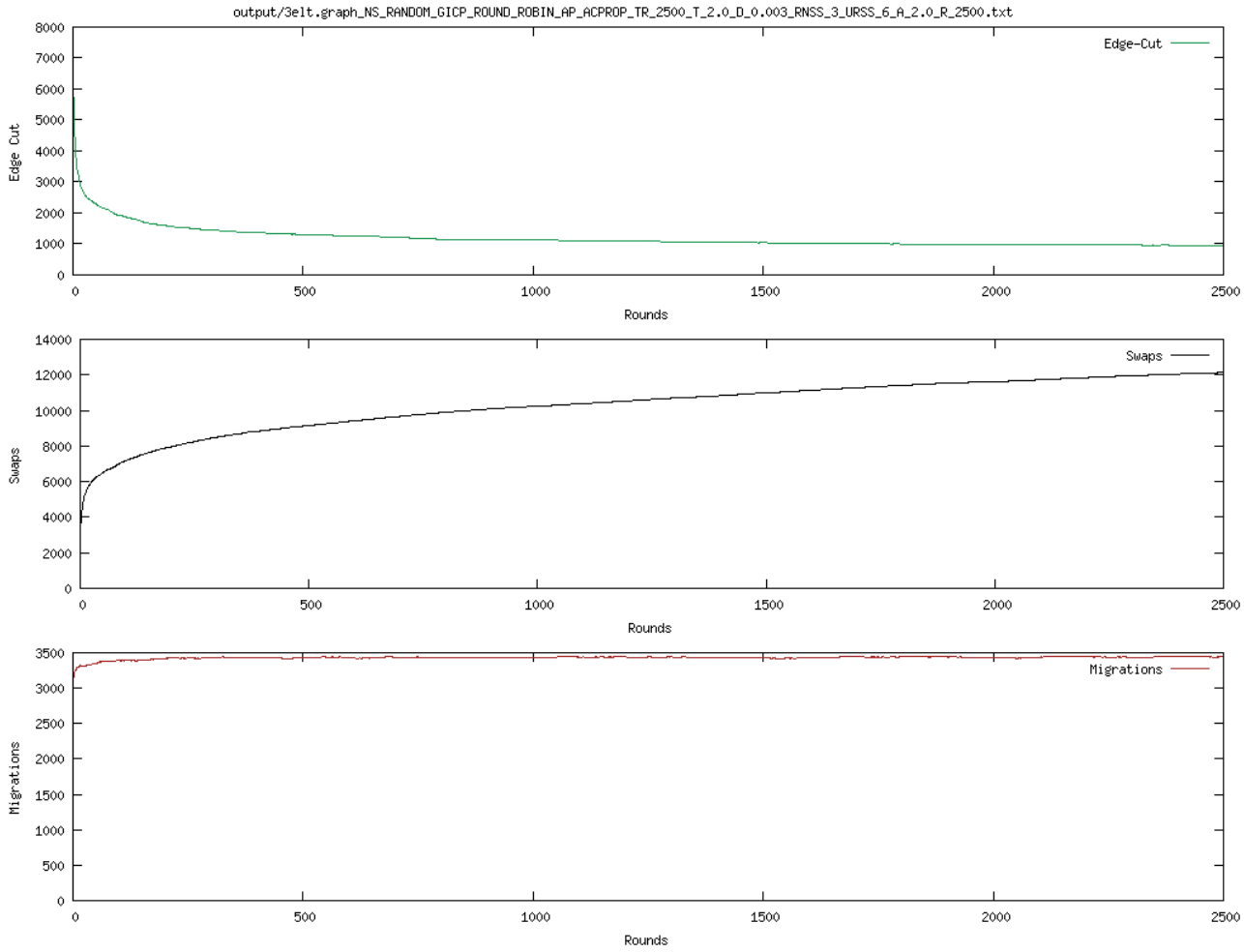


Acceptance probability anealing conclusions – both graphs

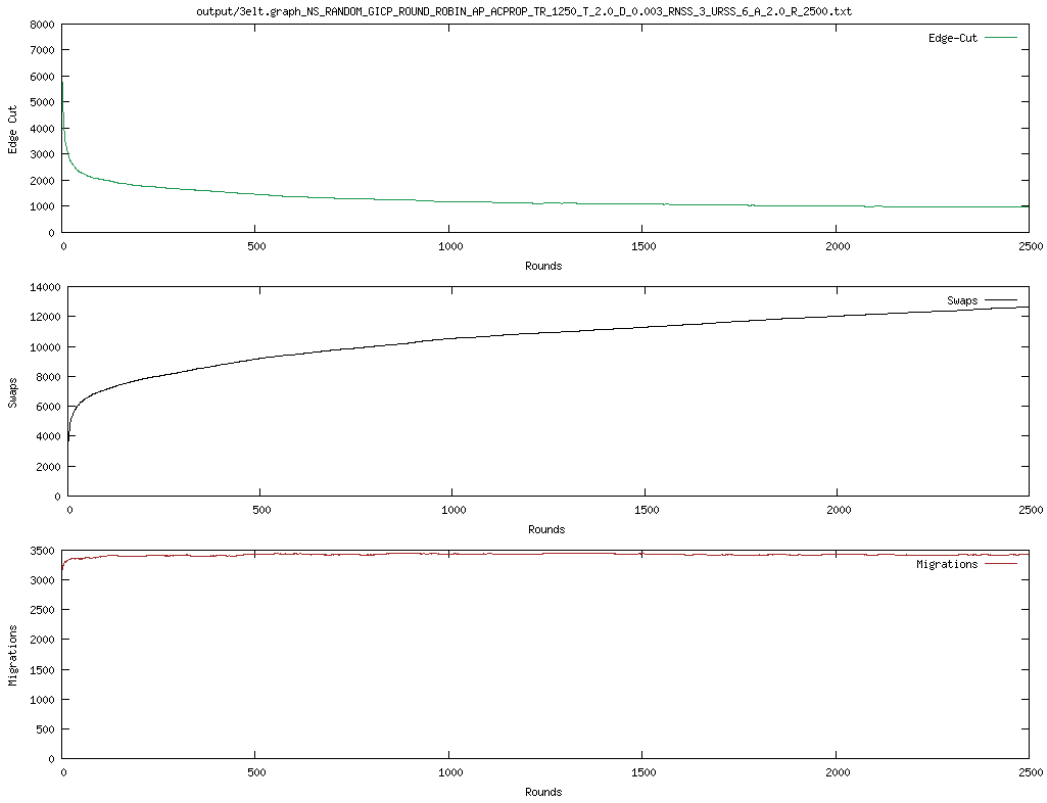
Using acceptance probability in both graphs convergence is achieved much faster, and with better edge-cuts. It is due to avoiding a problem of stuck in the local optima solution to do the fact with certain (decreasing) probability we allow bad-cuts to be able to get into global optima.

3elt.graph – parameters testing

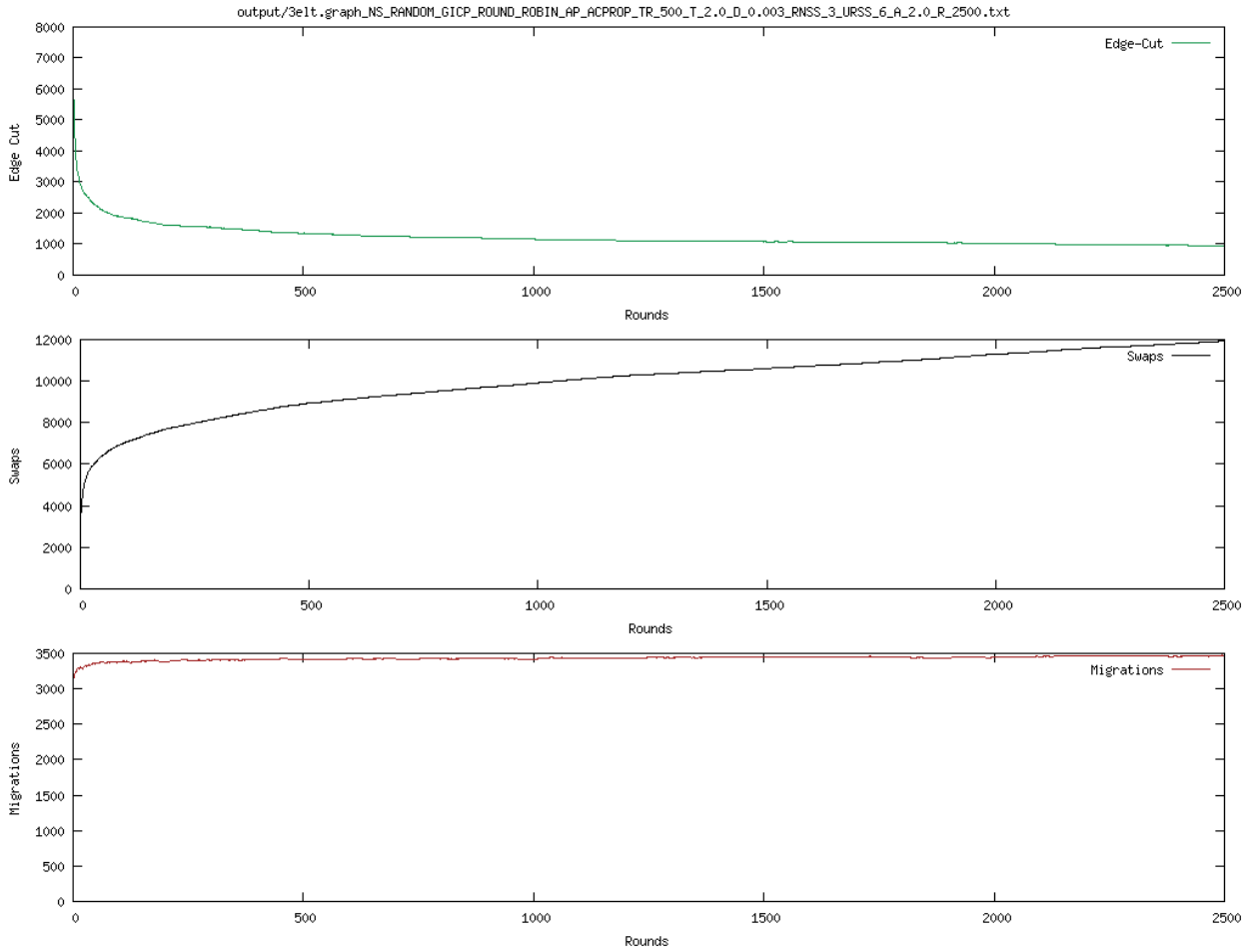
2500 rounds, random, acceptance probability, reset 2500 – ~1000 edge cuts



2500 rounds, random, acceptance probability, reset 1250 – ~980 edge cuts

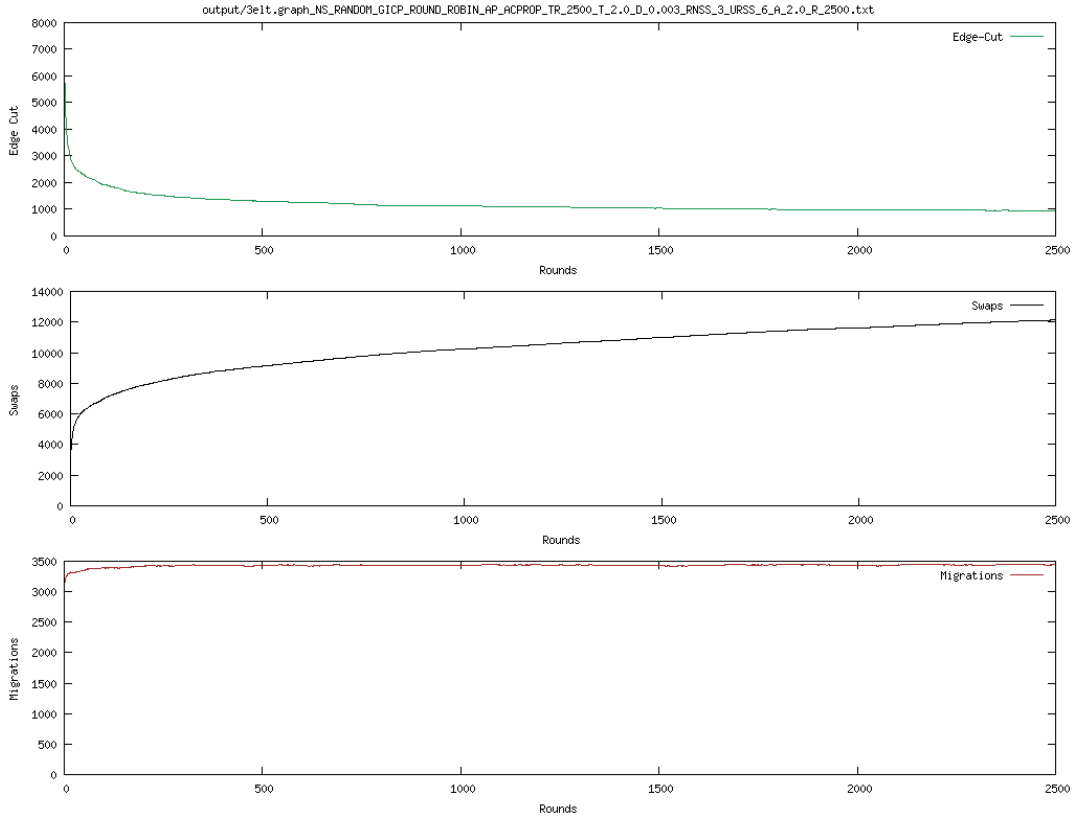


2500 rounds, random, acceptance probability, reset 500 – ~930 edge cuts


output/3elt.graph_NS_RANDOM_GICP_ROUND_ROBIN_AP_ACPROP_TR_500_T_2.0_D_0.003_RNSS_3_URSS_6_A_2.0_R_2500.txt
Edge-Cut
Edge Cut
Rounds
Swaps
Swaps
Rounds
Migrations
Migrations
Rounds

3elt.graph – parameters performance

random – acceptance probability – slow and fast converge, low number of swaps



hybrid – linear – fast and slow converge, high number of swaps