The JaBeJa Algorithm Report: An Explanation and Evaluation

Dario Del Gaizo Filippo Caliò
dariodg@kth.se | calio@kth.se

December 2022

1 Introduction

The JaBeJa [1] algorithm is a distributed algorithm for solving the graph coloring problem for graph partition (same color = same partition). In this report, we will explain the main steps of the JaBeJa algorithm and describe some tests that we performed to evaluate different parameters.

2 The JaBeJa Algorithm

The JaBeJa algorithm proceeds in rounds, where each round consists of the following main steps:

1. Each node selects a random neighbor and exchanges its color with that neighbor.
2. Each node updates its color to the most common color among its neighbors.
3. Repeat until convergence.

There are different ways of selecting the candidate set for swapping:

1. Local (L): every node considers its directly connected nodes (neighbors) as candidates for color exchange.
2. Random (R): every node selects a uniform random sample of the nodes in the graph.
3. Hybrid (H): in this policy first the immediate neighbor nodes are selected (i.e., the local policy). If this selection fails to improve the pair-wise utility, the node is given another chance for improvement, by letting it to select nodes from its random sample (i.e., the random policy).

3 Tests and Results

3.1 Task 1

After implementing the required methods, we performed a series of tests to evaluate the performance of the JaBeJa algorithm tweaking the parameters. In each test, we ran the algorithm on the graphs proposed in the assignment.

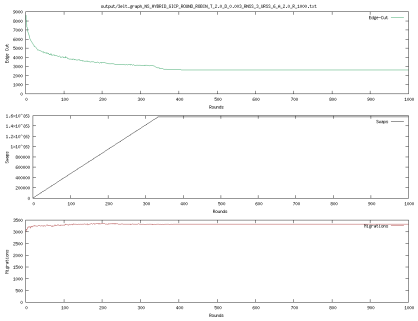

Figure 1: Default settings on the graph *3elt*.

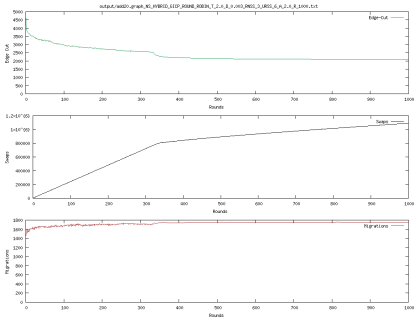

Figure 2: Default settings on the graph *add20*.

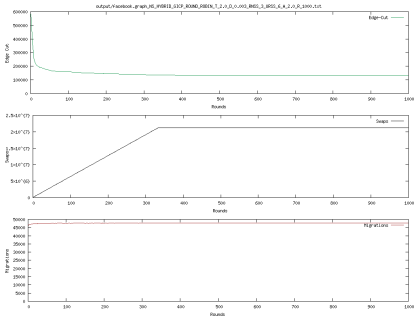

Figure 3: Default settings on the graph *facebook*.

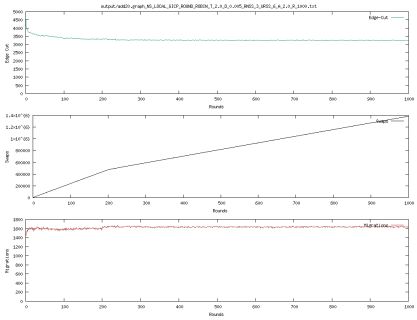

Figure 4: Default settings on the graph *add20* except for $\delta = 0.005$ and LOCAL nodeSelectionPolicy.

The default settings are:

Variable	Value
rounds	1000
numPartitions	4
uniformRandSampleSize	6
temp	2
delta	0.003
seed	0
alpha	2
randNeighborsSampleSize	3
graphInitColorSelectionPolicy	ROUND_ROBIN
nodeSelectionPolicy	HYBRID
annealingPolicy	LINEAR

3.2 Task 2

For the second task, an exponentially decreasing annealing policy is implemented instead of the linear one. The formula used for the *acceptance probability* is $a_p = e^{(new-old)/T}$. This time the parameters have been set to:

Variable	Value
temp	1
delta	0.9
annealingPolicy	EXPONENTIAL

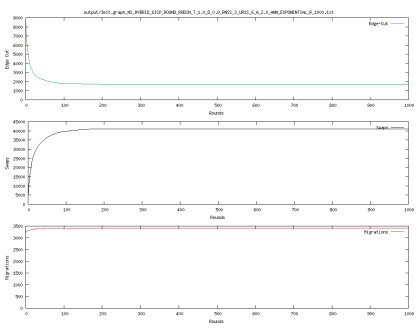

Figure 5: Exponential annealing on the graph *3elt*.

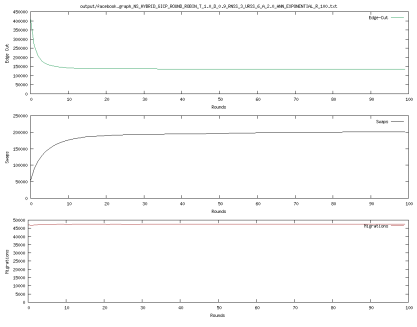

Figure 6: Exponential annealing on the graph *facebook* stopped at round 100.

4 Conclusion

In this paper, we have explained the main steps of the JaBeJa algorithm and presented some test results that demonstrate its performance with different

parameters. To increase exploration and exit local minima, the restart method has been implemented every $200 rounds$ with $\delta = 0.08$, $\alpha = 6$, $T = 5$ and has the following effect on *3elt*. With different parameters it could significantly yield another solution that results in a different edge cut.

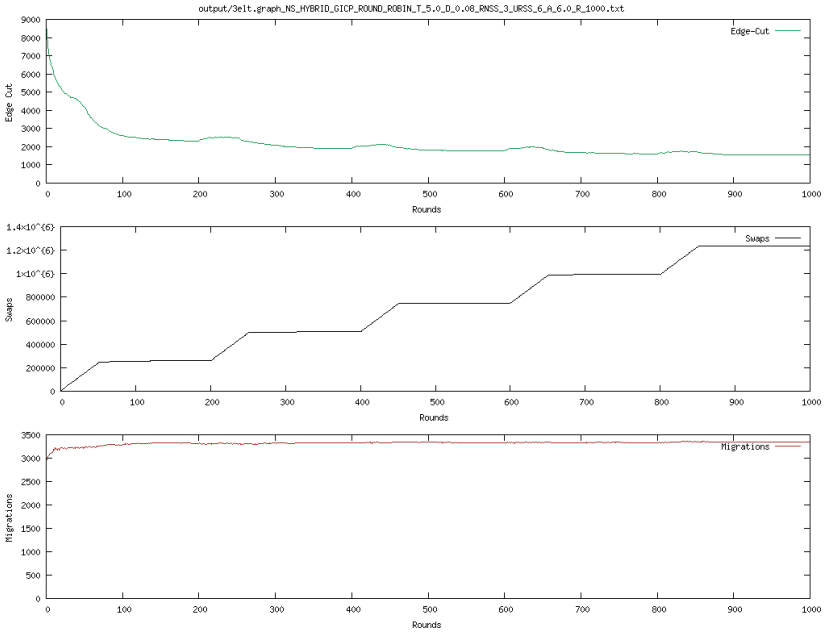

Figure 7: Linear annealing on the graph *3elt* with restart.

References

[1] Fatemeh Rahimian, Amir H. Payberah, Sarunas Girdzijauskas, Mark Jelasity, and Seif Haridi. Ja-be-ja: A distributed algorithm for balanced graph partitioning.