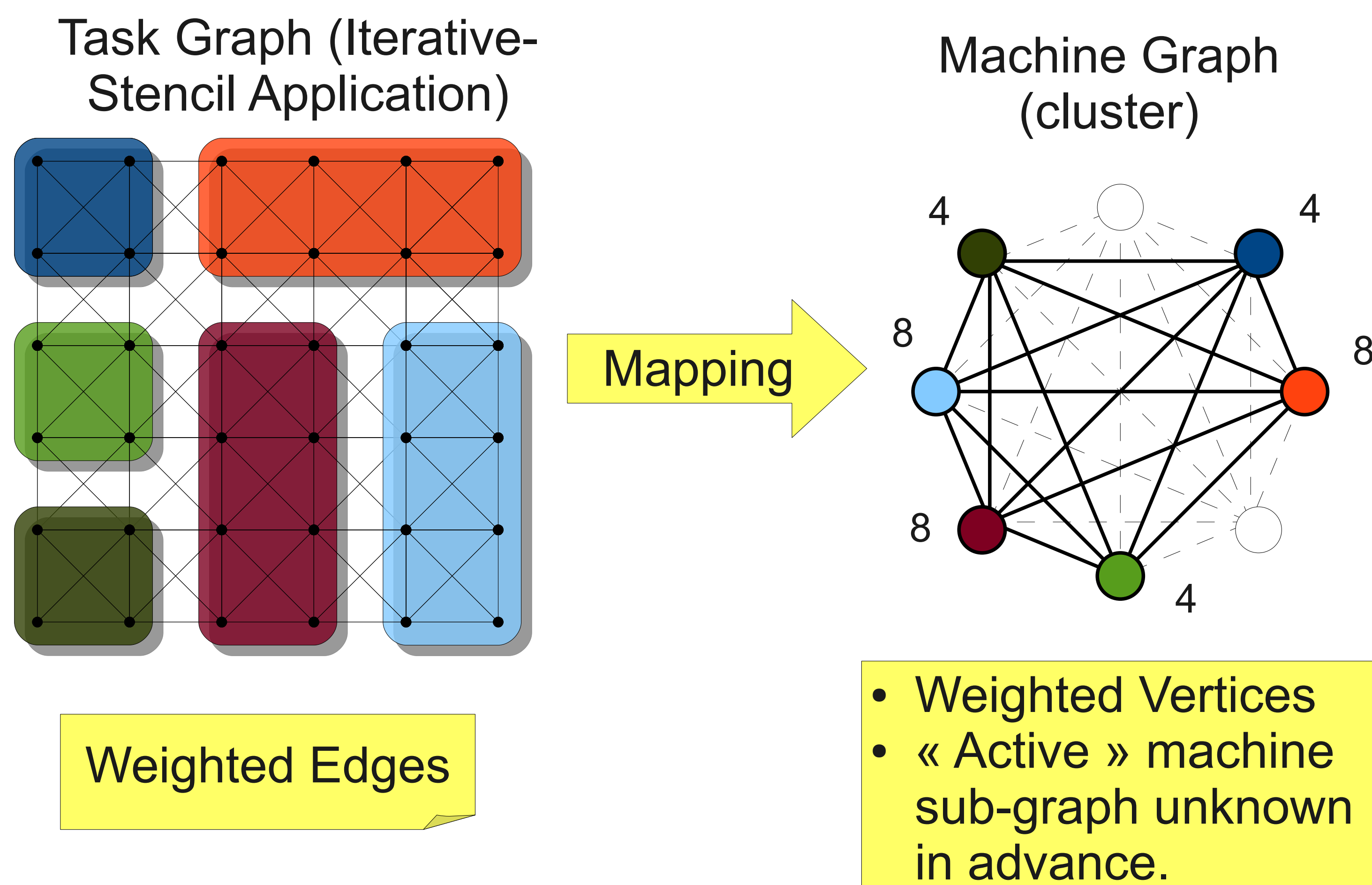
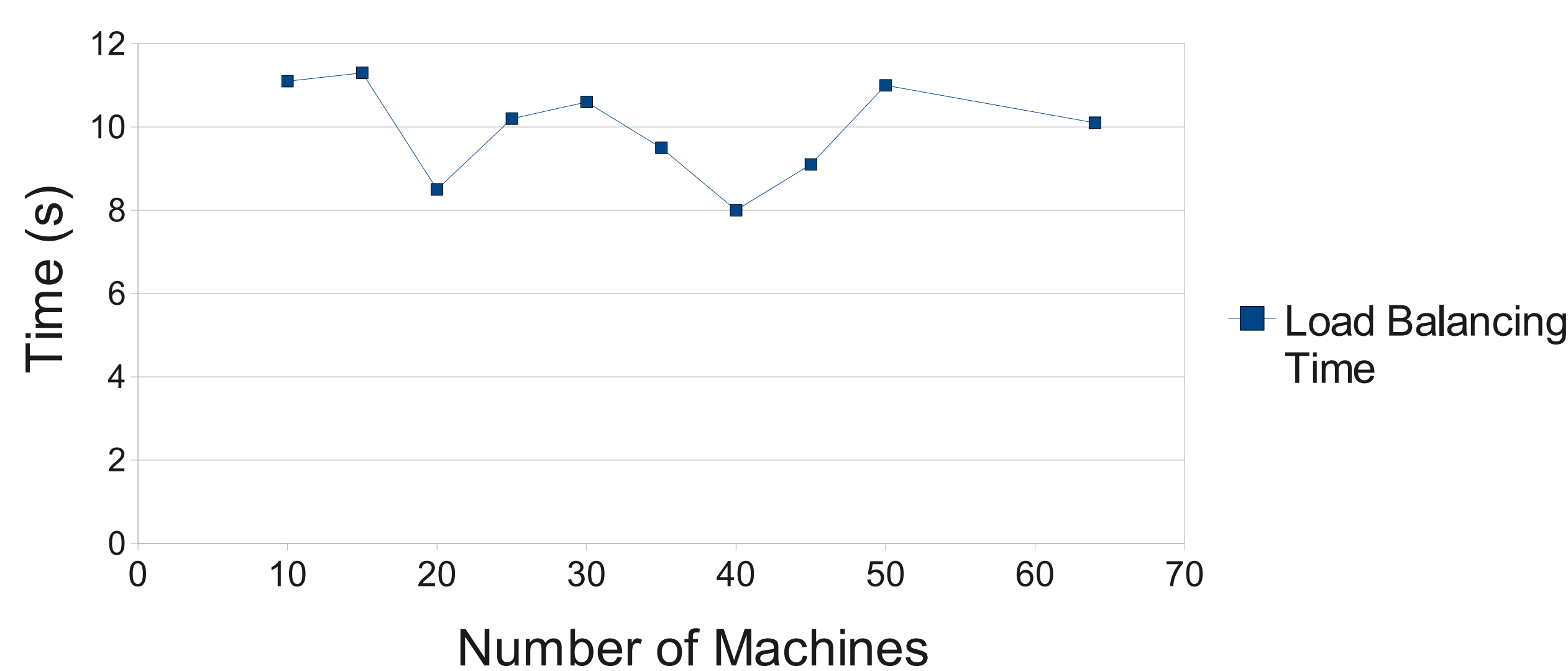


1. Problem Description



3. Results

3.1 Performance



The load balancing time remains roughly constant while the number of machines increases. The time variations are caused by different tree layouts of the machines.

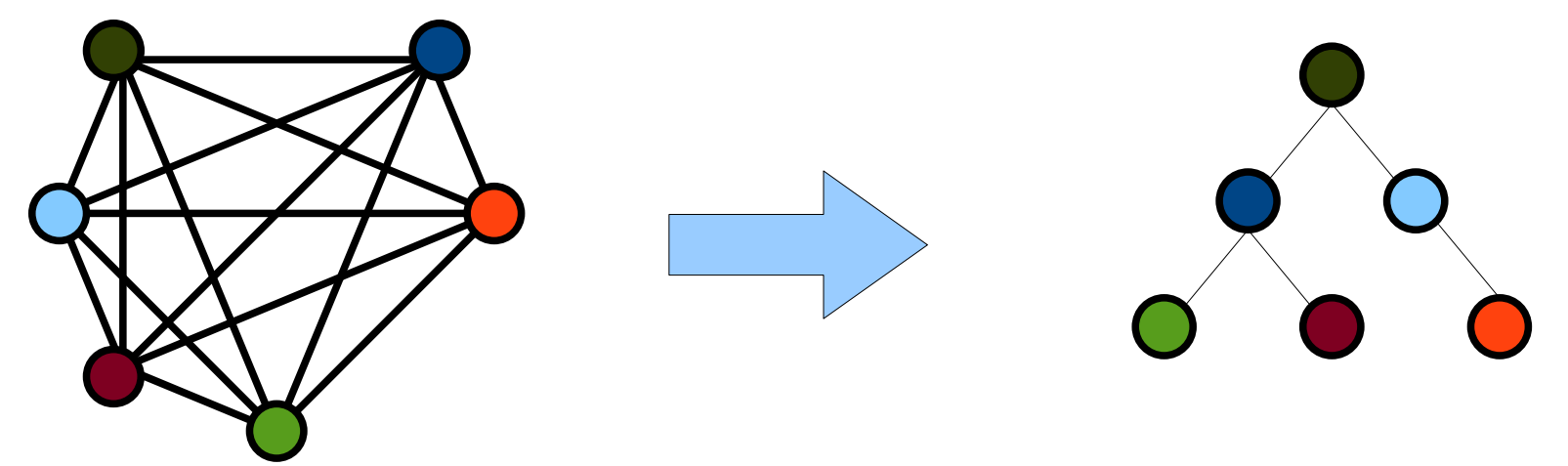
3.2 Partition Quality

- 2 cluster configurations are used
 - C1 = 25 x 2Ghz + 25 x 3Ghz
 - C2 = 25 x 1Ghz + 25 x 3Ghz
- Quality is estimated by comparing the execution time of a specific Iterative-Stencil Application with and without load balancing.
- The execution times given here are obtained using simulations.

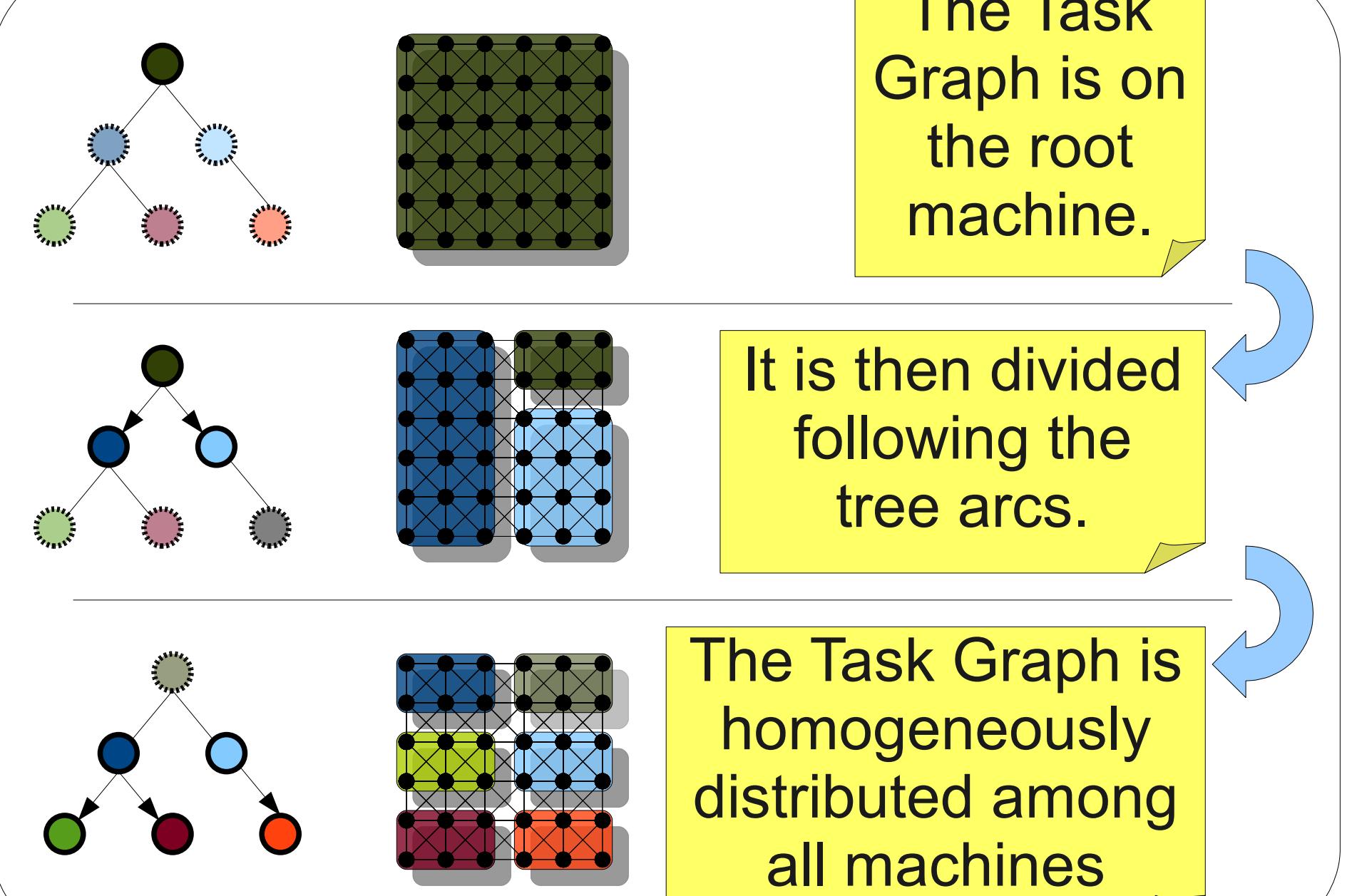
Cluster	Execution time (s)		Gain
	Without Load Balancing	With Load Balancing	
C1	200,22	96,375	52%
C2	217,54	117,71	46%

2. Algorithm

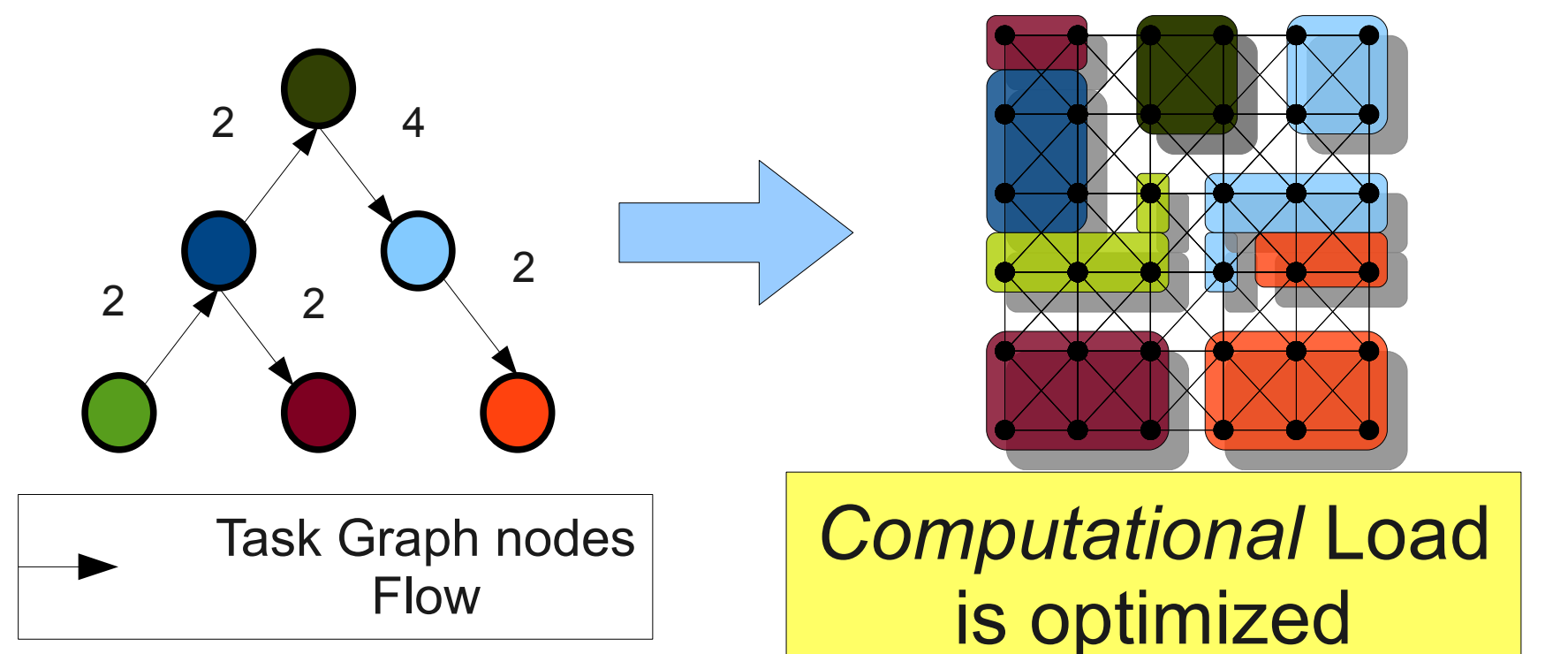
2.1. Tree Self Organization



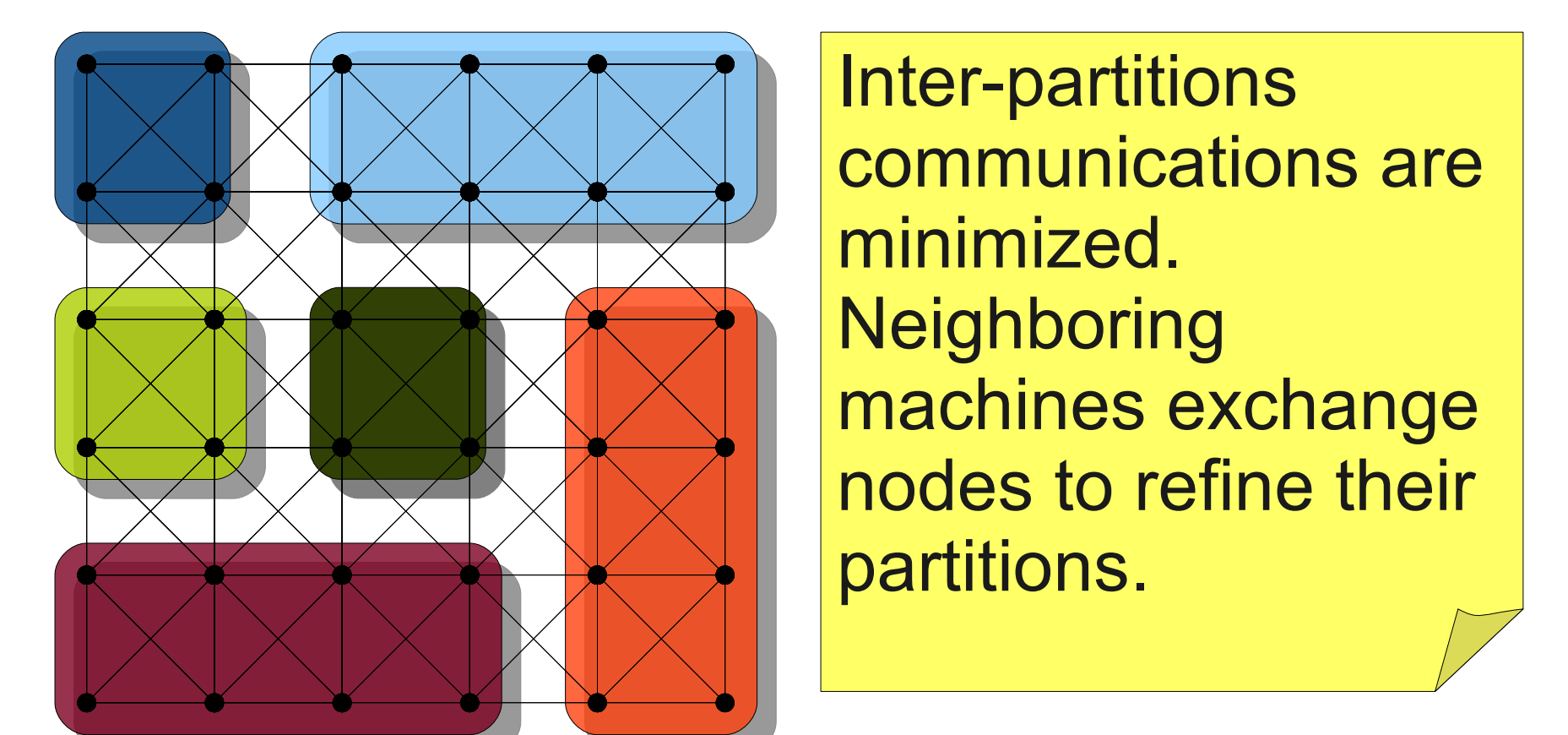
2.2. Initial Data Distribution



2.3. Load Balancing



2.4. Partitions Refinement



4. Conclusion

- A distributed method to balance load among an heterogeneous cluster has been presented.
- Load balancing time is bounded for the number of tested machines. Tests with more machines could prove the scalability of the method.
- The gain in execution time obtained with load balancing is good ; the overhead in execution time caused by load balancing leads to a total execution time well under the execution time without load balancing.