

Parallel Framework for Updating Large Scale Dynamic Networks

Sudharshan Srinivasan¹, Aashish Pandey², Arindam Khanda³, Sriram Srinivasan⁴,⁵ and Sajal K. Das³

¹University of Oregon, ²University of North Texas, ³Missouri University of Science and Technology, ⁴Virginia Commonwealth University, ⁵National Radiation Oncology Program, Department of Veteran Affairs, Richmond, VA

Parallel Algorithms for Updating Large Dynamic Networks

- Analysis of dynamic networks has become an important tool for studying large-scale systems of interacting entities that change over time.

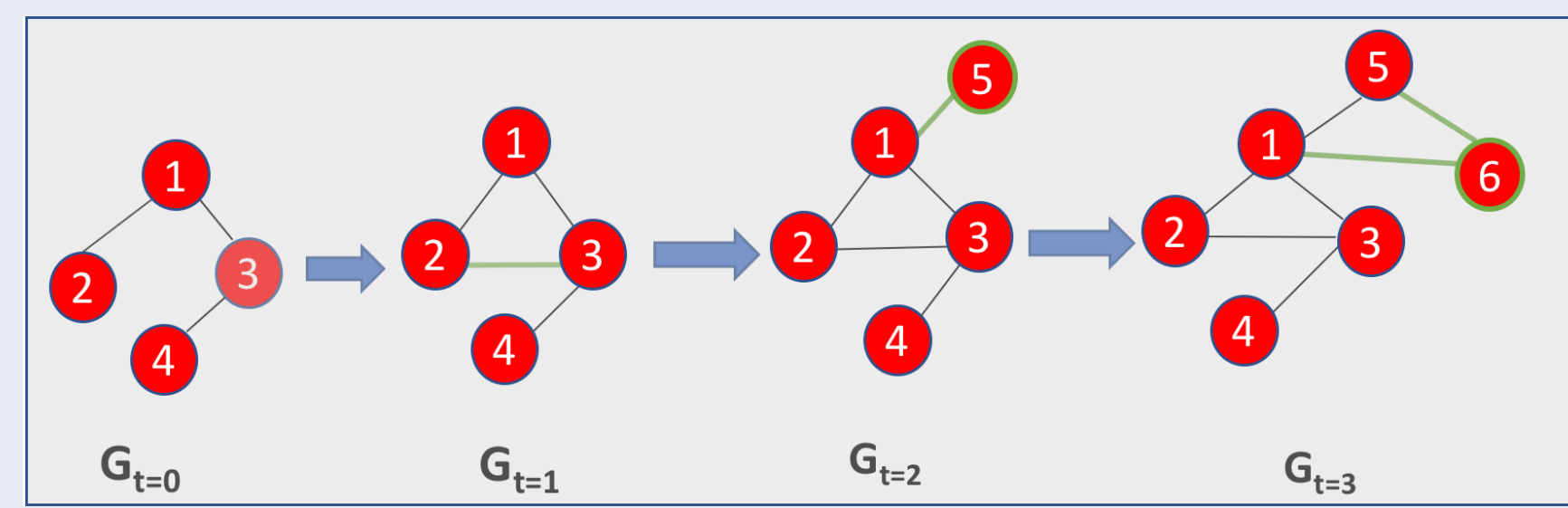


Figure: Changing of graph over time

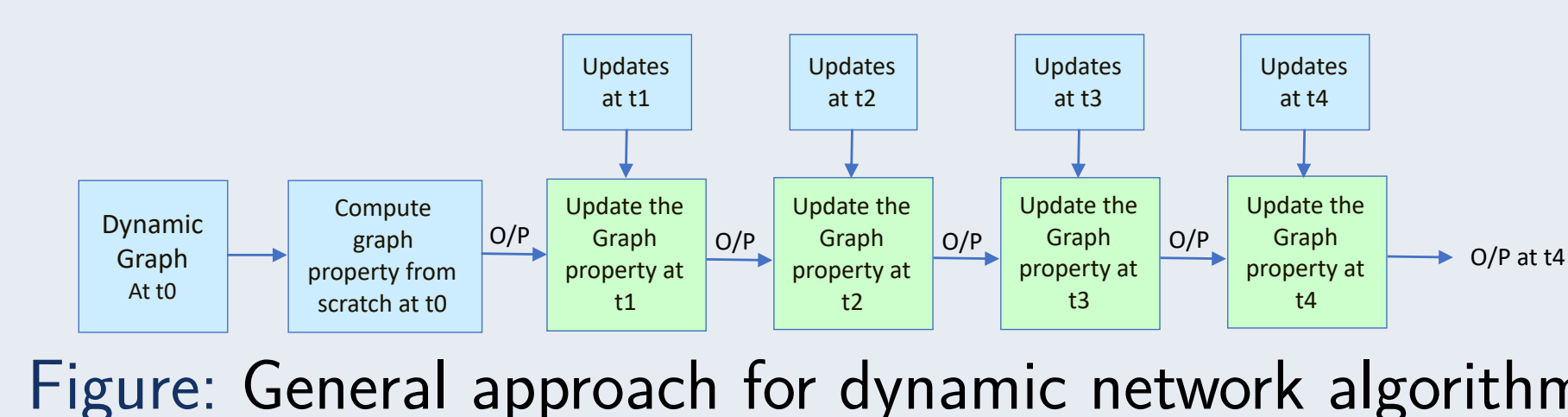


Figure: General approach for dynamic network algorithms

Updating Over the Sparsification Tree

- Graph sparsification** The original network is recursively divided into several small subgraphs over a structure called the sparsification tree.
- Based on the observation that the edges that pertain to the property to be computed(key edges), often form a very small portion of the total edges of the network.
- Updating graph properties can be done by processing only key edges.

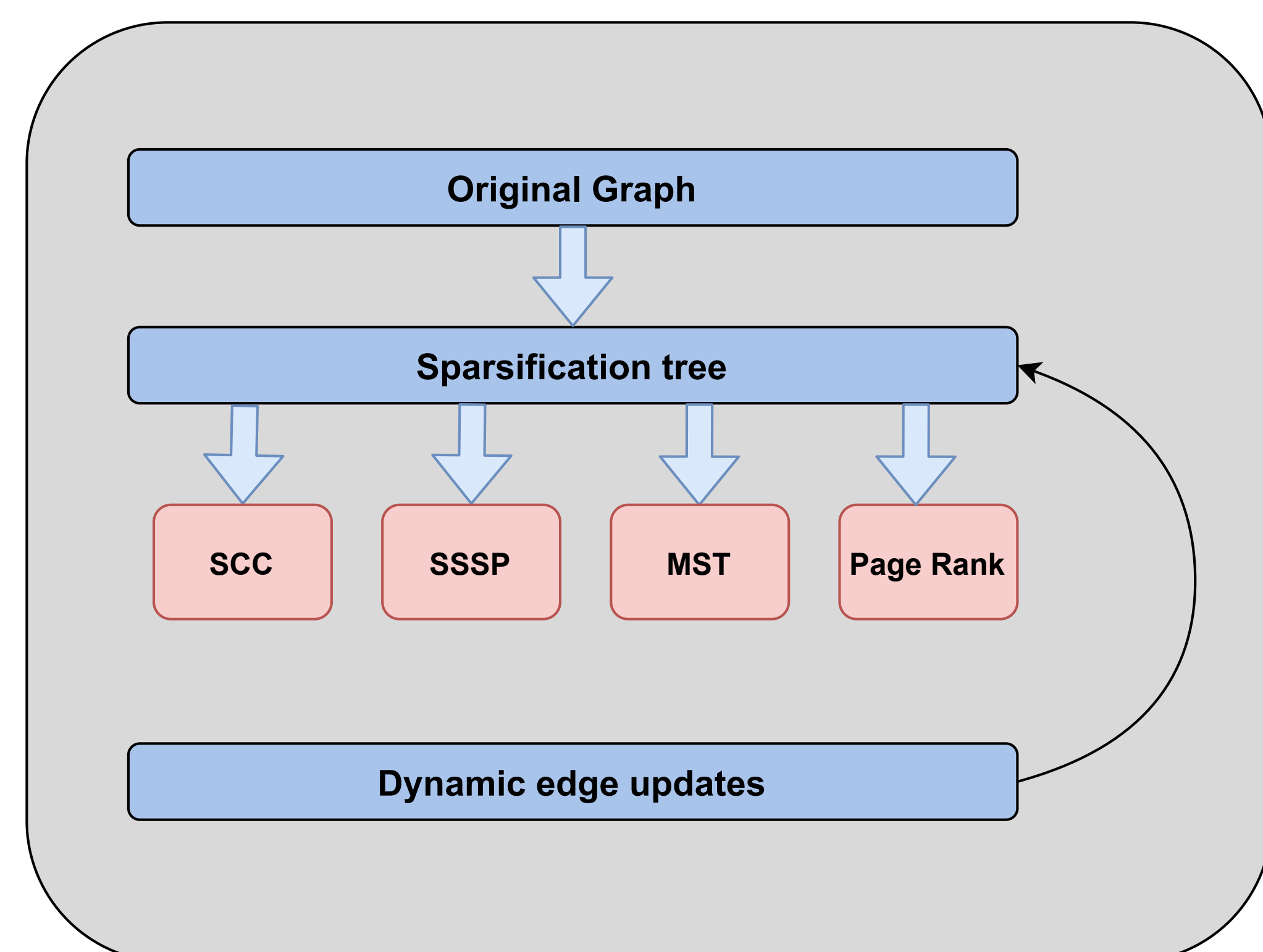


Figure: Overview of the framework

Previous Works and Challenges

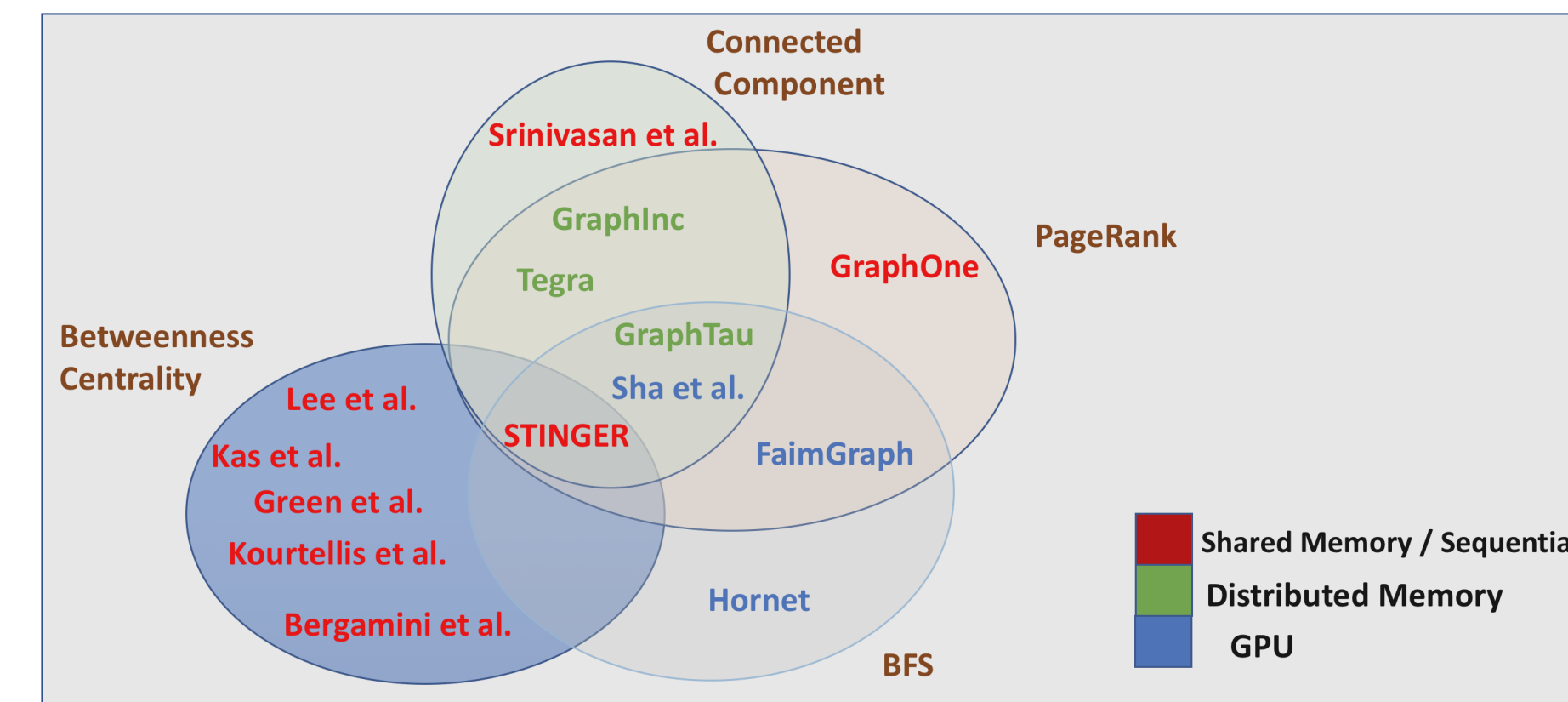


Figure: State of the art works on dynamic graphs.

Challenges

- Designing an efficient algorithm to update with minimal graph traversal.
- Determining the impact of changes and consider which approach is best re-computation or update algorithm.
- Metrics to determine the efficacy of the parallel dynamic algorithms such as TEPS, speedup when compared to its recompute counterpart.

Graph Properties

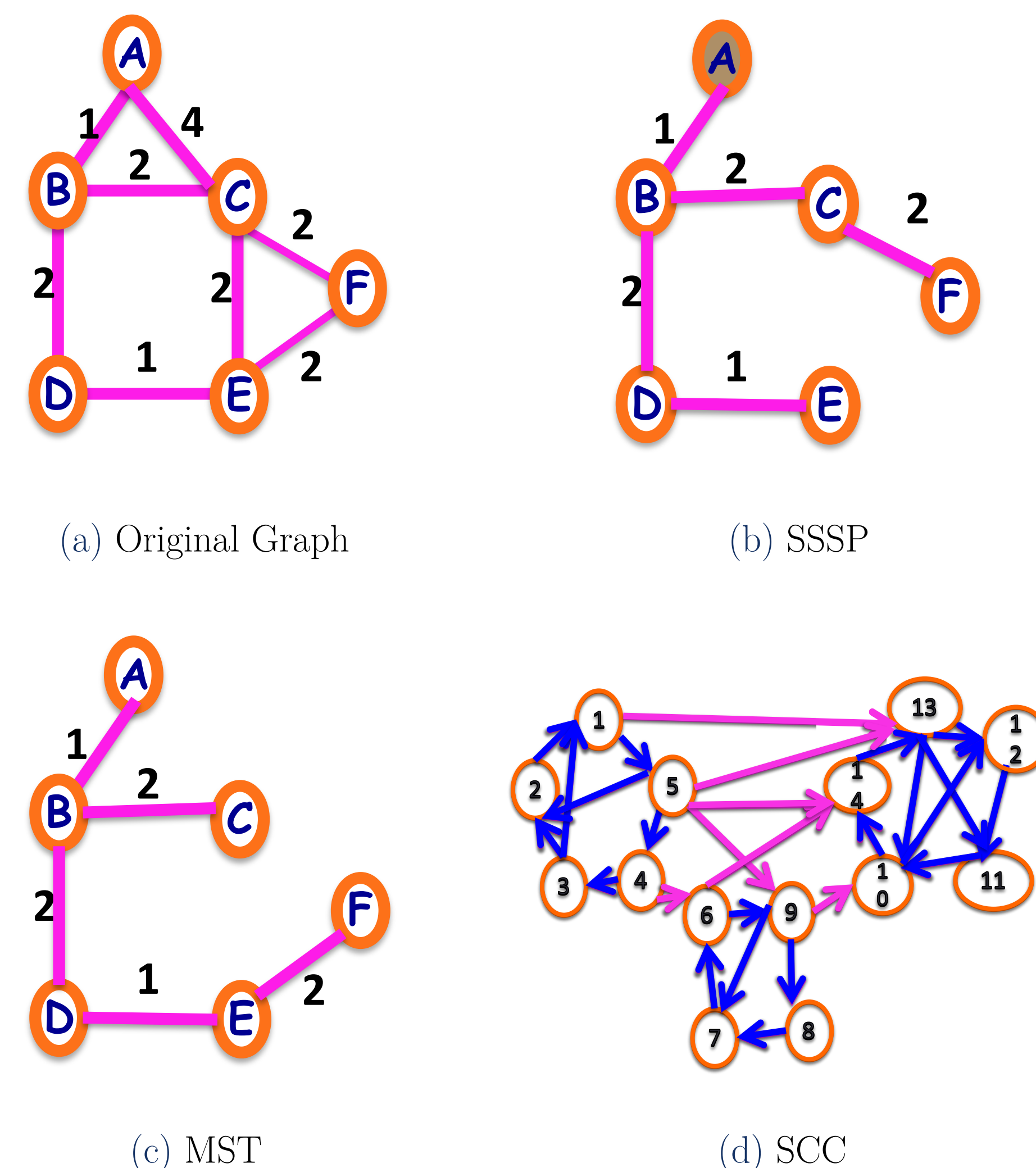


Figure: Example of common graph properties

Detecting SCCs on Dynamic Networks

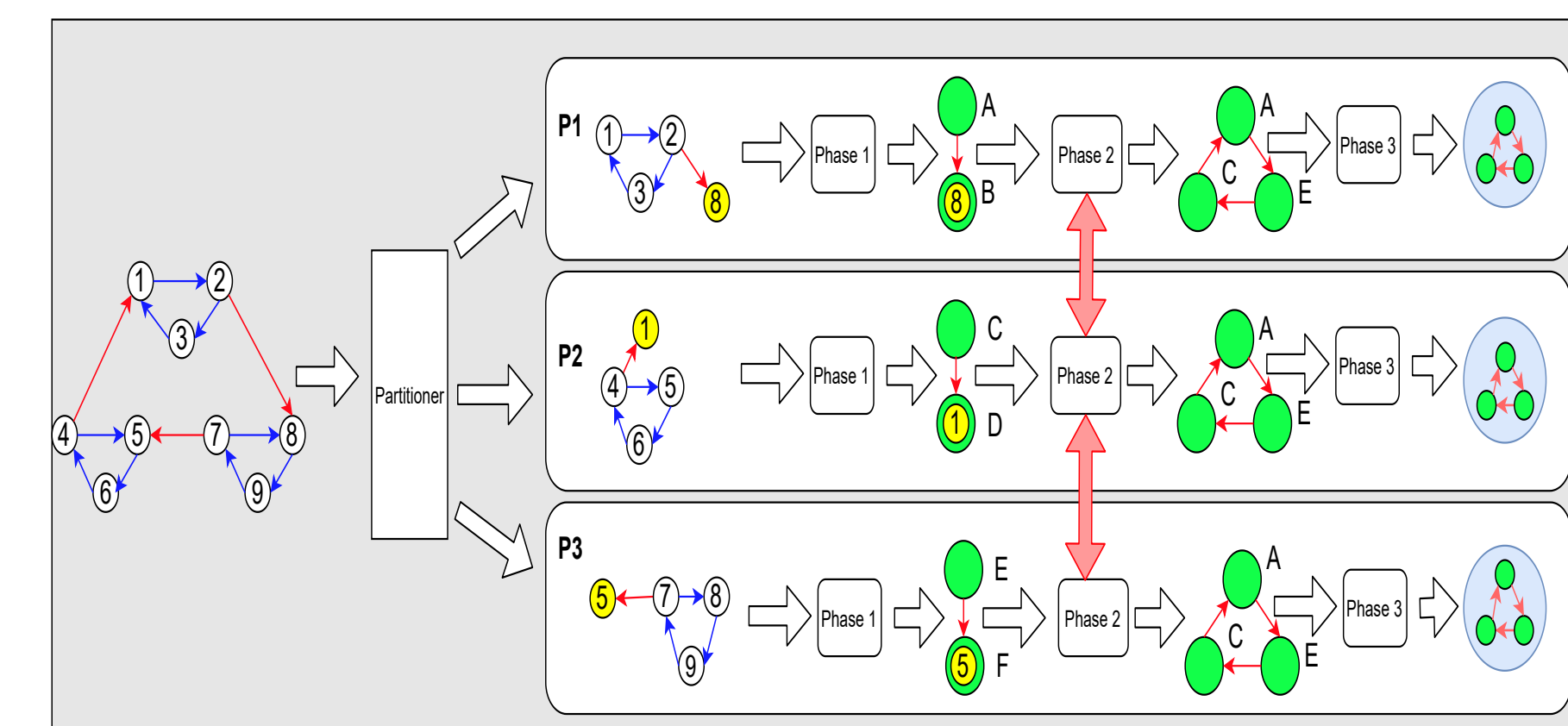


Figure: Workflow of SCC algorithm

- Graph is K-way partitioned across K processes.
- Each partition creates multiple meta-graphs with local SCCs as vertices and connections between SCCs as edges during phase one.
- SCC is locally recomputed on the meta-graphs during phase three.
- SCC computation is done over reduced-sized meta-graphs, while communication is done over updated edges.
- Recomputation for dynamic networks are isolated to new batch of edges along with border vertices of meta-nodes.

Results

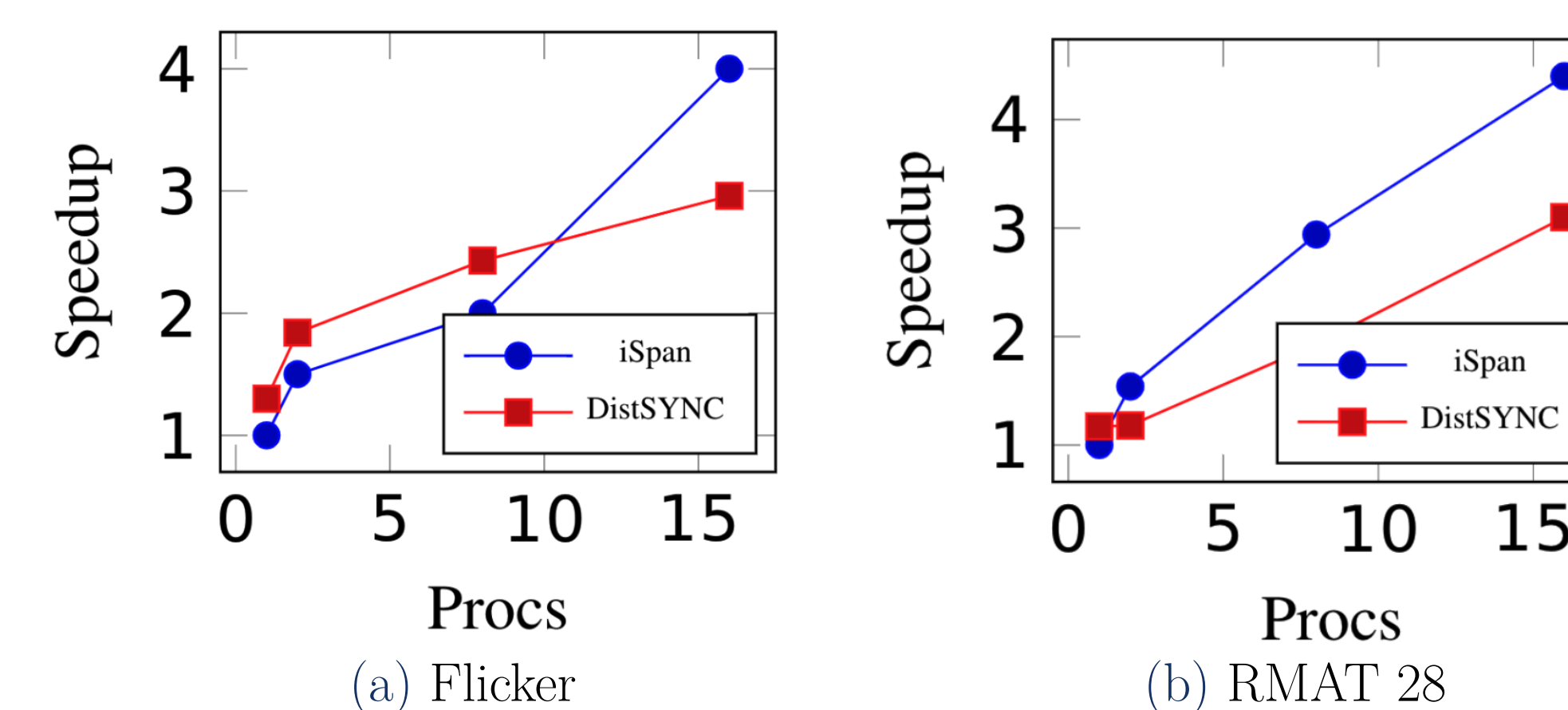


Figure: Strong scaling results for our SCC framework (red) and baseline iSpan (blue) with respect to single threaded Tarjan's implementation.

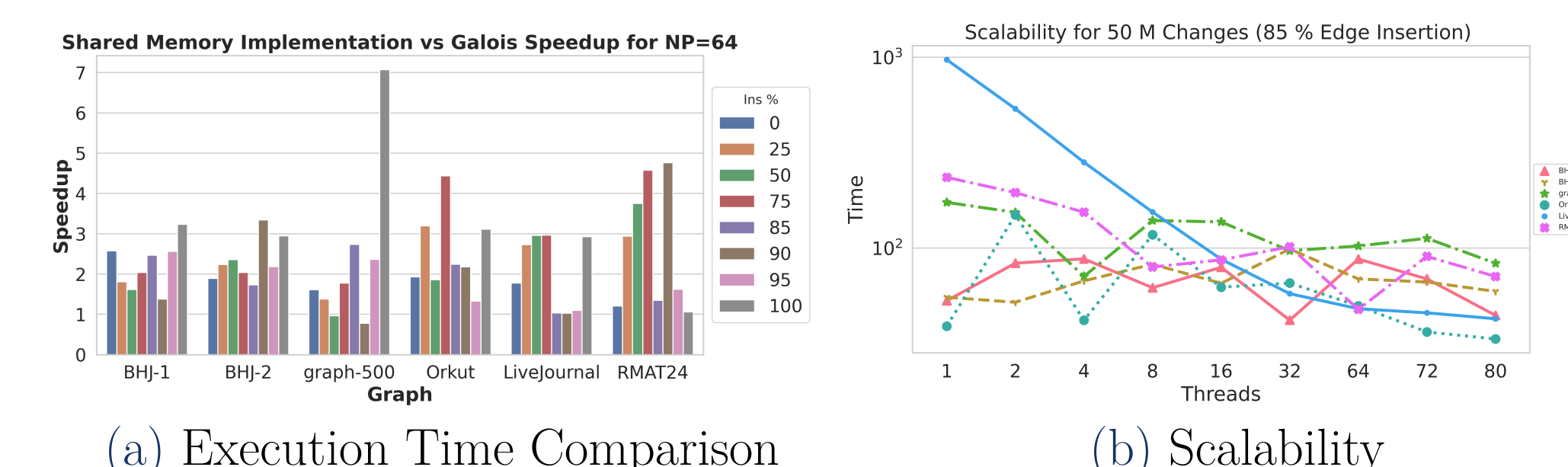


Figure: Scalability plot for SSSP implementation and speedup comparison with baseline

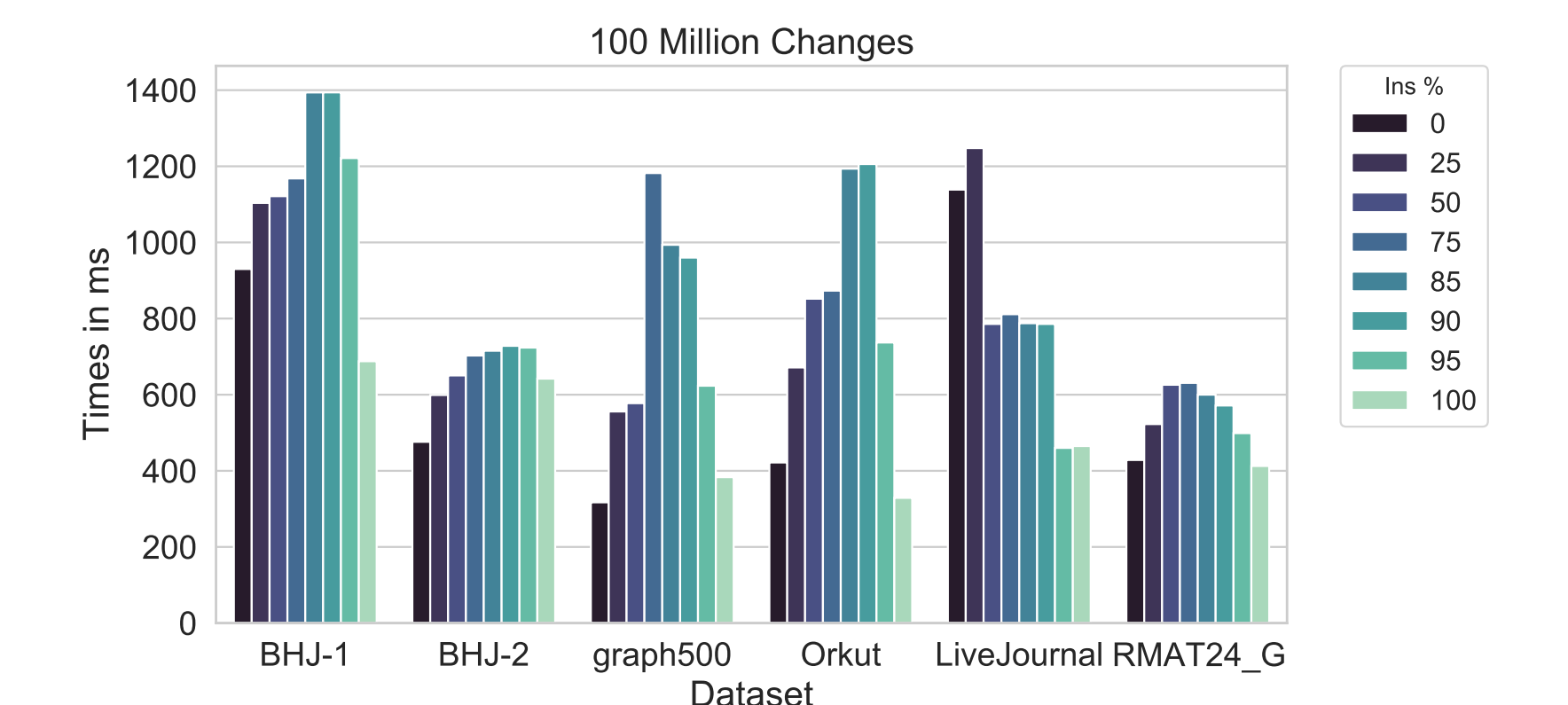


Figure: GPU implementation for updating SSSP on 50M changes

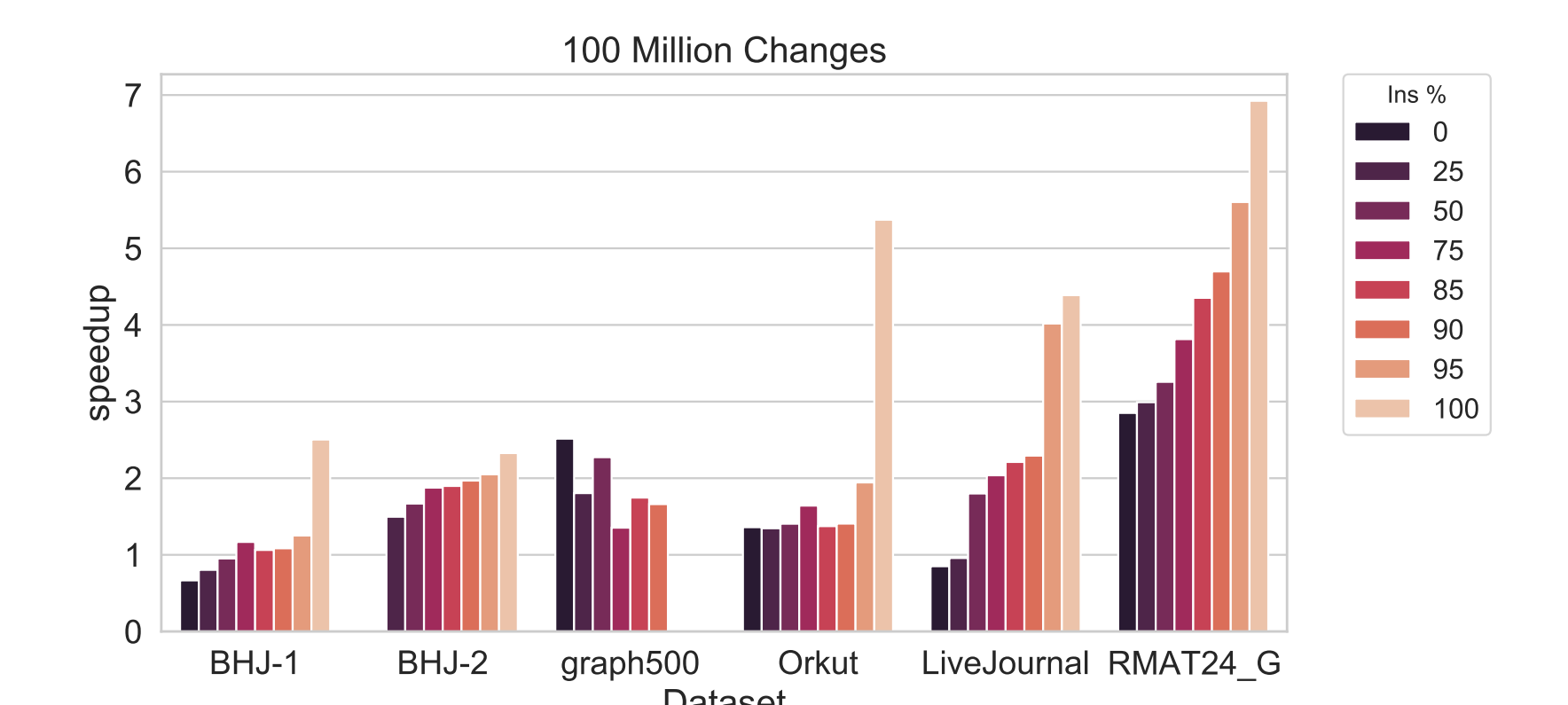


Figure: Speed up comparison for GPU implementation of updating SSSP vs GPU implementation of recomputing SSSP from Scratch using Gunrock

Conclusion

- Parallel implementation of graph sparsification which is used to divide the graphs into subgraphs and used to detect which portion of the network is affected due to the changes
- New shared-memory algorithm for updating SSSP, MST, Page Rank, and SCC in dynamic networks.
- Distributed algorithm for updating SCC in dynamic networks.

References

- Ediger, D., McColl, R., Riedy, J., Bader, D. A. (2012, September). Stinger: High performance data structure for streaming graphs. In 2012 IEEE Conference on High Performance Extreme Computing (pp. 1-5). IEEE.
- Wang, Y., Pan, Y., Davidson, A., Wu, Y., Yang, C., Wang, L., ... Owens, J. D. (2017). Gunrock: GPU graph analytics. ACM Transactions on Parallel Computing (TOPC), 4(1), 1-49.
- Khanda, A., Srinivasan, S., Bhowmick, S., Norris, B., Das, S. K. (2021). A Parallel Algorithm Template for Updating Single-Source Shortest Paths in Large-Scale Dynamic Networks. IEEE Transactions on Parallel and Distributed Systems.
- Srinivasan, S., Pollard, S., Das, S. K., Norris, B., Bhowmick, S. (2018). A shared-memory algorithm for updating tree-based properties of large dynamic networks. IEEE Transactions on Big Data.

Acknowledgment

This research was supported by the NSF OAC grants 2104078 and 1725755.