CANDY: An Efficient Framework for Updating Properties on Large Scale Dynamic Networks
Aashish Pandey¹, Arindam Khandak², Sriram Srinivasan³, Sanjukta Bhowmick¹, Sajal K. Das², and Boyana Norris³
¹University of North Texas, ²Missouri University of Science and Technology, ³University of Oregon

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.
- Support parallel dynamic network algorithm development on distributed memory, shared memory, and GPUs, and their use through user-friendly interfaces.
- Comprehensive cyberinfrastructure supporting innovative research challenges in large-scale, complex, dynamic networks.

CANDY Architecture

- CANDY: a parallel, scalable, extendable, and user-friendly software platform for updating important properties of dynamic networks.
- Support parallel dynamic network algorithm development on distributed memory, shared memory, and GPUs, and their use through user-friendly interfaces.
- Comprehensive cyberinfrastructure supporting innovative research challenges in large-scale, complex, dynamic networks.

Challenges and Intellectual Merits

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.
- Creating parallel graph computation software infrastructure for modern heterogeneous architectures.

Intellectual Merits
- Includes templates for creating new scalable, parallel algorithms for dynamic network analysis.
- Provides functionality and tools to create new algorithms or modify existing ones, catering to users with varying expertise.
- Provides algorithms to partition streaming sets of nodes and edges into network snapshots at changing points.

Graph Properties

- High PageRank (PR) nodes have a cascading effect on a large number of remaining nodes.
- Adaptive Switch evaluates the total percentage of affected and high PR nodes.
- Adaptive Switch decides if updating property will be more efficient than property recomputation.
- Our update algorithm uses the previous property and iteratively computes the updated property in the affected subgraph.

We empirically tested our adaptive implementation on real-world and synthetic networks and based on our observation we saw if the changes impact above 50% of nodes with high PR, the overall nodes affected are always above 95% of the entire network.

When we compare our implementation with the recomputation approach if the changes impact less than 50% of nodes with high PR values update algorithm outperforms the recomputation approach.

Results

- Presented an adaptive template for creating new scalable, parallel algorithms for dynamic network property updates.

References


Acknowledgment

This work was partially supported by the NSF projects SANDY (Award # OAC-1727575) and CANDY (Award # OAC-2104115, 2104070, 2104078).