Improved Massively Parallel Computation Algorithms for MIS, Matching, and Vertex Cover

Themis Gouleakis

MIT

June 14, 2018

Joint work with: Mohsen Ghaffari (ETH), Christian Konrad (University of Bristol), Slobodan Mitrović (EPFL) and Ronitt Rubinfeld (MIT and Tel Aviv University)
CONGESTED CLIQUE model [Lotker Pavlov Patt-Shamir Peleg 03]

- Complete communication graph: $G' = K_{|V|}$.
- Synchronous messages of size $O(\log n)$ bits
- complexity=$\#$ rounds.
Massively Parallel Computation (MPC) model

[Karloff Suri Vassilvitskii 10]

- Inspired by MapReduce
- Input size=$N$
- $m$ machines
- Space $S \in \left[ \frac{N}{m}, N \right]$ per machine
- unbounded internal computation
- Total communication per node bounded by $S$ each round.
- complexity=$\#$ rounds.
RESULTS

**Theorem:** There is an algorithm that with high probability computes an MIS in $O(\log \log \Delta)$ rounds of the MPC model, with $\tilde{O}(n)$-bits of memory per machine.

**Theorem:** There is an algorithm that with high probability computes a $(2 + \varepsilon)$-approximate integral maximum matching and a $(2 + \varepsilon)$-approximate integral minimum vertex cover in $O(\log \log n)$ rounds of the MPC model, with $\tilde{O}(n)$-bits of memory per machine.
Improved Massively Parallel Computation Algorithms for MIS, Matching, and Vertex Cover

Mohsen Ghaffari\textsuperscript{1}, Themis Gouleakis\textsuperscript{2}, Christian Konrad\textsuperscript{3}, Slobodan Mitrovic\textsuperscript{4} and Ronitt Rubinfeld\textsuperscript{2,5}

ETH\textsuperscript{1}, MIT\textsuperscript{2}, University of Bristol\textsuperscript{3}, EPFL\textsuperscript{4}, Tel Aviv University\textsuperscript{5}

Models

CONGESTED CLIQUE model
[Loekh Pavlov Patt-Shamir Peleg 03]

- Complete communication graph: \( G' = K_{|V|} \)
- Synchronous messages of size \( O(\log n) \) bits
- Complexity: \( \leq \) rounds

Massively Parallel Computation model
[Karloff Suri Vassilvitskii 10]

- Input size: \( N \)
- \( m \) machines
- Space \( S = \lceil \frac{\log N}{m} \rceil \) per machine
- Total communication per node per round: \( \leq S \)
- Complexity: \( \leq \) rounds

Greedy randomized Algorithm for MIS

Greedy Randomized Maximal Independent Set
- Choose a permutation \( \pi : [n] \rightarrow [n] \) u.a.r.
- Repeat until the next rank is at least \( n/\log^{10} n \) and the maximum degree is at most \( \log^{10} n \):
  - Add smallest rank vertex \( v \) to the MIS.
  - Remove all neighbors of \( v \).
  - Run \( O(\log \log \Delta) \) rounds of the Sparsiﬁed MIS Algorithm of [Ghaffari 17] in the remaining graph. Remove from the graph the constructed MIS and its neighborhood.
  - Find MIS of the residual graph (single machine)
  - Output the union of the constructed MIS sets.

MPC simulation

- Choose a permutation \( \pi : [n] \rightarrow [n] \) u.a.r.
- Repeat for each chunk \( V_i \):
  - Machine \( \Pi \) receives all \( G(V_i) \) edges.
  - Broadcasts MIS(\( G(V_i) \))
  - Update local memories

Until: \( \Delta < \log^{10} n \)
- Run \( O(\log \log \Delta) \) rounds of sparsiﬁed MIS algorithm [Ghaffari 17]

CONGESTED CLIQUE vs MPC

MPC memory: \( S = O(n) \)

- \textbf{CONGESTED CLIQUE} \( \Rightarrow \textbf{MPC} \) [BDH18]
  - (1 MPC machine per graph node)
  - Assign each MPC machine to a vertex.
  - For each edge \((u,v)\), send message to machines \( u,v \).
  - \textbf{Simulate} \textbf{CONGESTED CLIQUE} algorithm (\(< n \) MPC machines: assign multiple vertices to each one)

- \textbf{MPC} \( \Rightarrow \textbf{CONGESTED CLIQUE} \)
  - Each node needs to send/receive up to \( O(n) \) bits.
  - Split messages into \( O(\log n) \) bit chunks.
  - Exploit unused edges of the communication graph using routing scheme in [Lemma13]
  - We can simulate one MPC round using \( O(1) \)-congested clique rounds.

Theorem (inf.) [GGKMR18, [BDH18]

\textbf{CONGESTED CLIQUE} \( \approx \) \textbf{MPC} model with \( S = O(n) \)

Analysis

**Lemma:** Let \( G_r \) be the remaining graph after \( r \) vertices are simulated. Then \( \Delta_r = O(\frac{\log n}{\log \log n}) \)

- \( O(n) \) edges sent to the central machine each round w.h.p.
- 1st round: For \( \alpha < 1/2 \), we get:
  \( P_r[\{i,j\} \in G_r] = \left(\frac{1}{2}\right)^{\alpha} \leq 2^{-\log \log n} \)
- \( k \)-th round:
  \( E[|\text{edges}|] = \Delta_r \left(\frac{1}{2}\right)^k \leq O(n) \)
- After \( \tau = O(\log \log \Delta) \) rounds the max degree \( \Delta_r \leq \log^{10} n \)
- **Key idea:** High degree vertices are much more likely to be removed!

Results

**Theorem:** There is an algorithm that with high probability computes an MIS in \( O(\log \log \Delta) \) rounds of the MPC model, with \( O(n) \)-bits of memory per machine.

**Theorem:** There is an algorithm that with high probability computes a \( (2 + c) \)-approximate integral maximum matching and a \( (2 + c) \)-approximate integral minimum vertex cover in \( O(\log \log n) \) rounds of the MPC model, with \( O(n) \)-bits of memory per machine.

References

Brief announcement: Semi-mapreduce meets congested clique.
Distributed mis via all-to-all communication.
A model of computation for mapreduce.
Optimal deterministic routing and sorting on the congested clique.
MST construction in \( O(\log \log n) \) communication rounds.