Checkpoint 2

Hamza Abdullah - ha07194

1 Introduction and Problem Statement

In many distributed systems, information must be broadcast or multicast efficiently over a network where each informed node can inform at most one neighbor per round. The *Telephone Model* captures this: given a directed graph $G = (V, E)$, a root $r \in V$ initially holds a message, and in each synchronous round, informed vertices send the message along one outgoing edge to uninformed neighbors.

The k-*Multicast* problem (k-**MTM**) asks: given root r, terminal set $S \subset V$ of size t, and target $k \leq t$, find a schedule to inform any k terminals in the fewest rounds. This generalizes broadcast ($k = |V|$) and multicast ($k = |S|$) in directed and undirected graphs.

2 Main Contributions

1. **Directed k-MTM: Additive $\tilde{O}(\sqrt{k})$ Approximation.** Theorem 1.2 presents a polynomial-time algorithm with additive $O(\sqrt{k}\,\text{polylog}(k))$ guarantee. Key steps:
   - *Greedy Packing:* Extract disjoint "good" trees of size $\lceil \sqrt{k} \rceil$ via BFS to depth D^*.
   - *Partition Matroid Cover (PMCover):* Formulate covering remaining k as submodular maximization under a partition matroid (degree budgets B^*); implement via a simple half-approximation or the $(1 - 1/e)$-approx routine.
   - *Complete:* Merge partial trees and selected cover edges, then extract a shortest-path tree spanning k terminals.
2. **Undirected k-MTM: Multiplicative $\tilde{O}(t^{1/3})$ Approximation.** Theorem 1.3 achieves $O(t^{1/3}\,\text{polylog}(t))$ approximation by:
   - *Small-Tree Packing:* Greedily find disjoint trees of size $t^{1/3}$, contract each to a super-terminal.
   - *Alternation:* Either locate a "large" tree covering $t^{1/3}$ super-terminals, or apply PMCover and discard covered terminals, repeating $O(t^{1/3})$ steps.
3. **Novel Use of Matroid-Constrained Submodular Coverage.** First application of partition-matroid submodular maximization in telephone-multicast scheduling, allowing finer degree control in directed k-MTM.
4. **Bridging the Approximability Gap.** Complements prior $O(\sqrt{t})$-additive and $O(\log t / \log \log t)$ bounds by tuning with parameter k.

3 Directed-Case Algorithm Breakdown

3.1 Greedy Packing of $\sqrt{k}$-Good Trees

We maintain $A = \{r\}$, $C = V \setminus \{r\}$. While there exists $c \in C$ whose BFS tree to depth D^* covers $\geq \sqrt{k}$ terminals, extract a subtree of exactly $\lceil \sqrt{k} \rceil$ leaves, add its vertices to A, and remove from C.

3.2 Handling Many vs. Few Trees

Many Trees ($\geq \sqrt{k}$): Select any $\sqrt{k}$ trees, connect each root to r via a shortest path, and take a shortest-path tree of the union. Guarantees max-degree $\leq 2\sqrt{k}$ and height $\leq 2D^*$.

Few Trees ($< \sqrt{k}$): Use PMCover on the additive partition (A, C) to cover remaining terminals, then Complete to build the final tree with additive $O(\log k\, B^*)$ degree overhead.

3.3 Partition Matroid Cover (PMCover)

Ground set: uncovered terminals in C. Sets: $S_{a,c} = (a, c)$ covering terminals within depth D^* of c. Matroid: allow $\leq B^*$ sets per a. Greedy 1/2-approx picks highest-marginal-gain sets until k covered.

3.4 Complexity and Guarantees

Runs in $\tilde{O}(m\sqrt{k} + k \cdot m_{\text{cover}})$, achieving an additive $O(\sqrt{k} + \log k\, B^*)$ degree bound and $O(D^*)$ height, i.e. additive $\tilde{O}(\sqrt{k})$ rounds above optimal.

4 Comparison to Prior Work and Key Data Structures

4.1 Relation to Previous Algorithms

- Elkin–Kortsarz's $O(\sqrt{t})$-additive algorithm for directed MTM [EK06a] specializes to $k = t$; we improve to $k \ll t$.
- EK06a's multiple set-cover differs from our matroid-constrained coverage, giving tighter degree control.
- Undirected gap: from $O(\log t / \log\log t)$ to $\tilde{O}(t^{1/3})$ via subtree contraction strategies.

4.2 Key Data Structures

- *Coverage Trees:* BFS trees to depth D^* for each $c \in C$.
- *Partition Matroid:* Maps each $a \in A$ to its incident cover-sets; per-a counters enforce budgets.

- *Greedy Cover Heap:* Max-heap on marginal gains updates in $O(\log |C|)$ per pick.
- *Shortest-Path Extraction:* Dijkstra/BFS on final union graph.

4.3 Implementation Challenges

- Maintaining vertex-disjointness in greedy packing.
- Optimizing marginal-gain updates in PMCover.
- Estimating D^* via binary search or baseline broadcast.

5 Experimental Design and Evaluation Metrics

5.1 Graph Datasets

Directed ER ($n = 100, 500, 1000, 5000$, $p = 0.05$) and directed power-law (Barabási–Albert, $m = 3$), with $t = 0.1n$ terminals and $k = \{0.2t, 0.5t, 0.8t\}$.

5.2 Baseline Algorithms

SPT broadcast, directed-MST broadcast, and greedy matching per round.

5.3 Metrics

Rounds to inform k, tree poise (max-degree+height), and algorithm runtime.

5.4 Visualization

Coverage curves (informed vs. round), poise vs. $\sqrt{k}$, runtime vs. n comparing half-approx vs. $(1 - 1/e)$ PMCover.

5.5 Reproducibility

Unit tests on toy graphs ($n \leq 10$), modular code organization, and environment captured in `requirements.txt`.