Joint Scheduling and Multiflow Maximization in Wireless Networks

Yanxiao Liu[1], Shenghao Yang[2] and Cheuk Ting Li[3]

[1] Imperial College London
[2] The Chinese University of Hong Kong, Shenzhen
[3] The Chinese University of Hong Kong

2026 International Zurich Seminar on Information and Communication

February 2026

This work was done when Yanxiao Liu was with The Chinese University of Hong Kong

Maximum Multiflow Problem: Background

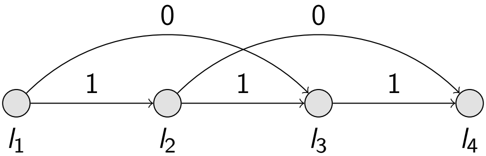
- We consider the **maximum multiflow** problem.
- To maximize the total (or concurrent) flow values between multiple sources and sinks, we usually have two steps:
  1. Model the wireless interference as a conflict graph to compute the scheduling rates (Jain et al., 2005);
  2. Find the maximum throughput supported by achievable scheduling rate.
- However, the first step is NP-hard and also difficult to approximate (Baker, 1994).
- The MMF problem is NP-hard even in simple settings (Wan, 2009).
- Existing joint optimization methods usually focus on either *approximate* solutions or *resitricted* networks.

Maximum Multiflow Problem: Our Contribution

- We investigate the exact-optimal (rather than approximate) solutions for general multi-hop networks.
- We *jointly* calculate the maximum multiflow and the scheduling rates by employing a *decomposition* method.
- Usually only a small subset of the scheduling rate region is needed.
- We prove the output solution is optimal.
- We consider the most general setting: *multiple multicast* with network coding (Ahlswede et al., 2000) allowed.

Network Model

- The network is modeled by $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{I}, D)$, where
  - $\mathcal{V}$ is the *node set*;
  - $\mathcal{L} \subseteq \mathcal{V}^2$ is the *link set*; each link l is associated with a *collision set* $\mathcal{I}(l)$;
  - $D : \mathcal{L}^2 \to \mathbb{Z}$ is a link-wise delay matrix.
  - We assume each link has a unit bandwidth and allow parallel links between nodes.
- $(\mathcal{L}, \mathcal{I}, D)$ can form a weighted, directed graph $\mathcal{N}$ where (l, l') is an edge if $l' \in \mathcal{I}(l)$, and $D(l, l')$ is the weight.
- For example, for a 4-link line network with unit delay and 1-hop interference model, we can draw:



Scheduling Rate Region

- Assuming time is slotted, when link l is active at time slot t and link $l' \in \mathcal{I}(l)$ is active at time slot $t + D(l, l')$, a *collision occurs*.
- We use a binary matrix $S : \mathcal{L} \times \mathbb{N} \to \{0, 1\}$ to specify a *schedule*: $S(l, t) = 1$ indicates that link l is active in time slot t.
- $S(l, t)$ has collision if $S(l, t) = S(l', t + D(l, l')) = 1$ for some $l' \in \mathcal{I}(l)$.
- S is collision free if $S(l, t)$ is collision free for all (l, t).
- For a collision-free schedule S and a link l, the link rate is

$$R_S(l) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} S(l, t). \tag{1}$$

- $R_S = (R_S(l), l \in \mathcal{L})$ is the *rate vector* of S, and it is *achievable* if there is a collision-free schedule achieves S it.
- The collection $\mathcal{R}(\mathcal{N})$ of all the achievable rate vectors is called the *scheduling rate region*.

Problem Setting

- We now define the maximum multiflow (MMF) problem.
- We say $F = (F(l) \in \mathbb{N}_{\geq 0} : l \in \mathcal{L})$ is a valid flow from source $s \in \mathcal{V}$ to sink $t \in \mathcal{V}$ w.r.t. R if
  - $0 \leq F(l) \leq R(l)$ for all $l \in \mathcal{L}$;
  - $\sum_{l \in \mathrm{In}(v)} F(l) = \sum_{l \in \mathrm{Out}(v)} F(l)$ for all $v \in \mathcal{V} \setminus \{s, t\}$.
- Flow $\sum_{l \in \mathrm{Out}(s)} F(l)$ out of s equals to $\sum_{l \in \mathrm{In}(t)} F(l)$ into t, and it is called the *value* of F, denoted by $\mathrm{val}(F)$.
- **MMF problem**: Consider a source s_i multicasts a message to nodes in the set $\mathcal{D}_{s_i} = \{t_{i,1}, \ldots, t_{i,k_i}\}$ (with network coding) at a rate

$$v_i = \min_j \mathrm{val}(F_{i,j}),$$

  where $F_{i,j}$ is a flow from s_i to $t_{i,j}$ for each j.
- Fix rate vector R. Link l has to accommodate all k flow requirements simultaneously, i.e., $\sum_{i=1}^{k} \max_j F_{i,j}(l) \leq R(l)$ for all $l \in \mathcal{L}$.

Problem Setting

- LP-MMF($\tilde{\mathcal{R}}$): maximize the sum of rates of multicasting k sources:

$$\begin{aligned}
\text{maximize} \quad & \sum_{i=1}^{k} v_i && (2)\\
\text{subject to} \quad & F_{i,j} \text{ is a valid flow}, \forall i \in [k], j \in [k_i], \\
& \sum_{l \in \mathrm{Out}(s_i)} F_{i,j}(l) = \sum_{l \in \mathrm{In}(t_{i,j})} F_{i,j}(l) = v_i, \ \forall i \in [k], j \in [k_i], \\
& G_i(l) \geq F_{i,j}(l), \ \forall l \in \mathcal{L}, i \in [k], j \in [k_i], \\
& \sum_{i=1}^{k} G_i(l) \leq R(l), \ \forall l \in \mathcal{L}, && (3)\\
& R \in \tilde{\mathcal{R}},
\end{aligned}$$

where $[k] := \{1, \ldots, k\}$, $k, k_i \in \mathbb{N}_+, \forall i$.

- It takes polytope $\tilde{\mathcal{R}}$ as an input.
- (3) gives a dual vector μ; $\mu(l)$ indicates how sensitive the optimization objective is to the rate constraint $R(l)$.

Algorithm

- Our joint algorithm computes the optimal MMF while only requires a subset of the scheduling rate region.
- We attach $a_i \geq 0$ to link l_i, and with $\mathbf{a} = (a_i)_{i=1,\ldots,|\mathcal{L}|}$ we solve

$$\mathrm{argmax}_{R\in\mathcal{R}}\langle \mathbf{a}, R\rangle, \tag{4}$$

- This corresponds to a weighted maximal independent set problem

$$\begin{aligned} \text{ILP:} \quad & \text{maximize} \quad \sum_{i=1}^{|\mathcal{L}|} a_i S(l_i) \\ & \text{subject to} \quad S(l_i) + S(l_j) \leq 1, \quad \forall\, l_i, l_j : \; l_j \in \mathcal{I}(l_i). \end{aligned}$$

- The solution gives us an achievable rate vector, i.e., a vertex of $\mathcal{R}$.
- The ILP can have high complexity; we deal with this later.
- Based on (4), we iteratively search the MMF and the scheduling rates.

Algorithm

Algorithm 1

Input: a network $(\mathcal{V}, \mathcal{L}, \mathcal{I})$ **Output**: maximum multiflow v

Start with any rate vector $R_1 \in \mathcal{R}$, $\mathcal{R}_1 \leftarrow \{R_1\}$
for $i = 1, 2, \ldots$ **do**
  Run $v_i \leftarrow$LP-MMF$(\mathcal{R}_i)$ and obtain the dual vector μ_i
  Run ILP to find $R_{i+1} \leftarrow \mathrm{argmax}_{R \in \mathcal{R}} \langle \mu_i, R \rangle$
  $\mathcal{R}_{i+1} \leftarrow \mathrm{conv}\left(\mathcal{R}_i \cup \{R_{i+1}\}\right)$
  if $\langle \mu_i, R_{i+1} \rangle = \max_{R \in \mathcal{R}_i} \langle \mu_i, R \rangle$ **then**
    return v_i
  end if
end for

From $i = 1$, in each iteration, the algorithm works as follows:

1. We start with a known subset of the scheduling region $\mathcal{R}_i$. E.g., we start with $\mathcal{R}_1 = \{R_1\}$.
2. By the ILP, we find a new rate vector R_{i+1} by

$$R_{i+1} = \operatorname{argmax}_{R \in \mathcal{R}} \langle \mu_i, R \rangle. \tag{5}$$

3. We update the subset by $\mathcal{R}_{i+1} = \operatorname{conv}\left(\mathcal{R}_i \cup \{R_{i+1}\}\right)$.
4. If $\langle \mu_i, R_{i+1} \rangle = \max_{R \in \mathcal{R}_i} \langle \mu_i, R \rangle$, the algorithm terminates and outputs the last optimal value of LP-MMF($\mathcal{R}_i$).

The algorithm can terminate quite quickly; the remaining question is whether the output value is optimal.

Algorithm: Proof Sketch of Optimality

1. Since the scheduling region is a polytope with finite number of vertices, the Algorithm 1 will eventually terminate.
2. $\mathrm{flow}(R)$ is the function that outputs the MMF w.r.t. a rate vector R.
3. $\mathcal{R}$ is the entire scheduling region and $\mathcal{R}_i$ is the subset until iteration i.
4. If the algorithm ends at iteration i', $\langle \mu_{i'}, R_{i'+1} \rangle = \max_{R \in \mathcal{R}_{i'}} \langle \mu_{i'}, R \rangle$.
5. Combining with $R_{i+1} = \mathrm{argmax}_{R \in \mathcal{R}} \langle \mu_i, R \rangle$, we have

$$\max_{R \in \mathcal{R}} \langle \mu_{i'}, R \rangle = \max_{R \in \mathcal{R}_{i'}} \langle \mu_{i'}, R \rangle. \tag{6}$$

6. Suppose the optimal R in LP-MMF($\mathcal{R}_{i'}$) is R^*.
7. We can verify R^* maximizes $\mathrm{flow}(R) - \langle \mu_{i'}, R \rangle$ for $R \in \mathbb{R}_{\geq 0}^{|\mathcal{L}|}$, hence

$$\begin{aligned} \mathrm{flow}(R^*) - \langle \mu_{i'}, R^* \rangle &\geq \mathrm{flow}(R') - \langle \mu_{i'}, R' \rangle \\ &\overset{(a)}{\geq} \mathrm{flow}(R') - \langle \mu_{i'}, R^* \rangle, \end{aligned}$$

   where (a) is by (6) and the R^* maximizes $\langle \mu_{i'}, R \rangle$ for $R \in \mathcal{R}_{i'}$.
8. Therefore R^* maximizes $\mathrm{flow}(R)$ for $R \in \mathcal{R}$.

Algorithm

- The algorithm requires integer linear programming, hence it does not have a polynomial time complexity in theory, which is expected since this problem is NP-hard.
- We use simulation on random networks to show the efficiency.

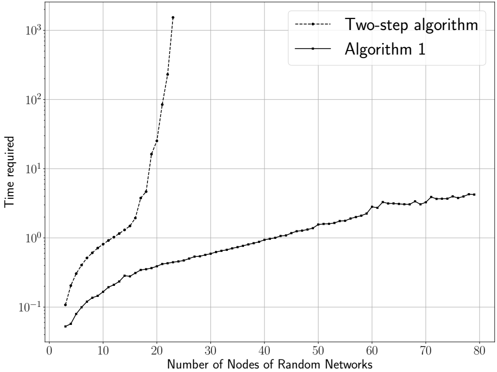
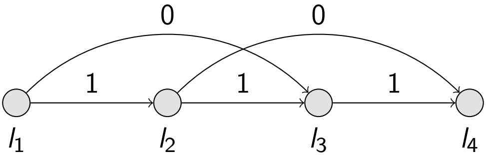


Figure: Results of 1-hop interference and multicast case of 1 source and 2 sinks.

Networks with Non-negligible Delays

- Recent results show that by utilizing non-negligible propagation delays, the scheduling region can be greatly improved (Yang et al., 2023).
- However, the scheduling problem suffers doubly-exponential complexity.
- We adopt a similar graphical formulation and extend our algorithm.
- We first review the scheduling problem.
- Recall for $|\mathcal{L}| = 4$ and unit delay and 1-hop interference, we have:



Networks with Non-negligible Delays

- For a collision-free schedule matrix S and integers $T \in \mathbb{N}_+$, $k \in \mathbb{Z}$, $S[T,k]$ denotes the submatrix of S consisting of columns $kT, kT+1, \ldots, (k+1)T-1$.

Definition (Scheduling Graph (Ma et al., 2021))

Given a network $\mathcal{N}$ and an integer $T > 0$, a *scheduling graph* $(\mathcal{M}_T, \mathcal{E}_T)$ satisfies: the vertex set $\mathcal{M}_T$ includes all the $|\mathcal{L}| \times T$ binary matrices A such that $A = S[T,0]$ for a certain collision-free schedule S. The edge set $\mathcal{E}_T$ includes all the vertex pairs (A,B) such that $A = S[T,0]$ and $B = S[T,1]$ for a certain collision-free schedule S.

- By choosing $T \geq \max_{l\in\mathcal{L}} \max_{l'\in\mathcal{I}(l)} |D(l,l')|$, the scheduling problem is searching all the cycles in the scheduling graph, which is NP-hard.

Networks with Non-negligible Delays

- To solve the MMF problem, the key is to have a function similar to $\operatorname{argmax}_{R\in\mathcal{R}}\langle \mathbf{a}, R\rangle$ that gives a rate vector fast.
- We hence define a *weighted* scheduling graph:

Definition

Given a weight vector $\mathbf{a} \in \mathbb{R}^{|\mathcal{L}|}$ and a scheduling graph $(\mathcal{M}_T, \mathcal{E}_T)$, a *weighted scheduling graph* $(\mathcal{M}_T, \mathcal{E}_T, w_\mathbf{a})$ is a directed, weighted graph such that: the vertex set is $\mathcal{M}_T$, and for each directed edge (v_1, v_2) in $(\mathcal{M}_T, \mathcal{E}_T)$, there is a weighed, directed edge (v_1, v_2) in $(\mathcal{M}_T, \mathcal{E}_T, w_\mathbf{a})$ with weight $w_\mathbf{a}(v_1, v_2) = \mathbf{a}^\intercal v_2 \mathbf{1}$.

- Given a vector $\mathbf{a}$, finding a vector that solves $\operatorname{argmax}_{R\in\mathcal{R}}\langle \mathbf{a}, R\rangle$ is equivalent to finding the maximum-mean-cycle in $(\mathcal{M}_T, \mathcal{E}_T, w_\mathbf{a})$.
- This can be solved with a polynomial complexity $\Theta(|\mathcal{M}_T|\,|\mathcal{E}_T|)$ by the Karp's algorithm (Karp, 1978).

Algorithm

Algorithm 2 Algorithm for Networks with Delays

Input: a network $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{I}, D)$
Output: maximum multiflow v

Start with any rate vector $R_1 \in \mathcal{R}$, $\mathcal{R}_1 \leftarrow \{R_1\}$
Construct $(\mathcal{M}_T, \mathcal{E}_T)$ from $\mathcal{N}$
for $i = 1, 2, \ldots$ **do**
  Run $v_i \leftarrow$LP-MMF$(\mathcal{R}_i)$ and obtain the dual vector μ_i
  Construct $(\mathcal{M}_T, \mathcal{E}_T, w_{\mu_i})$ by $(\mathcal{M}_T, \mathcal{E}_T)$ and μ_i
  Find maximum-mean-cycle in $(\mathcal{M}_T, \mathcal{E}_T, w_{\mu_i})$ and obtain $R_{i+1} \leftarrow \operatorname{argmax}_{R \in \mathcal{R}} \langle \mu_i, R \rangle$
  $\mathcal{R}_{i+1} \leftarrow \operatorname{conv}(\mathcal{R}_i \cup \{R_{i+1}\})$
  if $\langle \mu_i, R_{i+1} \rangle = \max_{R \in \mathcal{R}_i} \langle \mu_i, R \rangle$ **then**
    return v_i
  end if
end for

Networks with Non-negligible Delays

- We can similar prove the optimality of the output.
- We present simulation results on simple settings:

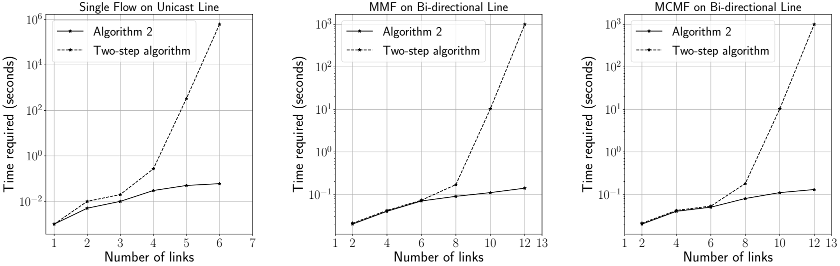


Figure: Line networks with unit delay

- For $\mathcal{N}_{4,1}^{D=1}$, the algorithm in (Ma et al., 2021) needs to search 7653 rate vectors, while we only need to find 2 of them for solving the MMF problem.

Conclusion

- We revisited the maximum multiflow problem that is NP-hard in theory.
- We provided an efficient joint algorithm that outputs optimal value in a finite number of iterations, while only requires a subset of the scheduling region.
- We studied the general multi-source multi-sink network with network coding.
- Our framework involves networks with non-negligible delay.
- We used simulation results to demonstrate our advantages.
- Future work: our maximum-mean-cycle algorithm could improve the calculation of the scheduling rate region (Ma et al., 2021; Yang et al., 2023).

Acknowledgement

The work of Shenghao Yang was partially supported by the National Key R&D Program of China under Grant 2022YFA1005000. The work of Cheuk Ting Li was partially supported by two grants from the Research Grants Council of the Hong Kong Special Administrative Region, China [Project No.s: CUHK 24205621 (ECS), CUHK 14209823 (GRF)]. This work was done when Yanxiao Liu was with The Chinese University of Hong Kong.

References

Ahlswede, R., Cai, N., Li, S.-Y., and Yeung, R. W. (2000). Network information flow. *IEEE Transactions on information theory*, 46(4):1204–1216.

Baker, B. S. (1994). Approximation algorithms for np-complete problems on planar graphs. *Journal of the ACM (JACM)*, 41(1):153–180.

Jain, K., Padhye, J., Padmanabhan, V. N., and Qiu, L. (2005). Impact of interference on multi-hop wireless network performance. *Wireless networks*, 11(4):471–487.

Karp, R. M. (1978). A characterization of the minimum cycle mean in a digraph. *Discrete mathematics*, 23(3):309–311.

Ma, J., Liu, Y., and Yang, S. (2021). Rate region of scheduling a wireless network with discrete propagation delays. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pages 1–10. IEEE.

Wan, P.-J. (2009). Multiflows in multihop wireless networks. In *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 85–94.

Yang, S., Ma, J., and Liu, Y. (2023). Wireless network scheduling with discrete propagation delays: Theorems and algorithms. *arXiv 2107.03083*.