Multiflows under Physical Interference Model

(draft)

Consider a multihop wireless network $(V, A, \mathcal{I})$, where V is the set of networking nodes, A is the set of communication links, and $\mathcal{I}$ is the collection of independent set of links specified by an interference model. Each link has a *unit* communication data rate. For any positive integer k, let $[k]$ denote the set $\{1, 2, \cdots, k\}$. Suppose that we are given k end-to-end unicast communication requests. For each $j \in [k]$, $\mathcal{P}_j$ denotes the set of simple paths of the request j, $\mathcal{F}_j$ denotes the set of flows of the request j, and the value of a flow $f_j \in \mathcal{F}_j$ is denoted by $val(f_j)$. A multiflow is a sequence $f = \langle f_1, f_2, \cdots, f_k \rangle$ with $f_j \in \mathcal{F}_j$ for each $j \in [k]$. Let $f = \langle f_1, f_2, \cdots, f_k \rangle$ be a multiflow. The transmission time of each link $a \in A$ required by f is

$$\textstyle\sum_{j \in [k]} f_j(a).$$

The total value of f is

$$\textstyle\sum_{j \in [k]} val(f_j).$$

Given that each request j has a positive traffic demand d_j, the concurrency of f is

$$\min_{j \in [k]} \frac{val(f_j)}{d_j}.$$

This chapter studies the following two variants of the multiflow problems:

- **Maximum Multiflow** (**MMF**): The problem **MMF** seeks a multiflow f and a MAC-layer link schedule $\mathcal{S}$ of $\sum_{j \in [k]} f_j$ such that the length of $\mathcal{S}$ is at most one and the total value of f is maximized.
- **Maximum Concurrent Multiflow** (**MCMF**): Given that each request has a traffic demand, the problem **MCMF** seeks a multiflow f and a MAC-layer link schedule $\mathcal{S}$ of $\sum_{j \in [k]} f_j$ such that the length of $\mathcal{S}$ is at most one and the concurrency of f is maximized.

While both **MMF** and **MCMF** admit polynomial-time approximation-preserving reduction to **MWIS** based on ellipsoid method, such reduction is quite inefficient in practice. This chapter presents faster and simpler combinatorial approximation algorithms for **MMF** and **MCMF** which involve a sequence of computations of shortest paths and independent sets. These algorithms offer nice trade-off between accuracy in terms the approximation bound and efficiency in terms of the running time. Specifically, let $\mathcal{A}$ be a μ-approximation algorithm for **MWIS**, and $\varepsilon \in (0, 1/2]$ be an accuracy-efficiency trade-off parameter. The approximation algorithms presented in this chapter achieve an approximation bound $(1 + 2\varepsilon)\,\mu$ in a running time growing with $1/\varepsilon$ in the at most square order.

When applied to **MMF** and **MCMF** under the physical interference model, the above algorithms together the approximation algorithms for **MWISL** developed in Chapter 7 immediately lead to the following algorithmic results:

- With linear power assignment, there are constant-approximation algorithms for **MMF** and **MCMF** respectively under the physical interference model.
- With any other monotone and sublinear power assignment, there are $O(\ln \alpha)$-approximation algorithm for **MMF** and **MCMF** respectively under the physical interference model.
- With power control, there are $O(\ln \alpha)$-approximation algorithm for **MMF** and **MCMF** respectively under the physical interference model.

The remainder of this chapter is organized as follows. Section 1 establishes weak dualities of **MMF** and **MCMF**, which reveal the intrinsic relations among them, shortest paths, and maximum-weighted independent sets. Section 2 introduces a generic adaptive coupled game. Section 3 and Section 4 describe the design and analyses of the approximation algorithms for **MCMF** and **MMF** respectively, which offer nice trade-off between accuracy in terms the approximation bound and efficiency in terms of the running time.

1 Weak Dualities

Suppose that y is positive function on A. For each $j \in [k]$, let $dist_j(y)$ be the length of a shortest of the j-th request with respect to y. The problem **MCMF** has the following weak duality.

Theorem 1.1 *The concurrency of the maximum concurrent multiflow is at most*

$$\frac{\max_{I \in \mathcal{I}} y(I)}{\sum_{j \in [k]} d_j dist_j(y)}$$

The problem **MMF** has the following weak duality.

Theorem 1.2 *The total value of the maximum multiflow is at most*

$$\frac{\max_{I\in\mathcal{I}} y(I)}{\min_{j\in[k]} dist_j(y)}$$

Consider a multiflow $f = \langle f_1, \cdots, f_k \rangle$. A non-negative function x on $\bigcup_{j\in[k]} \mathcal{P}_j$ is said to be a path-flow decomposition of f if for each $a \in A$ and each $j \in [k]$,

$$f_j(a) = \sum_{P\in\mathcal{P}_j} |\{a\} \cap P| \, x(P).$$

A non-negative function z on $\mathcal{I}$ is said to be a (fractional) link schedule of $f = \langle f_1, \cdots, f_k \rangle$ if for each $a \in A$,

$$\sum_{j=1}^{k} f_j(a) = \sum_{I\in\mathcal{I}} |\{a\} \cap I| \, z(I).$$

The following invariant property holds.

Lemma 1.3 *For any path-flow decomposition x of f and any link schedule z of f,*

$$\sum_{j=1}^{k} \sum_{P\in\mathcal{P}_j} x(P) \, y(P) = \sum_{I\in\mathcal{I}} y(I) \, z(I)$$

Proof. Since

$$\begin{aligned}
&\sum_{j=1}^{k} \sum_{P\in\mathcal{P}_j} x(P) \, y(P) \\
&= \sum_{j=1}^{k} \sum_{P\in\mathcal{P}_j} x(P) \sum_{a\in A} y(a) \, |\{a\} \cap P| \\
&= \sum_{a\in A} y(a) \sum_{j=1}^{k} \sum_{P\in\mathcal{P}_j} x(P) \, |\{a\} \cap P| \\
&= \sum_{a\in A} y(a) \sum_{j=1}^{k} f_j(a) \\
&= \sum_{a\in A} y(a) \sum_{I\in\mathcal{I}} |\{a\} \cap I| \, z(I) \\
&= \sum_{I\in\mathcal{I}} z(I) \sum_{a\in A} y(a) \, |\{a\} \cap I| \\
&= \sum_{I\in\mathcal{I}} y(I) \, z(I),
\end{aligned}$$

the lemma holds. ■

Lemma 1.4 *If f is feasible, then*

$$\sum_{j=1}^{k} dist_j(y)\, val(f_j) \leq \max_{I \in \mathcal{I}} y(I)$$

Proof. Consider any path-flow decomposition x of f and any shortest link schedule z of f. On one hand,

$$\begin{aligned}
&\sum_{j=1}^{k} \sum_{P \in \mathcal{P}_j} x(P)\, y(P) \\
&\geq \sum_{j=1}^{k} \sum_{P \in \mathcal{P}_j} x(P)\, dist_j(y) \\
&= \sum_{j=1}^{k} dist_j(y) \sum_{P \in \mathcal{P}_j} x(P) \\
&= \sum_{j=1}^{k} dist_j(y)\, val(f_j)
\end{aligned}$$

On the other hand,

$$\sum_{I \in \mathcal{I}} y(I)\, z(I) \leq \left(\max_{I \in \mathcal{I}} y(I) \right) \sum_{I \in \mathcal{I}} z(I) \leq \max_{I \in \mathcal{I}} y(I).$$

By Lemma 1.3,

$$\sum_{j=1}^{k} dist_j(y)\, val(f_j) \leq \max_{I \in \mathcal{I}} y(I).$$

So, the lemma holds. ■

Next, we prove Theorem 1.1. Let $f = \langle f_1, \cdots, f_k \rangle$ be a maximum concurrent multiflow. By Lemma 1.4,

$$\begin{aligned}
\max_{I \in \mathcal{I}} y(I) &\geq \sum_{j=1}^{k} dist_j(y)\, val(f_j) \\
&= \sum_{j=1}^{k} d_j dist_j(y) \frac{val(f_j)}{d_j} \\
&\geq \left(\min_{j \in [k]} \frac{val(f_j)}{d_j} \right) \sum_{j=1}^{k} d_j dist_j(y).
\end{aligned}$$

Thus,

$$\min_{j\in[k]} \frac{val\left(f_j\right)}{d_j} \leq \frac{\max_{I\in\mathcal{I}} y\left(I\right)}{\sum_{j=1}^{k} d_j dist_j\left(y\right)}.$$

So, Theorem 1.1 holds.

Finally, we prove Theorem 1.2. Let $f = \langle f_1, \cdots, f_k \rangle$ be a maximum multiflow. By Lemma 1.4,

$$\begin{aligned}\max_{I\in\mathcal{I}} y\left(I\right) &\geq \sum_{j\in[k]} dist_j\left(y\right) val\left(f_j\right)\\ &\geq \left(\min_{j\in[k]} dist_j\left(y\right)\right) \sum_{j\in[k]} val\left(f_j\right).\end{aligned}$$

Thus,

$$\sum_{j\in[k]} val\left(f_j\right) \leq \frac{\max_{I\in\mathcal{I}} y\left(I\right)}{\min_{j\in[k]} dist_j\left(y\right)}.$$

So, Theorem 1.2 holds.

2 An Adaptive Coupled Game

In this section, we introduce a sequential game played between an adversary and an agent who is advised by a set E of experts. Prior to the first round of the game, the agent chooses a positive weight $w\left(e\right)$ for each expert $e \in E$. In each round, the adversary determines a non-negative profit $p\left(e\right)$ and a non-negative loss $l\left(e\right)$ for each $e \in E$ subjected to two rules:

- **Normalization Rule**:

$$\max_{e\in E} \left|p\left(e\right) - l\left(e\right)\right| = 1.$$

- **Generalized Zero-Sum Rule**:

$$\sum_{e\in E} y\left(e\right)\left(p\left(e\right) - l\left(e\right)\right) \geq 0$$

The agent may then update the weight $y\left(e\right)$ of each expert $e \in E$ after observing the profits and loss of the experts. Given a parameter $\eta \in (0, 1)$, the objective of the agent is to maintain that at the end of each round for *each* expert e, the cumulative profit of e minus η times the cumulative loss of e is lower-bounded by some value invariant to the round number; the objective of the adversaries is exactly the opposite.

Now, we describe the **Multiplicative Weights Update (MWU)** strategy for the agent, while leaving the strategy for the adversary to specific applications. Fix an $\varepsilon \in (0, 1)$. For each expert $e \in E$,

- $y(e) = 1$ initially;
- in each game round, if e earns a profit $p(e)$ and incurs a loss $l(e)$, $y(e)$ is updated by a multiplicative factor

$$1 - \varepsilon (p(e) - l(e)).$$

An implementation of the game playing with this strategy is described as follows. Let $P(e)$, $L(e)$, and $y(e)$ be cumulative profit, cumulative loss, and weight respectively of each $e \in E$, which are initially 0, 0, and 1 respectively. Repeat following rounds:

1. **Generation of Profits/Losses**: The adversary determines a non-negative profit $p(e)$ and loss $l(e)$ for each $e \in E$ subjected to the **Normalization Rule** and **Generalized Zero-Sum Rule**. As the result,

$$L(e) \leftarrow L(e) + l(e);$$
$$P(e) \leftarrow P(e) + p(e).$$

2. **Multiplicative Weights Update**: The agent updates $y(e)$ for each $e \in E$ by setting

$$y(e) \leftarrow y(e)(1 - \varepsilon (p(e) - l(e))).$$

The effectiveness of above implementation of the game is asserted in the theorem below.

Theorem 2.1 *At the end of each round, for any $e \in E$,*

$$\frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}} L(e) \leq P(e) + \frac{\ln m}{\ln(1-\varepsilon)^{-1}}$$

Proof. For each round r and each $e \in E$, let $p_r(e)$ (respectively, $l_r(e)$, $y_r(e)$) denote the profit (respectively, loss, weight) of e received in round r; let $P_r(e)$ (respectively, $L_r(e)$, $y_r(e)$) denote the cumulative profit (respectively, cumulative loss, weight) of e at the end of the round r. In addition, for each $e \in E$, let

$$P_0(e) = L_0(e) = 0,$$
$$y_0(e) = 1.$$

We first claim that for any round t,

$$y_t(E) \leq y_{t-1}(E).$$

Indeed, by the **Generalized Zero-Sum Rule** we have

$$\begin{aligned} y_t(E) &= \sum_{e\in E} y_t(e) \\ &= \sum_{e\in E} y_{t-1}(e)\left(1-\varepsilon\left((p_t(e)-l_t(e)\right)\right)) \\ &= \sum_{e\in E} y_{t-1}(e) - \varepsilon \sum_{e\in E} y_{t-1}(e)\left((p_t(e)-l_t(e)\right)) \\ &\le \sum_{e\in E} y_{t-1}(e) \\ &\le y_{t-1}(E). \end{aligned}$$

So, the claim holds.

Next, we claim that for any round t and any expert e,

$$y_t(e) \ge (1-\varepsilon)^{P_t(e)}(1+\varepsilon)^{L_t(e)}.$$

Indeed, let

$$\begin{aligned} [t]^+ &= \{i \in [t] : p_i(e) \ge l_i(e)\}, \\ [t]^- &= \{i \in [t] : p_i(e) < l_i(e)\}. \end{aligned}$$

By the **Normalization Rule** and the inequality

$$(1\pm\varepsilon)^x \le 1 \pm \varepsilon x$$

for any $x \in [0,1]$, we have

$$\begin{aligned} & y_t(e) \\ &= \prod_{i\in[t]} \left(1-\varepsilon\left(p_i(e)-l_i(e)\right)\right) \\ &= \prod_{i\in[t]^+} \left(1-\varepsilon\left(p_i(e)-l_i(e)\right)\right) \cdot \prod_{i\in[t]^-} \left(1+\varepsilon\left(l_i(e)-p_i(e)\right)\right) \\ &\ge \prod_{i\in[t]^+} (1-\varepsilon)^{p_i(e)-l_i(e)} \cdot \prod_{i\in[t]^-} (1+\varepsilon)^{l_i(e)-p_i(e)} \\ &= (1-\varepsilon)^{\sum_{i\in[t]^+}(p_i(e)-l_i(e))}(1+\varepsilon)^{\sum_{i\in[t]^-}(l_i(e)-p_i(e))} \\ &= (1-\varepsilon)^{\sum_{i\in[t]^+} p_i(e)} \left(\frac{1}{1-\varepsilon}\right)^{\sum_{i\in[t]^+} \ell_i(e)} \left(\frac{1}{1+\varepsilon}\right)^{\sum_{i\in[t]^-} p_i(e)} (1+\varepsilon)^{\sum_{i\in[t]^-} l_i(e)} \\ &\ge (1-\varepsilon)^{\sum_{i\in[t]^+} p_i(e)}(1+\varepsilon)^{\sum_{i\in[t]^+} \ell_i(e)}(1-\varepsilon)^{\sum_{i\in[t]^-} p_i(e)}(1+\varepsilon)^{\sum_{i\in[t]^-} l_i(e)} \\ &= (1-\varepsilon)^{\sum_{i\in[t]} p_i(e)}(1+\varepsilon)^{\sum_{i\in[t]} l_i(e)} \\ &= (1-\varepsilon)^{P_t(e)}(1+\varepsilon)^{L_t(e)}. \end{aligned}$$

Thus, the claim holds.

The previous two claims yield that for any round t and any expert e,

$$(1-\varepsilon)^{P_t(e)}(1+\varepsilon)^{L_t(e)} \leq y_t(e) \leq y_t(E) \leq y_0(E) = m.$$

Taking the logarithm of both sides, we have

$$P_t(e)\ln(1-\varepsilon) + L_t(e)\ln(1+\varepsilon) \leq \ln m,$$

Hence,

$$\frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}} L_t(e) \leq P_t(e) + \frac{\ln m}{\ln(1-\varepsilon)^{-1}}.$$

So, the theorem holds. ■

3 Maximum Concurrent Mulitflow

This section presents a $(1+2\varepsilon)$-approximation algorithm **CMF-LS**(ε) for **MCMF** where $\varepsilon \in (0, 1/2]$ is a fixed parameter. We first give an overview on the design of the algorithm **CMF-LS**(ε). Let *opt* be the concurrency of a maximum concurrent multiflow. The algorithm is iterative. Each iteration first computes a multiflow f of concurrency ℓ, a scaling factor $\lambda \geq \frac{opt}{\mu}$, and a "primary" link schedule Γ of length ℓ and its corresponding link transmission-time function g. With respect to the triple (f, λ, g), the *deficit* of a link a is defined to be

$$d'(a) = \max\left\{0, \frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}}\lambda \sum_{j\in[k]} f_j(a) - g(a)\right\}.$$

For such deficit demand d', a "complementary" link schedule Γ' is then computed. Clearly, $\Gamma \cup \Gamma'$ is a link schedule of the multiflow

$$\frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}}\lambda f.$$

Let ℓ' be the length of Γ'. If

$$\frac{\ell'}{\ell} \leq \varepsilon' := \frac{(1+2\varepsilon)\ln(1+\varepsilon) + \ln(1-\varepsilon)}{\ln(1-\varepsilon)^{-1}},$$

then the scaled multiflow

$$\frac{1}{\ell+\ell'}\frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}}\lambda f$$

and the scaled schedule

$$\frac{1}{\ell+\ell'}(\Gamma\cup\Gamma')$$

are returned as the output. The concurrency of the returned multiflow is

$$\begin{aligned}
&\frac{\ell}{\ell+\ell'}\frac{\ln\left(1+\varepsilon\right)}{\ln\left(1-\varepsilon\right)^{-1}}\lambda \\
&=\frac{1}{1+\ell'/\ell}\frac{\ln\left(1+\varepsilon\right)}{\ln\left(1-\varepsilon\right)^{-1}}\lambda \\
&\geq\frac{\lambda}{1+2\varepsilon} \\
&\geq\frac{opt}{\left(1+2\varepsilon\right)\mu}.
\end{aligned}$$

In order to compute the complementary link schedule quickly, a preprocessing is performed to partition all links into independent sets. On such partition $\mathcal{J}$ can be obtained by applying the algorithm **GreedyFC** with $\mathcal{A}$, and let $l=|\mathcal{J}|$. Then, a complementary link schedule Γ' of a deficit demand d' is computed as follows. Initially, Γ' is empty. For each $J\in\mathcal{J}$, let

$$\delta'=\max_{a\in J}d'\left(a\right),$$

and if $\delta'>0$ then add the pair (J,δ') to Γ'. The length of Γ' is

$$\ell'=\sum_{J\in\mathcal{J}}\max_{a\in J}d'\left(a\right)\leq|\mathcal{J}|\max_{a\in A}d'\left(a\right)=l\max_{a\in A}d'\left(a\right).$$

The algorithm **CMF-LS**(ε) is outlined in Table 1. The computation of the primary schedule Γ (together with its length ℓ and its link transmission time function g), the multiflow f, and the scaling factor λ in each round follows the general framework of the coupled game introduced in Section 2. Each link is regarded as an an expert, and each iteration corresponds to a game round. The agent plays exactly with the strategy on maintaining the expert/link weight function y described in Section 2. The profit (respectively, loss) generation strategy of the adversary is coupled with the primary schedule (respectively, multiflow) augmentation with the following *invariant* property maintained throughout the game: At the end of each round, the cumulative profit of each link a is exactly $g\left(a\right)$, and the cumulative loss of each link is at least $\lambda\sum_{j\in[k]}f_j\left(a\right)$. Initially, both the multiflow f and the primary schedule Γ are empty, and the scaling factor λ is infinity. In each round, the adversary generates the profits/losses as follows. Let I be the independent set w.r.t. y output by the algorithm $\mathcal{A}$, and P_j be a shortest path of the j-th request w.r.t. y for each $j\in[k]$. If λ is greater than

$$\frac{y\left(I\right)}{\sum_{j\in[k]}d_jy\left(P_j\right)}=\frac{y\left(I\right)}{\sum_{j\in[k]}d_jdist_j\left(y\right)},$$

Algorithm $\mathbf{CMF\text{-}LS}(\varepsilon)$:

$\forall j \in [k]$, $f_j \leftarrow \mathbf{0}$; $\Gamma \leftarrow \emptyset$, $\ell \leftarrow 0$, $g \leftarrow \mathbf{0}$; $\lambda \leftarrow \infty$;
$y \leftarrow \mathbf{1}$, $\varepsilon' \leftarrow \frac{(1+2\varepsilon)\ln(1+\varepsilon)+\ln(1-\varepsilon)}{\ln(1-\varepsilon)^{-1}}$;
$\mathcal{J} \leftarrow$ a link partition output by **GreedyFC** with $\mathcal{A}$;

repeat forever

  $I \leftarrow$ a y-weighted IS in $\mathcal{I}$ output by $\mathcal{A}$;
  $\forall j \in [k]$, $P_j \leftarrow$ a shortest j-path w.r.t. y;
  $\lambda \leftarrow \min\left\{\lambda, \frac{y(I)}{\sum_{j\in[k]} d_j y(P_j)}\right\}$; //truncation
  $\delta \leftarrow \frac{1}{\max_{a\in A}\left||\{a\}\cap I| - \lambda\sum_{j=1}^{k} d_j |\{a\}\cap P_j|\right|}$;
  $\Gamma \leftarrow \Gamma \cup \{(I,\delta)\}$, $\ell \leftarrow \ell + \delta$;
  for each $a \in I$ do $g(a) \leftarrow g(a) + \delta$;
  for each $j \in [k]$ do
    for each $a \in P_j$ do $f_j(a) \leftarrow f_j(a) + \delta d_j$;

  $\Gamma' \leftarrow \emptyset, \ell' \leftarrow 0$;
  for each $J \in \mathcal{J}$ do
    $\delta' \leftarrow \max_{a\in J}\max\left\{0, g(a) - \frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}}\lambda\sum_{j\in[k]} f_j(a)\right\}$;
    if $\delta' > 0$ then $\Gamma' \leftarrow \Gamma' \cup \{(J,\delta')\}$, $\ell' \leftarrow \ell' + \delta'$;
  if $\ell' \le \varepsilon'\ell$ then return $\frac{1}{\ell+\ell'}\frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}} f$ and $\frac{1}{\ell+\ell'}(\Gamma\cup\Gamma')$.

  for each $a \in A$ do
    $y(a) \leftarrow y(a)\left(1 - \varepsilon\delta\left(|\{a\}\cap I| - \lambda\sum_{j\in[k]} d_j |\{a\}\cap P_j|\right)\right)$;

Table 1: Outline of the algorithm $\mathbf{CMF\text{-}LS}(\varepsilon)$.

then λ is reset to

$$\frac{y(I)}{\sum_{j\in[k]} d_j dist_j(y)}.$$

Thus, λ is non-increasing with the round number. For some positive length δ to be determined shortly, Γ is augmented by the pair (I,δ), and both ℓ and g are updated accordingly. The profit of each link $a \in I$ is exactly its transmission time received from the augmenting pair (I,δ), which is equal to $|\{a\} \cap I|$. Thus, the cumulative profit of each link a is exactly $g(a)$. For each $j \in [k]$, f_j is augmented by a flow of value δd_j along the path P_j. The loss of each link a is exactly the λ times the total amount of the augmenting flow through a, which is equal to

$$\delta\lambda \sum_{j=1}^{k} d_j |\{a\} \cap P_j|.$$

Since λ is non-increasing with the round number, the cumulative loss of each link a is at least

$$\lambda \sum_{j\in[k]} f_j(a).$$

The **Normalization Rule** dictates that

$$\delta = \frac{1}{\max_{a\in A} \left| |\{a\} \cap I| - \lambda \sum_{j=1}^{k} d_j |\{a\} \cap P_j| \right|}.$$

Since

$$\begin{aligned}
&\sum_{a\in A} y(a) \left(\delta |\{a\} \cap I| - \delta\lambda \sum_{j\in[k]} d_j |\{a\} \cap P_j| \right) \\
&= \delta \left(\sum_{a\in A} y(a) |\{a\} \cap I| - \lambda \sum_{a\in A} y(a) \sum_{j=1}^{k} d_j |\{a\} \cap P_j| \right) \\
&= \delta \left(y(I) - \lambda \sum_{j=1}^{k} d_j \sum_{a\in A} y(a) |\{a\} \cap P_j| \right) \\
&= \delta \left(y(I) - \lambda \sum_{j=1}^{k} d_j y_{t-1}(P_j) \right) \\
&\geq 0,
\end{aligned}$$

the **Generalized Zero-Sum Rule** is also satisfied by the profit/loss assignment in this round.

Subsequently, the complementary link schedule Γ' of the deficit demand is computed. By Theorem 2.1, for each link a,

$$\frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}} \lambda \sum_{j\in[k]} f_j(a) \leq g(a) + \frac{\ln m}{\ln(1-\varepsilon)^{-1}}.$$

This means that the deficit of each link is at most

$$\frac{\ln m}{\ln (1-\varepsilon)^{-1}}.$$

Hence,

$$\ell' \leq \frac{l \ln m}{\ln (1-\varepsilon)^{-1}}.$$

If $\ell' \leq \varepsilon' \ell$, both

$$\frac{1}{\ell+\ell'} \frac{\ln (1+\varepsilon)}{\ln (1-\varepsilon)^{-1}} f$$

and

$$\frac{1}{\ell+\ell'} \left(\Gamma \cup \Gamma'\right)$$

are returned as the output. Otherwise, the agent updates the weight function y using the multiplicative weight update method, and the algorithm moves on to the next round.

We proceed to derive the upper bounds on both the number of iterations and the approximation ratio of the algorithm **CMF-LS**(ε). Let h_j be the hop number of a minimum-hop path of the j-th request for each $j \in [k]$, and α be the size of a maximum-sized independent set.

Theorem 3.1 *The algorithm* ***CMF-LS***(ε) *runs in*

$$O\left(\varepsilon^{-2} \max\left\{1, \frac{\alpha}{\min_{j\in[k]} h_j}\right\} l \ln m\right).$$

iterations and has an approximation bound $(1+2\varepsilon)\,\mu$.

The following essential properties of λ and δ are the cornerstone to the above theorem

Lemma 3.2 *The scaling factor* λ *and the length* δ *computed in each iteration satisfy that*

$$\frac{opt}{\mu} \leq \lambda \leq \frac{\alpha}{\sum_{j\in[k]} d_j h_j},$$
$$\delta \geq \frac{1}{\max\left\{1, \frac{\alpha}{\min_{j\in[k]} h_j}\right\}}.$$

Proof. Let λ_0 and y_0 be the initial values of λ and y. For each round $t \geq 1$, let λ_t, δ_t and y_t be the value of λ, δ and y respectively computed in the round t; let I_t be the independent set I, computed in the iteration t. Then,

$$\lambda_1 = \frac{y_0\left(I_1\right)}{\sum_{j\in[k]} d_j dist_j\left(y_0\right)} = \frac{\left|I_1\right|}{\sum_{j\in[k]} d_j h_j} \leq \frac{\alpha}{\sum_{j\in[k]} d_j h_j}.$$

Since λ_t is non-increasing with t, for each iteration t we have

$$\lambda_t \leq \frac{\alpha}{\sum_{j\in[k]} d_j h_j}.$$

Next, we prove by induction on t that

$$\lambda_t \geq \frac{opt}{\mu}.$$

Consider the (first) iteration $t = 1$. By the weak duality given in Theorem 1.1,

$$\lambda_1 = \frac{y_0\left(I_1\right)}{\sum_{j\in[k]} d_j dist_j\left(y_0\right)} \geq \frac{1}{\mu}\frac{\max_{I\in\mathcal{I}} y_0\left(I\right)}{\sum_{j\in[k]} d_j dist_j\left(y_0\right)} \geq \frac{opt}{\mu}.$$

Now consider any iteration $t > 1$ and assume that

$$\lambda_{t-1} \geq \frac{opt}{\mu}.$$

and show that

$$\lambda_t \geq \frac{opt}{\mu}.$$

If $\lambda_t = \lambda_{t-1}$, then it is trivial that

$$\lambda_t \geq \frac{opt}{\mu}.$$

If $\lambda_t < \lambda_{t-1}$, then by the weak duality given in Theorem 1.1,

$$\lambda_t = \frac{y_{t-1}\left(I_t\right)}{\sum_{j\in[k]} d_j dist_j\left(y_{t-1}\right)} \geq \frac{1}{\mu}\frac{\max_{I\in\mathcal{I}} y_{t-1}\left(I\right)}{\sum_{j\in[k]} d_j dist_j\left(y_{t-1}\right)} \geq \frac{opt}{\mu}.$$

So, in either case,

$$\lambda_t \geq \frac{opt}{\mu}.$$

Finally, we show that for each iteration $t \geq 1$,

$$\delta_t \geq \frac{1}{\max\left\{1, \frac{\alpha}{\min_{j\in[k]} h_j}\right\}}.$$

For each round $t \geq 1$, let $P_j^{(t)}$ be the shortest path of the j-th request computed in the round t for each $j \in [k]$. Then, for each link a,

$$\begin{aligned}
&\left| |\{a\} \cap I_t| - \lambda_t \sum_{j=1}^{k} d_j \left| \{a\} \cap P_j^{(t)} \right| \right| \\
&\leq \max \left\{ |\{a\} \cap I_t| , \lambda_t \sum_{j=1}^{k} d_j \left| \{a\} \cap P_j \right| \right\} \\
&\leq \max \left\{ 1, \lambda_t \sum_{j=1}^{k} d_j \right\} \\
&\leq \max \left\{ 1, \frac{\alpha}{\sum_{j\in[k]} d_j h_j} \sum_{j=1}^{k} d_j \right\} \\
&\leq \max \left\{ 1, \frac{\alpha}{\min_{j\in[k]} h_j} \right\}
\end{aligned}$$

Thus,

$$\begin{aligned}
\delta_t &= \frac{1}{\max_{a\in A} \left| |\{a\} \cap I_t| - \lambda_t \sum_{j=1}^{k} d_j \left| \{a\} \cap P_j^{(t)} \right| \right|} \\
&\geq \frac{1}{\max \left\{ 1, \frac{\alpha}{\min_{j\in[k]} h_j} \right\}}.
\end{aligned}$$

This completes the proof of the lemma. ■

Next, we prove by contradiction that the number of iterations is at most

$$t = \left\lceil \frac{l \ln m}{(1+2\varepsilon) \ln (1+\varepsilon) + \ln (1-\varepsilon)} \max \left\{ 1, \frac{\alpha}{\min_{j\in[k]} h_j} \right\} \right\rceil .$$

Assume to the contrary the number of iterations is more than t. Consider the iteration t. By Lemma 3.2, the primary schedule Γ in the iteration t has length

$$\begin{aligned}
\ell &\geq \frac{t}{\max \left\{ 1, \frac{\alpha}{\min_{j\in[k]} h_j} \right\}} \\
&\geq \frac{l \ln m}{(1+2\varepsilon) \ln (1+\varepsilon) + \ln (1-\varepsilon)} \\
&= \frac{l \ln m}{\varepsilon' \ln (1-\varepsilon)^{-1}}.
\end{aligned}$$

On the other hand, the the complementary schedule Γ' in the iteration t has length

$$\ell' \leq \frac{l \ln m}{\ln (1-\varepsilon)^{-1}}.$$

Thus, $\ell' \leq \varepsilon' \ell$. This implies that the algorithm would terminate at the end the iteration t, which is a contradiction. So, the number of iterations is at most t, which is

$$O\left(\varepsilon^{-2} \max\left\{1, \frac{\alpha}{\min_{j\in[k]} h_j}\right\} l \ln m\right).$$

Finally, the approximation bound $(1+2\varepsilon)\mu$ follows from the property

$$\lambda \geq \frac{opt}{\mu}$$

in each iteration which is asserted in Lemma 3.2 and the argument given in the overview of the algorithm at the beginning of this section. This completes the proof of Theorem 3.1.

4 Maximum Mulitflow

The algorithm **MF-LS**(ε) for **MMF** is outlined in Table 2. It is very similar to the algorithm **CMF-LS**(ε) for **MCMF**, and consequently we only highlight the difference in this section. In each iteration, let P_j be the shortest one among the k shortest paths with respect to y. Then, only f_j is augmented in this iteration. Accordingly, λ is computed by

$$\lambda \leftarrow \min\left\{\lambda, \frac{y(I)}{y(P_j)}\right\};$$

and δ is chosen to be

$$\frac{1}{\max_{a\in A} \left||\{a\}\cap I| - \lambda |\{a\}\cap P_j|\right|}.$$

So, f_j is augmented by a flow of value δ along P_j, the weight $y(a)$ of each link a is updated by the multiplicative factor

$$1 - \varepsilon\delta\left(|\{a\}\cap I| - \lambda |\{a\}\cap P_j|\right).$$

For the performance analysis, we can show that in each iteration,

$$\frac{opt}{\mu} \leq \lambda \leq \frac{\alpha}{\min_{j\in[k]} h_j},$$
$$\delta \geq \min\left\{1, \frac{\min_{j\in[k]} h_j}{\alpha}\right\},$$
$$\ell' \leq l \frac{\ln m}{\ln(1-\varepsilon)^{-1}}.$$

Using these properties, we can prove the following performance of The algorithm **MF-LS**(ε).

Algorithm **MF-LS**(ε):

$\forall j \in [k]$, $f_j \leftarrow \mathbf{0}$; $\Gamma \leftarrow \emptyset$, $\ell \leftarrow 0$, $g \leftarrow \mathbf{0}$; $\lambda \leftarrow \infty$;
$y \leftarrow \mathbf{1}$, $\varepsilon' \leftarrow \frac{(1+2\varepsilon)\ln(1+\varepsilon)+\ln(1-\varepsilon)}{\ln(1-\varepsilon)^{-1}}$;
$\mathcal{J} \leftarrow$ a link partition output by **GreedyFC** with $\mathcal{A}$;

repeat forever
  $I \leftarrow$ a y-weighted IS in $\mathcal{I}$ output by $\mathcal{A}$;
  $\forall j \in [k]$, $P_j \leftarrow$ a shortest j-path w.r.t. y;
  $j \leftarrow \arg\min_{j\in[k]} y(P_j)$;
  $\lambda \leftarrow \min\left\{\lambda, \frac{y(I)}{y(P_j)}\right\}$;
  $\delta \leftarrow \frac{1}{\max_{a\in A} ||\{a\}\cap I| - \lambda|\{a\}\cap P_j||}$;
  $\Gamma \leftarrow \Gamma \cup \{(I,\delta)\}$, $\ell \leftarrow \ell + \delta$;
  for each $a \in I$ do $g(a) \leftarrow g(a) + \delta$;
  for each $a \in P_j$ do $f_j(a) \leftarrow f_j(a) + \delta$;

  $\Gamma' \leftarrow \emptyset, \ell' \leftarrow 0$;
  for each $J \in \mathcal{J}$ do
    $\delta' \leftarrow \max_{a\in J} \max\left\{0, g(a) - \frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}}\lambda \sum_{j\in[k]} f_j(a)\right\}$;
    if $\delta' > 0$ then $\Gamma' \leftarrow \Gamma' \cup \{(J,\delta')\}, \ell' \leftarrow \ell' + \delta'$;
  if $\ell' \leq \varepsilon'\ell$ then return $\frac{1}{\ell+\ell'}\frac{\ln(1+\varepsilon)}{\ln(1-\varepsilon)^{-1}} f$ and $\frac{1}{\ell+\ell'}(\Gamma \cup \Gamma')$.

  for each $a \in A$ do
    $y(a) \leftarrow y(a)\left(1 - \varepsilon\delta\left(|\{a\}\cap I| - \lambda|\{a\}\cap P_j|\right)\right)$;

Table 2: Outline of the algorithm **MF-LS**(ε).

Theorem 4.1 *The algorithm **MF-LS**(ε) runs in*

$$O\left(\varepsilon^{-2}\max\left\{1,\frac{\alpha}{\min_{j\in[k]}h_j}\right\}l\ln m\right).$$

iterations and has an approximation bound $(1+2\varepsilon)\,\mu$.