Maximum-Weighted Subset of Communication Requests Schedulable without Spectral Splitting

P.-J. Wan, H.Q. Yuan, X.H. Jia, J.L. Wang, and Z. Wang

wan@cs.iit.edu

Outline

- Problem Description
- A Finer Treatment of Conflicts
- Light Requests
- Heavy Requests
- Summary

Problem Instance

- A: a set of point-to-point communication requests

Problem Instance

- A: a set of point-to-point communication requests
  - $d \in (0,1]^A$: demands in transmission time

Problem Instance

- A: a set of point-to-point communication requests
  - $d \in (0,1]^A$: demands in transmission time
  - $w \in \mathbb{R}_+^A$: a weight function

Problem Instance

- A: a set of point-to-point communication requests
  - $d \in (0, 1]^A$: demands in transmission time
  - $w \in \mathbb{R}_+^A$: a weight function
- λ: number of channels

Problem Instance

- A: a set of point-to-point communication requests
  - $d \in (0,1]^A$: demands in transmission time
  - $w \in \mathbb{R}_+^A$: a weight function
- λ: number of channels
- Protocol interference model

Maximum-Weighted Feasible Subset (MWFS)

- *Feasibility of $F \subseteq A$:*

Maximum-Weighted Feasible Subset (MWFS)

- *Feasibility of* $F \subseteq A$:
  - $\exists$ a channel assignment $\pi : F \to [\lambda]$.

Maximum-Weighted Feasible Subset (MWFS)

- *Feasibility of* $F \subseteq A$:
  - $\exists$ a channel assignment $\pi : F \rightarrow [\lambda]$.
  - $\exists$ a transmission schedule of length ≤ 1 for F under π

Maximum-Weighted Feasible Subset (MWFS)

- *Feasibility of $F \subseteq A$*:
  - $\exists$ a channel assignment $\pi : F \to [\lambda]$.
  - $\exists$ a transmission schedule of length ≤ 1 for F under π
- **MWFS**: find a feasible $F \subseteq A$ maximizing $w(F) := \sum_{a \in F} w(a)$, and a witness

Maximum-Weighted Feasible Subset (MWFS)

- *Feasibility of $F \subseteq A$*:
  - $\exists$ a channel assignment $\pi : F \to [\lambda]$.
  - $\exists$ a transmission schedule of length ≤ 1 for F under π
- **MWFS**: find a feasible $F \subseteq A$ maximizing $w(F) := \sum_{a \in F} w(a)$, and a witness
  - channel assignment

Maximum-Weighted Feasible Subset (MWFS)

- *Feasibility of $F \subseteq A$*:
  - $\exists$ a channel assignment $\pi : F \to [\lambda]$.
  - $\exists$ a transmission schedule of length ≤ 1 for F under π
- **MWFS**: find a feasible $F \subseteq A$ maximizing $w(F) := \sum_{a \in F} w(a)$, and a witness
  - channel assignment
  - transmission schedule

Hardness & Challenges

- Maximum-Weight Independent Set of Links (**MWIS**): unit-demand

Hardness & Challenges

- Maximum-Weight Independent Set of Links (**MWIS**): unit-demand
  - NP-hard even with single channel

Hardness & Challenges

- Maximum-Weight Independent Set of Links (**MWIS**): unit-demand
  - NP-hard even with single channel
- **MWFS** is harder than **MWIS**:

Hardness & Challenges

- Maximum-Weight Independent Set of Links (**MWIS**): unit-demand
  - NP-hard even with single channel
- **MWFS** is harder than **MWIS**:
  - Single-channel, single-interference domain: **MWIS** is trivially solvable, **MWFS** $\equiv$ Knapsack

Hardness & Challenges

- Maximum-Weight Independent Set of Links (**MWIS**): unit-demand
  - NP-hard even with single channel
- **MWFS** is harder than **MWIS**:
  - Single-channel, single-interference domain: **MWIS** is trivially solvable, **MWFS** $\equiv$ Knapsack
- Feasibility test is NP-hard alone!

Impact of Spectral Non-Splitting

- **Instance**: $2\lambda - 1$ disjoint but mutually conflicting requests with uniform demand $\frac{\lambda}{2\lambda-1}$.

Impact of Spectral Non-Splitting

- **Instance**: $2\lambda - 1$ disjoint but mutually conflicting requests with uniform demand $\frac{\lambda}{2\lambda-1}$.
- **Non-splitting**: λ feasible requests

Impact of Spectral Non-Splitting

- **Instance**: $2\lambda - 1$ disjoint but mutually conflicting requests with uniform demand $\frac{\lambda}{2\lambda-1}$.
- **Non-splitting**: λ feasible requests
- **Splitting**: $2\lambda - 1$ feasible requests

Divide & Conquer

- **Division**:

$$A_1 \leftarrow \left\{ a \in A : d(a) \leq \frac{1}{2} \right\} \text{ // light requests}$$

$$A_2 \leftarrow \left\{ a \in A : d(a) > \frac{1}{2} \right\} \text{ // heavy requests}$$

- **Conquer**: Apply a ρ_i-approx. alg. to select a feasible subset F_i of A_i for $i = 1, 2$.
- **Combination**: return the heavier one.
    - $(\rho_1 + \rho_2)$-approximate solution.

Roadmap

- Problem Description
- **A Finer Treatment of Conflicts**
- Light Requests
- Heavy Requests
- Summary

A Finer Treatment of Conflicts

- $a \in A$

A Finer Treatment of Conflicts

- $a \in A$
- $N(a) := \{\text{requests having conflict with } a\}$

A Finer Treatment of Conflicts

- $a \in A$
- $N(a) :=$ {requests having conflict with a}
- $\dot{N}(a) :=$ {requests having primary conflict with a}

A Finer Treatment of Conflicts

- $a \in A$
- $N(a) :=$ {requests having conflict with a}
- $\dot{N}(a) :=$ {requests having primary conflict with a}
- $\ddot{N}(a) :=$ {requests having secondary conflict with a}

A Finer Treatment of Conflicts

- $a \in A$
- $N(a) :=$ {requests having conflict with a}
- $\dot{N}(a) :=$ {requests having primary conflict with a}
- $\ddot{N}(a) :=$ {requests having secondary conflict with a}
- $N^{in}(a) :=$ {requests interfering a}

A Finer Treatment of Conflicts

- $a \in A$
- $N(a) :=$ {requests having conflict with a}
- $\dot{N}(a) :=$ {requests having primary conflict with a}
- $\ddot{N}(a) :=$ {requests having secondary conflict with a}
- $N^{in}(a) :=$ {requests interfering a}
- $N^{out}(a) :=$ {requests interfered by a}

A Finer Treatment of Conflicts

$$\dot{N}[a] := \dot{N}(a) \cup \{a\}$$
$$\ddot{N}^{in}(a) := \ddot{N}(a) \cap N^{in}(a)$$
$$\ddot{N}^{out}(a) := \ddot{N}(a) \cap N^{out}(a)$$

Inward Local Independence number (ILIN)

- $\mu :=$ the maximum $\#$ of pairwise conflict-free requests in $N^{in}(a)$ for all $a \in A$.

Inward Local Independence number (ILIN)

- $\mu :=$ the maximum $\#$ of pairwise conflict-free requests in $N^{in}(a)$ for all $a \in A$.
  - constant-bounded in most settings

Inward Local Independence number (ILIN)

- $\mu :=$ the maximum $\#$ of pairwise conflict-free requests in $N^{in}(a)$ for all $a \in A$.
  - constant-bounded in most settings
- $\mu_\lambda := \mu + \left(1 - \frac{1}{\lambda}\right)$.

Roadmap

- Problem Description
- A Finer Treatment of Conflicts
- **Light Requests**
- Heavy Requests
- Summary

"Optimal" Partial Demand

$$\begin{array}{ll} \max & \sum_{a\in A} \frac{w(a)}{d(a)} x(a) \\ s.t. & x\left(\dot{N}[a]\right) + \frac{2}{\lambda} x\left(\ddot{N}^{in}(a)\right) \leq 1, \forall a \in A \\ & 0 \leq x(a) \leq d(a), \forall a \in A \end{array}$$

“Optimal” Partial Demand

$$
\begin{array}{ll}
\max & \sum_{a\in A}\frac{w(a)}{d(a)}x(a) \\
s.t. & x\left(\dot{N}[a]\right)+\frac{2}{\lambda}x\left(\ddot{N}^{in}(a)\right)\leq 1, \forall a\in A \\
 & 0\leq x(a)\leq d(a), \forall a\in A
\end{array}
$$

$$
\sum_{a\in A}\frac{w(a)}{d(a)}x(a)\geq\frac{1}{2\mu_{\lambda}}opt.
$$

Surplus-preserving Ordering

surplus-preserving ordering w.r.t. x

$B \leftarrow A$;
for $i = \lvert A \rvert$ **down to** 1 do
    $a_i \leftarrow$ an x-surplus request in B;
        $//x\left(\ddot{N}^{in}(a_i) \cap B\right) \geq x\left(\ddot{N}^{out}(a_i) \cap B\right)$.
    $B \leftarrow B \setminus \{a_i\}$;

return $\left\langle a_1, a_2, \cdots, a_{\lvert A \rvert} \right\rangle$

Demand-Aware Conflict Factors

$$\varrho(a,b)=\begin{cases}\frac{d(a)}{1-d(b)}, & \text{if } b\in\dot{N}(a);\\ \frac{1}{\lambda}\frac{d(a)}{1-d(b)}, & \text{if } b\in\ddot{N}(a);\\ 0, & \text{if } b\notin N(a).\end{cases}$$

Greedy Candidate Subset

greedy candidate subset S of A in $\prec$
$S \leftarrow \emptyset$; for each $a \in A$ in the *reverse* order of $\prec$ do $\quad \overline{w}(a) \leftarrow w(a) - \sum_{b \in S} \varrho(a,b)\, \overline{w}(b)$; $\quad$ if $\overline{w}(a) > 0$, $S \leftarrow S \cup \{a\}$;
return S.

Greedy Candidate Subset

greedy candidate subset S of A in $\prec$
$S \leftarrow \emptyset$; for each $a \in A$ in the *reverse* order of $\prec$ do $\quad \overline{w}(a) \leftarrow w(a) - \sum_{b \in S} \varrho(a,b)\, \overline{w}(b)$; $\quad$ if $\overline{w}(a) > 0$, $S \leftarrow S \cup \{a\}$;
return S.

$$\overline{w}(S) \geq \frac{1}{2} \sum_{a \in A} \frac{w(a)}{d(a)} x(a).$$

Inductive Feasibility

$$\max_{a \in F} \varrho \left(F_{\prec a}, a \right) \leq 1$$

Inductive Feasibility

$$\max_{a \in F} \varrho \left(F_{\prec a}, a \right) \leq 1$$

$$\Delta \left(F \right) := \max_{a \in F} \left[d\left(a \right) + d\left(\dot{N}\left(a \right) \cap F_{\prec a} \right) + \frac{1}{\lambda} d\left(\ddot{N}\left(a \right) \cap F_{\prec a} \right) \right] \leq 1$$

Maximal Inductively Feasible Subset

maximal inductively feasible subset $F \subseteq S$ in $\prec$
$F \leftarrow \emptyset$; for each $a \in S$ in $\prec$ do   if $\varrho(F, a) \leq 1$ then $F \leftarrow F \cup \{a\}$;
return F

maximal inductively feasible subset $F \subseteq S$ in $\prec$
$F \leftarrow \emptyset$; for each $a \in S$ in $\prec$ do   if $\varrho(F, a) \leq 1$ then $F \leftarrow F \cup \{a\}$;
return F

$$w(F) \geq \overline{w}(S).$$

Greedy Channel Assignment

greedy channel assignment π to F in $\prec$
for each $a \in F$ in $\prec$ do $\quad\pi(a) \leftarrow$ the *first* channel with the *least* **secondary** congestion;
return π

Greedy Channel Assignment

greedy channel assignment π to F in $\prec$
for each $a \in F$ in $\prec$ do $\quad\pi(a) \leftarrow$ the *first* channel with the *least* **secondary** congestion;
return π

$$\Delta_\pi(F) := \max_{a\in F}\left[d(a) + d\left(\dot{N}(a) \cap F_{\prec a}\right) + \sum_{b\in \ddot{N}(a)\cap F_{\prec a}:\pi(b)=\pi(a)} d(b)\right]$$
$$\leq \Delta(F) \leq 1$$

Greedy Transmission Scheduling

First-Fit schedule of F in $\prec$ under π
for each $a \in F$ in $\prec$ do $\quad$ a transmits over channle $\pi(a)$ in the first available periods.

Greedy Transmission Scheduling

First-Fit schedule of F in $\prec$ under π
for each $a \in F$ in $\prec$ do $\quad$ a transmits over channle $\pi(a)$ in the first available periods.

schedule length $\leq \Delta_{\pi}(F) \leq 1$

Putting Together

1. Compute an optimal partial demand x and a surplus-preserving ordering $\prec$ of A w.r.t. x.
2. Compute the greedy candidate subset S of A in $\prec$.
3. Compute the maximal inductively feasible subset F of S in $\prec$.
4. Compute the greedy channel assignment π to F in $\prec$.
5. Compute the greedy schedule of F in $\prec$ under π.

Putting Together

1. Compute an optimal partial demand x and a surplus-preserving ordering $\prec$ of A w.r.t. x.
2. Compute the greedy candidate subset S of A in $\prec$.
3. Compute the maximal inductively feasible subset F of S in $\prec$.
4. Compute the greedy channel assignment π to F in $\prec$.
5. Compute the greedy schedule of F in $\prec$ under π.

$$w(F) \geq \overline{w}(S) \geq \frac{1}{2} \sum_{a \in A} \frac{w(a)}{d(a)} x(a) \geq \frac{1}{4\mu_\lambda} opt.$$

Roadmap

- Problem Description
- A Finer Treatment of Conflicts
- Light Requests
- **Heavy Requests**
- Summary

Modifications

- Round-up all demands to **one**

Modifications

- Round-up all demands to **one**
- Simplified conflict factors

$$\varrho(a,b) = \begin{cases} 1, & \text{if } b \in \dot{N}(a); \\ \frac{1}{\lambda}, & \text{if } b \in \ddot{N}(a); \\ 0, & \text{if } b \notin N(a). \end{cases}$$

Modifications

- Round-up all demands to **one**
- Simplified conflict factors

$$\varrho(a,b)=\begin{cases}1, & \text{if } b\in \dot{N}(a);\\ \frac{1}{\lambda}, & \text{if } b\in \ddot{N}(a);\\ 0, & \text{if } b\notin N(a).\end{cases}$$

- Same algorithm with better approximation bound

$$w(F)\geq \overline{w}(S)\geq \sum_{a\in A} w(a)\,x(a)\geq \frac{1}{2\mu_\lambda}opt.$$

Roadmap

- Problem Description
- A Finer Treatment of Conflicts
- Light Requests
- Heavy Requests
- **Summary**

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation
- Restriction: inductive feasibility, inductive compatibility

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation
- Restriction: inductive feasibility, inductive compatibility
- Fractional local-ratio (primal-dual)

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation
- Restriction: inductive feasibility, inductive compatibility
- Fractional local-ratio (primal-dual)
  - optimal partial demand via LP and surplus-preserving ordering

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation
- Restriction: inductive feasibility, inductive compatibility
- Fractional local-ratio (primal-dual)
  - optimal partial demand via LP and surplus-preserving ordering
  - greedy candidate set

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation
- Restriction: inductive feasibility, inductive compatibility
- Fractional local-ratio (primal-dual)
  - optimal partial demand via LP and surplus-preserving ordering
  - greedy candidate set
  - maximal inductively feasible subset

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation
- Restriction: inductive feasibility, inductive compatibility
- Fractional local-ratio (primal-dual)
  - optimal partial demand via LP and surplus-preserving ordering
  - greedy candidate set
  - maximal inductively feasible subset
- Greedy channel assignment

Summary

- Divide & Conquer: $6\mu_\lambda$-approximation
- Restriction: inductive feasibility, inductive compatibility
- Fractional local-ratio (primal-dual)
  - optimal partial demand via LP and surplus-preserving ordering
  - greedy candidate set
  - maximal inductively feasible subset
- Greedy channel assignment
- First-fit transmission schedule