Distributed Scheduling for Multi-Hop Wireless Networks

Nachiket Sahasrabudhe, Joy Kuri

Centre for Electronics Design and Technology, Indian Institute of Science, Bangalore
Network Utility Maximization (NUM) Problem

Given the end-to-end flows that are going to operate in the network, our aim is to maximize the aggregate network utility.
Network Model

- A network is represented by a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$.
- Multiple end-to-end multi-hop flows are present in the network.
- All source nodes are saturated.
- If $x_f$ is the data-rate associated with a flow $f$, we associate a nondecreasing, concave utility function $U(x_f)$, with $U(0) = 0$, with every flow $f$ in the network.
Interference in Wireless Scenario

- Activity on one link affects the activity on the other.
- Only a subset of links can be activated simultaneously.
- Link operates at effective link capacity which is lesser than the actual link capacity.
- $K$-hop interference model: $K + 1$ consecutive links interfere with each other.
Distributed Scheduling for Multi-Hop Wireless Networks

Nachiket Sahasrabudhe, Joy Kuri

Problem Background
System Model
Interference Model

Mathematical formulation
Convex Optimization Problem

Scheduling Problem
Distributed Scheduling
Greedy Heuristic

Numerical Evaluation

Figure: A wireless network

Figure: Independent set of links:
\[ l_1 = [(0, 1), (6, 7)], l_2 = [(4, 7), (3, 5)], l_3 = [(0, 3), (6, 7)], l_4 = [(1, 4), (5, 6)] \]
Primal Problem

- Associate a concave utility function $U_f(x_f)$, with each end-to-end flow.
- Network Utility Maximization (NUM) Problem:

$$
\text{maximize: } f(x) = \sum_{f \in F} U(x_f)
$$

Subject to:

$$
\mathbf{A}\mathbf{y}_f = \mathbf{u}_f, \forall f \in F \tag{1}
$$

$$
\sum_{f \in F} y_f \leq M_i a_i \tag{2}
$$

$$
\sum_{i=1}^{K} a_i = 1 \tag{3}
$$

$$
\mathbf{x} \geq 0 \tag{4}
$$

- Convex optimization problem with affine constraints: Duality gap is zero.

$\nabla$
A solution to the congestion control and routing problem

\[ D_1(p) = \max \sum_{f \in F} (U(x_f) - p_R(f)x_f) \]  
subject to: \( y_f \geq 0 \), \( \forall f \in F \)  
\( x_f \geq 0 \), \( \forall f \in F \).

The data is routed along the path for which the path price is minimum.

Congestion control:

\[ x_f = U'^{-1}(p_{R_{\text{min}}}(f)), \forall f \in F \]
Utility Function

\[ U(x_f) = \begin{cases} 
\text{P1.x}_f & \text{if } x_f \leq x_1 \\
\text{P2.x}_f & \text{if } x_f > x_1
\end{cases} \]

\[ x_2 = 0 \]

\[ x_1 \]

\[ x_f \]
Optimal Schedule

- Optimal schedule $\mathbf{a}$, maximizes $\mathbf{p}^T \mathbf{c}_{\text{eff}}$

$$D_2(\mathbf{p}) = \max_{i=K} (\mathbf{p}^T \mathbf{M}_i \mathbf{a}) \quad (7)$$

subject to: $$\sum_{i=1}^{K} a_i = 1$$

$$\mathbf{a} \geq 0$$

- Columns of $\mathbf{M}_i$ represent the independent sets in the network.

- $M_{ij} = 1$ if link $i$ is part of $j^{th}$ independent set, else $M_{ij} = 0$.

- The fraction of time a $j^{th}$ independent set is activated is given by $a_j$. 
Optimal scheduling scheme

- The optimal schedule for a given price vector is maximum weighted independent set.
- Optimal scheduling algorithm needs a centralized entity that is aware of the network topology.
- Finding an optimal schedule is an NP-hard problem.
Distributed Greedy Heuristics

- **Key notion:** Schedule an independent set containing a maximum weighted link, rather than scheduling a maximum weighted independent set.

- **Greedy Algorithm:**
  1. Price dissemination over 2-hop neighbourhood.
  2. Identification of interfering links.
  3. Price sorting.
  4. Link scheduling.
  5. Price updation.
Solution of the Dual problem by $\varepsilon$-subgradient algorithm

- Price updation equation:
  \[
  p = p - \delta h_\varepsilon(p) \tag{8}
  \]
  \[
  p_l[j + 1] = (p_l[j] - \delta(c_l[j] - y_l[j]))^+ \tag{9}
  \]

Definition
Given a convex function $D(p) : \mathbb{R}^n \to \mathbb{R}$ and $\varepsilon \geq 0$, a vector $h(p_0) \in \mathbb{R}^n$ is an $\varepsilon$-subgradient of $D(p)$ at point $p_0 \in \mathbb{R}^n$ if $D(p) \geq D(p_0) - \varepsilon + (p - p_0)^T h(p_0)$, $\forall p \in \mathbb{R}^n$.

- $h_\varepsilon(p) = c_\varepsilon(p) - y(p) \tag{10}$
- $\varepsilon = 0$ case represents a case of the optimal solution.
Degree of Suboptimality

- If we select an independent set in some arbitrary manner, can we guarantee that the solution of Network Utility Maximization problem will approach the optimal solution?
Proposition

There exist an $0 < M < \infty$ and $j_0 < \infty$, such that $p_{\text{max}}[j] \leq M \ \forall j \geq j_0$, under any scheduling policy that schedules an independent set in each time slot.

Proposition

Suppose at each iteration $j$ an $\epsilon_j$-subgradient is used. If $\epsilon_j \leq \epsilon_0 \ \forall j$ or $\lim_{j \to \infty} \epsilon_j = \epsilon_0$ and $\|h(j)\|_2 \leq H \ \forall j$, then $\epsilon$-subgradient algorithm converges within $\delta H^2 / 2 + \epsilon_0$ of the optimal value.
$p_{max}$ remains upper bounded

![Diagram](image)

Figure: An example illustrating variation of optimal $x_f$ under different values of $S_{min}$.

\[ p_{l}[j + 1] = (p_{l}[j] - \delta(c_{l}[j] - y_{l}[j]))^+ \]  

(11)
An example of interference degree $d_K(G)$, for $K = 2$

**Figure:** A network graph

- Set of links that interfere with link $l_{(1,2)}$: $\{l_{(0,1)}, l_{(1,3)}, l_{(2,4)}, l_{(3,4)}, l_{(4,5)}\}$.
- $d_K(l_{(1,2)}) = 2$.
- $d_K(G) = \max_{l \in \mathcal{L}} d_K(l) = 2$
Claim
Consider a line graph, with sufficiently large number of links $L$, each with capacity $C$. Then the path-price $S$ satisfies the inequality

$$\frac{K + 1}{K + 1 + C} \leq S \leq \frac{2K + 1}{2K + 1 + C} \quad (12)$$

Proposition
Under the same set of assumptions for utility function, interference model and number of end-to-end flows, as for the claim above, in an arbitrary network graph $S \leq 1$ under any scheduling policy that schedules an independent set in each time slot.
Proposition

Any scheduling algorithm that schedules a set of non-interfering links at each time instant leads to $\epsilon$-subgradient algorithm.

Proposition

The greedy scheduling policy leads to $LC \frac{d_K(G)^{-1}}{d_K(G)}$ - subgradient.
Numerical Evaluation

Figure: Network $G_1$. Flow is present between following pairs of nodes: $(0, 4), (5, 1), (3, 6)$. 
Figure: Network $G_2$. Flow is present between following pairs of nodes: $(0, 1), (5, 4), (6, 1)$. 
Utility Plot

Figure: Utility function under optimal and greedy schedules for networks $G_1$ and $G_2$. 
Figure: $p_{\text{max}}$ remains bounded in networks $G_1$ and $G_2$
THANK YOU!