Online Generalized Network Design Under (Dis)Economies of Scale

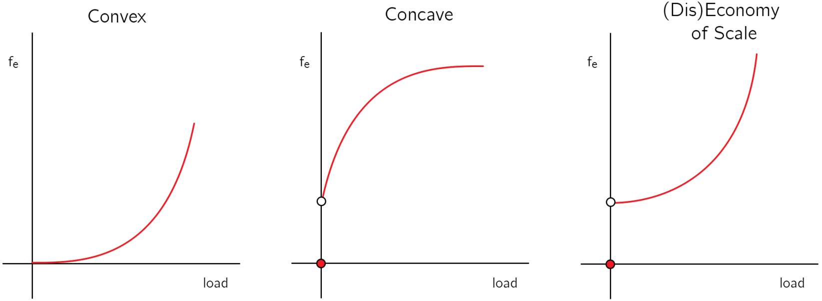
Viswanath Nagarajan, Lily Wang

Industrial and Operations Engineering
University of Michigan

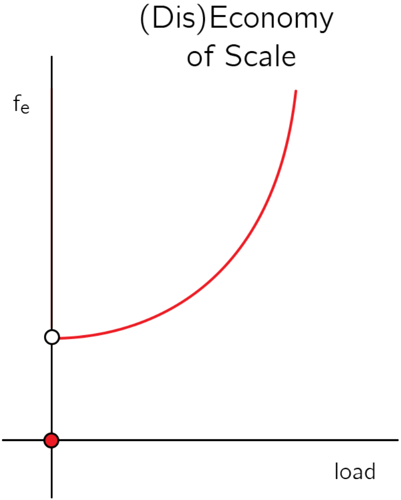
viswa@umich.edu
lilyxy@umich.edu

Online Multicommodity Flow

- Directed graph $G = (V, E)$
- Each request i has a
  - source s_i
  - destination t_i
  - demand $d_i \geq 1$
- Energy costs are sum of cost functions $f_e(\ell_e)$, where ℓ_e is the total demand going through edge e
- Goal: minimize total energy cost to route requests



(D)oS Function



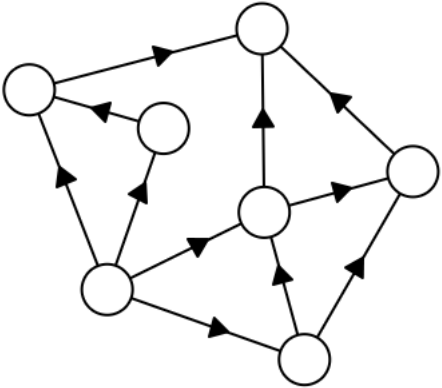
In our setting, each edge cost is

$$f_e(\ell_e) = \begin{cases} 0 & \text{if } \ell_e = 0 \\ \sigma_e + \ell_e^{\alpha_e} & \text{if } \ell_e > 0 \end{cases}$$

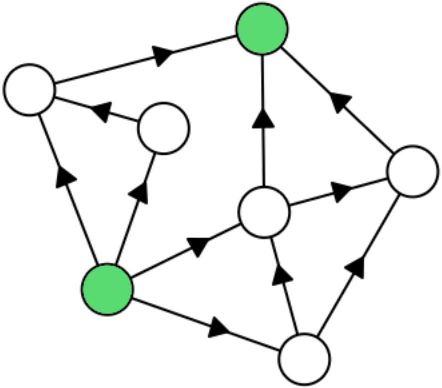
We define $q_e = \sigma_e^{1/\alpha_e}$ and

$$q = \max_{e \in E} q_e$$

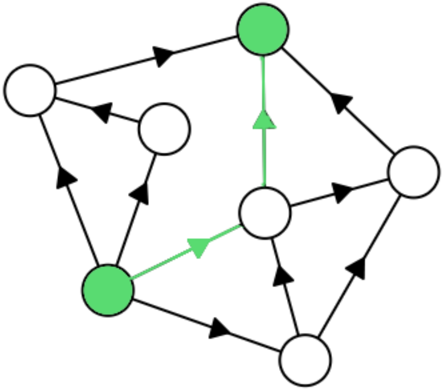
Online Multicommodity Flow Example



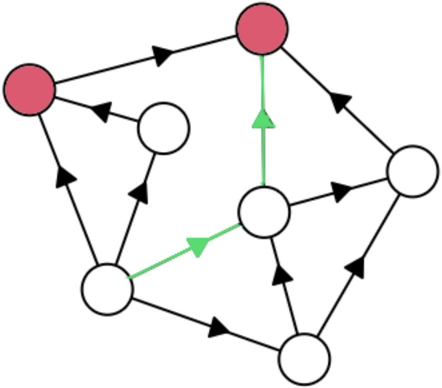
Online Multicommodity Flow Example



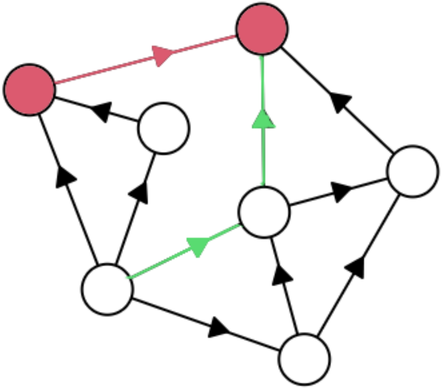
Online Multicommodity Flow Example



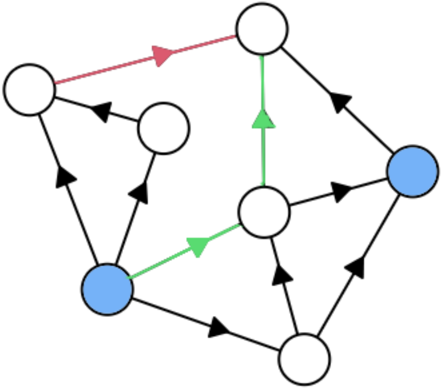
Online Multicommodity Flow Example



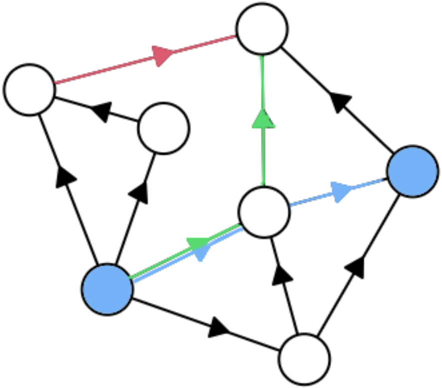
Online Multicommodity Flow Example



Online Multicommodity Flow Example



Online Multicommodity Flow Example



Previous Results: Multicommodity Flow

Function	Offline	Online
Concave	[Chekuri, Hajiaghayi, Kortsarz, Salavatipour, '10]	[Awerbuch, Azar '97]
Convex	[Andrews, Anta, Zhang, Zhao '12] [Makarychev, Sviridenko '18]	[Gupta, Krishnaswamy, Pruhs '12]
(D)oS	[Andrews, Antonakopoulos, Zhang '16]	[Antoniadis, Im, Krishnaswamy, Moseley, Nagarajan, Pruhs, Stein, '20]

- All the above results are for undirected networks.
- Recently [Emek, Kutten, Lavi, Shi '20] obtained (offline) approximation algorithms even for directed networks.

Result for Multicommodity flow

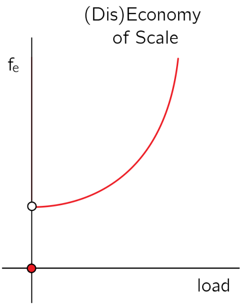
Theorem

Deterministic online algorithm for multicommodity flow with competitive ratio $O(q + (\mathrm{e}\alpha)^{\alpha})$ where cost functions are (Dis)Economies of scale.

- First online result for (D)oS costs in directed graphs
- Matches the *offline* bound of [Emek, Kutten, Lavi, Shi '20]
- Tight result in online setting

Our Setting - Online Generalized Network Design

- Input: set of E resources and N requests
- Each request i is associated with replies $\mathcal{P}_i$ which can satisfy i and weight vector w_i with $w_{i,e}$ representing load on resource e.
- Load on each resource e is $\ell_e = \sum_{i:e \in p_i} w_{i,e}$
- Want to minimize total cost $\sum_{e \in E} f_e(\ell_e)$ with (D)oS functions



Our Results

Theorem (upper bound)

There is a polynomial time $O(q\tau + (e\alpha\tau)^\alpha)$-competitive deterministic online algorithm for GND assuming a τ-approximation algorithm for the min-cost oracle.

- The min cost oracle finds a τ-approximate min-cost reply from $\mathcal{P}_i$ for any single request i efficiently.
- The previous online multicommodity flow competitive ratio of $O(q + (e\alpha)^\alpha)$ follows from this result and having an exact min-cost oracle for online multicommodity flow.

Theorem (lower bound)

Any deterministic online algorithm has competitive ratio $\Omega(q + (1.44\alpha)^\alpha)$.

Overview

- Simplify (D)oS cost functions
- Convex program and dual
- Fractional online algorithm
- Integrality gap
- Integer online algorithm
- Lower bound

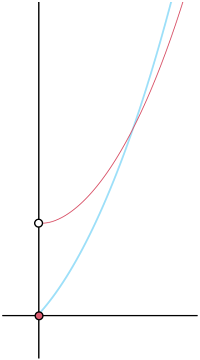
Simplifying DoS cost functions

We model (D)oS cost functions as

$$f(x) = \begin{cases} 0 & \text{if } x = 0 \\ \sigma + x^{\alpha} & \text{if } x > 0 \end{cases},$$

Can approximate by a convex power function by a loss factor of $2q$, with $q := \sigma^{1/\alpha}$ using the function

$$h(x) := q^{\alpha-1} \cdot x + x^{\alpha}$$



Simplifying (D)oS cost functions

Lemma

For $x \in \{0\} \cup \mathbb{R}_{\geq 1}$*, we have*

$$\frac{1}{2} \cdot h(x) \leq f(x) \leq \max\{q, 1\} q^{\alpha - 1} \cdot x + x^{\alpha} = \max\{q, 1\} \cdot h(x).$$

Proof.

Simplifying (D)oS cost functions

Lemma

For $x \in \{0\} \cup \mathbb{R}_{\geq 1}$, we have

$$\frac{1}{2} \cdot h(x) \leq f(x) \leq \max\{q, 1\} q^{\alpha-1} \cdot x + x^{\alpha} = \max\{q, 1\} \cdot h(x).$$

Proof.

If $x < q$, $h(x) = q^{\alpha-1} \cdot x + x^{\alpha} \leq q^{\alpha} + x^{\alpha} = \sigma + x^{\alpha} = f(x)$.

Simplifying (D)oS cost functions

Lemma

For $x \in \{0\} \cup \mathbb{R}_{\geq 1}$, we have

$$\frac{1}{2} \cdot h(x) \leq f(x) \leq \max\{q, 1\} q^{\alpha-1} \cdot x + x^{\alpha} = \max\{q, 1\} \cdot h(x).$$

Proof.

If $x < q$, $h(x) = q^{\alpha-1} \cdot x + x^{\alpha} \leq q^{\alpha} + x^{\alpha} = \sigma + x^{\alpha} = f(x)$.
If $x \geq q$, $h(x) = q^{\alpha-1} \cdot x + x^{\alpha} \leq 2x^{\alpha} \leq 2(x^{\alpha} + \sigma) = 2f(x)$

Simplifying (D)oS cost functions

Lemma

For $x \in \{0\} \cup \mathbb{R}_{\geq 1}$, we have

$$\frac{1}{2} \cdot h(x) \leq f(x) \leq \max\{q, 1\} q^{\alpha-1} \cdot x + x^{\alpha} = \max\{q, 1\} \cdot h(x).$$

Proof.

If $x < q$, $h(x) = q^{\alpha-1} \cdot x + x^{\alpha} \leq q^{\alpha} + x^{\alpha} = \sigma + x^{\alpha} = f(x)$.
If $x \geq q$, $h(x) = q^{\alpha-1} \cdot x + x^{\alpha} \leq 2x^{\alpha} \leq 2(x^{\alpha} + \sigma) = 2f(x)$

For the second inequality, we have

$$\max\{q, 1\} q^{\alpha-1} \cdot x + x^{\alpha} \geq q^{\alpha} \cdot x + x^{\alpha} = \sigma x + x^{\alpha} \geq \sigma + x^{\alpha} = f(x).$$

□

Algorithm for Convex Sums of Powers

- We simplified (D)oS into sum of power functions (convex)

Algorithm for Convex Sums of Powers

- We simplified (D)oS into sum of power functions (convex)
- Consider cost functions $g_e(\ell_e) = c_e \cdot \ell_e^{\alpha_e}$
  Assume that $\alpha_e \geq 1$

Algorithm for Convex Sums of Powers

- We simplified (D)oS into sum of power functions (convex)
- Consider cost functions $g_e(\ell_e) = c_e \cdot \ell_e^{\alpha_e}$
  Assume that $\alpha_e \geq 1$
- Obtain an α^α competitive algorithm for a fractional relaxation.

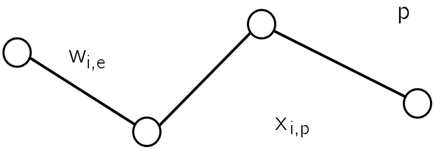
Algorithm for Convex Sums of Powers

- We simplified (D)oS into sum of power functions (convex)
- Consider cost functions $g_e(\ell_e) = c_e \cdot \ell_e^{\alpha_e}$
  Assume that $\alpha_e \geq 1$
- Obtain an α^α competitive algorithm for a fractional relaxation.
- Extend to get $(e\alpha)^\alpha$ competitive algorithm for the integral problem

Fractional Online Algorithm

Convex program relaxation for GND:

$$
\begin{aligned}
\min \quad & \sum_{e \in E} c_e \cdot \left(\sum_{i=1}^{N} w_{i,e} \sum_{p \in \mathcal{P}_i : e \in p} x_{i,p} \right)^{\alpha_e} \\
\text{s.t.} \quad & \sum_{p \in \mathcal{P}_i} x_{i,p} \geq 1, \qquad \forall i \in [N] \\
& x \geq 0.
\end{aligned}
$$



Fractional Online Algorithm

For the analysis, will use dual convex program for GND:

$$\begin{aligned} \max \quad & \sum_{i=1}^{N} y_i - \sum_{e \in E \setminus E_1} \frac{c_e \alpha_e}{\beta_e} \cdot z_e^{\beta_e} \\ \text{s.t.} \quad & \sum_{e \in p} w_{i,e} c_e \alpha_e \cdot z_e \geq y_i, \qquad && \forall p \in \mathcal{P}_i, \forall i \in [N] \\ & z_e \leq 1, && \forall e \in E_1 \\ & \mathrm{y}, \mathrm{z} \geq 0. \end{aligned}$$

Lemma

For any primal $x \in (P)$ and dual $(y, z) \in (D)$ solutions,

$$\sum_{e \in E} c_e \cdot \left(\sum_{i=1}^{N} w_{i,e} \sum_{p \in \mathcal{P}_i : e \in p} x_{i,p} \right)^{\alpha_e} \geq \sum_{i=1}^{N} y_i - \sum_{e \in E \setminus E_1} \frac{c_e \alpha_e}{\beta_e} \cdot z_e^{\beta_e}.$$

Fractional Online Algorithm

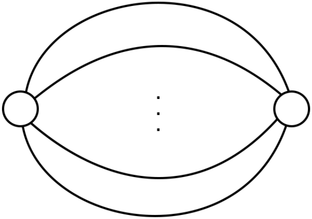
Upon arrival of request i, do the following:
For each continuous time $t \in [0, 1]$:

- Choose reply $p^* \in \mathcal{P}_i$ using the min-cost oracle under costs $d_e = \alpha_e c_e \cdot \ell_e^{\alpha_e - 1} \cdot w_{i,e}$ for each $e \in E$, where ℓ_e is the current fractional load on e.
- Raise primal variable x_{i,p^*} at rate one, i.e. $\frac{\partial}{\partial t} x_{i,p^*} = 1$

The fractional online algorithm has competitive ratio at most α^α where $\alpha = \max_{e \in E} \alpha_e$.

Integrality Gap

Example where fractional relaxation has a large integrality gap:



Let all costs be $f(x) = x^2$ with a single request, and n edges. If we divide the request up uniformly, total cost is $1/n$. In the integer setting, the total cost is 1.

Stronger Relaxation

Convex program relaxation for GND:

$$
\begin{aligned}
\min \quad & \sum_{e \in E} c_e \cdot \left(\sum_{i=1}^{N} w_{i,e} \sum_{p \in \mathcal{P}_i : e \in p} x_{i,p} \right)^{\alpha_e} + \sum_{e \in E} \frac{c_e \alpha_e}{e^{\alpha}} \cdot \sum_{i=1}^{N} w_{i,e}^{\alpha_e} \sum_{p \in \mathcal{P}_i : e \in p} x_{i,p} \\
\text{s.t.} \quad & \sum_{p \in \mathcal{P}_i} x_{i,p} \geq 1, \qquad \forall i \in [N] \\
& x \geq 0.
\end{aligned}
$$

- Optimal convex program value at most $(1 + \alpha e^{-\alpha}) \cdot OPT \leq 2 \cdot OPT$.

Integer Online Algorithm

Upon the arrival of request i, we do the following:

- Choose reply $p_i \in \mathcal{P}_i$ using the min-cost oracle under the costs

$$\psi_e = \alpha_e c_e \cdot \ell_e^{\alpha_e - 1} \cdot w_{i,e} + \frac{\rho}{\mathrm{e}^{\alpha}} \cdot c_e \alpha_e w_{i,e}^{\alpha_e}, \text{ for each } e \in E,$$

where $\ell_e := \sum_{j<i:e\in p_j} w_{j,e}$ is the current load on e.

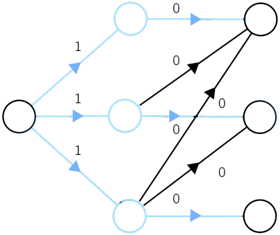
Lower Bound

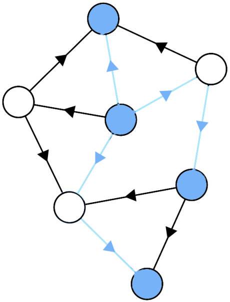
Machine Scheduling

- $\Omega((1.44\alpha)^{\alpha})$ lower bound from restricted assignment scheduling with ℓ_p-norm load balancing (Caragiannis, 2008)

Online Directed Steiner Tree

- Similar idea as $\Omega(q)$ lower bound (Faloutsos, Pankaj, Sevcik, 2002)





Conclusion

Conclusion

- GND and multicommodity flow

Conclusion

- GND and multicommodity flow
- Convex program and dual

Conclusion

- GND and multicommodity flow
- Convex program and dual
- Fractional and Integer algorithms
- Future work: primal-dual algorithms with convex programs for more general problems?

Conclusion

- GND and multicommodity flow
- Convex program and dual
- Fractional and Integer algorithms
- Future work: primal-dual algorithms with convex programs for more general problems?
- Thanks for watching!