A Strongly Polynomial Algorithm for the Minimum Cost Generalized Flow Problem

Zhuan Khye (Cedric) Koh

Centrum Wiskunde & Informatica

Joint work with

Daniel Dadush, Bento Natura, Neil Olver, László A. Végh

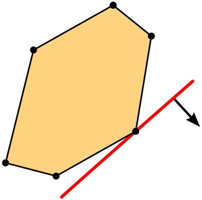
Talk Overview

1. Strongly Polynomial Landscape of Linear Programming
2. LPs with ≤ 2 variables per Inequality
3. Minimum Cost Generalized Flow
4. A Strongly Polynomial Interior Point Method
5. Some Proof Ideas

Linear Program (LP)

Primal:

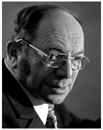
$$\min \ c^\top x$$
$$\text{s.t.} \ Ax = b$$
$$x \geq \mathbf{0}$$

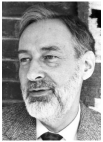


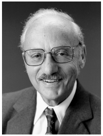
Dual:

$$\max \ b^\top y$$
$$\text{s.t.} \ A^\top y \leq c$$

- Introduced by [Kantorovich '39] [Hitchcock '41] [Koopmans '42] [Dantzig '47].





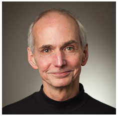


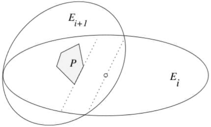
LP Algorithms

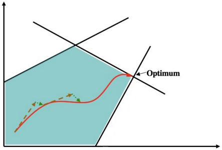
Input: $A \in \mathbb{R}^{n \times m}$, $b \in \mathbb{R}^n$, $c \in \mathbb{R}^m$. Total bit length L.

Def: A polynomial algorithm runs in $\mathrm{poly}(m, n, L)$ time.

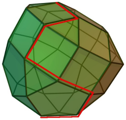
- Polynomial algorithms for LP:
  - Ellipsoid method [Khachiyan '79]
  - Interior point method [Karmarkar '84] [Renegar '88]







- Simplex method [Dantzig '47]
  - Not known to be polynomial, but efficient in practice.



Strongly Polynomial

Input: $A \in \mathbb{R}^{n \times m}$, $b \in \mathbb{R}^n$, $c \in \mathbb{R}^m$. Total bit length L.

Def: An algorithm is strongly polynomial if it uses

1. $\mathrm{poly}(m, n)$ elementary arithmetic operations $(+, -, \times, \div, <?)$, and
2. $\mathrm{poly}(m, n, L)$ space.

Smale's 9th Problem [Megiddo '83]

Is there a strongly polynomial algorithm for linear programming?

The Zoo of LP Subclasses

General LP $\equiv$ LP with ≤ 3 variables per inequality

Strongly polynomial (Since 2024) (Before 2024)

Combinatorial LP:

- Shortest path
- Bipartite matching
- Maximum flow
- Minimum cost flow

- LP feasibility with ≤ 2 variables per inequality
- Discounted MDP
- Maximum generalized flow
- LP with ≤ 2 variables per inequality

- Undiscounted MDP
- LP with ≤ 2 variables per inequality

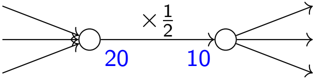
2-Variables-per-Inequality LP

- [Hochbaum '04] Every 2-variables-per-inequality LP can be reduced to

$$\begin{aligned} \min\ & c^\top x \\ \text{s.t.}\ & \sum_{e\in\delta^{\mathrm{in}}(v)} \gamma_e x_e - \sum_{e\in\delta^{\mathrm{out}}(v)} x_e = b_v \quad \forall v \in V \\ & x \geq \mathbf{0} \end{aligned}$$

Interpretation: Given directed graph $G = (V, E)$, node demands $b \in \mathbb{R}^V$, arc costs $c \in \mathbb{R}^E$ and gain factors $\gamma \in \mathbb{R}^E_{>0}$,

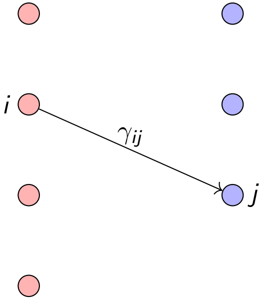
Find a **minimum cost generalized flow** satisfying all node demands.



Models leaky pipes, currency exchange etc.

Example: Production with Different Machines

- Variant of a problem proposed by Kantorovich in his 1939 paper introducing Linear Programming.



M: machines P: parts

- Machine i can produce γ_{ij} units of part j in one day at cost c_{ij}.

- Daily demand d_j for part j.

$$
\begin{aligned}
\min \quad & \sum_{i \in M, j \in P} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{j \in P} x_{ij} \leq 1 \qquad \forall i \in M \\
& \sum_{i \in M} \gamma_{ij} x_{ij} \geq d_j \quad \forall j \in P \\
& x \geq \mathbf{0}
\end{aligned}
$$

Previous Algorithms for Generalized Flow

- Algorithms for **primal** feasibility:
  - Polynomial [Goldberg, Plotkin, Tardos '91]
  - Strongly polynomial [Végh '13] [Olver, Végh '20]
- Algorithms for **dual** feasibility:
  - Polynomial [Aspvall, Shiloach '80]
  - Strongly polynomial [Megiddo '83] [Cohen, Megiddo '94] [Hochbaum, Naor '94] [Dadush, K, Natura, Végh '21] [Karczmarz '22]
- Algorithms for optimization:
  - Polynomial [Wayne '02]

Main Result

Theorem [Dadush, K, Natura, Olver, Végh '24]

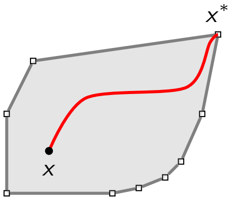
There is a strongly polynomial algorithm for the minimum cost generalized flow problem, and consequently, for LPs with at most 2 variables per inequality.

- The algorithm is the interior point method by [Allamigeon, Dadush, Loho, Natura, Végh '22].
- What we'll need for this talk:
  1. Interior point method
  2. Straight line complexity

Central Path

• For each $\mu > 0$, there exists a unique optimal solution $x^{\mathrm{cp}}(\mu)$ to

$$\begin{aligned} &\min\ c^\top x - \mu \sum_{i=1}^{n} \log(x_i) \\ &\text{s.t. } Ax = b. \end{aligned}$$



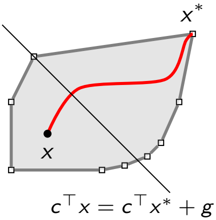
Def: The central path is the curve

$$\{x^{\mathrm{cp}}(\mu) : \mu > 0\}.$$

• As $\mu \to 0$, $x^{\mathrm{cp}}(\mu)$ converges to an optimal solution x^* of the LP.

• Interior Point Method (IPM): Walk down the central path with geometrically decreasing μ.

Max Central Path



- Let us reparameterize x^{cp} by the optimality gap:

$$c^\top x^{\mathrm{cp}}(g) = c^\top x^* + g \qquad \forall g \geq 0.$$

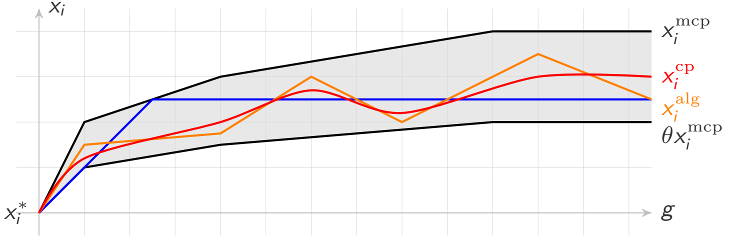
- For every $g \geq 0$ and $i \in [m]$, define

$$\begin{aligned} x_i^{\mathrm{mcp}}(g) := \max\ & x_i \\ \text{s. t.}\ & x \text{ feasible} \\ & \text{optimality gap} \leq g. \end{aligned}$$

Def: The max central path is the curve $\{x^{\mathrm{mcp}}(g) : g \geq 0\}$,

Theorem: $\frac{1}{2m}\, x^{\mathrm{mcp}} \leq x^{\mathrm{cp}} \leq x^{\mathrm{mcp}}$.

Straight Line Complexity



- IPM generates a piecewise-affine curve x^{alg} near the central path

$$\theta x^{\mathrm{mcp}} \leq x^{\mathrm{alg}} \leq x^{\mathrm{mcp}}.$$

Def: The straight line complexity of x_i^{mcp}, $\mathrm{SLC}_\theta(x_i^{\mathrm{mcp}})$, is the minimum number of pieces of a continuous piecewise-affine function h such that

$$\theta x_i^{\mathrm{mcp}} \leq h \leq x_i^{\mathrm{mcp}}.$$

Straight Line Complexity

- # iterations required by any IPM is at least

$$\max_{i \in [m]} \mathsf{SLC}_\theta(x_i^{\mathrm{mcp}}).$$

Theorem [Allamigeon, Dadush, Loho, Natura, Végh '22]

There is an interior point method which solves LP in

$$O\left(\min_{\theta \in (0,1]} \sqrt{m} \log\left(\frac{m}{\theta}\right) \sum_{i=1}^{m} \mathsf{SLC}_\theta(x_i^{\mathrm{mcp}})\right)$$

iterations.

Main Result

Theorem [Dadush, K, Natura, Olver, Végh '24]

For the minimum-cost generalized flow problem on $G = (V, E)$ with n nodes and m arcs,

$$\mathsf{SLC}_{\frac{1}{m}}(x_e^{\mathrm{mcp}}) = O(mn \log(mn)) \qquad \forall e \in E.$$

- Key ingredient: **Circuits**

Theorem [Dadush, K, Natura, Olver, Végh '24]

There is a strongly polynomial algorithm for the minimum cost generalized flow problem.

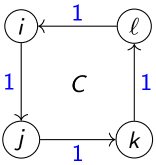
Circuits

Def: Let $W = \ker(A)$. A circuit is any vector $f \in W \setminus \{\mathbf{0}\}$ such that $\nexists h \in W \setminus \{\mathbf{0}\}$ with $\mathrm{supp}(h) \subsetneq \mathrm{supp}(f)$.

Example: Network flow

$$Ax = b \qquad \Longleftrightarrow \qquad \sum_{e \in \delta^{\mathrm{in}}(v)} x_e - \sum_{e \in \delta^{\mathrm{out}}(v)} x_e = b_v \quad \forall v \in V$$

- $\ker(A)$ = set of circulations. Circuits correspond to directed cycles.



$$\begin{array}{cc} & \begin{array}{cccc} 1 & 1 & 1 & 1 \end{array} \\ \begin{array}{c} i \\ j \\ k \\ \ell \end{array} & \boxed{\begin{array}{cccc} -1 & & & 1 \\ 1 & -1 & & \\ & 1 & -1 & \\ & & 1 & -1 \end{array}} \end{array} \quad f^C = \mathbf{0}$$

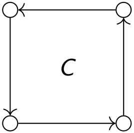
Circuits of Generalized Flow

- For generalized flow,

$$Ax = b \qquad \Longleftrightarrow \qquad \sum_{e \in \delta^{\mathrm{in}}(v)} \gamma_e x_e - \sum_{e \in \delta^{\mathrm{out}}(v)} x_e = b_v \quad \forall v \in V$$

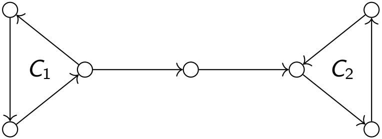
- $\ker(A)$ = set of generalized circulations. 2 types of circuits:

Conservative cycle



$\prod_{e \in C} \gamma_e = 1$

Bicycle



$\prod_{e \in C_1} \gamma_e > 1$

flow-generating

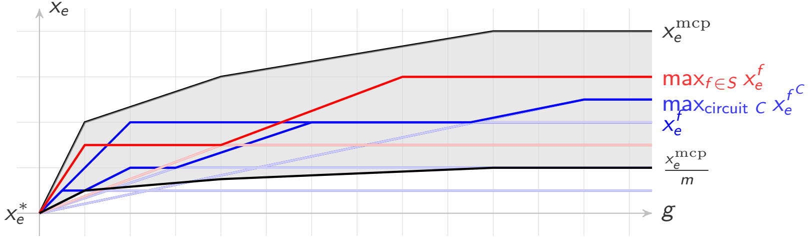
$\prod_{e \in C_2} \gamma_e < 1$

flow-absorbing

Upper Bounding the SLC

- Every $f \in \ker(A)$ with $c^\top f > 0$ induces a line segment in the feasible region:

$$x^f(g) := x^* + \frac{g}{c^\top f} f.$$



Fact: $\max_{\text{circuit } C} x_e^{f^C} \geq \frac{x_e^{\mathrm{mcp}}}{m}$.

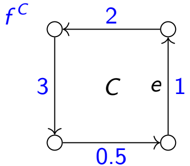
Strategy: Find $S \subseteq \ker(A)$ such that $\max_{f \in S} x_e^f \geq \max_{\text{circuit } C} x_e^{f^C}$.

$$\implies \mathrm{SLC}_{\frac{1}{m}}(x_e^{\mathrm{mcp}}) \leq 2|S|.$$

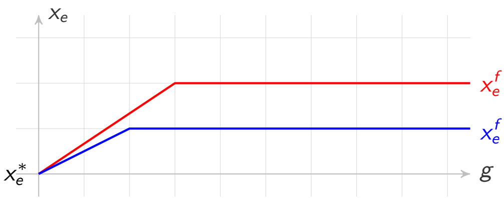
Dominating Circuits of Generalized Flow

Goal: Find a small $S \subseteq \ker(A)$ such that $\max_{f \in S} x_e^f \geq \max_{\text{circuit } C} x_e^{f^C}$.

- Consider the residual graph G_{x^*} with capacities $u \in \mathbb{R}_{>0}^{E \cup \overleftarrow{\mathrm{supp}(x^*)}}$.
- Let C be a conservative cycle or bicycle in G_{x^*} containing e.



conservative



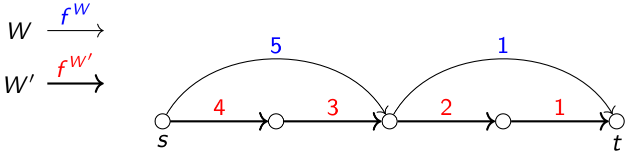
- Given $f \in \ker(A)$, what does $x_e^f \geq x_e^{f^C}$ mean?
  - For any bound g on the cost of the flow, f can send more flow on e than f^C.

Dominating Paths

- We reduce dominating circuits to dominating paths.

Def: Given s-t walks W and W', W' dominates W if for any bound g on the ℓ_∞-cost of the flow, W' can send more flow to t than W.

W' dominates W assuming $\nabla_t f^W = \nabla_t f^{W'}$



$c = \mathbb{1}$

$u = 10 \cdot \mathbb{1}$

Core Problem

Find a small set $\mathcal{W}$ of s-t walks such that every s-t path is dominated by some walk in $\mathcal{W}$.

Dominating Paths

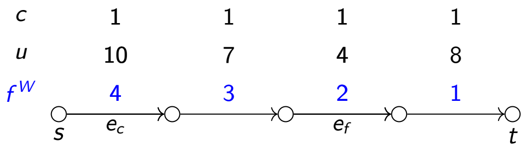
Theorem [Dadush, K, Natura, Olver, Végh '24]

For every $s, t \in V$, there exists a set $\mathcal{W}$ of $O(m^2)$ s-t walks such that every s-t path is dominated by some walk in $\mathcal{W}$.

- For every walk W, assign a signature (e_c, e_f) where

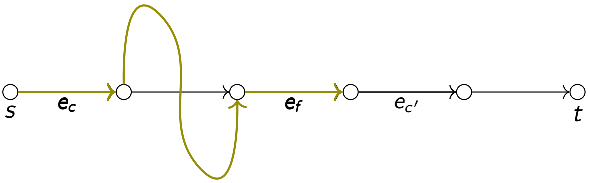
$$e_c := \arg\max_{e \in E(W)} c_e f_e^W \qquad\qquad e_f := \arg\min_{e \in E(W)} \frac{u_e}{f_e^W}.$$

We call e_c the cost bottleneck, and e_f the flow bottleneck of W.



Path Patching

- Let P be an s-t path with signature (e_c, e_f).



Def: Let patch(P) be the walk obtained from P by replacing the e_c-e_f subpath with a max gain e_c-e_f path of signature (e_c, e_f).

Patching Lemma:

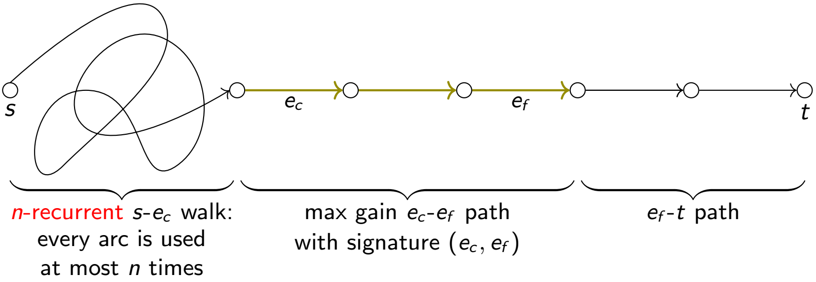
1. patch(P) dominates P.
2. The signature of patch(P) is either (e_c, e_f) or (e_c', e_f), where e_c' comes after e_f.

Dominating Paths

- For an s-t path P, let $W_1, W_2, \ldots, W_k$ be the sequence of walks obtained by repeatedly patching until the signature stops changing, i.e.

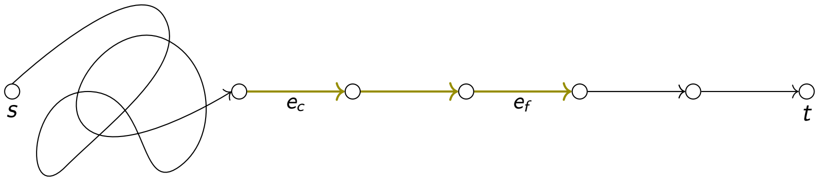
$$W_1 = \text{patch}(P) \qquad\qquad W_i = \text{patch}(W_{i-1}) \quad \forall i \geq 2.$$

- By patching lemma, W_k dominates P and $k \leq n$.



The Dominating Set of Walks $\mathcal{W}$

- For every signature (e_c, e_f),
  1. Start with a max gain e_c-e_f path with signature (e_c, e_f).
  2. Append a max gain n-recurrent s-e_c walk which preserves signature.
  3. Append a max gain e_f-t path which preserves signature.



- Analogous construction for the case where e_c comes after e_f.

- $|\mathcal{W}| = O(m^2)$.

Conclusion

- SLC of minimum cost generalized flow is $\mathrm{poly}(m, n)$.
- Strongly polynomial algorithm for LPs with ≤ 2 variables per inequality.
- [Allamigeon, Benchimol, Gaubert, Joswig '18] There exist LPs with

$$\mathrm{SLC}_\theta(x_i^{\mathrm{mcp}}) = 2^{\Omega(m)}.$$

- Future directions:
  - Develop a theory of SLC for LPs.
  - Undiscounted MDP: strongly polynomial solvability/straight line complexity open.
  - Faster strongly polynomial algorithm for minimum cost generalized flow.

Thank you!