School of Mathematics

Applications of Interior Point Methods: From Sparse Approximations to Discrete Optimal Transport

Jacek Gondzio
J.Gondzio@ed.ac.uk
http://www.maths.ed.ac.uk/~gondzio

Outline

- Interior Point Methods for Optimization
- Sparse Approximations
- Discrete Optimal Transport
- Final Remarks

Thanks to my collaborators:

Stefano Cipolla (Edinburgh, UK),
Valentina De Simone (Caserta, Italy),
Daniela di Serafino (Naples, Italy),
Spyros Pougkakiotis (Yale, USA),
Marco Viola (Dublin, Ireland),
Filippo Zanetti (Edinburgh, UK).

Professor Daniela di Serafino (1966-2022)

https://sites.google.com/view/danieladiserafino/

Observation

Numerous practical (engineering) problems can be cast as the following optimization problems

LP:

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b, \\ & x \geq 0. \end{array}$$

QP:

$$\begin{array}{ll} \min & c^T x + \frac{1}{2} x^T Q x \\ \text{s.t.} & Ax = b, \\ & x \geq 0. \end{array}$$

SDP:

$$\begin{array}{ll} \min & C \bullet X \\ \text{s.t.} & \mathcal{A}X = b, \\ & X \succeq 0, \end{array}$$

where $X \in \mathcal{SR}^{n \times n}$.

Logarithmic Barrier Functions

For the nonnegativity constraint $\{x \in \mathcal{R},\ x \geq 0\}$ use $f : \mathcal{R} \mapsto \mathcal{R}$

$$f(x) = \begin{cases} -\ln(x) & \text{if } x > 0 \\ +\infty & \text{otherwise.} \end{cases}$$

For the quadratic cone

$$K_q = \{(x,t) : x \in \mathcal{R}^{n-1}, t \in \mathcal{R},\ t^2 \geq \|x\|^2,\ t \geq 0\},$$

use $f : \mathcal{R}^n \mapsto \mathcal{R}$

$$f(x,t) = \begin{cases} -\ln(t^2 - \|x\|^2) & \text{if } \|x\| < t \\ +\infty & \text{otherwise.} \end{cases}$$

For the cone $\mathcal{SR}_+^{n\times n}$ of pos. definite matrices, use $f : \mathcal{SR}_+^{n\times n} \mapsto \mathcal{R}$

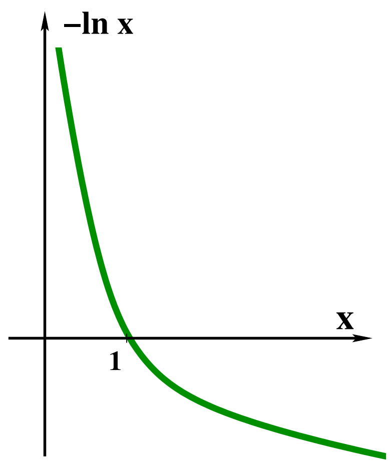
$$f(X) = \begin{cases} -\ln\det X & \text{if } X \succ 0 \\ +\infty & \text{otherwise.} \end{cases}$$

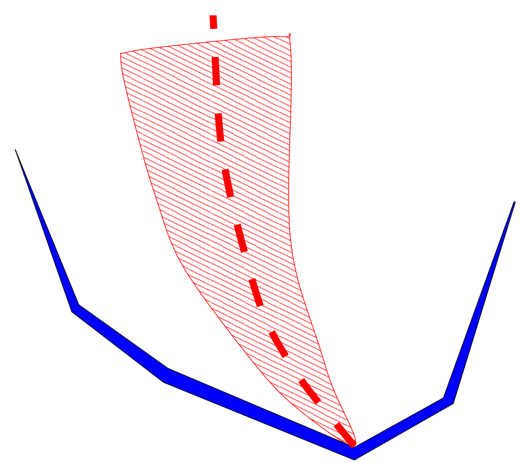
LP: Replace $x \geq 0$ with $-\mu\sum_{j=1}^{n} \ln x_j$.

SDP: Replace $X \succeq 0$ with $-\mu\sum_{j=1}^{n} \ln \lambda_j = -\mu \ln(\prod_{j=1}^{n} \lambda_j)$.

IPM Framework

- use logarithmic barrier to "replace" the inequality $x \geq 0$
- write down the first-order optimality conditions
- apply Newton method to FOC





Enjoyable Features of IPMs

- solve LP, QP, NLP, SOCP, SDP
- deliver polynomial complexity and unequalled efficiency
- are well suited to large scale optimization

Why IPMs? Why Logarithmic Barrier?

The use of logarithmic barrier function (in LP) has several enjoyable consequences:

- it is a self-concordant barrier
- it is mildly nonlinear hence Newton method can approximate it very accurately
- it transforms a difficult equation

$$XSe = 0, \quad (\text{ i.e., } \; x_j \cdot s_j = 0 \quad \forall j)$$

into an easier one:

$$XSe = \boldsymbol{\mu} e, \quad (\text{ i.e., } \; x_j \cdot s_j = \boldsymbol{\mu} \quad \forall j)$$

There is no need to "guess" the activity of variables.
It will be gradually revealed as $\boldsymbol{\mu}$ approaches zero.

Computational view of IPMs (for LP)

Primal	Dual
$\min \; c^T x$ s.t. $Ax = b$, $x \geq 0$;	$\max \; b^T y$ s.t. $A^T y + s = c$, $s \geq 0$.

At each iteration of IPM solve

$$\begin{bmatrix} \Theta^{-1} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix},$$

where $\boldsymbol{\Theta = XS^{-1}}$ is an **ill conditioned** diagonal scaling matrix.

Eliminate Δx from the first equation and get

$$(A\Theta A^T)\Delta y = h.$$

At optimality: some elements $\Theta_j \approx 0$, others $\Theta_j \rightarrow \infty$.

Overarching Feature of IPMs

They possess an unequalled ability to identify
the **"essential subspace"**
in which the optimal solution is hidden.

Sparse Approximations

- Statistics: Estimate x from observations
- Machine Learning: Classifications, SVMs, etc
- Compressed Sensing (Signal Processing)
- Sparse portfolio selection
- Inverse Problems
- Wavelet-based signal/image reconstruction & restoration
- Classification models for funct'l Magnetic Resonance Imaging

Such problems lead to some *dense*, often *structured*, possibly *very large* optimization instances (LP, QP or NLP):

$$\begin{array}{ll} \min_x & f(x) + \tau_1 \|x\|_1 + \tau_2 \|Lx\|_1 \\ \text{s.t.} & Ax \;=\; b. \end{array}$$

ℓ_1-regularization

$$\min_x \quad \tau \|x\|_1 + \phi(x).$$

think of LASSO:

$$\min_x \; f(x) = \tau \|x\|_1 + \|Ax - b\|_2^2$$

Unconstrained optimization $\Rightarrow$ easy

Serious Issue: nondifferentiability of $\|.\|_1$

Two possible tricks:

- Splitting $x = u - v$ with $u, v \geq 0$
- Smoothing with pseudo-Huber approximation

  replaces $\|x\|_1$ with $\psi_\mu(x) = \sum_{i=1}^n (\sqrt{\mu^2 + x_i^2} - \mu)$

Continuation

Embed inexact Newton Method into a *homotopy* approach:

- Inequalities $u \geq 0,\ v \geq 0$ $\longrightarrow$ use **IPM**

  replace $z \geq 0$ with $-\mu\, log z$ and drive μ to zero.

- pseudo-Huber regression $\longrightarrow$ use **continuation**

  replace $|x_i|$ with $\mu(\sqrt{1+\frac{x_i^2}{\mu^2}}-1)$ and drive μ to zero.

Questions:

- Theory?
- Practice?

1st-order methods vs 2nd-order methods

The 2nd-order methods are sometimes criticised as unsuitable:
“computing/using the 2nd-order information is too expensive”.

An **unfounded criticism** based on an **unfair comparison**:
specialised 1st-order methods compared with
general (of-the-shelf) 2nd-order methods.

The 1st-order methods have clear drawbacks:

- they struggle with accuracy, and
- they work only for trivial, well conditioned problems.

The **specialised 2nd-order methods**
overcome these drawbacks and are very competitive.

This talk will demonstrate why.

How to Specialize an IPM?

- Simplify the linear algebra:
  - use inexact Newton method
  - build efficient preconditioners for iterative methods
    $\longrightarrow$ use *matrix-free* IPM
- Exploit expected sparsity of the solution
  - ignore "long" matrix A
  - do not update all variables x
    $\longrightarrow$ use *column-generation-type* approach

Gondzio,
Convergence analysis of an inexact feasible IPM for convex QP, *SIOPT* 23 (2013) No 3, 1510–1527.

Gondzio, Pougkakiotis and Pearson,
General-purpose preconditioning for regularized interior point methods, *COAP* 83 (2022) pp. 727–757.

Zanetti and Gondzio,
A new stopping criterion for Krylov solvers applied in interior point methods, *SISC* 45 (2023), No. 2.

Zanetti and Gondzio,
An interior-point-inspired algorithm for linear programs arising in discrete optimal transport, *INFORMS J on Computing* 35 (2023) No 5 pp. 1061–1078. `https://doi.org/10.1287/ijoc.2022.0184`

Main Tool: Inexact Newton Method

Replace an *exact* Newton direction

$$\nabla^2 f(x)\Delta x = -\nabla f(x)$$

with an *inexact* one:

$$\nabla^2 f(x)\Delta x = -\nabla f(x) + \boldsymbol{r},$$

where the *residual* $\boldsymbol{r}$ is small: $\|\boldsymbol{r}\| \leq \boldsymbol{\eta}\|\nabla f(x)\|$, $\boldsymbol{\eta} \in (0,1)$.

The NLP community usually writes it as:

$$\|\nabla^2 f(x)\Delta x + \nabla f(x)\|_2 \leq \boldsymbol{\eta}\|\nabla f(x)\|_2, \quad \boldsymbol{\eta} \in (0,1).$$

Bellavia,
Inexact Interior Point Method, *JOTA* 96 (1998) 109–121.

Dembo, Eisenstat & Steihaug,
Inexact Newton Methods, *SIAM J. on Numerical Analysis* 19 (1982) 400–408.

Theorem: Suppose the feasible IPM for QP is used.

If the method operates in the *small* neighbourhood

$$\mathcal{N}_2(\theta) := \{(x,y,s) \in \mathcal{F}^0 : \|XSe - \mu e\|_2 \leq \theta\mu\}$$

and uses the *inexact* Newton direction with $\boldsymbol{\eta} = 0.3$, then it converges in at most

$$K = \mathcal{O}(\sqrt{\boldsymbol{n}}\ \ln(1/\epsilon)) \quad \text{iterations.}$$

If the method operates in the *symmetric* neighbourhood

$$\mathcal{N}_S(\gamma) := \{(x,y,s) \in \mathcal{F}^0 : \gamma\mu \leq x_i s_i \leq (1/\gamma)\mu\}$$

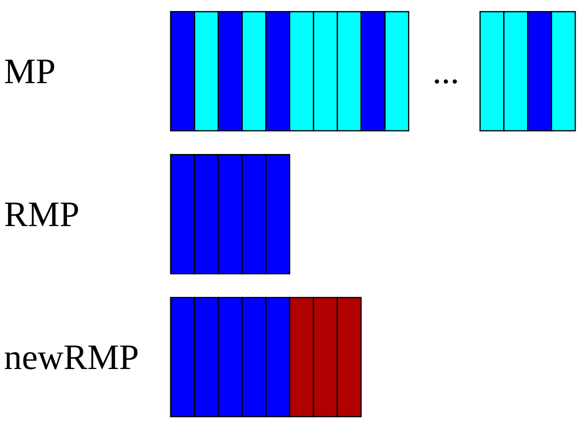
and uses the *inexact* Newton direction with $\boldsymbol{\eta} = 0.05$, then it converges in at most

$$K = \mathcal{O}(\boldsymbol{n}\ \ln(1/\epsilon)) \quad \text{iterations.}$$

“Long” LP $\rightarrow$ use Column Generation (CG)

Replace an LP with a “shorter” one $(\overline{N} \subset N)$:

Master LP	Restricted Master LP
$\min \quad \sum_{j \in N} c_j x_j$	$\min \quad \sum_{j \in \overline{N}} c_j x_j$
$\text{s.t.} \quad \sum_{j \in N} A_j x_j = b$	$\text{s.t.} \quad \sum_{j \in \overline{N}} A_j x_j = b$
$x_j \geq 0, \ \forall j \in N;$	$x_j \geq 0, \ \forall j \in \overline{N}.$



set LB $= -\infty$, UB $= \infty$, gap $= \infty$, $\varepsilon = 0.5$;

while (gap $> \delta$) do
    find a well-centred ε-opt $(\tilde{\lambda}, \tilde{u})$ of the RMP;
    UB $= \min\{\text{UB}, \tilde{z}_{RMP}\}$;
    call the oracle with the query point $\tilde{u}$;
    LB $= \max\{\text{LB}, \kappa \tilde{z}_{SP} + b^T \tilde{u}\}$;
    gap $= (\text{UB} - \text{LB})/(1 + |\text{UB}|)$;
    $\varepsilon = \min\{\varepsilon_{\max}, \ \text{gap}/D\}$;
    if $(\tilde{z}_{SP} < 0)$ then add new columns to the RMP;

end (while)

IPM Specialized for Column Generation

- Ignore "long" matrix A; work with "short" A and x
  $\longrightarrow$ do not update all variables x; use "sparse" x
- Use simplex-type pricing mechanism (*Oracle in CG*)
  $\longrightarrow$ update dual slacks only for a subset of variables x
- Simplify normal equations
  $\longrightarrow$ replace $\sum_{j=1}^{N} \theta_j A_j A_j^T$ with $\sum_{j \in \overline{N}} \theta_j A_j A_j^T$,
  where $\overline{N}$ is likely to contain the "sparse" solution set
- Build a preconditioner for the normal equations matrix
  $\longrightarrow$ keep it sparse at all times

Use IPMs in Sparse Approximations

Problems of the form

$$\begin{array}{ll} \min & f(x) + \tau_1 \|x\|_1 + \tau_2 \|Lx\|_1 \\ \text{s.t.} & Ax = b. \end{array}$$

- Sparse portfolio selection
  comparison with Split Bregman method
- Classification models for funct'l Magnetic Resonance Imaging
  comparison with FISTA and ADMM
- TV-based Poisson Image Restoration
  comparison with PDAL
- Linear Classification via Regularized Logistic Regression
  comparison with newGLMNET and ADMM

De Simone, di Serafino, Gondzio, Pougkakiotis, Viola,
Sparse Approximations with Interior Point Methods,
SIAM Review 64 (2022) pp. 954–988. `https://doi.org/10.1137/21M1401103`

Modeling trick

$$\begin{array}{ll} \min\limits_x & f(x) + \tau_1 \|x\|_1 + \tau_2 \|Lx\|_1 \\ \text{s.t.} & Ax = b, \end{array}$$

where $L \in \mathcal{R}^{l \times n},\ A \in \mathcal{R}^{m \times n},\ b \in \mathcal{R}^m,\ m \leq n.$

Let $|x| = x^+ + x^-$, where $x^+ = \max\{x, 0\}$ and $x^- = \max\{-x, 0\}$.
Set $d = Lx \in \mathcal{R}^l$ and also write $|d| = d^+ + d^-$.
New formulation:

$$\begin{array}{ll} \min\limits_{x^+, x^-, d^+, d^-} & f(x^+ - x^-) + \tau_1 (e_n^T x^+ + e_n^T x^-) + \tau_2 (e_l^T d^+ + e_l^T d^-) \\ \text{s.t.} & A(x^+ - x^-) = b \\ & L(x^+ - x^-) = d^+ - d^- \\ & x^+, x^-, d^+, d^- \geq 0, \end{array}$$

where $e_p \in \mathcal{R}^p$ is a vector of all 1's
Larger smooth problem, but IPMs are able to efficiently handle large sets of linear equality and non-negativity constraints!

Binary Classification of fMRI Data

$$\min_{w} \frac{1}{2s} \|Dw - \hat{y}\|^2 + \tau_1 \|w\|_1 + \tau_2 \|Lw\|_1$$

where: $\tau_1, \tau_2 > 0$, $\|Lw\|_1$ is a discrete anisotropic TV of w,

and $L = [L_x^T \; L_y^T \; L_z^T]^T \in \mathcal{R}^{l \times q}$ are the first-order forward finite differences in x, y, z.

Baldassarre, Pontil & Mouraõ-Miranda,

Sparsity Is Better with Stability: Combining Accuracy and Stability for Model Selection in Brain Decoding,

Frontiers of Neuroscience 2017. https://doi.org/10.3389/fnins.2017.00062

Classification models for fMRI

Comparison of IPM, FISTA and ADMM (opt tol 10^{-5}). We report:

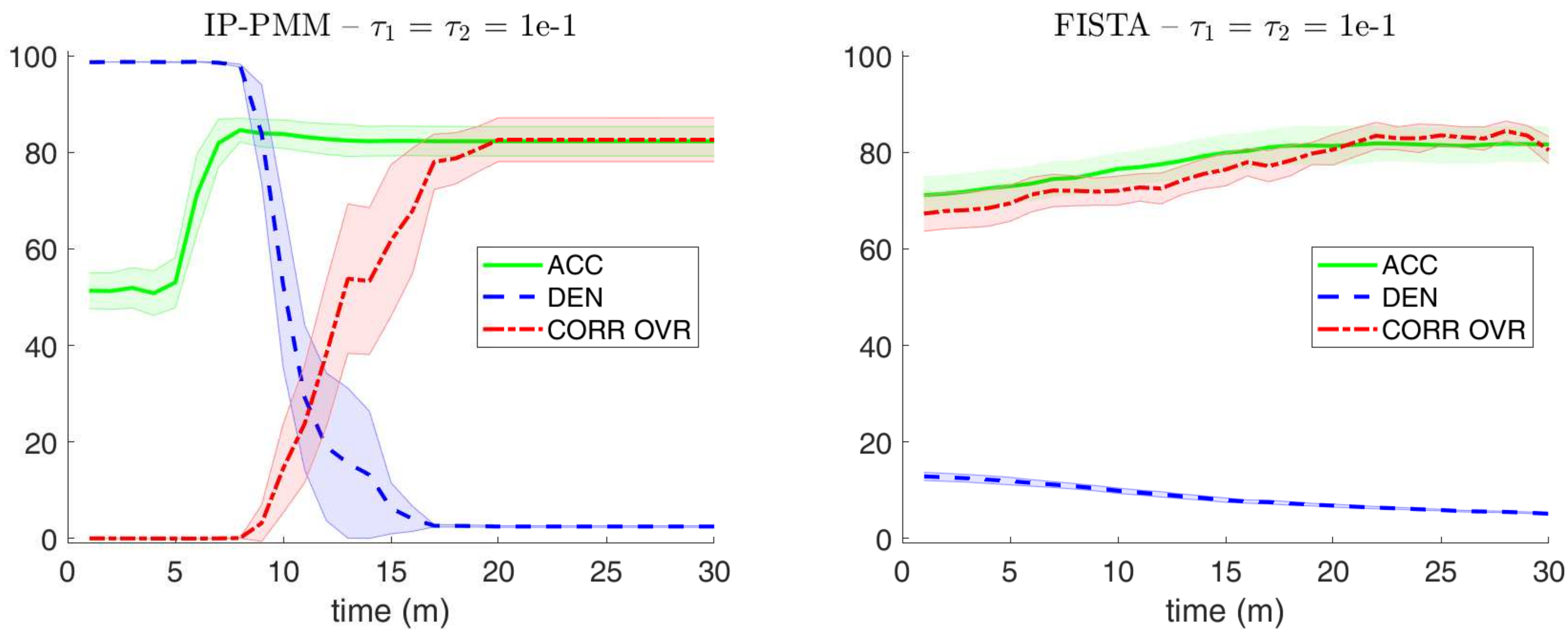
- *classification accuracy* (ACC),
- *corrected pairwise overlap* (CORR OVR); measures the "stability" of each voxel selection,
- *solution density* (DEN).

Algorithm	$\tau_1 = \tau_2$	ACC	CORR OVR	DEN	
IP-PMM	10^{-2}	86.16 ± 7.11	43.47 ± 9.09	20.56 ± 6.63	
	$5 \cdot 10^{-2}$	84.90 ± 4.80	62.70 ± 10.39	3.77 ± 0.84	
	10^{-1}	82.29 ± 6.22	82.60 ± 9.24	2.49 ± 0.34	good
FISTA	10^{-2}	86.90 ± 5.01	5.43 ± 0.43	88.97 ± 0.71	
	$5 \cdot 10^{-2}$	84.15 ± 5.92	65.50 ± 2.68	19.36 ± 0.86	
	10^{-1}	81.62 ± 7.58	80.44 ± 5.72	5.14 ± 0.44	acceptable
ADMM	10^{-2}	86.46 ± 6.91	0.03 ± 0.01	98.70 ± 0.03	
	$5 \cdot 10^{-2}$	85.57 ± 5.37	0.15 ± 0.04	97.97 ± 0.05	
	10^{-1}	82.07 ± 6.51	0.26 ± 0.13	97.50 ± 0.19	unacceptable

We want: ACC and CORR OVR *close to 100*, and *small* DEN.

Classification models for fMRI (cont'd)

Performance comparison in terms of elapsed time:



Evolution of ACC, DEN and CORR OVR with time;
IP-PMM (*left*) and FISTA (*right*).
We report average measures with 95% confidence intervals.

Optimal Transport

Significant research interest:

Gaspard Monge (1781)
Leonid Kantorovich (1942) **Nobel Prize in 1975**
Alessio Figalli (2008) **Fields Medal in 2018**

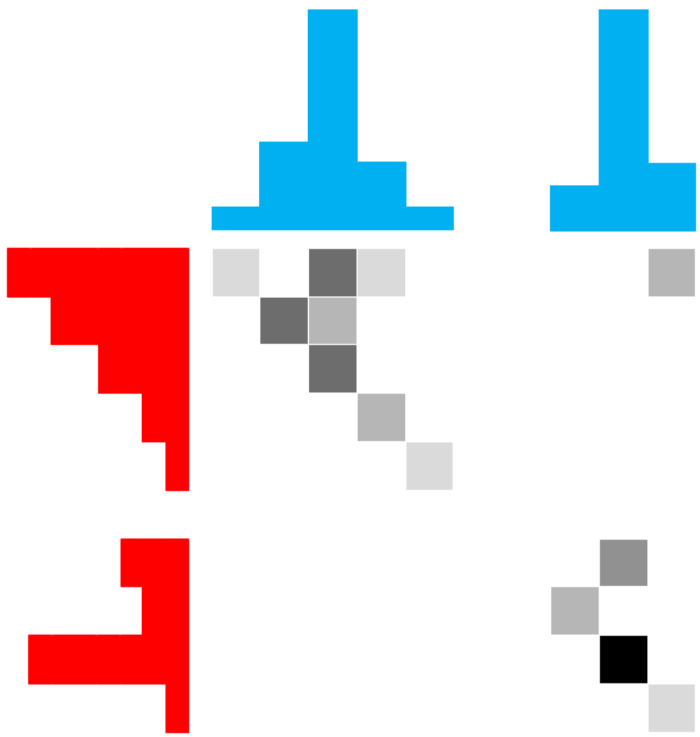
F. Santambrogio,
Optimal Transport for Applied Mathematicians, Birkhauser Basel, 2016.

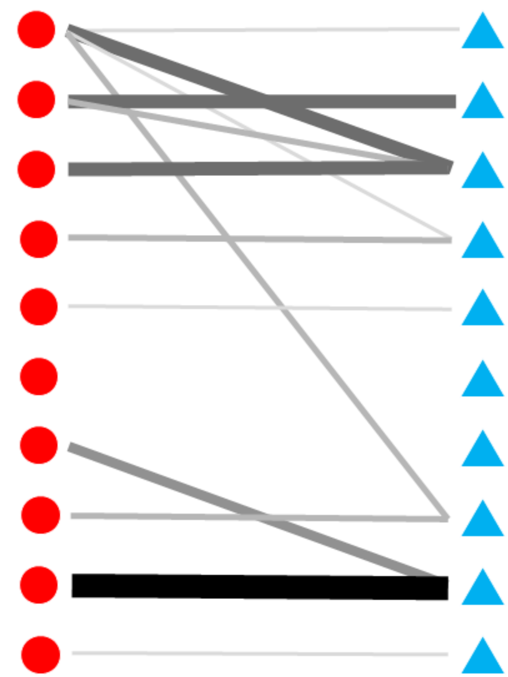
G. Peyré and M. Cuturi,
Computational Optimal Transport: With Applications to Data Science. *Foundations and Trends in Machine Learning* 11 (2019) No 5-6, pp. 355–607.

Discrete OT problem has an obvious connection to
Network Flow Problem

Small OT Example

Move the mass in the red configuration into the blue configuration.
Right figure: the corresponding bipartite graph. → Sparse solution!





Discrete Optimal Transport

We write the OT problem as a standard LP:

$$\begin{aligned} \min_{\mathbf{p}\in\mathcal{R}^{mn}} \quad & \mathbf{c}^T\mathbf{p} \\ \text{s.t.} \quad & \begin{bmatrix} \mathbf{e}_n^T \otimes I_m \\ I_n \otimes \mathbf{e}_m^T \end{bmatrix} \mathbf{p} = \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \\ & \mathbf{p} \geq 0, \end{aligned}$$

where $\otimes$ denotes the Kronecker product,
$\mathbf{c} \in \mathcal{R}^{mn}$ and $\mathbf{p} \in \mathcal{R}^{mn}$ are the vectorized versions of $\mathcal{C}$ and $\mathcal{P}$, respectively, $\mathbf{c} = \text{vec}(\mathcal{C})$ and $\mathbf{p} = \text{vec}(\mathcal{P})$.

LP with *$m + n$ constraints* and *$m \times n$ variables*.

Zanetti and Gondzio,
An interior-point-inspired algorithm for linear programs arising in discrete optimal transport,
INFORMS J on Computing 35 (2023) No 5 pp. 1061–1078. https://doi.org/10.1287/ijoc.2022.0184

Cipolla, Gondzio and Zanetti,
A regularized interior point method for sparse optimal transport on graphs, (submitted 3 March 2023, revised 11 July 2023). https://arxiv.org/abs/2307.05186

Graph interpretation of Optimal Transport

OT problem is an example of an LP:

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b, \\ & x \geq 0. \end{array}$$

Constraint matrix $A \in R^{m \times n}$ is a graph node-arc incidence matrix

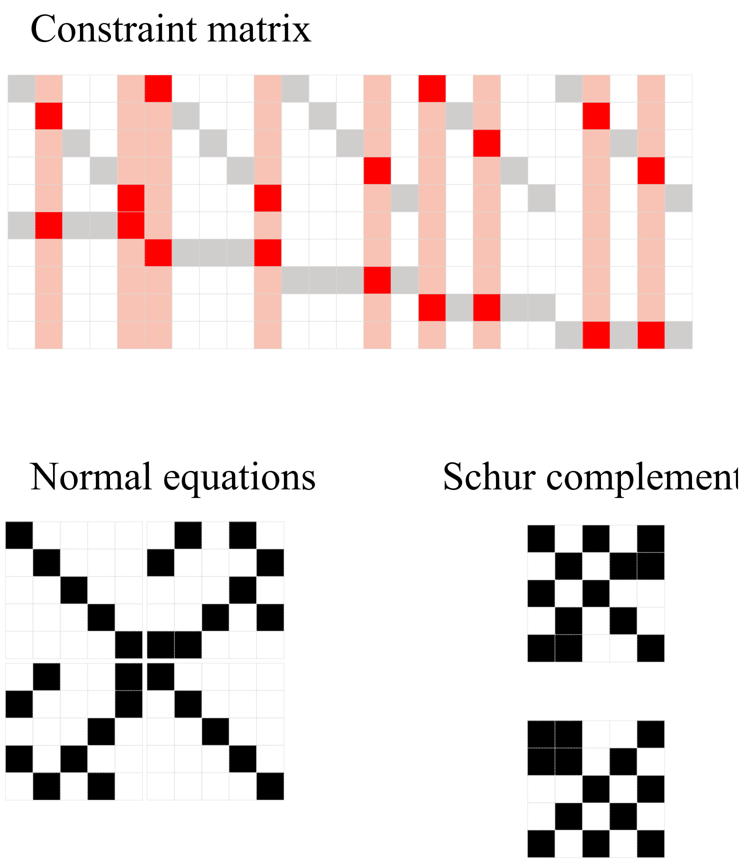
- Every column of A has only two nonzero entries
- A is very sparse
- But AA^T may be quite dense

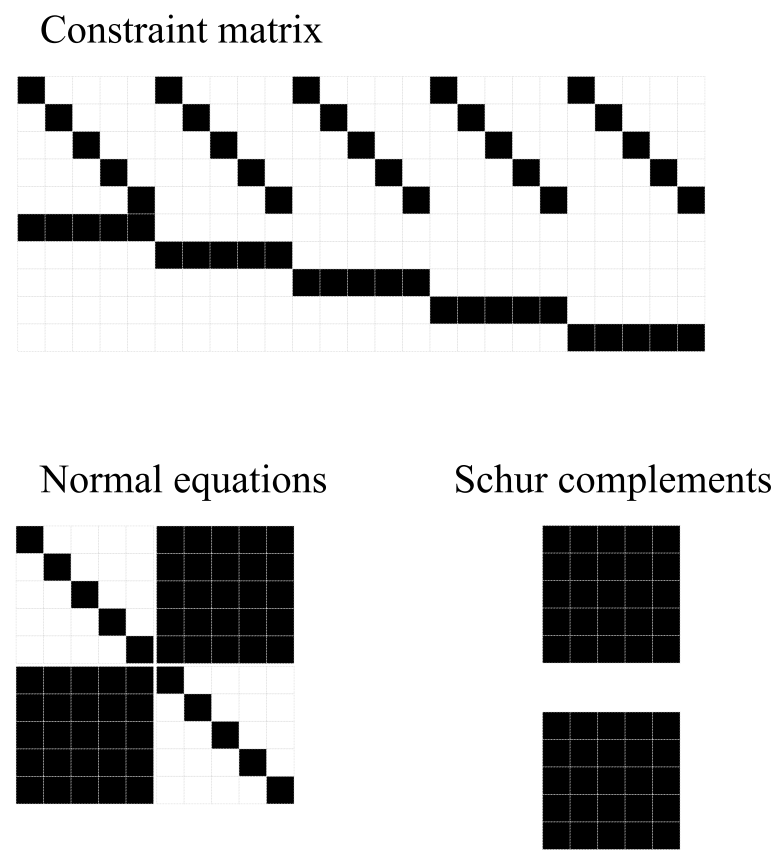
The main computational effort of IPM for LP (for OT) is:

- Building $A \Theta A^T$, where Θ is a diagonal matrix
- Solving equations with $A \Theta A^T$

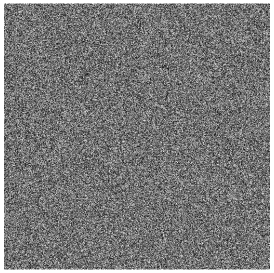
Structure of A and $A\Theta A^T$

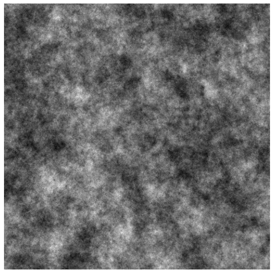
Restricted Master Problem, $\overline{N}$ (Full) Master, N

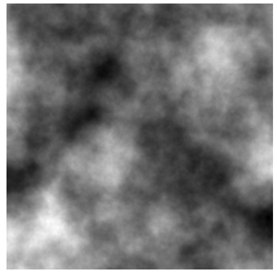




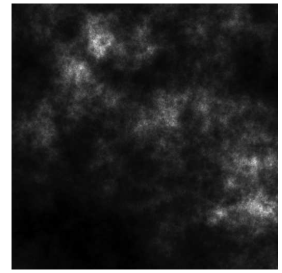
Test examples from DOTmark collection


Class 1

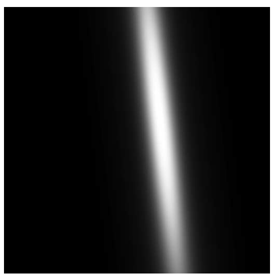

Class 2

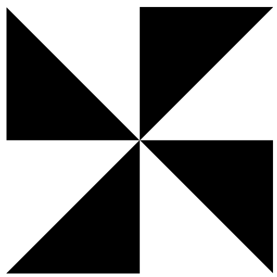

Class 3


Class 4

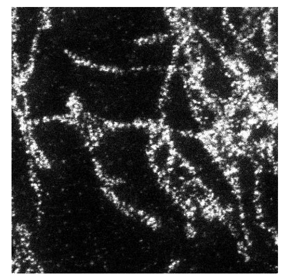

Class 5


Class 6


Class 7


Class 8


Class 9


Class 10

For the resolution r, the LP has $2r^2$ *constraints* and r^4 *variables*.
For $r = 32$: 2,048 constraints and 1 million variables;
For $r = 64$: 8,192 constraints and 16.8 million variables;
For $r = 128$: 32,768 constraints and 268.4 million variables;
For $r = 256$: 131,072 constraints and 4.295 billion variables.

Discrete Optimal Transport (cont'd)

DOTmark test collection (*dense* graphs, *sparse* solutions):

Schrieber, Schuhmacher, and Gottschlich,

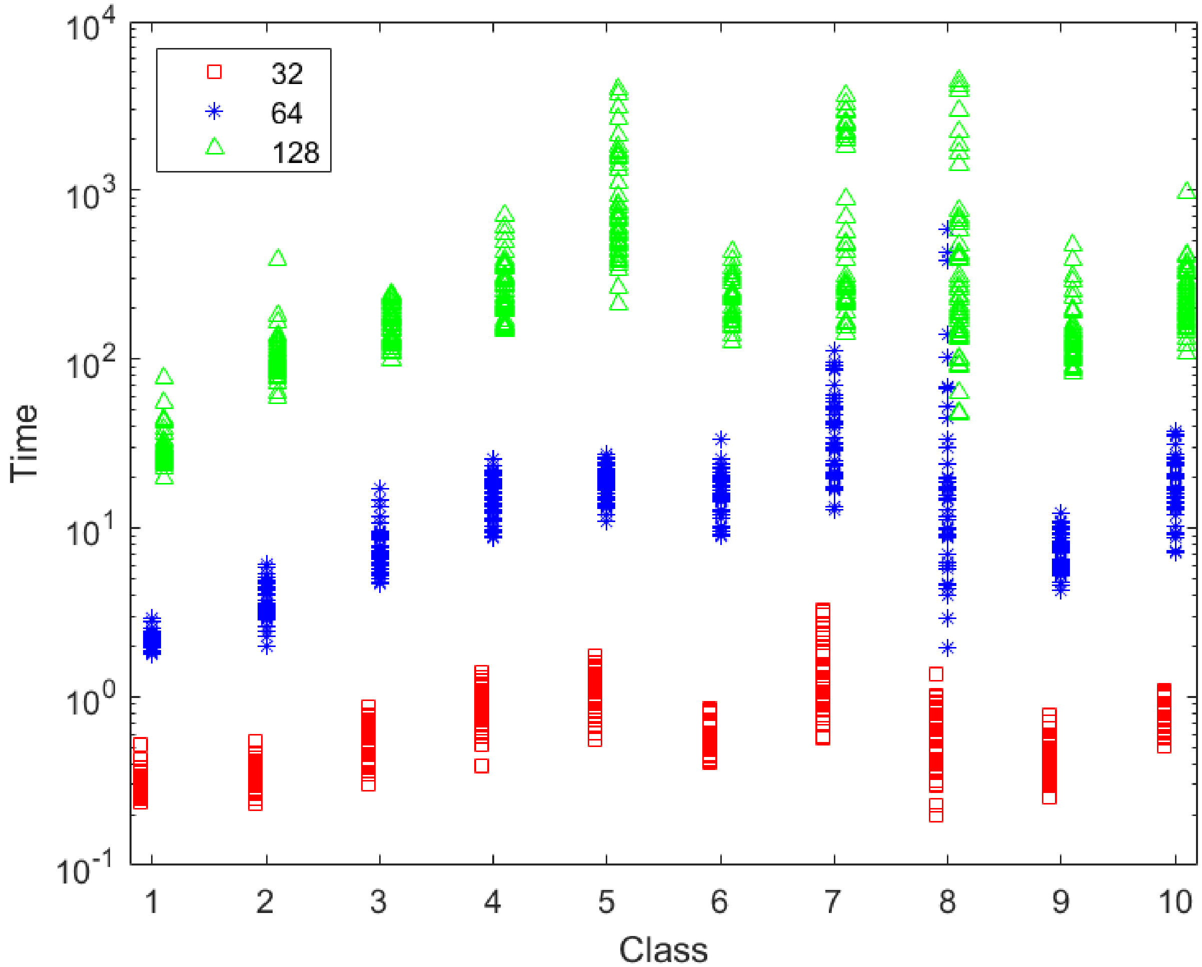
DOTmark - A Benchmark for Discrete Optimal Transport, *IEEE Access*, 5 (2017), pp. 271–282.

Softwares compared:

- **Cuturi**,
  Sinkhorn distances: Lightspeed computation of optimal transport,
  Proc. NIPS, (2013), pp. 2292–2300.
- **Gottschlich and Schuhmacher**,
  The Shortlist Method for Fast Computation of the Earth Mover's Distance and Finding Optimal Solutions to Transportation Problems, *PLoS ONE*, 9 (2014), p. e110214.
- **Merigot**,
  A Multiscale Approach to Optimal Transport,
  Computer Graphics Forum, 30 (2011), pp. 1583–1592.
- Network Simplex Method, IBM ILOG CPLEX.
  `https://www.ibm.com/analytics/cplex-optimizer`.

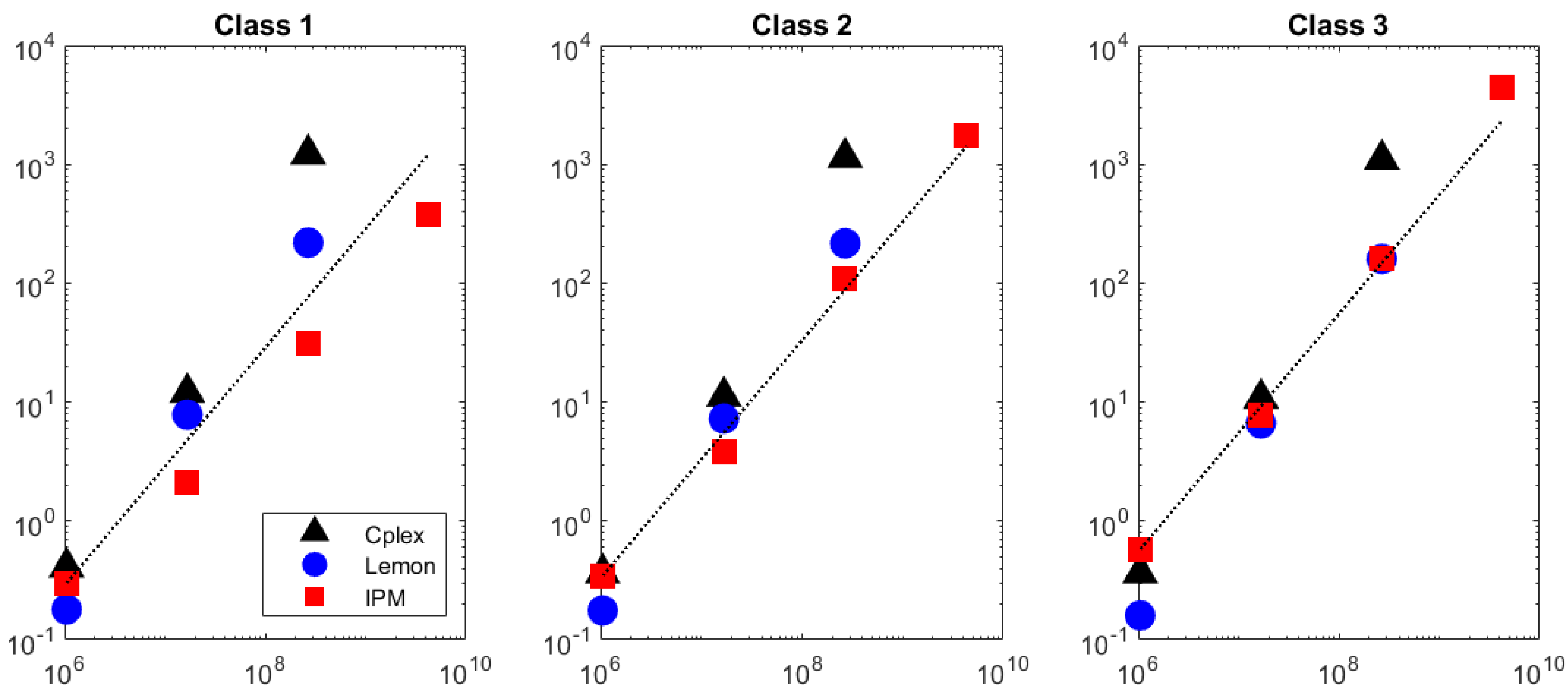
Cplex seems to be the most reliable and usually the festest of them.

CPU time of SparseIPM (1–norm, 128 pixels)



$$m = 2r^2$$
$$n = r^4$$

Comparison: Scalability of 3 solvers ($r = 256$)



Cplex (Simplex Method for Network Problems)
LEMON (Specialized Network Algorithm)
SparseIPM for Discrete OT

Optimal Transport on Sparse Graphs

DOTmark test collection (**dense** graphs, **sparse** solutions)
number of nodes $m = 2r^2$,
number of edges $n = r^4$,
where r is the picture resolution.
Case when $\boldsymbol{m} \ll \boldsymbol{n}$. Very "long" LP.

But what happens on **sparse** graphs with **sparse** solutions?
number of nodes m,
number of edges $n = \alpha m$,
where α is the average number of edges per node, say, $2 \leq \alpha \leq 10$.
Case when $\boldsymbol{m} \leq \boldsymbol{n}$. More "usual" LP shape.

Network Structure

- Discrete OT is equivalent to a *single commodity network flow*
  $\longrightarrow$ LP matrix A is a *graph node-arc incidence matrix*
- Use special form of normal equations $\sum_{j=1}^{N} \theta_j A_j A_j^T$
  $\longrightarrow$ exploit its *Laplacian structure*
- Use IPM matrix $\Theta = XS^{-1}$ to select a subset of variables and simplify normal equations
  $\longrightarrow$ replace $\sum_{j=1}^{N} \theta_j A_j A_j^T$ with $\sum_{j \in \mathcal{S}} \theta_j A_j A_j^T$,
  where $\mathcal{S}$ contains variables with "large" θ_j
- EITHER use it as an *approximation* of the normal equations OR use it to *precondition* normal equations

Cipolla, Gondzio and Zanetti,

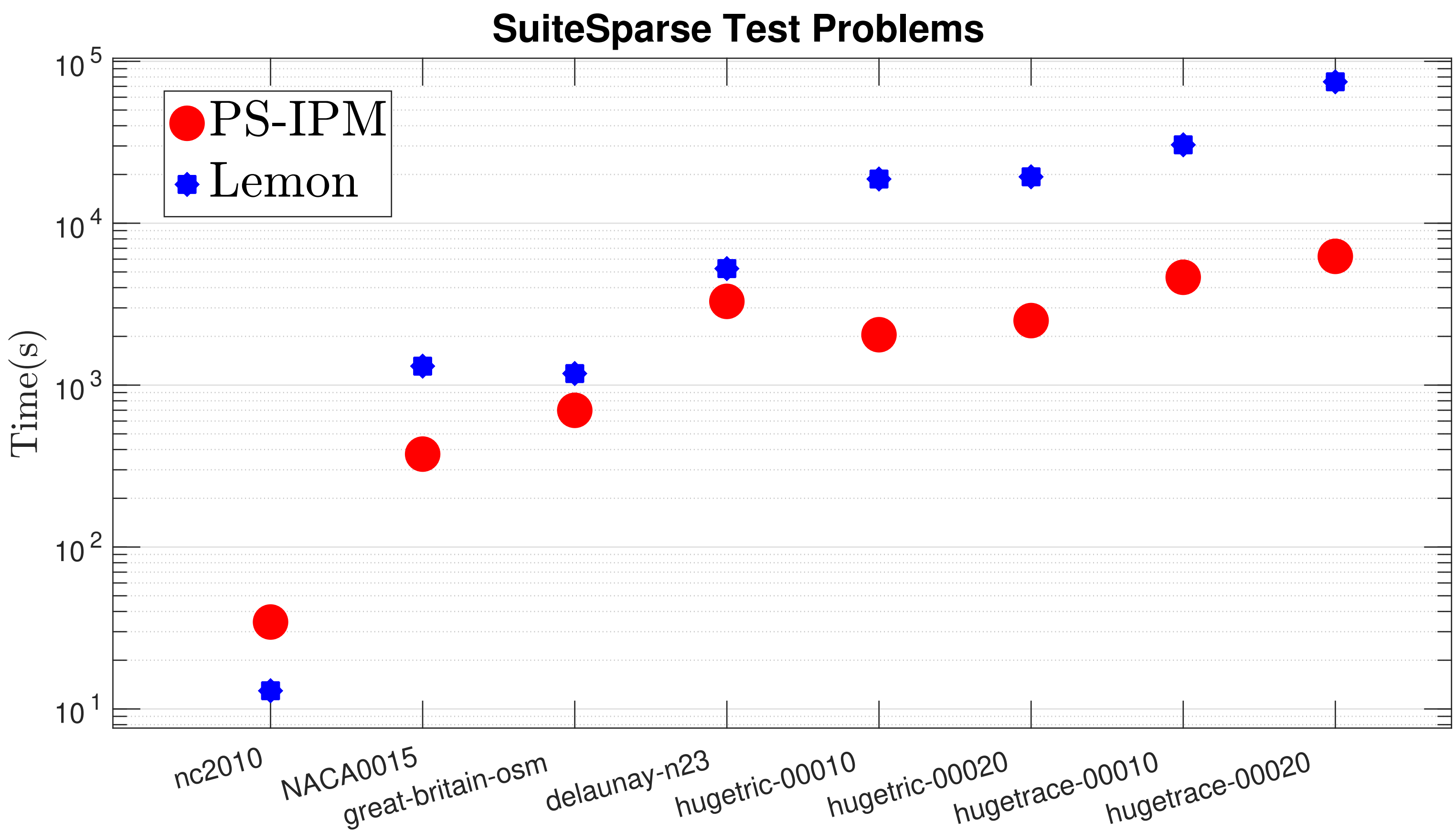
A regularized interior point method for sparse optimal transport on graphs, (submitted 3 March 2023, revised 11 July 2023). https://arxiv.org/abs/2307.05186

Large Sparse Graphs

Graphs from the SuiteSparse matrix collection

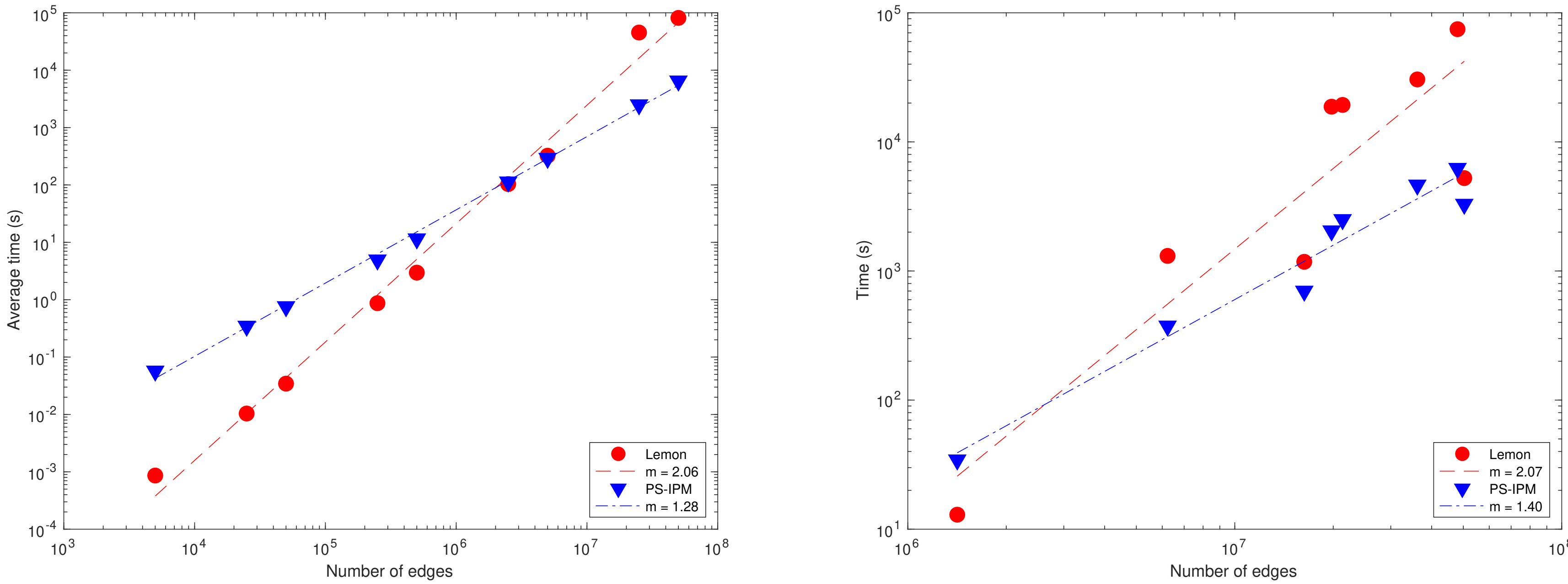
Name	Nodes	Edges	Density
`delaunay-n23`	8,388,608	50,331,568	6.0
`great-britain-osm`	7,733,822	16,313,034	2.1
`hugetric-00010`	6,592,765	19,771,708	3.0
`hugetric-00020`	7,122,792	21,361,554	3.0
`hugetrace-00010`	12,057,441	36,164,358	3.0
`hugetrace-00020`	16,002,413	47,997,626	3.0
`NACA0015`	1,039,183	6,229,636	6.0
`nc2010`	288,987	1,416,620	4.9

PS-IPM vs LEMON



PS-IPM vs LEMON (cont'd)

Performance comparison in terms of elapsed time:



Random graphs
LEMON: $m = 2.06$
PS-IPM: $m = 1.28$

SuiteSparse graphs
LEMON: $m = 2.07$
PS-IPM: $m = 1.40$

Conclusions

- IPMs are well-suited to solving large scale optimization problems
  - *enjoy predictable behaviour*
  - *deliver high accuracy*
- When applied to sparse approximations, IPMs
  - *compete with/outperform the 1st-order methods*
- When applied to optimal transport problems, IPMs
  - *are the methods to beat*

IPMs possess an unequalled ability to identify
the **"essential subspace"**
in which the optimal solution is hidden.