Solving Laplacian Systems in Nearly-Linear Time

Maguy Trefois   Jean-Charles Delvenne   Paul Van Dooren

Université catholique de Louvain

December 2014

Goal: Solving

$$Av = b$$

when $A \in \mathbb{R}^{n \times n}$ is symmetric diagonally-dominant (SDD).

Several methods:

- Gauss elimination: time $\mathcal{O}(n^3)$
- Fast matrix inversion (Strassen 1969, Coppersmith-Winograd 1987, ...): time $\mathcal{O}(n^{2.37})$
- ...

$\Rightarrow$ Too slow in case of huge matrix A

Goal: Solving any SDD system

$$Av = b$$

in nearly-linear time, i.e. in

$$\mathcal{O}(m \log^c n) \text{ time,}$$

where

- n is the size of the system
- m is the number of nonzero entries in A

Particular case of interest: when A is a Laplacian matrix.

Laplacian systems for...

- solving any SDD linear system

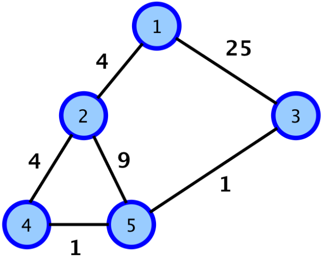
but also...

- computing effective resistances in a network
- computing dominant eigenvectors of graphs (by inverse power method)
- ...

Outline

G : undirected graph: n nodes, m edges

positive weights along the edges



Laplacian matrix:

$$L = \begin{vmatrix} 29 & -4 & -25 & 0 & 0 \\ -4 & 17 & 0 & -4 & -9 \\ -25 & 0 & 26 & 0 & -1 \\ 0 & -4 & 0 & 5 & -1 \\ 0 & -9 & -1 & -1 & 11 \end{vmatrix}$$

L is SDD and positive semidefinite.

G : undirected graph: n nodes, m edges
positive weights along the edges

L : Laplacian matrix of G

Goal: solving the Laplacian system

$$Lv = b$$

in nearly-linear time, namely in time $\mathcal{O}(m \log^c n)$.

More precisely, given $\epsilon > 0$, find $v_K \in \mathbb{R}^n$ such that

$$||v_K - v_{opt}||_L \leq \epsilon \cdot ||v_{opt}||_L,$$

where $Lv_{opt} = b$.

Laplacian systems: definition

Some fast solvers

Overview of the Kelner method

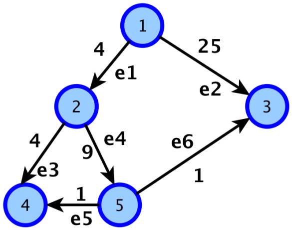
Spanning tree and Kaczmarz's algorithm

Approximate solution

Running time

Conclusion

- Spielman and Teng, 2004
  - Tools: low-stretch spanning trees, spectral sparsifiers, local clustering algorithm, preconditioned Chebyshev method, ...
  - Time: $\mathcal{O}(m \log^c n)$ with $c \simeq 15$
- Koutis, Miller and Peng, 2011
  - Based on the same ideas as Spielman's algorithm
  - Running time reduced to $\mathcal{O}(m \log n)$
- Kelner *et al.*, 2013
  - ONE low-stretch spanning tree, a Kaczmarz method
  - Simplest algorithm
  - Time: $\mathcal{O}(m \log^2 n)$
- Lee and Sidford, 2013
  - Based on the same ideas as Kelner *et al.*'s algorithm
  - Use of an accelerated coordinate gradient descend method
  - Time: $\mathcal{O}(m \log^{3/2} n)$
- Cohen *et al.*, 2014
  - Based on the ideas of Spielman's algorithm
  - Fastest algorithm
  - Time: $\mathcal{O}(m \log^{1/2} n)$



$$B = \begin{array}{c|ccccc|} & 1 & 2 & 3 & 4 & 5 \\ e_1 & 1 & -1 & 0 & 0 & 0 \\ e_2 & 1 & 0 & -1 & 0 & 0 \\ e_3 & 0 & 1 & 0 & -1 & 0 \\ e_4 & 0 & 1 & 0 & 0 & -1 \\ e_5 & 0 & 0 & 0 & -1 & 1 \\ e_6 & 0 & 0 & -1 & 0 & 1 \end{array} \quad R = \left|\begin{array}{cccccc} 1/4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/25 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right|$$

Factorize L as:

$$L = B^T R^{-1} B$$

The Kelner method

Laplacian System:

$$B^T R^{-1} B v = b$$

Pose $f = R^{-1} B v$.

- Step 1: Approximate $f_{opt} \in \mathbb{R}^m$ the unique solution of minimal R-norm to

$$B^T f = b,$$

i.e. given $\epsilon > 0$, find $f_K \in \mathbb{R}^m$ such that $B^T f_K = b$ and

$$||f_K - f_{opt}||_R \leq \epsilon \cdot ||f_{opt}||_R$$

- Step 2: given f_K and from the variable change $f_K = R^{-1} B v$, given $\epsilon > 0$, find $v_K \in \mathbb{R}^n$ such that,

$$||v_K - v_{opt}||_L \leq \epsilon \cdot ||v_{opt}||_R,$$

where $L v_{opt} = b$.

The Kelner method: Step 1

$f_{opt} \in \mathbb{R}^m$ is the solution to $B^T f = b$ of minimal R-norm, namely

- $B^T f_{opt} = b$
- f_{opt} is orthogonal to the kernel of B^T

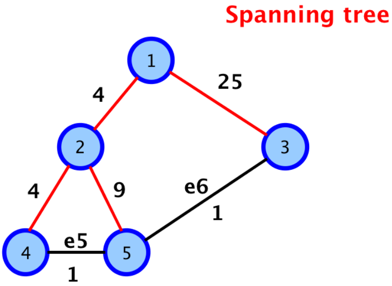
Goal: given $\epsilon > 0$, find $f_K \in \mathbb{R}^m$ such that $B^T f_K = b$ and

$$||f_K - f_{opt}||_R \leq \epsilon \cdot ||f_{opt}||_R$$

Needed:

- A basis of the kernel of B^T $\Rightarrow$ spanning tree
- An iterative algorithm $\Rightarrow$ a Kaczmarz method

Low-stretch spanning tree

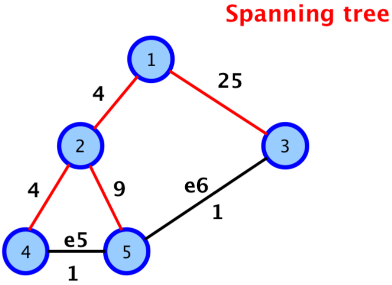


- Stretch of an off-tree edge = path length in the tree, over weight

  Stretch of edge $e_5 = 13/1$

- Stretch of tree T = sum of stretch of all off-tree edges

  Stretch of $T = 13/1 + 38/1 = 51$



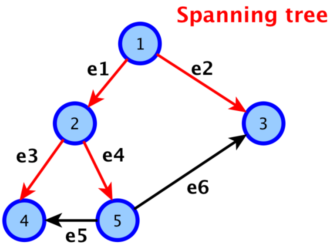
In the Kelner method, compute the stretch with weight of edge $e = R_{ee} = 1/weight(e)$.
In our example,

$$\text{stretch}(T) = (1/4 + 1/9)/1 + (1/4 + 1/9 + 1/25)/1 \simeq 0.7622$$

Roles of the spanning tree:

- provides a basis of the kernel of B^T
- plays a role of preconditioner in the Kaczmarz method

A basis of the kernel of B^T

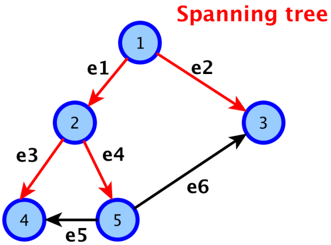


- Any off-tree edge $e = \{a, b\}$ defines a unique cycle of the form:

$$e + \text{ path between } a \text{ and } b \text{ in } T$$

For edge e_5: cycle $= \{e_3, e_4, e_5\} \Rightarrow$ pose $Q_{e_5} = \begin{vmatrix} 0 \\ 0 \\ -1 \\ 1 \\ 1 \\ 0 \end{vmatrix}$

A basis of the kernel of B^T



- The vectors Q_e's where e is an off-tree edge form a basis of the kernel of B^T
  In our example, a basis is:

$$Q_{e_5} = \begin{vmatrix} 0 \\ 0 \\ -1 \\ 1 \\ 1 \\ 0 \end{vmatrix} \qquad Q_{e_6} = \begin{vmatrix} 1 \\ -1 \\ 0 \\ 1 \\ 0 \\ 1 \end{vmatrix}$$

Stretch and condition number

The condition number of the spanning tree T is:

$$\tau(T) := \mathrm{stretch}(T) + (m - 2n + 2)$$

In the Kaczmarz method, the convergence rate depends on $\tau(T)$

$$\mathbb{E}\left[||f_k||_R^2 - ||f_{opt}||_R^2\right] \leq \left(1 - \frac{1}{\tau(T)}\right)^k \left(||f_0||_R^2 - ||f_{opt}||_R^2\right)$$

$\Rightarrow$ find a spanning tree with stretch as low as possible

Theorem (Abraham-Neiman, 2012)

One can find a spanning tree with stretch $\mathcal{O}(m \log n)$ in $\mathcal{O}(m \log n)$ time.

The Kaczmarz method

For any off-tree edge e, a hyperplane

$$P(Q_e) := \{f \in \mathbb{R}^m | f \text{ is orthogonal to } Q_e\}$$

f_{opt} is in the intersection of all these hyperplanes

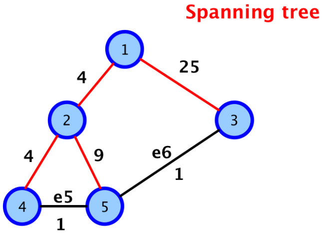
Ideas of the Kaczmarz method:

- start with $f_0 \in \mathbb{R}^m$ solution to $B^T f_0 = b$
- Do iteratively:
  1. pick randomly one basis vector Q_e
  2. project current f_k onto the hyperplane $P(Q_e)$, i.e.

$$f_{k+1} := f_k - \frac{\langle f_k, Q_e \rangle_R}{\langle Q_e, Q_e \rangle_R} \cdot Q_e$$

The Kaczmarz method

- compute $f_0 \in \mathbb{R}^m$ solution to $B^T f = b$, nonzero only on the edges of T



$$f_0 = \begin{vmatrix} \star \\ \star \\ \star \\ \star \\ 0 \\ 0 \end{vmatrix}$$

$\Rightarrow f_0$ in $\mathcal{O}(n)$ time

The Kaczmarz method

- Do iteratively:
  1. pick randomly one basis vector Q_e

$$proba(Q_e) := \frac{1}{R_{ee}} \cdot \frac{Q_e^T R Q_e}{\tau(T)}$$

  2. project current f_k onto the hyperplane $P(Q_e)$, i.e.

$$f_{k+1} := f_k - \frac{\langle f_k, Q_e \rangle_R}{\langle Q_e, Q_e \rangle_R} \cdot Q_e$$

any f_k is such that $B^T f_k = b$

  3. Number of iterations: $K = \mathcal{O}\left(\lceil m \log n \cdot \log(n/\epsilon) \rceil\right)$

     Tricky point: to get a running time in $\mathcal{O}(m \log^c n)$: any projection in logarithmic time

     Idea: work with two particular bases of $\mathbb{R}^m$ simultaneously

At the end of the Kaczmarz method, $f_K \in \mathbb{R}^m$ is such that

- $B^T f_K = b$
- $||f_K - f_{opt}||_R \leq \epsilon \cdot ||f_{opt}||_R$

Expected running time for Step 1:

$\mathcal{O}(m \log^2 n \log(n/\epsilon)) = \sharp$ of iterations $\times \mathcal{O}(\log n)$ time per iteration

The Kelner method: Step 2

Goal: from $f_K \in \mathbb{R}^m$ and the variable change $f = R^{-1}Bv$, given $\epsilon > 0$, find $v_K \in \mathbb{R}^n$ such that

$$||v_K - v_{opt}||_L \leq \epsilon \cdot ||v_{opt}||_L$$

Idea:

- solve $f_K = R^{-1}Bv$ exactly on the edges of T
- choose v_K the solution of minimal euclidean norm.

$f_K = R^{-1}Bv$ can be written as

$$\begin{vmatrix} f_K(T) \\ f_K(C) \end{vmatrix} = \begin{vmatrix} R_T^{-1} & 0 \\ 0 & R_C^{-1} \end{vmatrix} \cdot \begin{vmatrix} B_T \\ B_C \end{vmatrix} \cdot v$$

Solve $f_K = R^{-1}Bv$ on the edges of T, namely solve exactly

$$f_K(T) = R_T^{-1} B_T v \tag{1}$$

Choose v_K the solution to (1) of minimal euclidean norm.

- Since B_T is the incidence matrix of a tree, find any solution to

$$f_K(T) = R_T^{-1} B_T v$$

  in $\mathcal{O}(n)$ time.
- Since the kernel of B_T has dimension 1, find the euclidian minimal norm solution in $\mathcal{O}(n)$ time.

Running time for Step 2:

$$\mathcal{O}(n)$$

Expected running time

- Step 1:
  - Spanning tree T and condition number $\tau(T)$: $\mathcal{O}(m \log n)$ time
  - Initial vector $f_0 \in \mathbb{R}^m$: $\mathcal{O}(n)$ time
  - Two particular bases of $\mathbb{R}^m$: implemented via a data structure based on a tree decomposition: $\mathcal{O}(n \log n)$ time
  - Kaczmarz's method:

    $\mathcal{O}(m \log^2 n \log(n/\epsilon)) = \sharp$ of iterations $\times \mathcal{O}(\log n)$ time per iteration

- Step 2: $\mathcal{O}(n)$ time

$$\boxed{\Rightarrow \text{The Kelner method : } \mathcal{O}(m \log^2 n \log(n/\epsilon)) \text{ time}}$$

Conclusion

- Goal: solving any Laplacian system in time which is nearly linear in the number of nonzero entries
- Motivation: solving any SDD system in nearly-linear time, among others
- Best running time: $\mathcal{O}(m \log^{1/2} n)$ [Cohen, 2014]. Method: complicated !!!
- The Kelner method (2013): $\mathcal{O}(m \log^2 n)$ time, method: simple (one low-stretch spanning tree, a Kaczmarz's method)