

Preconditioned Iterative Methods

15-859NN
3/24/21

Basic Idea to get faster convergence:

Our Goal: Solve $Ax=b$ where A SPD or SPSd.

As an example consider Richardson Iteration.

Pick a matrix B and solve

$$B^{-1}Ax = B^{-1}b \quad (B \text{ is called a preconditioner})$$

Note: $A\bar{x}=b$ iff $B^{-1}A\bar{x}=B^{-1}b$

$$\begin{aligned} \text{Richardson's gives } x_{n+1} &= x_n + (B^{-1}b - B^{-1}Ax_n) \\ &= x_n + B^{-1}(b - Ax_n) \end{aligned}$$

Thus Preconditioned Richardson is:

Given x_n

- 1) Compute $r = b - Ax_n$
- 2) Solve $By = r$
- 3) Return $x_{n+1} = x_n + y$

(Step 2 \equiv Preprocess $B=LU$ or)
Recursively solve

Examples of Preconditioners

1) Extrapolated Method
 $B = \alpha I$ where $\alpha = \frac{2}{\lambda_{\min}(A) + \lambda_{\max}(A)}$

2) Jacobi $B = \text{Diagonal}(A)$

3) Gauss-Seidel $B = \text{Lower Tri}(A)$

4) SSOR?

5) Graph Theoretic (We will consider)

Convergence Rates per iteration!

Thm 1) Pre-Richardson Rate $\approx \frac{1}{K(B^{-1}A)}$

2) Pre-CG Rate $\approx \frac{1}{\sqrt{K(B^{-1}A)}}$

Goal: Find B st

1) Better convergence rate

2) Solve $By=r$ not too expensive

Estimating $\kappa(B^{-1}A)$

3

Note: $B^{-1}Ax = \lambda x$ iff $Ax = \lambda Bx$ (B invertible)

Def λ & x are called generalized eigenvalues & eigenvectors.

If A & B are not invertible but $\text{Null}(A) = \text{Null}(B)$ then

Def $\lambda_f(A, B) = \{ \lambda : \exists x Ax = \lambda Bx \text{ \& } x \in \text{Null}(A) = \text{Null}(B) \}$

Note: $Ax = \lambda Bx$ iff $Bx = (\frac{1}{\lambda})Ax$ $\lambda \neq 0$

If B spd then $Ax = \lambda Bx$ iff $B^{-\frac{1}{2}}AB^{-\frac{1}{2}}x = \lambda x$

thus if A is also spd then λ in real.

Claim $\kappa(B^{-1}A) = \frac{\text{Max } \lambda_f(A, B)}{\text{min } \lambda_f(A, B)} = \text{Max } \lambda_f(A, B) \text{Max } \lambda_f(B, A)$

Def $\gamma(A/B) = \underset{x}{\text{argmin}} Ax \preceq xB$

thus $\kappa(B^{-1}A) = \gamma(A/B) \cdot \gamma(B/A)$

Subgraph Preconditioners

Pick $H \subseteq G$ to be a subgraph of G .

Set $B = L_H$ & $A = L_G$

Now $B \preceq A$ thus $\lambda_{\mathcal{F}}(A, B) \leq 1$

& $\kappa(B^{-1}A) \leq \nabla(A/B) = \underset{\mathcal{F}}{\operatorname{argmin}} \{A \preceq \tau B\}$

For any subgraph preconditioner!

Spanning Tree Preconditioners

5

Upside: If T is a tree then solver in $O(n)$ time.

Vaidya: Pick T to be a max weight spanning tree of G .

Question: What is $\kappa(G/T) \geq \kappa(G, T)$

$$\kappa(G/T) = \min_{\tau} \{L_G \preceq \tau L_T\}$$

lets use path embedding argument to bd τ .

$$\psi: G \xrightarrow{\text{path}} T$$

$$\text{Congestion} \leq m \quad m = \# \text{edge in } G.$$

$$\text{Dilation} \leq n-1 \leq n$$

$$\text{Thus. } L_G \preceq c \cdot D L_T \leq m \cdot n L_T \Rightarrow \kappa(G, T) \leq m \cdot n$$

Worst case the best possible

$$G = \begin{array}{c} \boxed{\quad \dots \quad} \\ \downarrow \qquad \qquad \qquad \downarrow \\ 1 \qquad \qquad \qquad n/2 \end{array}$$

$$x^T L x \approx n$$

$$x^T G x \approx n^3$$

$$T = \begin{array}{c} \overset{-1}{\downarrow} \qquad \qquad \qquad \overset{-n/2}{\downarrow} \\ \boxed{\quad \dots \quad} \\ \downarrow \qquad \qquad \qquad \downarrow \\ 1 \qquad \qquad \qquad n/2 \end{array}$$

$$\kappa(G/T) = \Theta(n^2)$$

Thm Max Spanning Tree PCG is $O(m^{3/2} n^{1/2}) / \text{bit.}$

- 1) $O(m)$ time to find tree
- 2) $O(m)$ time per iteration
- 3) $O((m \cdot n)^{1/2})$ iteration per bit using PCG

Finding A Better Spanning Tree

6

Two ways: 1) Better condition number
2) Better eigen value distribution.

eg Let $G = C_n$ the cycle graph &
 $T = P_n$ its spanning tree

Claim! $\kappa(C_n, P_n) = \Theta(n)$

(\Leftarrow) $\sigma(P_n/C_n) = 1$ & $\sigma(C_n/P_n) \leq C/D = O(n)$

(\Rightarrow) Consider $x = (x_1, \dots, x_n)$ $x_i = i$

$$\frac{x^T C_n x}{x^T P_n x} = \frac{(n-1)^2 + (n-1)}{(n-1)} = n$$

$$\lambda_{\max}(P_n^+ C_n) = n \quad \lambda_{\min}(P_n^+ C_n) = 1$$

What is the average? $\sum \lambda_i$?

ie $\text{Tr}(P_n^+ C_n)$

Trace-101

7

$$1) \operatorname{Tr}(\alpha A) = \alpha \operatorname{Tr}(A)$$

$$2) \operatorname{Tr}(A) = \sum_{i=1}^n A_{ii} = \sum \lambda_i$$

$$3) \operatorname{Tr}(ABC) = \operatorname{Tr}(CAB)$$

Let $A = L_G$ & $B = L_H$ $H \not\subseteq G$

Def: $ER_B(u, v)$ = Effective Resistance in B from u to v .

Claim $\operatorname{Tr}(B^+ A) = \sum_{e \in A} w_A(e) ER_B(e)$, $e = (u, v)$

Pf Let $L_{uv} = \chi_{uv} \chi_{uv}^T$

$$\chi_{uv} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \left. \vphantom{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}} \right\} u \\ \left. \vphantom{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}} \right\} v \end{matrix}$$

$$\operatorname{Tr}(B^+ A) = \operatorname{Tr}(B^+ \sum_{(u,v) \in A} w_A(u,v) L_{uv})$$

$$= \operatorname{Tr}(\sum w_A(u,v) B^+ \chi_{uv} \chi_{uv}^T) \quad (\text{Def of } L_{uv})$$

$$= \sum w_A(u,v) \operatorname{Tr}(B^+ \chi_{uv} \chi_{uv}^T) \quad (\text{Tr of Sum})$$

$$= \sum w_A(u,v) \operatorname{Tr}(\chi_{uv}^T B^+ \chi_{uv}) \quad 3)$$

$$= \sum w_A(u,v) \chi_{uv}^T B^+ \chi_{uv} \quad (\text{Check})$$

$$= \sum w_A(u,v) ER_B(e=(u,v)) \quad (\text{Def of Resistance})$$

□

Back to C_n with tree P_n

8

$$\text{By Thm } \sum_{\lambda_i \in \lambda(C_n, P_n)} \lambda_i = \text{Tr}(P_n^+ C_n) = \sum_{e \in C_n} ER_T(e)$$

$$\text{note: } \forall e \in P_n \quad ER_T(e) = 1$$

$$e \notin P_n \quad ER_T(e) = n-1$$

$$\text{Thus } \sum_{\lambda_i \in \lambda(C_n, P_n)} \lambda_i = 2(n-1)$$

HW: λ : 1 of size $n-1$
1 of size 0
 $n-1$ of size 1

Consider poly $P(z) = (1-z)(1-z/n)$

$$\text{Thus } P(P_n^+ C_n) = 0$$

PCG on (C_n, P_n) converges in 3 iterations.

Low Stretch Spanning Trees (LSST)

9

Def Let T be spanning tree of G & $e \in E(G)$

Let w_1, \dots, w_k be edge weights in T from u to v where $e = (u, v)$

$$\text{Stretch}(e) = w_G(e) \sum_{i=1}^k \frac{1}{w_i}$$

Thm $\forall G \exists$ spanning tree T st

$$\sum_{e \in E(G)} \text{stretch}(e) = O(m \log n (\log \log n)^2)$$

pt Algorithms & proof non-trivial.

Chebyshev Polys & Acceleration 10

Def Chebyshev Polynomials

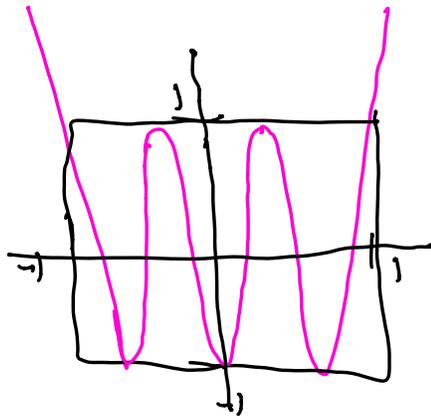
$$T_0(w) = 1 \quad T_1(w) = w$$

$$T_{n+1}(w) = 2wT_n(w) - T_{n-1}(w)$$

eg $T_2(w) = 2w \cdot w - 1 = 2w^2 - 1$

$$\begin{aligned} T_3(w) &= 2w(2w^2 - 1) - w = \\ &= 4w^3 - 2w - w = 4w^3 - 3w \end{aligned}$$

eg $T_6(w) =$



Some Simple Facts:

11

$$1) T_n(1) = 1 \Rightarrow \sum \text{coef} = 1$$

$$2) T_n(-1) = (-1)^n$$

$$3) n \text{ odd} \Rightarrow T_n(w) \text{ is odd}$$

$$n \text{ even} \Rightarrow T_n(w) \text{ is even}$$

Harder Facts

$$1) T_n(w) = \frac{1}{2} \left[(w + \sqrt{w^2 - 1})^n + (w + \sqrt{w^2 - 1})^{-n} \right]$$

(HW)

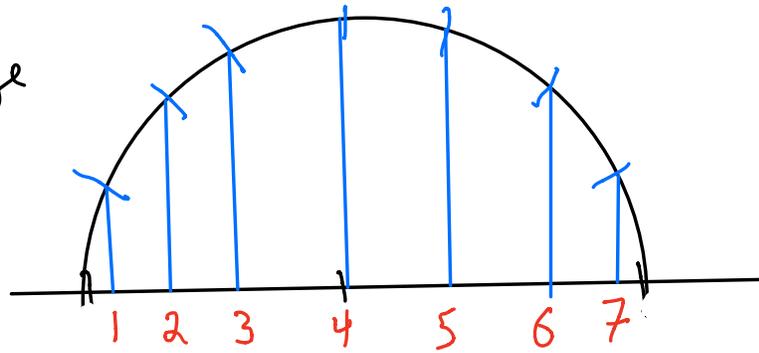
$$2) \text{Unique Poly Satisfying}$$
$$T_n(\cos(\theta)) = \cos(n\theta)$$

$$3) \text{For } |x| \leq 1$$

$$T_n(x) = \cos(n \arccos(x))$$

$$4) \text{Roots } x_k = \cos\left(\frac{k}{n+1}\pi\right)$$

Image



12

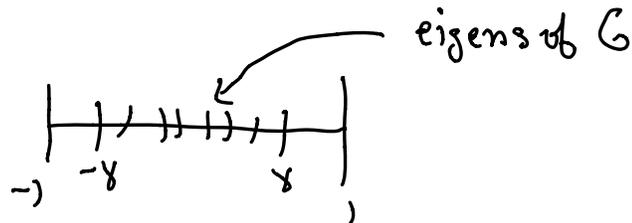
Thm $d > 1$ Let $H_n(w) = \frac{T_n(w)}{T_n(1/d)}$

1) $\max_{-1 \leq w \leq 1} |H_n(w)| = \frac{1}{T_n(1/d)}$

2) $H_n(w)$ is min deg such poly.

Chebyshev Plus Extrapolated Method 13

$$\text{let } \kappa = \kappa(A) \text{ \& } \gamma = \frac{\kappa-1}{\kappa+1}$$

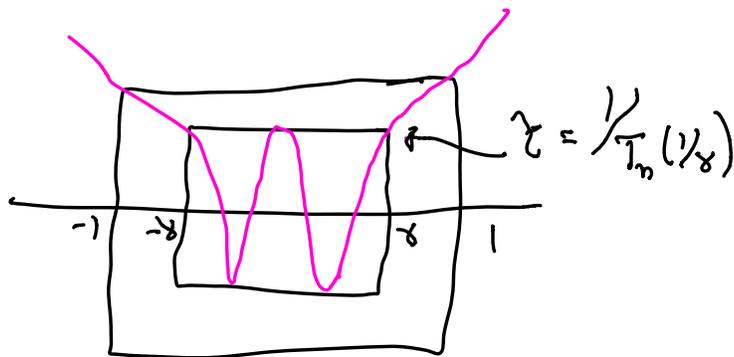


$$\text{We will use } \bar{T}_n = T_n(x/\gamma) / T_n(1/\gamma)$$

for our poly acceleration!

$$\text{Note } \bar{T}_n(1) = 1 \text{ thus } \sum \text{coef} = 1$$

picture



$$\text{Note: } |\lambda(G)| \leq \gamma \text{ thus } |\lambda(Q_n(G))| < \tau$$

Goal bd from below $T_n(1/\gamma)$

$$1/\gamma = \frac{\kappa+1}{\kappa-1} = 1 + \frac{2}{\kappa-1} = 1 + 2\mu \text{ where } \mu = \frac{1}{\kappa-1}$$

Note $T_n(x) = \frac{1}{2} \left[(x + \sqrt{x^2 - 1})^n + (x - \sqrt{x^2 - 1})^{-n} \right]$
 $\geq \frac{1}{2} (x + \sqrt{x^2 - 1})^n$ for $x \geq 1$

$$\begin{aligned} T_n(1+2\mu) &\geq \frac{1}{2} \left(1+2\mu + \sqrt{(1+2\mu)^2 - 1} \right)^n \\ &= \frac{1}{2} \left(1+2\mu + 2\sqrt{\mu(\mu+1)} \right)^n \\ &= \frac{1}{2} \left(\sqrt{\mu} + \sqrt{\mu+1} \right)^{2n} \quad (*) \end{aligned}$$

note

$$\left(\sqrt{\mu} + \sqrt{\mu+1} \right)^2 = \left(\frac{1}{\sqrt{k-1}} + \frac{\sqrt{k}}{\sqrt{k-1}} \right)^2 = \frac{(1+\sqrt{k})^2}{k-1} = \frac{\sqrt{k+1}}{\sqrt{k-1}}$$

thus $(*) = \frac{1}{2} \left(\frac{\sqrt{k+1}}{\sqrt{k-1}} \right)^n$

Finally: $T_n(1/8) \geq \frac{1}{2} \left(\frac{\sqrt{k+1}}{\sqrt{k-1}} \right)^n$

or: $r \leq 2 \left(\frac{\sqrt{k-1}}{\sqrt{k+1}} \right)^n$

Thm (SW) Suppose we have

$$\lambda_1 \geq \dots \geq \lambda_k \geq \beta \geq \lambda_{k+1} \geq \dots \geq \lambda_n \geq \alpha > 0$$

then $\forall t \geq k \exists P(x)$ of degree t s.t.

$$1) P(0) = 1$$

$$2) \forall \lambda_i \quad |P(\lambda_i)| \leq 2 \left(\frac{\sqrt{\beta/\alpha} - 1}{\sqrt{\beta/\alpha} + 1} \right)^{t-k}$$

pf Let $r(x)$ be the Chebychev poly

of deg = $t-k$ st $\forall \alpha \leq x \leq \beta$

$$|r(x)| \leq 2 \left(\frac{\sqrt{\beta/\alpha} - 1}{\sqrt{\beta/\alpha} + 1} \right)^{t-k}$$

$$\text{Set } P(x) = r(x) \prod_{\substack{\lambda_i > \beta \\ \lambda_i < \alpha}} (1 - x/\lambda_i)$$

to show: $\forall i [1 \leq i \leq n] |P(\lambda_i)| \leq 2 \left(\frac{\sqrt{\beta/\alpha} - 1}{\sqrt{\beta/\alpha} + 1} \right)^{t-k}$.

For $\lambda_i > \beta$ $P(\lambda) = 0$

For $\lambda_i < \beta$

Note: $(1 - x/\lambda_j) \leq 1$ for $0 \leq x \leq \lambda_j$

Thus $|P(\lambda_i)| \leq |r(\lambda_i)|$ \square

Consider PCG using LSST.

Let $\lambda_1 \leq \dots \leq \lambda_n \equiv \lambda(L_T^{-1} L_G)$

Note $\lambda_1 \geq 1$

$$\text{Tr}(L_T^{-1} L_G) = O(m \log n \ell^2 n)$$

We claim most eigens must be small!

Thm PCG with LSST preon

$O(m^{1/3} \log n)$ iteration per bit.

Pf We will use Thm(SW) to show:

$\exists P(z)$,

1) $\deg P = O(m^{1/3} \log n)$

2) $\forall i \lambda_i \quad |P(\lambda_i)| \leq 2/e^{z/2}$.

In Thm(SW) set $\beta = \text{Tr}(L_T^{-1} L_G)^{2/3}$

Claim: # eigens $\geq \beta$, say k $k = \sqrt{\beta}$

pf $k \cdot \beta \leq \text{Tr} \Rightarrow k \geq \sqrt{\beta}$

We have that $\lambda_1 = 0$ & $\lambda_1 \geq 1$ thus $\alpha = 1$

Set t in Thm(SW) to $t = 2\sqrt{\beta}$

Thus Thm(SW) gives

$$\forall i \quad |P(\lambda_i)| \leq 2 \left(\frac{\sqrt{\beta} - 1}{\sqrt{\beta} + 1} \right)^{2\sqrt{\beta} - \sqrt{\beta}} \quad \text{where } \deg(P) \leq 2\sqrt{\beta}$$

$$\text{Thus } \forall i \quad |P(\lambda_i)| \leq 2 \left(1 - \frac{1}{\sqrt{\beta}}\right)^{\sqrt{\beta}} \approx 2/e \quad 18$$

Thus $O(\text{Tr}(L_T^{-1} L_G)^{1/3})$ iter per bit

or $O(m^{1/3} \log n \ell^3 n)$ iter per bit.

□