An introduction to fractional calculus

Fundamental ideas and numerics

Fabio Durastante
Università di Pisa
fabio.durastante@unipi.it
fdurastante.github.io

October, 2022

Variable coefficients cases

We now want to solve the *slightly* more complex case

$$\begin{cases} \frac{\partial W}{\partial t} = d^{+}(x,t)\,{}^{RL}D^{\alpha}_{[0,x]}W(x,t) + d^{-}(x,t)\,{}^{RL}D^{\alpha}_{[x,1]}W(x,t), \\ W(0,t) = W(1,t) = 0, \qquad W(x,t) = W_0(x). \end{cases}$$

with $d^{+}(x,t), d^{-}(x,t) \geq 0$ and **not identically** zero.

Variable coefficients cases

We now want to solve the *slightly* more complex case

$$\begin{cases} \frac{\partial W}{\partial t} = d^{+}(x,t)\, {}^{RL}D^{\alpha}_{[0,x]} W(x,t) + d^{-}(x,t)\, {}^{RL}D^{\alpha}_{[x,1]} W(x,t), \\ W(0,t) = W(1,t) = 0, \qquad W(x,t) = W_0(x). \end{cases}$$

with $d^{+}(x,t), d^{-}(x,t) \geq 0$ and **not identically** zero.

1. We go through all the **same discretization procedure**: from Riemann–Liouville to (shifted) Grünwald–Letnikov, then series truncation, *etc.*

Variable coefficients cases

We now want to solve the *slightly* more complex case

$$\begin{cases} \frac{\partial W}{\partial t} = d^{+}(x,t)\, {}^{RL}D^{\alpha}_{[0,x]} W(x,t) + d^{-}(x,t)\, {}^{RL}D^{\alpha}_{[x,1]} W(x,t), \\ W(0,t) = W(1,t) = 0, \qquad W(x,t) = W_0(x). \end{cases}$$

with $d^{+}(x,t), d^{-}(x,t) \geq 0$ and **not identically** zero.

1. We go through all the **same discretization procedure**: from Riemann–Liouville to (shifted) Grünwald–Letnikov, then series truncation, *etc.*
2. we obtain a matrix sequence of the form

$$A_N = \nu I_N - \left(D_N^{+} G_N + D_N^{-} G_N^{T} \right),$$

where $D_N^{\pm}$ are **diagonal matrices** whose entries **sample the functions** $d_N^{\pm}(x,t)$ on the finite difference grid.

Variable coefficients cases

We now want to solve the *slightly* more complex case

$$\begin{cases} \frac{\partial W}{\partial t} = d^{+}(x,t)\, {}^{RL}D^{\alpha}_{[0,x]} W(x,t) + d^{-}(x,t)\, {}^{RL}D^{\alpha}_{[x,1]} W(x,t), \\ W(0,t) = W(1,t) = 0, \qquad W(x,t) = W_0(x). \end{cases}$$

with $d^{+}(x,t), d^{-}(x,t) \geq 0$ and **not identically** zero.

1. We go through all the **same discretization procedure**: from Riemann–Liouville to (shifted) Grünwald–Letnikov, then series truncation, *etc.*
2. we obtain a matrix sequence of the form

$$A_N = \nu I_N - \left(D_N^{+} G_N + D_N^{-} G_N^{T} \right),$$

   where $D_N^{\pm}$ are **diagonal matrices** whose entries **sample the functions** $d_N^{\pm}(x,t)$ on the finite difference grid.

We **no longer have Toeplitz matrices**!

Not all hope is lost

- We can still perform **fast matrix-vector products**:

$$A_N\mathbf{x} = \nu\mathbf{x} - D_N^{+}(G_N\mathbf{x}) - D_N^{-}(G_N^T\mathbf{x})$$

  still $O(N\log N)$ cost.

- Maybe we can use some trick to reuse circulant preconditioners
  1. If $d_N^{\pm}(x,t)$ do not vary much maybe we can **average them**, i.e.,

$$P(t) = \nu I - \widehat{d^{+}}(t)s(G_N) - \widehat{d^{-}}(t)s(G_N^T),$$

  with $\widehat{d^{\pm}}(t) = {}^{1}\!/_{N}\sum_{i=1}^{N} d^{\pm}(x_i,t)$

The averaging trick

Does it work?

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

```
w0 = @(x) 5*x.*(1-x);
hN = 1/(N-1); x = 0:hN:1; dt = hN; t = 0:dt:1;
dplus = @(x,t) gamma(3-alpha).*x.^alpha;
dminus = @(x,t) gamma(3-alpha).*(2-x).^alpha;
% Discretize
G = glmatrix(N,alpha); Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x,0)); Dminus = diag(dminus(x,0));
% Left-hand side
nu = hN^alpha/dt;
A = nu*I - (Dplus*Gr + Dminus*Grt);
```

The averaging trick

Does it work?

$$d^+(x,t) = \Gamma(3-\alpha)x^\alpha, \qquad d^-(x,t) = \Gamma(3-\alpha)(2-x)^\alpha$$

```
% Solve
[ev,evt] = sunprec(N,alpha);
c = nu + mean(dplus(x,0))*ev + mean(dminus(x,0))*evt;
P = @(x) cprec(c,x);
[X,FLAGsun,RELRESsun,ITERsun,RESVECsun] = gmres(A,(nu*w),[],1e-9,N,P);
```

The averaging trick

Does it work?

$$d^+(x,t) = \Gamma(3-\alpha)x^\alpha, \qquad d^-(x,t) = \Gamma(3-\alpha)(2-x)^\alpha$$

α	N	GMRES	P
	2^5	31	13
	2^6	50	14
	2^7	64	14
1.2	2^8	75	15
	2^9	84	15
	2^{10}	91	14
	2^{11}	96	14

α	N	GMRES	P
	2^5	31	13
	2^6	59	14
	2^7	92	15
1.4	2^8	127	15
	2^9	161	15
	2^{10}	196	15
	2^{11}	231	15

α	N	GMRES	P
	2^5	32	13
	2^6	62	13
	2^7	112	14
1.6	2^8	183	14
	2^9	262	14
	2^{10}	353	14
	2^{11}	456	14

α	N	GMRES	P
	2^5	32	12
	2^6	64	12
	2^7	126	13
1.8	2^8	225	13
	2^9	378	13
	2^{10}	559	12
	2^{11}	779	12

The averaging trick

Does it work?

$$d^+(x,t) = \Gamma(3-\alpha)x^\alpha, \qquad d^-(x,t) = \Gamma(3-\alpha)(2-x)^\alpha$$

α	N	GMRES	P
	2^5	31	13
	2^6	50	14
	2^7	64	14
1.2	2^8	75	15
	2^9	84	15
	2^{10}	91	14
	2^{11}	96	14

α	N	GMRES	P
	2^5	31	13
	2^6	59	14
	2^7	92	15
1.4	2^8	127	15
	2^9	161	15
	2^{10}	196	15
	2^{11}	231	15

α	N	GMRES	P
	2^5	32	13
	2^6	62	13
	2^7	112	14
1.6	2^8	183	14
	2^9	262	14
	2^{10}	353	14
	2^{11}	456	14

α	N	GMRES	P
	2^5	32	12
	2^6	64	12
	2^7	126	13
1.8	2^8	225	13
	2^9	378	13
	2^{10}	559	12
	2^{11}	779	12

We have **doubled the number of iterations** but things still seem reasonable...

Can we prove anything?

What did we actually prove for the *constant coefficient case*?

Can we prove anything?

What did we actually prove for the *constant coefficient case*?

- We computed the **asymptotic spectral distribution** of the matrix sequence $\{\nu A_N\}_N$ (*eigenvalues* for the symmetric case, *singular values* for the general case);

Can we prove anything?

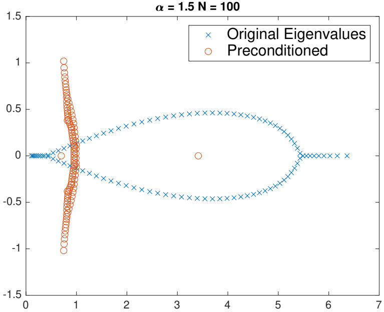
What did we actually prove for the *constant coefficient case*?

- We computed the **asymptotic spectral distribution** of the matrix sequence $\{\nu A_N\}_N$ (*eigenvalues* for the symmetric case, *singular values* for the general case);
- We proved that $P^{-1}A_N - I =$ "small norm" $+$ "small rank", i.e., that the preconditioner delivered a **clustering of the eigenvalues**.

Can we prove anything?

What did we actually prove for the *constant coefficient case*?

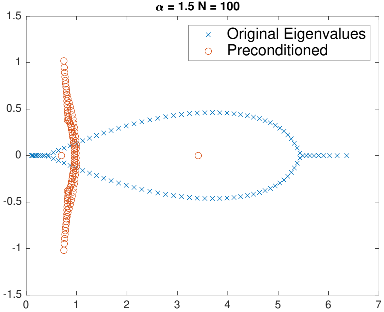
- We proved that $P^{-1}A_N - I =$ "small norm" + "small rank", i.e., that the preconditioner delivered a **clustering of the eigenvalues**.

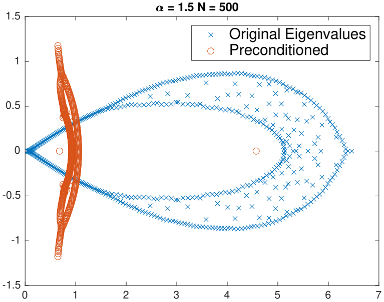


Can we prove anything?

What did we actually prove for the *constant coefficient case*?

- We proved that $P^{-1}A_N - I =$"small norm"+"small rank", i.e., that the preconditioner delivered a **clustering of the eigenvalues**.

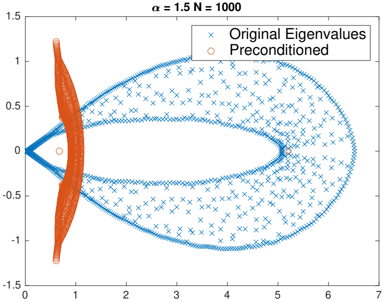


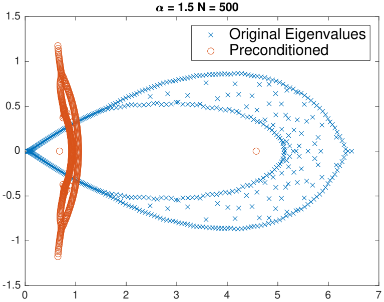


Can we prove anything?

What did we actually prove for the *constant coefficient case*?

- We proved that $P^{-1}A_N - I =$"small norm"+"small rank", i.e., that the preconditioner delivered a **clustering of the eigenvalues**.

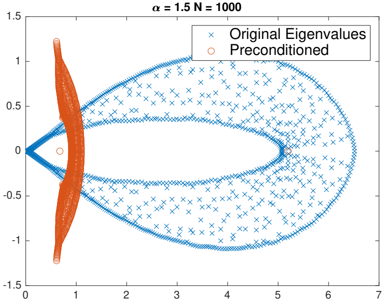


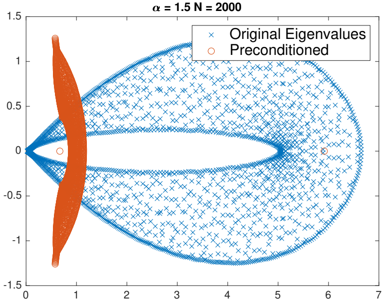


Can we prove anything?

What did we actually prove for the *constant coefficient case*?

- We proved that $P^{-1}A_N - I =$"small norm"$+$"small rank", i.e., that the preconditioner delivered a **clustering of the eigenvalues**.

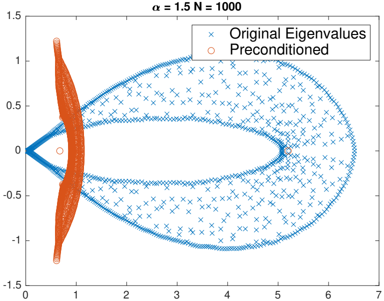


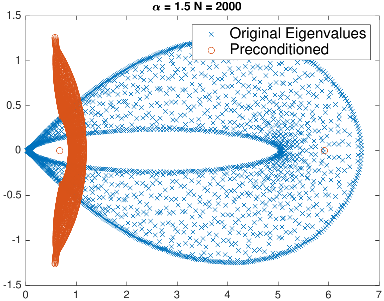


Can we prove anything?

What did we actually prove for the *constant coefficient case*?

- We proved that $P^{-1}A_N - I =$"small norm"$+$"small rank", i.e., that the preconditioner delivered a **clustering of the eigenvalues**.



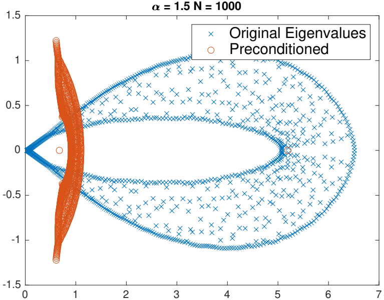


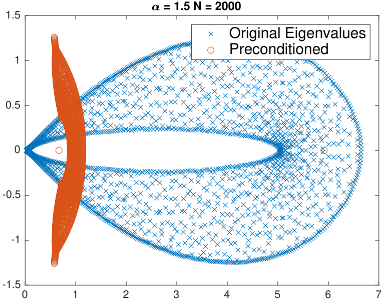
- We **don't have a cluster**, yet eigenvalues are in a fairly small region.

Can we prove anything?

What did we actually prove for the *constant coefficient case*?

- We proved that $P^{-1}A_N - I =$"small norm"$+$"small rank", i.e., that the preconditioner delivered a **clustering of the eigenvalues**.





We **don't have a cluster**, yet eigenvalues are in a fairly small region.

let's investigate!

Having a cluster: $C_n - A_n$

For two matrix sequences $\{C_n\}_n$ and $\{A_n\}_n$ (both of order n) we say that they are ε-close by rank if

$$\forall\, \varepsilon > 0 \; A_n - C_n = E_{n,\varepsilon} + R_{n,\varepsilon}, \qquad \begin{array}{l} \|E_{n,\varepsilon}\|_2 \leq \varepsilon, \\ \mathrm{rank}(R_{n,\varepsilon}) \leq r(n,\varepsilon) = o(n) \text{ for } n \to +\infty, \end{array} \tag{ε-close}$$

Having a cluster: $C_n - A_n$

For two matrix sequences $\{C_n\}_n$ and $\{A_n\}_n$ (both of order n) we say that they are ε-close by rank if

$$\forall\, \varepsilon > 0 \; A_n - C_n = E_{n,\varepsilon} + R_{n,\varepsilon}, \quad \begin{array}{l} \|E_{n,\varepsilon}\|_2 \leq \varepsilon, \\ \mathrm{rank}(R_{n,\varepsilon}) \leq r(n,\varepsilon) = o(n) \text{ for } n \to +\infty, \end{array} \qquad (\varepsilon\text{-close})$$

Let $\gamma_n(\varepsilon)$ count how many singular values $\sigma(A_n - C_n)$ are greater than ε, i.e.,

$$\gamma_n(\varepsilon) = \left|\{j \,:\, \sigma_j(A_n - C_n) > \varepsilon, \quad j = 1, \ldots, n\}\right|,$$

Having a cluster: $C_n - A_n$

For two matrix sequences $\{C_n\}_n$ and $\{A_n\}_n$ (both of order n) we say that they are ε-close by rank if

$$\forall\, \varepsilon > 0 \; A_n - C_n = E_{n,\varepsilon} + R_{n,\varepsilon}, \qquad \begin{array}{l} \|E_{n,\varepsilon}\|_2 \leq \varepsilon, \\ \mathrm{rank}(R_{n,\varepsilon}) \leq r(n,\varepsilon) = o(n) \text{ for } n \to +\infty, \end{array} \tag{ε-close}$$

Let $\gamma_n(\varepsilon)$ count how many singular values $\sigma(A_n - C_n)$ are greater than ε, i.e.,

$$\gamma_n(\varepsilon) = \left|\{j \,:\, \sigma_j(A_n - C_n) > \varepsilon, \quad j = 1, \ldots, n\}\right|,$$

$\Rightarrow$ (ε-close) is telling us that $\gamma_n(\varepsilon) = o(n)$ for $n \to +\infty$.

Having a cluster: $C_n - A_n$

For two matrix sequences $\{C_n\}_n$ and $\{A_n\}_n$ (both of order n) we say that they are ε-close by rank if

$$\forall\, \varepsilon > 0 \; A_n - C_n = E_{n,\varepsilon} + R_{n,\varepsilon}, \quad \begin{array}{l} \|E_{n,\varepsilon}\|_2 \leq \varepsilon, \\ \operatorname{rank}(R_{n,\varepsilon}) \leq r(n,\varepsilon) = o(n) \text{ for } n \to +\infty, \end{array} \tag{ε-close}$$

Let $\gamma_n(\varepsilon)$ count how many singular values $\sigma(A_n - C_n)$ are greater than ε, i.e.,

$$\gamma_n(\varepsilon) = \left|\{j \,:\, \sigma_j(A_n - C_n) > \varepsilon, \quad j = 1, \ldots, n\}\right|,$$

$\Rightarrow$ (ε-close) is telling us that $\gamma_n(\varepsilon) = o(n)$ for $n \to +\infty$.

Then we know that $\{A_n - C_n\}_n$ has a singular value **cluster** at zero, if $\gamma_n(\varepsilon) = O(1)$ which holds equally with $r(n,\varepsilon) = r(\varepsilon) = O(1)$ for any $\varepsilon > 0$ then we have a **proper cluster** by the definition we have seen during the last lecture.

Having a cluster: $C_n^{-1}A_n - I_n$

To **estimate the convergence rate** we have shown that $C_n^{-1}A_n$ and I_n are (ε-close) matrix sequences, one usually use the **following nomenclature**

- C_n is **superlinear** for A_n if $r(n, \varepsilon) = O(1)$,
- C_n is **sublinear** for A_n if $r(n, \varepsilon) = o(n)$.

Having a cluster: $C_n^{-1}A_n - I_n$

To **estimate the convergence rate** we have shown that $C_n^{-1}A_n$ and I_n are (ε-close) matrix sequences, one usually use the **following nomenclature**

- C_n is **superlinear** for A_n if $r(n, \varepsilon) = O(1)$,
- C_n is **sublinear** for A_n if $r(n, \varepsilon) = o(n)$.

Strategy

It is usually easier to prove that C_n and A_n are (ε-close), rather than $C_n^{-1}A_n$ and I_n.

Having a cluster: $C_n^{-1}A_n - I_n$

To **estimate the convergence rate** we have shown that $C_n^{-1}A_n$ and I_n are (ε-close) matrix sequences, one usually use the **following nomenclature**

- C_n is **superlinear** for A_n if $r(n, \varepsilon) = O(1)$,
- C_n is **sublinear** for A_n if $r(n, \varepsilon) = o(n)$.

Strategy

It is usually easier to prove that C_n and A_n are (ε-close), rather than $C_n^{-1}A_n$ and I_n.

Proposition

If C_n and C_n^{-1} are **bounded uniformly** in n, then A_n and C_n are (ε-close) by $O(1)$ rank if and only if $C_n^{-1}A_n$ and I_n are.

Having a cluster: $C_n^{-1}A_n - I_n$

To **estimate the convergence rate** we have shown that $C_n^{-1}A_n$ and I_n are (ε-close) matrix sequences, one usually use the **following nomenclature**

- C_n is **superlinear** for A_n if $r(n, \varepsilon) = O(1)$,
- C_n is **sublinear** for A_n if $r(n, \varepsilon) = o(n)$.

Strategy

It is usually easier to prove that C_n and A_n are (ε-close), rather than $C_n^{-1}A_n$ and I_n.

Proposition

If C_n and C_n^{-1} are **bounded uniformly** in n, then A_n and C_n are (ε-close) by $O(1)$ rank if and only if $C_n^{-1}A_n$ and I_n are.

Proof.

$$A_n - C_n = C_n(C_n^{-1}A_n - I_n), \text{ and } C_n^{-1}A_n - I_n = C_n^{-1}(A_n - C_n).$$

Having a cluster: $C_n^{-1}A_n - I_n$

To **estimate the convergence rate** we have shown that $C_n^{-1}A_n$ and I_n are (ε-close) matrix sequences, one usually use the **following nomenclature**

- C_n is **superlinear** for A_n if $r(n,\varepsilon) = O(1)$,
- C_n is **sublinear** for A_n if $r(n,\varepsilon) = o(n)$.

Strategy

It is usually easier to prove that C_n and A_n are (ε-close), rather than $C_n^{-1}A_n$ and I_n.

Proposition

If C_n and C_n^{-1} are **bounded uniformly** in n, then A_n and C_n are (ε-close) by $O(1)$ rank if and only if $C_n^{-1}A_n$ and I_n are.

Proof.

$$C_n^{-1}A_n - I_n = C_n^{-1}(E_{n,\varepsilon} + R_{n,\varepsilon}) = C_n^{-1}E_{n,\varepsilon} + C_n^{-1}R_{n,\varepsilon}$$

Having a cluster: $C_n^{-1}A_n - I_n$

To **estimate the convergence rate** we have shown that $C_n^{-1}A_n$ and I_n are (ε-close) matrix sequences, one usually use the **following nomenclature**

- C_n is **superlinear** for A_n if $r(n,\varepsilon) = O(1)$,
- C_n is **sublinear** for A_n if $r(n,\varepsilon) = o(n)$.

Strategy

It is usually easier to prove that C_n and A_n are (ε-close), rather than $C_n^{-1}A_n$ and I_n.

Proposition

If C_n and C_n^{-1} are **bounded uniformly** in n, then A_n and C_n are (ε-close) by $O(1)$ rank if and only if $C_n^{-1}A_n$ and I_n are.

Proof.

$$C_n^{-1}A_n - I_n = C_n^{-1}E_{n,\varepsilon} + C_n^{-1}R_{n,\varepsilon}, \quad \|C_n^{-1}E_{n,\varepsilon}\| \leq \varepsilon/\|C_n\|_2, \quad \operatorname{rank}(C_n^{-1}R_{n,\varepsilon}) \leq r(n,\varepsilon) = O(1). \quad \square$$

Having a cluster: $C_n^{-1}A_n - I_n$

The connection between boundedness and ε-closeness can also be inverted, i.e.,

Proposition

Let C_n be non singular. If C_n is bounded uniformly in n and A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n **is not superlinear** for A_n.

Proof.

- Both propositions makes assumption on C_n, can we say something without having to impose anything on C_n, $\|C_n\|_2$ or $\|C_n^{-1}\|_2$?

Having a cluster: $C_n^{-1}A_n - I_n$

The connection between boundedness and ε-closeness can also be inverted, i.e.,

Proposition

Let C_n be non singular. If C_n is bounded uniformly in n and A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n **is not superlinear** for A_n.

Proof. By *contradiction*, if C_n is *superlinear* for A_n, then $C_n^{-1}A_n - I_n$ is the sum of a term of bounded norm $\varepsilon/\|C_n\|_2$ and a term of rank bounded by $O(1)$.

- Both propositions makes assumption on C_n, can we say something without having to impose anything on C_n, $\|C_n\|_2$ or $\|C_n^{-1}\|_2$?

Having a cluster: $C_n^{-1}A_n - I_n$

The connection between boundedness and ε-closeness can also be inverted, i.e.,

Proposition

Let C_n be non singular. If C_n is bounded uniformly in n and A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n **is not superlinear** for A_n.

Proof. By *contradiction*, if C_n is *superlinear* for A_n, then $C_n^{-1}A_n - I_n$ is the sum of a term of bounded norm $\varepsilon/\|C_n\|_2$ and a term of rank bounded by $O(1)$. Therefore,

$$A_n - C_n = C_n(C_n^{-1}A_n - I_n),$$

is the sum of a term of norm bounded by ε and a term of *constant rank*: this contradicts the assumption that A_n and C_n are not (ε-close) by $O(1)$ rank. $\square$

- Both propositions makes assumption on C_n, can we say something without having to impose anything on C_n, $\|C_n\|_2$ or $\|C_n^{-1}\|_2$?

Having a cluster: $C_n^{-1}A_n - I_n$

Proposition

Let A_n and C_n be non singular. If A_n is bounded uniformly in n and if A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n is not superlinear for A_n.

Proof.

Having a cluster: $C_n^{-1}A_n - I_n$

Proposition

Let A_n and C_n be non singular. If A_n is bounded uniformly in n and if A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n is not superlinear for A_n.

Proof. We prove it again by contradiction.

Having a cluster: $C_n^{-1}A_n - I_n$

Proposition

Let A_n and C_n be non singular. If A_n is bounded uniformly in n and if A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n is not superlinear for A_n.

Proof. We prove it again by contradiction. If C_n is superlinear for A_n, then (ε-close) holds for $C_n^{-1}A_n - I_n$ with $r(n, \varepsilon) = O(1)$.

Having a cluster: $C_n^{-1}A_n - I_n$

Proposition

Let A_n and C_n be non singular. If A_n is bounded uniformly in n and if A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n is not superlinear for A_n.

Proof. We prove it again by contradiction. If C_n is superlinear for A_n, then (ε-close) holds for $C_n^{-1}A_n - I_n$ with $r(n, \varepsilon) = O(1)$. We use Sherman-Morrison-Woodbury formula to show that

$$A_n^{-1}C_n - I_n = E_{n,\varepsilon} + R_{n,\varepsilon}, \quad \|E_{n,\varepsilon} < \varepsilon \text{ and } R_{n,\varepsilon} = O(1).$$

Having a cluster: $C_n^{-1}A_n - I_n$

Proposition

Let A_n and C_n be non singular. If A_n is bounded uniformly in n and if A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n is not superlinear for A_n.

Proof. We prove it again by contradiction. If C_n is superlinear for A_n, then (ε-close) holds for $C_n^{-1}A_n - I_n$ with $r(n, \varepsilon) = O(1)$. We use Sherman-Morrison-Woodbury formula to show that

$$A_n^{-1}C_n - I_n = E_{n,\varepsilon} + R_{n,\varepsilon}, \quad \|E_{n,\varepsilon} < \varepsilon \text{ and } R_{n,\varepsilon} = O(1).$$

Therefore,

$$-(A_n - C_n) = A_n(A_n^{-1}C_n - I_n)$$

is the sum of a term of norm bounded by $O(\varepsilon)$ and a term o constant rank this contradicts A_n and C_n non being (ε-close) by $O(1)$ rank. $\square$

Having a cluster: $C_n^{-1}A_n - I_n$

Proposition

Let A_n and C_n be non singular. If A_n is bounded uniformly in n and if A_n and C_n are not (ε-close) by $O(1)$ rank, then C_n is not superlinear for A_n.

Proof. We prove it again by contradiction. If C_n is superlinear for A_n, then (ε-close) holds for $C_n^{-1}A_n - I_n$ with $r(n,\varepsilon) = O(1)$. We use Sherman-Morrison-Woodbury formula to show that

$$A_n^{-1}C_n - I_n = E_{n,\varepsilon} + R_{n,\varepsilon}, \quad \|E_{n,\varepsilon} < \varepsilon \text{ and } R_{n,\varepsilon} = O(1).$$

Therefore,

$$-(A_n - C_n) = A_n(A_n^{-1}C_n - I_n)$$

is the sum of a term of norm bounded by $O(\varepsilon)$ and a term o constant rank this contradicts A_n and C_n non being (ε-close) by $O(1)$ rank. □

- If we have information on the *spectral distribution* of the involved sequences, can we conclude something?

Asymptotic spectral distribution for non-Toeplitz sequences

For **Toeplitz matrices** we discovered that the following definitions holds for suitably chosen generating functions f.

Asymptotic eigenvalue distribution

Given a sequence of matrices $\{X_n\}_n \in \mathbb{C}^{d_n \times d_n}$ with $d_n = \{\dim X_n\}_n \xrightarrow{n \to +\infty} \infty$ monotonically and a μ-measurable function $f : D \to \mathbb{R}$, with $\mu(D) \in (0, \infty)$, we say that the sequence $\{X\}_n$ is distributed in the sense of the eigenvalues as the function f and write $\{X_n\}_n \sim_\lambda f$ if and only if,

$$\lim_{n \to \infty} \frac{1}{d_n} \sum_{j=0}^{d_n} F(\lambda_j(X_n)) = \frac{1}{\mu(D)} \int_D F(f(t))dt, \;\; \forall F \in \mathcal{C}_c(D),$$

where $\lambda_j(\cdot)$ indicates the j-th eigenvalue.

Asymptotic spectral distribution for non-Toeplitz sequences

For **Toeplitz matrices** we discovered that the following definitions holds for suitably chosen generating functions f.

Asymptotic singular value distribution

Given a sequence of matrices $\{X_n\}_n \in \mathbb{C}^{d_n \times d_n}$ with $d_n = \{\dim X_n\}_n \overset{n\to+\infty}{\longrightarrow} \infty$ monotonically and a μ-measurable function $f : D \to \mathbb{R}$, with $\mu(D) \in (0, \infty)$, we say that the sequence $\{X\}_n$ is distributed in the sense of the singular values as the function f and write $\{X_n\}_n \sim_\sigma f$ if and only if

$$\lim_{n\to\infty} \frac{1}{d_n} \sum_{j=0}^{d_n} F(\sigma_j(X_n)) = \frac{1}{\mu(D)} \int_D F(|f(t)|)dt, \quad \forall F \in \mathcal{C}_c(D),$$

where $\sigma_j(\cdot)$ is the j-th singular value.

Asymptotic spectral distribution for non-Toeplitz sequences

For **Toeplitz matrices** we discovered that the following definitions holds for suitably chosen generating functions f.

- ❓ Are there any other matrix sequences for which these definitions hold?

Asymptotic spectral distribution for non-Toeplitz sequences

For **Toeplitz matrices** we discovered that the following definitions holds for suitably chosen generating functions f.

- ❓ Are there any other matrix sequences for which these definitions hold?

1. Sequence of matrices describing the **energy on fractals**, *e.g.*, a version of the Szegö limit theorems on the Sierpiński gasket (Okoudjou, Rogers, and Strichartz 2010);

Asymptotic spectral distribution for non-Toeplitz sequences

For **Toeplitz matrices** we discovered that the following definitions holds for suitably chosen generating functions f.

- ❓ Are there any other matrix sequences for which these definitions hold?

1. Sequence of matrices describing the **energy on fractals**, *e.g.*, a version of the Szegö limit theorems on the Sierpiński gasket (Okoudjou, Rogers, and Strichartz 2010);
2. **L**ocally **T**oeplitz Sequences (Tilli 1998);

Asymptotic spectral distribution for non-Toeplitz sequences

For **Toeplitz matrices** we discovered that the following definitions holds for suitably chosen generating functions f.

- ? Are there any other matrix sequences for which these definitions hold?

1. Sequence of matrices describing the **energy on fractals**, *e.g.*, a version of the Szegö limit theorems on the Sierpiński gasket (Okoudjou, Rogers, and Strichartz 2010);
2. **L**ocally **T**oeplitz Sequences (Tilli 1998);
3. **G**eneralized **L**ocally **T**oeplitz Sequences (Garoni and Serra-Capizzano 2017, 2018).

Asymptotic spectral distribution for non-Toeplitz sequences

For **Toeplitz matrices** we discovered that the following definitions holds for suitably chosen generating functions f.

? Are there any other matrix sequences for which these definitions hold?

1. Sequence of matrices describing the **energy on fractals**, *e.g.*, a version of the Szegö limit theorems on the Sierpiński gasket (Okoudjou, Rogers, and Strichartz 2010);
2. **L**ocally **T**oeplitz Sequences (Tilli 1998);
3. **G**eneralized **L**ocally **T**oeplitz Sequences (Garoni and Serra-Capizzano 2017, 2018).

GLT Sequences

They are a $*$-algebra of matrix sequences $\{A_N\}_N$ to which we can extend some of the techniques and results we have briefly discussed for Toeplitz sequences. They can be used to describe **asymptotic spectral properties** of matrix sequences coming from the **discretization of differential equations** on **highly regular meshes**.

GLT Sequences (Garoni and Serra-Capizzano 2017, 2018)

The **machinery** and the **relative notation** is unfortunately **cumbersome**.

GLT Sequences (without the agonizing pain)

☺ We need just **few tools** to get a couple of results for the case at hand.

GLT Sequences (without the agonizing pain)

☺ We need just **few tools** to get a couple of results for the case at hand.

Theorem (Axiomatic description) (Garoni and Serra-Capizzano 2017, 2018)

1. Each GLT sequence has a singular value symbol $f(x, \theta)$ for $(x, \theta) \in [0, 1] \times [-\pi, \pi]$. If the sequence is Hermitian, then the distribution also holds in the eigenvalue sense. If $\{A_N\}_N$ has a GLT symbol $f(x, \theta)$ we will write $\{A_N\}_N \sim_{\text{GLT}} f(x, \theta)$.
2. The set of GLT sequences form a $*$-algebra, i.e., it is closed under linear combinations, products, inversion (whenever the symbol is singular, at most, in a set of zero Lebesgue measure), and conjugation.
3. Every Toeplitz sequence generated by an $\mathbb{L}^1$ function $f = f(\theta)$ is a GLT sequence and its symbol is f. Every *diagonal sampling* matrix $(D_n)_{ii} = a(i/n)$ obtained from a continuous $a(x)$ is a GLT sequence and its symbol is a.
4. Every sequence which is distributed as the constant zero in the singular value sense is a GLT sequence with symbol 0.

GLT Sequences (without the agonizing pain)

☺ We need just **few tools** to get a couple of results for the case at hand.

Theorem (Axiomatic description) (Garoni and Serra-Capizzano 2017, 2018)

5. If $\{A_N\}_N \sim_{\mathrm{GLT}} \kappa$ and the matrices A_N are such that $A_N = X_N + Y_n$, where
   - every X_N is Hermitian,
   - the spectral norms of X_N and Y_N are uniformly bounded with respect to N,
   - the trace-norm of Y_N divided by the matrix size N converges to 0,

   then the distribution holds in the eigenvalue sense.

GLT Sequences (without the agonizing pain)

We need just **few tools** to get a couple of results for the case at hand.

Theorem (Axiomatic description) (Garoni and Serra-Capizzano 2017, 2018)

5. If $\{A_N\}_N \sim_{\mathrm{GLT}} \kappa$ and the matrices A_N are such that $A_N = X_N + Y_n$, where
   - every X_N is Hermitian,
   - the spectral norms of X_N and Y_N are uniformly bounded with respect to N,
   - the trace-norm of Y_N divided by the matrix size N converges to 0,

   then the distribution holds in the eigenvalue sense.

Okay, but what do we do with this stuff?

GLT Sequences (without the agonizing pain)

We need just **few tools** to get a couple of results for the case at hand.

Theorem (Axiomatic description) (Garoni and Serra-Capizzano 2017, 2018)

5. If $\{A_N\}_N \sim_{\mathrm{GLT}} \kappa$ and the matrices A_N are such that $A_N = X_N + Y_n$, where
   - every X_N is Hermitian,
   - the spectral norms of X_N and Y_N are uniformly bounded with respect to N,
   - the trace-norm of Y_N divided by the matrix size N converges to 0,

   then the distribution holds in the eigenvalue sense.

Okay, but what do we do with this stuff?

We take the sequence we have $\{A_n\}_n$ from our problem, and we try to show that it can be obtained via the $*$-algebra properties as the linear combination/product (with maybe some inversions and some zero distributed sequences) of GLT matrices of which we know the symbol (a.k.a., Toeplitz and diagonal matrices).

GLT Sequences (without the agonizing pain)

We need just **few tools** to get a couple of results for the case at hand.

Theorem (Axiomatic description) (Garoni and Serra-Capizzano 2017, 2018)

5. If $\{A_N\}_N \sim_{\mathrm{GLT}} \kappa$ and the matrices A_N are such that $A_N = X_N + Y_n$, where
   - every X_N is Hermitian,
   - the spectral norms of X_N and Y_N are uniformly bounded with respect to N,
   - the trace-norm of Y_N divided by the matrix size N converges to 0,

   then the distribution holds in the eigenvalue sense.

- Okay, but what do we do with this stuff?
- We take the sequence we have $\{A_n\}_n$ from our problem, and we try to show that it can be obtained via the $*$-algebra properties as the linear combination/product (with maybe some inversions and some zero distributed sequences) of GLT matrices of which we know the symbol (a.k.a., Toeplitz and diagonal matrices).
- If we are successful, then we know the spectral distribution of our sequence.

GLT stuff for the case at hand

We want to discover the **GLT symbol**, a.k.a., the **spectral distribution** for the discretization of:

$$\begin{cases} \frac{\partial W}{\partial t} = d^{+}(x,t)\, {}^{RL}D^{\alpha}_{[0,x]} W(x,t) + d^{-}(x,t)\, {}^{RL}D^{\alpha}_{[x,1]} W(x,t), \\ W(0,t) = W(1,t) = 0, \qquad W(x,t) = W_0(x). \end{cases}$$

GLT stuff for the case at hand

We want to discover the **GLT symbol**, a.k.a., the **spectral distribution** for:

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

GLT stuff for the case at hand

We want to discover the **GLT symbol**, a.k.a., the **spectral distribution** for:

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

Theorem (Donatelli, Mazza, and Serra-Capizzano 2016)

We assume $\nu = O(1)$, and that for a fixed instant of time t_m the functions $d^+(x,t) \equiv d^+(x)$ and $d^-(x,t) \equiv d^-(x)$ are both Riemann integrable over $[0,1]$, then

$$\{A_N\}_N \sim_{\mathrm{GLT}} h_\alpha(x,\theta) = d^+(x) f_\alpha(\theta) + d^-(x) f_\alpha(-\theta), \quad (x,\theta) \in [0,1] \times [-\pi,\pi].$$

GLT stuff for the case at hand

We want to discover the **GLT symbol**, a.k.a., the **spectral distribution** for:

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

Theorem (Donatelli, Mazza, and Serra-Capizzano 2016)

We assume $\nu = O(1)$, and that for a fixed instant of time t_m the functions $d^+(x,t) \equiv d^+(x)$ and $d^-(x,t) \equiv d^-(x)$ are both Riemann integrable over $[0,1]$, then

$$\{A_N\}_N \sim_{\text{GLT}} h_\alpha(x,\theta) = d^+(x) f_\alpha(\theta) + d^-(x) f_\alpha(-\theta), \quad (x,\theta) \in [0,1] \times [-\pi,\pi].$$

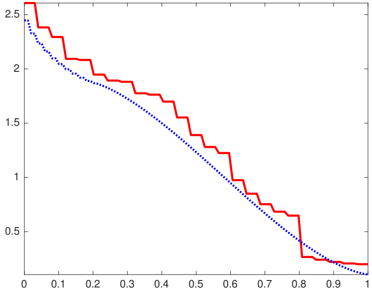
Proof. The diagonal elements of the matrices $D_N^\pm$ are a uniform sampling of the functions $d^\pm(x) \in [0,1]$, thus $D_N^\pm \sim_{\mathrm{GLT}} d^\pm(x)$. Toeplitz matrices G_N and G_N^T are also $\{G_N\}_N \sim_{\mathrm{GLT}} f_\alpha(\theta)$ and $\{G_N^T\}_N \sim_{\mathrm{GLT}} f_\alpha(-\theta)$. Finally $\{\nu I_N\}_N \sim_{\mathrm{GLT}} 0$ since $\nu = o(1)$ by hypothesis. The conclusion than follows from the $*$-algebra property, i.e.,

$$\{A_N\}_N \sim_{\mathrm{GLT}} 0 + d^+(x) p_\alpha(\theta) + d^-(x) p_\alpha(-\theta) = h_\alpha(x,\theta). \quad \square$$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```

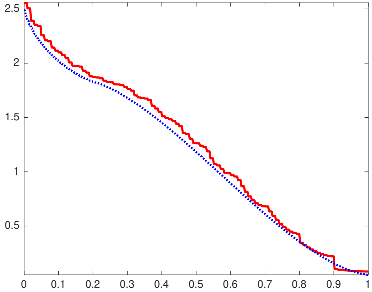


$N = 100$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```

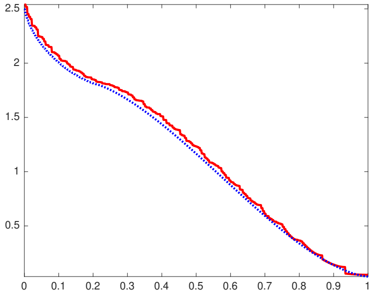


$N = 400$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```

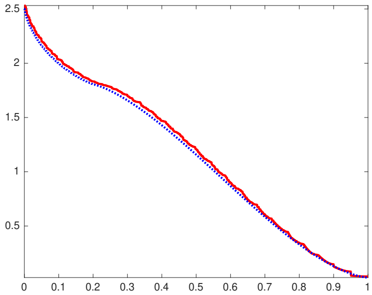


$N = 900$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```

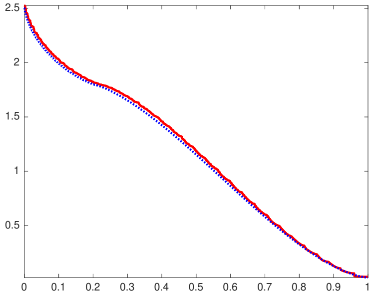


$N = 1600$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```

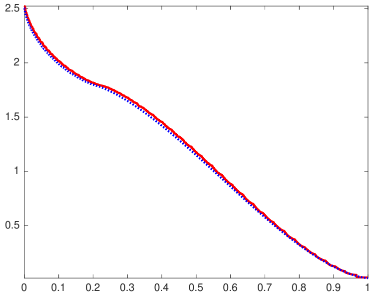


$N = 2500$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```

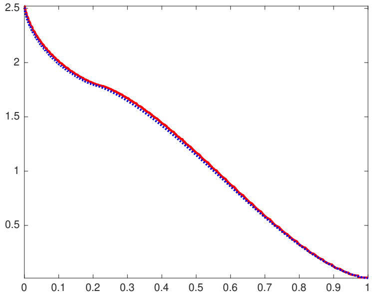


$N = 3600$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```

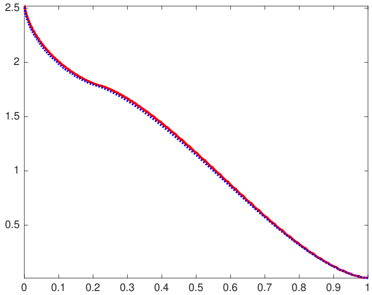


$N = 4900$

GLT stuff for the case at hand

```
alpha = 1.5; N = 100;
hN = 1/(N-1);  x = 0:hN:1; dt = hN;
dplus=@(x)gamma(3-alpha).*x.^alpha;
dminus=@(x)gamma(3-alpha).*(1-x).^alpha;
G = glmatrix(N,alpha); % Discretize
Gr = G; Grt = G.'; I = eye(N,N);
Dplus = diag(dplus(x));
Dminus = diag(dminus(x));
nu = hN^alpha/dt;
A = nu*I -(Dplus*Gr + Dminus*Grt);
f = @(theta) -exp(-1i*theta).*...
(1-exp(1i*theta)).^alpha;
xsq = linspace(0,1,sqrt(N));
tsq = linspace(-pi,pi,sqrt(N));
[X,THETA] = meshgrid(xsq,tsq);
```

```
h = @(x,theta) nu +
↪  (dplus(x).*f(theta) ...
+ dminus(x).*f(-theta));
sv = svd(full(A));
```



$N = 6400$

GLT: a negative result for circulant matrices

❓ And so we have the **asymptotic distribution** of our **singular values**, but what do we do with it?

GLT: a negative result for circulant matrices

❓ And so we have the **asymptotic distribution** of our **singular values**, but what do we do with it?

❗ The function

$$k(x,\theta) = d^{+}(x) f_{\alpha}(\theta) + d^{-}(x) f_{\alpha}(-\theta) \text{ for } (x,\theta) \in [0,1] \times [-\pi,\pi],$$

depends on **both** x and θ, on the other hand **any circulant preconditioner** will **depend only** on the θ **variable**!

GLT: a negative result for circulant matrices

❓ And so we have the **asymptotic distribution** of our **singular values**, but what do we do with it?

❗ The function

$$k(x,\theta) = d^{+}(x) f_{\alpha}(\theta) + d^{-}(x) f_{\alpha}(-\theta) \text{ for } (x,\theta) \in [0,1] \times [-\pi,\pi],$$

depends on **both** x and θ, on the other hand **any circulant preconditioner** will **depend only** on the θ **variable**!

$\Rightarrow$ No circulant preconditioner will ever cluster the singular values of a sequence with a "space variant" spectral distribution.

GLT: a negative result for circulant matrices

❓ And so we have the **asymptotic distribution** of our **singular values**, but what do we do with it?

❗ The function

$$k(x,\theta) = d^{+}(x) f_{\alpha}(\theta) + d^{-}(x) f_{\alpha}(-\theta) \text{ for } (x,\theta) \in [0,1] \times [-\pi,\pi],$$

depends on **both** x and θ, on the other hand **any circulant preconditioner** will **depend only** on the θ **variable**!

$\Rightarrow$ No circulant preconditioner will ever cluster the singular values of a sequence with a "space variant" spectral distribution.

❓ What type of preconditioner can we use to solve this issue?

Structure preserving preconditioners

The GLT class of sequences is a $*$-algebra, thus we can try to **proecondition** the sequence $\{A_N\}_N$ with **something from the same class**. We then look for:

Structure preserving preconditioners

The GLT class of sequences is a $*$-algebra, thus we can try to **proecondition** the sequence $\{A_N\}_N$ with **something from the same class**. We then look for:

- A sequence $\{P_N\}_N$ in the GLT class,

Structure preserving preconditioners

The GLT class of sequences is a $*$-algebra, thus we can try to **proecondition** the sequence $\{A_N\}_N$ with **something from the same class**. We then look for:

- A sequence $\{P_N\}_N$ in the GLT class,
- A sequence $\{P_N\}_N$ such that $\{P_N^{-1}A_N\}_N \sim_{\mathrm{GLT}} 1$,

Structure preserving preconditioners

The GLT class of sequences is a $*$-algebra, thus we can try to **proecondition** the sequence $\{A_N\}_N$ with **something from the same class**. We then look for:

- A sequence $\{P_N\}_N$ in the GLT class,
- A sequence $\{P_N\}_N$ such that $\{P_N^{-1}A_N\}_N \sim_{\mathrm{GLT}} 1$,
- A sequence $\{P_N\}_N$ that is *easy enough* to invert.

Structure preserving preconditioners

The GLT class of sequences is a $*$-algebra, thus we can try to **proecondition** the sequence $\{A_N\}_N$ with **something from the same class**. We then look for:

- A sequence $\{P_N\}_N$ in the GLT class,
- A sequence $\{P_N\}_N$ such that $\{P_N^{-1}A_N\}_N \sim_{\mathrm{GLT}} 1$,
- A sequence $\{P_N\}_N$ that is *easy enough* to invert.

An old idea anew

This a modification of an old idea, if we take a Toeplitz system $T_n(f)$ then we can use $T_n(1/f)$ as a preconditioner!

- $P_n^{-1} = T_n(1/f)$ **is not the inverse** of $T_n(f)$,
- If we have $T_n(1/f)$, its application cost is $O(n \log n)$,

Structure preserving preconditioners

The GLT class of sequences is a $*$-algebra, thus we can try to **proecondition** the sequence $\{A_N\}_N$ with **something from the same class**. We then look for:

- A sequence $\{P_N\}_N$ in the GLT class,
- A sequence $\{P_N\}_N$ such that $\{P_N^{-1}A_N\}_N \sim_{\mathrm{GLT}} 1$,
- A sequence $\{P_N\}_N$ that is *easy enough* to invert.

An old idea anew

This a modification of an old idea, if we take a Toeplitz system $T_n(f)$ then we can use $T_n(1/f)$ as a preconditioner!

- $P_n^{-1} = T_n(1/f)$ **is not the inverse** of $T_n(f)$,
- If we have $T_n(1/f)$, its application cost is $O(n \log n)$,
- Computing the Fourier coefficients of $1/f$ can be expensive.

Preconditioning Toeplitz with Toeplitz

We have expressed the Fourier coefficients of f as

$$t_k = \frac{1}{2\pi}\int_0^{2\pi} f(\theta)e^{-ik\theta}\,\mathrm{d}\theta, \qquad k = 0, \pm 1, \pm 2, \ldots,$$

we say that f is

- of **analytic type** if $t_k = 0$ for $k < 0$, or
- of **coanalytic type** if $t_k = 0$ for $k > 0$.

Lemma

Let f be of analytic type (or respectively coanalytic type) and $a_0 \neq 0$. Then $T_n(f)$ is invertible if and only if $1/f$ is bounded and of analytic type (or respectively coanalytic type). In either case, we have $T_n(1/f)T_n(f) = T_n(f)T_n(1/f) = I_n$, for I_n is the identity matrix.

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a **positive** trigonometric polynomial of degree K

$$f(\theta) = \sum_{k=-K}^{K} t_k e^{ik\theta}.$$

Then for $n > 2K$, $\operatorname{rank}(T_n(1/f)T_n(f) - I_n) \leq 2K$.

Proof. Let

$$\frac{1}{f(\theta)} = \sum_{k=-\infty}^{+\infty} \rho_k e^{ik\theta}$$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a **positive** trigonometric polynomial of degree K

$$f(\theta) = \sum_{k=-K}^{K} t_k e^{ik\theta}.$$

Then for $n > 2K$, $\mathrm{rank}(T_n(1/f)T_n(f) - I_n) \leq 2K$.

Proof. Let

$$\frac{1}{f(\theta)} = \sum_{k=-\infty}^{+\infty} \rho_k e^{ik\theta} \Rightarrow \sum_{k=-K}^{K} t_k \rho_{m-k} = \begin{cases} 1, & \text{if } m = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a **positive** trigonometric polynomial of degree K

$$f(\theta) = \sum_{k=-K}^{K} t_k e^{ik\theta}.$$

Then for $n > 2K$, $\mathrm{rank}(T_n(1/f)T_n(f) - I_n) \le 2K$.

Proof. Let

$$\frac{1}{f(\theta)} = \sum_{k=-\infty}^{+\infty} \rho_k e^{ik\theta} \Rightarrow \sum_{k=-K}^{K} t_k \rho_{m-k} = \begin{cases} 1, & \text{if } m = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Thus for $n > 2K$, the entries of $T_n(1/f)T_n(f) - I_n$ are all zeros except possibly entries in its first and last K columns. $\square$

Preconditioning Toeplitz with Toeplitz

Given $|\alpha| < 1$ consider

$$f(\theta) = \frac{1+\alpha^2 - \alpha e^{i\theta} - \alpha e^{-i\theta}}{1-\alpha^2}$$

$T_n(f)$ is tridiagonal and SPD.

```
function T = kacmatrix(n,alpha)
%KACMATRIX Kac-Murdock-Szego matrices
e = ones(n,1);
T = spdiags(([-alpha,1+alpha^2,-alpha]
↪   ./(1-alpha^2)).*e,-1:1,n,n);
end
```

We can express

$$\frac{1}{f(\theta)} = \sum_{k=-\infty}^{+\infty} t^{|k|} e^{ik\theta} = \frac{1-\alpha^2}{\left(1-\alpha e^{i\theta}\right)\left(1-\alpha e^{-i\theta}\right)},$$

and $T_n(1/f)$ is then a **dense Toeplitz matrix**.

Preconditioning Toeplitz with Toeplitz

We can compute the coefficients in an **inefficient way** and apply it to the CG/PCG

N	CG	PCG
32	20	2
64	20	2
128	20	2
256	20	2
512	20	2
1024	20	2
2048	20	2

$\alpha = 0.5$

```
function T = invkacmatrix(n,alpha)
%INVKACMATRIX Gives back the 1/Kac-Murdock-Szego
↪  matrices
f = @(th) (1 - alpha^2)./((1-alpha*exp(1i*th))
↪  .*(1-alpha*exp(-1i*th)));
c = zeros(n,1); r = zeros(1,n);
for k=1:n
 r(k) = integral(@(th) f(th).*exp(1i*th*(k-1)),0,2*pi)
↪  /(2*pi);
 c(k) = integral(@(th) f(th).*exp(-1i*th*(k-1)),0,2*pi)
↪  /(2*pi);
end
T = real(toeplitz(r,c));
end
```

Preconditioning Toeplitz with Toeplitz

We can compute the coefficients in an **inefficient way** and apply it to the CG/PCG

N	CG	PCG
32	6	2
64	6	2
128	6	2
256	6	2
512	6	2
1024	6	2
2048	6	2

$\alpha = 0.1$

```
function T = invkacmatrix(n,alpha)
%INVKACMATRIX Gives back the 1/Kac-Murdock-Szego
↪  matrices
f = @(th) (1 - alpha^2)./((1-alpha*exp(1i*th))
↪  .*(1-alpha*exp(-1i*th)));
c = zeros(n,1); r = zeros(1,n);
for k=1:n
 r(k) = integral(@(th) f(th).*exp(1i*th*(k-1)),0,2*pi)
↪  /(2*pi);
 c(k) = integral(@(th) f(th).*exp(-1i*th*(k-1)),0,2*pi)
↪  /(2*pi);
end
T = real(toeplitz(r,c));
end
```

Preconditioning Toeplitz with Toeplitz

We can compute the coefficients in an **inefficient way** and apply it to the CG/PCG

N	CG	PCG
32	20	3
64	20	2
128	20	2
256	20	2
512	20	2
1024	20	2
2048	20	2

$\alpha = 0.8$

```
function T = invkacmatrix(n,alpha)
%INVKACMATRIX Gives back the 1/Kac-Murdock-Szego
↪  matrices
f = @(th) (1 - alpha^2)./((1-alpha*exp(1i*th))
↪  .*(1-alpha*exp(-1i*th)));
c = zeros(n,1); r = zeros(1,n);
for k=1:n
 r(k) = integral(@(th) f(th).*exp(1i*th*(k-1)),0,2*pi)
↪  /(2*pi);
 c(k) = integral(@(th) f(th).*exp(-1i*th*(k-1)),0,2*pi)
↪  /(2*pi);
end
T = real(toeplitz(r,c));
end
```

Preconditioning Toeplitz with Toeplitz

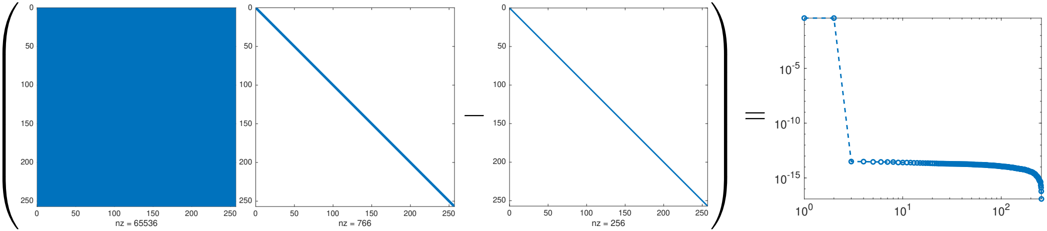
We can compute the coefficients in an **inefficient way** and apply it to the CG/PCG

$$\mathrm{rank}\left(T_n(1/f)\,T_n(f) - I_n\right) = 2$$

Preconditioning Toeplitz with Toeplitz

We can compute the coefficients in an **inefficient way** and apply it to the CG/PCG

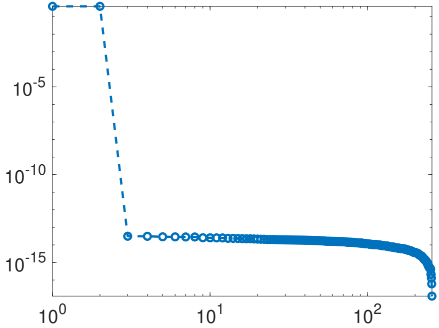
rank



Preconditioning Toeplitz with Toeplitz

We can compute the coefficients in an **inefficient way** and apply it to the CG/PCG

$$\mathrm{rank}\left(T_n(1/f)\,T_n(f) - I_n\right) =$$



An **exercise** to make the evaluation and construction of the involved quantities would be using the `fft` to compute the Fourier coefficients of $1/f(\theta)$.

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a positive 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exists positive integers M and N such that for all $n > N$,

$$T_n(1/f)\,T_n(f) = I_n + L_n + U_n, \text{where}\, \mathrm{rank}(L_n) \leq M \text{ and } \|U_n\|_2 < \varepsilon.$$

Proof. By the Weierstrass Theorem, there exists a positive trigonometric polynomial

$$p_K(\theta) = \sum_{k=-K}^{+K} \rho_k e^{ik\theta}, \quad \rho_{-k} = \overline{\rho}_k, \text{ such that } {}^{\mathrm{f_{min}}}/2 \leq p_K(\theta) \leq 2f_{\max}\; \forall\, \theta \in [0, 2\pi], \text{ and}$$

$$\max_{\theta\in[0,2\pi]} |f(\theta) - p_K(\theta)| \leq \frac{f_{\min}}{2}(-1 + \sqrt{1+\varepsilon})\min\left\{\frac{f_{\min}}{2f_{\max}}, 1\right\}.$$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a positive 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exists positive integers M and N such that for all $n > N$,

$$T_n(1/f)\,T_n(f) = I_n + L_n + U_n, \text{where}\,\mathrm{rank}(L_n) \leq M \text{ and } \|U_n\|_2 < \varepsilon.$$

Proof. We write

$$\begin{aligned} T_n(1/f)\,T_n(f) &= T_n(1/f)\,T_n^{-1}(1/p_K)\,T_n(1/p_K)\,T_n(p_K)\,T_n^{-1}(p_K)\,T_n(f) \\ &= (I_n + V_n)\,(T_n(1/p_K)\,T_n(p_K))\,(I_n + W_n) \end{aligned}$$

where $V_n = (T_n(1/f) - T_n(1/p_K)\,T_n^{-1}(1/p_K))$ and $W_n = T_n^{-1}(p_k)(T_n(f) - T_n(p_K))$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a positive 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exists positive integers M and N such that for all $n > N$,

$$T_n(1/f)T_n(f) = I_n + L_n + U_n, \text{where} \operatorname{rank}(L_n) \leq M \text{ and } \|U_n\|_2 < \varepsilon.$$

Proof. We write

$$T_n(1/f)T_n(f) = (I_n + V_n)\left(T_n(1/p_K)T_n(p_K)\right)(I_n + W_n)$$

and by the property of the generating functions and the Weierstrass Theorem

$$\|T_n^{-1}(p_K)\|_2 \leq \frac{2}{f_{\min}},\ \|T_n^{-1}(1/p_K)\|_2 \leq 2f_{\max},\ \|T_n(f) - T_n(p_K)\|_2 \leq \frac{(-1+\sqrt{1+\varepsilon})f_{\min}}{2},$$

$$\|T_n(1/f) - T_n(1/p_K)\|_2 \leq \max_{\theta in[0,2\pi]} \left|\frac{1}{f(\theta)} - \frac{1}{p_K(\theta)}\right| \leq \frac{2}{f_{\min}^2} \max_{\theta\in[0,2\pi]} |f(\theta) - p_K(\theta)|$$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a positive 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exists positive integers M and N such that for all $n > N$,

$$T_n(1/f)\,T_n(f) = I_n + L_n + U_n, \text{where}\,\mathrm{rank}(L_n) \leq M \text{ and } \|U_n\|_2 < \varepsilon.$$

Proof. We write

$$T_n(1/f)\,T_n(f) = (I_n + V_n)\,(T_n(1/p_K)\,T_n(p_K))\,(I_n + W_n)$$

and by the property of the generating functions and the Weierstrass Theorem

$$\|T_n^{-1}(p_K)\|_2 \leq \frac{2}{f_{\min}},\ \|T_n^{-1}(1/p_K)\|_2 \leq 2f_{\max},\ \|T_n(f) - T_n(p_K)\|_2 \leq \frac{(-1+\sqrt{1+\varepsilon})f_{\min}}{2},$$

$$\|T_n(1/f) - T_n(1/p_K)\|_2 \leq \frac{2}{f_{\min}^2} \max_{\theta\in[0,2\pi]} |f(\theta) - p_K(\theta)| \leq \frac{-1+\sqrt{1+\varepsilon}}{2f_{\max}}.$$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a positive 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exists positive integers M and N such that for all $n > N$,

$$T_n(1/f)\,T_n(f) = I_n + L_n + U_n, \text{where} \operatorname{rank}(L_n) \leq M \text{ and } \|U_n\|_2 < \varepsilon.$$

Proof. We write

$$T_n(1/f)\,T_n(f) = (I_n + V_n)\,(T_n(1/p_K)\,T_n(p_K))\,(I_n + W_n).$$

Using the **lemma on trigonometric polynomials** and using $n > 2K$ we have

$$T_n(1/p_K)\,T_n(p_K) = I_n + \tilde{L}_n \text{ with } \operatorname{rank}(\tilde{L}_n) \leq 2K.$$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a positive 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exists positive integers M and N such that for all $n > N$,

$$T_n(1/f)\, T_n(f) = I_n + L_n + U_n, \text{where } \operatorname{rank}(L_n) \leq M \text{ and } \|U_n\|_2 < \varepsilon.$$

Proof. We write

$$T_n(1/f)\, T_n(f) = (I_n + V_n)(I_n + \tilde{L}_n)(I_n + W_n).$$

Using the **lemma on trigonometric polynomials** and using $n > 2K$ we have

$$T_n(1/p_K)\, T_n(p_K) = I_n + \tilde{L}_n \text{ with } \operatorname{rank}(\tilde{L}_n) \leq 2K.$$

Preconditioning Toeplitz with Toeplitz

Lemma (Chan and Ng 1993)

Let f be a positive 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exists positive integers M and N such that for all $n > N$,

$$T_n(1/f)T_n(f) = I_n + L_n + U_n, \text{where} \operatorname{rank}(L_n) \leq M \text{ and } \|U_n\|_2 < \varepsilon.$$

Proof. We write

$$T_n(1/f)T_n(f) = (I_n + V_n)(I_n + \tilde{L}_n)(I_n + W_n) \equiv I_n + L_n + U_n,$$

where

$$U_n = V_n + W_n + V_n W_n, \quad L_n = \tilde{L}_n(I_n + W_n) + V_n\tilde{L}_n(I_n + W_n),$$

and using the previous relations

$$\operatorname{rank}(L_n) \leq 4K, \text{ and } \|U_n\|_2 \leq \varepsilon. \quad \square$$

Preconditioning Toeplitz with Toeplitz

Theorem (Chan and Ng 1993)

Let f be a **positive** 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exist positive integers M and N such that for all $n > N$, at most M eigenvalues of $T_n(1/f)\,T_n(f) - I_n$ have absolute value greater than ε.

Proof (idea). The HPD matrix $X_n = T_n^{1/2}(1/f)\,T_n(f)\,T_n^{1/2}(1/f) \sim T_n(1/f)\,T_n(f)$. Use the decomposition of the previous Theorem and the uniform boundedness of $T_n^{\pm 1/2}(1/f)$. $\square$

Preconditioning Toeplitz with Toeplitz

Theorem (Chan and Ng 1993)

Let f be a **positive** 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exist positive integers M and N such that for all $n > N$, at most M eigenvalues of $T_n(1/f)T_n(f) - I_n$ have absolute value greater than ε.

Proof (idea). The HPD matrix $X_n = T_n^{1/2}(1/f)T_n(f)T_n^{1/2}(1/f) \sim T_n(1/f)T_n(f)$. Use the decomposition of the previous Theorem and the uniform boundedness of $T_n^{\pm 1/2}(1/f)$. $\square$

- We still need **positive** generating functions,

Preconditioning Toeplitz with Toeplitz

Theorem (Chan and Ng 1993)

Let f be a **positive** 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exist positive integers M and N such that for all $n > N$, at most M eigenvalues of $T_n(1/f)T_n(f) - I_n$ have absolute value greater than ε.

Proof (idea). The HPD matrix $X_n = T_n^{1/2}(1/f)T_n(f)T_n^{1/2}(1/f) \sim T_n(1/f)T_n(f)$. Use the decomposition of the previous Theorem and the uniform boundedness of $T_n^{\pm 1/2}(1/f)$. $\square$

- We still need **positive** generating functions,
- If f is not given explicitly or the evaluation of $1/f(\theta)$ are costly the approach is infeasible.

Preconditioning Toeplitz with Toeplitz

Theorem (Chan and Ng 1993)

Let f be a **positive** 2π-periodic continuous function. Then for all $\varepsilon > 0$, there exist positive integers M and N such that for all $n > N$, at most M eigenvalues of $T_n(1/f)T_n(f) - I_n$ have absolute value greater than ε.

Proof (idea). The HPD matrix $X_n = T_n^{1/2}(1/f)T_n(f)T_n^{1/2}(1/f) \sim T_n(1/f)T_n(f)$. Use the decomposition of the previous Theorem and the uniform boundedness of $T_n^{\pm 1/2}(1/f)$. □

- We still need **positive** generating functions,
- If f is not given explicitly or the evaluation of $1/f(\theta)$ are costly the approach is infeasible.
- The **idea** from (Chan and Ng 1993) is to reduce the cost of working with f and $1/f$ by using convolution products with Kernel functions.

Preconditioning GLT with GLT

GLT sequences are a $*$-algebra, some of the analysis is therefore greatly simplified.

Theorem (Garoni and Serra-Capizzano 2017, Section 8.4)

Let $\{A_N\}_N$ be a sequence of Hermitian matrices such that $\{A_N\}_N \sim_{GLT} \kappa$, and let $\{P_N\}_N$ be a sequence of Hermitian positive definite matrices such that $\{P_N\}_N \sim_{GLT} \xi$ and $\xi \neq 0$ a.e. Then

$$\{P_N^{-1}A_N\}_N \sim_{\mathrm{GLT}} \xi^{-1}\kappa, \qquad \{P_N^{-1}A_N\}_N \sim_{\sigma,\lambda} (\xi^{-1}\kappa, \mathcal{I}^d).$$

Preconditioning GLT with GLT

GLT sequences are a $*$-algebra, some of the analysis is therefore greatly simplified.

Theorem (Garoni and Serra-Capizzano 2017, Section 8.4)

Let $\{A_N\}_N$ be a sequence of Hermitian matrices such that $\{A_N\}_N \sim_{GLT} \kappa$, and let $\{P_N\}_N$ be a sequence of Hermitian positive definite matrices such that $\{P_N\}_N \sim_{GLT} \xi$ and $\xi \neq 0$ a.e. Then

$$\{P_N^{-1} A_N\}_N \sim_{\text{GLT}} \xi^{-1}\kappa, \qquad \{P_N^{-1} A_N\}_N \sim_{\sigma,\lambda} (\xi^{-1}\kappa, \mathcal{I}^d).$$

- We need less than positive!

Preconditioning GLT with GLT

GLT sequences are a $*$-algebra, some of the analysis is therefore greatly simplified.

Theorem (Garoni and Serra-Capizzano 2017, Section 8.4)

Let $\{A_N\}_N$ be a sequence of Hermitian matrices such that $\{A_N\}_N \sim_{GLT} \kappa$, and let $\{P_N\}_N$ be a sequence of Hermitian positive definite matrices such that $\{P_N\}_N \sim_{GLT} \xi$ and $\xi \neq 0$ a.e. Then

$$\{P_N^{-1} A_N\}_N \sim_{\mathrm{GLT}} \xi^{-1}\kappa, \qquad \{P_N^{-1} A_N\}_N \sim_{\sigma,\lambda} (\xi^{-1}\kappa, \mathcal{I}^d).$$

- We need less than positive!
- If we move to the non-symmetric case, we are left just with a relation with respect to the singular values.

Preconditioning GLT with GLT

GLT sequences are a $*$-algebra, some of the analysis is therefore greatly simplified.

Theorem (Garoni and Serra-Capizzano 2017, Section 8.4)

Let $\{A_N\}_N$ be a sequence of Hermitian matrices such that $\{A_N\}_N \sim_{GLT} \kappa$, and let $\{P_N\}_N$ be a sequence of Hermitian positive definite matrices such that $\{P_N\}_N \sim_{GLT} \xi$ and $\xi \neq 0$ a.e. Then

$$\{P_N^{-1} A_N\}_N \sim_{\mathrm{GLT}} \xi^{-1}\kappa, \qquad \{P_N^{-1} A_N\}_N \sim_{\sigma,\lambda} (\xi^{-1}\kappa, \mathcal{I}^d).$$

- We need less than positive!
- If we move to the non-symmetric case, we are left just with a relation with respect to the singular values.
- The **general idea for a GLT preconditioner** is then to find a GLT sequence $\{P_N\}_N$
  - that is easy to invert,
  - and such that $\xi^1\kappa = 1$ or *at least* a quantity bounded and bounded away from zero.

Preconditioning GLT with GLT

Let us finally go back to our case of interest

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

we build a preconditioner with the **same structure** such that

- we have a *small bandwidth* $\Rightarrow$ a **small computational cost**,

Preconditioning GLT with GLT

Let us finally go back to our case of interest

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

we build a preconditioner with the **same structure** such that

- we have a *small bandwidth* $\Rightarrow$ a **small computational cost**,
- the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**.

Preconditioning GLT with GLT

Let us finally go back to our case of interest

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

we build a preconditioner with the **same structure** such that

- we have a *small bandwidth* $\Rightarrow$ a **small computational cost**,
- the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**.
- $P_{1,N} = \nu I + D_N^+ B_N + D_N^- B_N^T$, $B_n = T_n(1 - exp(-i\theta))$,

Preconditioning GLT with GLT

Let us finally go back to our case of interest

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

we build a preconditioner with the **same structure** such that

- we have a *small bandwidth* $\Rightarrow$ a **small computational cost**,
- the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**.
- $P_{1,N} = \nu I + D_N^+ B_N + D_N^- B_N^T,\ B_n = T_n(1 - exp(-i\theta)),$
- $P_{2,N} = \nu I + D_N^+ L_N + D_N^- L_N^T,\ B_n = T_n(2 - 2\cos(\theta))$

Preconditioning GLT with GLT

Let us finally go back to our case of interest

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

we build a preconditioner with the **same structure** such that

- we have a *small bandwidth* $\Rightarrow$ a **small computational cost**,
- the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**.
- $P_{1,N} = \nu I + D_N^+ B_N + D_N^- B_N^T$, $B_n = T_n(1 - exp(-i\theta))$,
- $\{P_{1,N}\}_N \sim_{GLT} p_1(x,\theta) = d_+(x)(1 - e^{-i\theta}) + d_-(x)(1 - e^{i\theta})$, holds only in the singular value sense!
- $P_{2,N} = \nu I + D_N^+ L_N + D_N^- L_N^T$, $B_n = T_n(2 - 2\cos(\theta))$

Preconditioning GLT with GLT

Let us finally go back to our case of interest

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$

we build a preconditioner with the **same structure** such that

- we have a *small bandwidth* $\Rightarrow$ a **small computational cost**,
- the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**.
- $P_{1,N} = \nu I + D_N^+ B_N + D_N^- B_N^T$, $B_n = T_n(1 - exp(-i\theta))$,
- $\{P_{1,N}\}_N \sim_{GLT} p_1(x, \theta) = d_+(x)(1 - e^{-i\theta}) + d_-(x)(1 - e^{i\theta})$, holds only in the singular value sense!
- $P_{2,N} = \nu I + D_N^+ L_N + D_N^- L_N^T$, $B_n = T_n(2 - 2\cos(\theta))$
- $\{P_{2,N}\}_N \sim_{GLT} p_2(x, \theta) = (d_+(x) + d_-(x))(2 - 2\cos(\theta))$, holds also in the eigenvalue sense!

Preconditioning GLT with GLT

Since the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**:

$$d_{\pm}(x,t) = d > 0 \, : \, \lim_{\theta \to 0} \frac{h(x,\theta)}{p_k(x,\theta)} = +\infty, \qquad k \in \{1,2\}.$$

Preconditioning GLT with GLT

Since the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**:

$$d_{\pm}(x,t) = d > 0 \, : \, \lim_{\theta \to 0} \frac{h(x,\theta)}{p_k(x,\theta)} = +\infty, \qquad k \in \{1,2\}.$$

Theorem (Serra 1995, Theorem 3.1)

Let f be an integrable function defined on $[-\pi,\pi]$ having in $x = x_0$ the unique zero of order ρ. Then, by choosing $2k$ the even number which minimizes the distance from ρ and setting $g = |x - x_0|^{2k}$, the condition number of $T_n(g)^{-1}T_n(f)$ is asymptotical to $n^{2k-\rho}$.

Preconditioning GLT with GLT

Since the symbol of a bandwidth Toeplitz matrix is a trigonometric polynomial, hence the **zero of the symbol cannot be of fractional order**:

$$d_{\pm}(x,t) = d > 0 \, : \, \lim_{\theta \to 0} \frac{h(x,\theta)}{p_k(x,\theta)} = +\infty, \qquad k \in \{1,2\}.$$

Theorem (Serra 1995, Theorem 3.1)

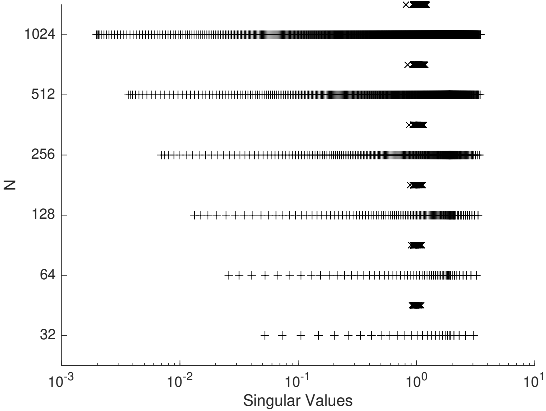
Let f be an integrable function defined on $[-\pi,\pi]$ having in $x = x_0$ the unique zero of order ρ. Then, by choosing $2k$ the even number which minimizes the distance from ρ and setting $g = |x - x_0|^{2k}$, the condition number of $T_n(g)^{-1} T_n(f)$ is asymptotical to $n^{2k-\rho}$.

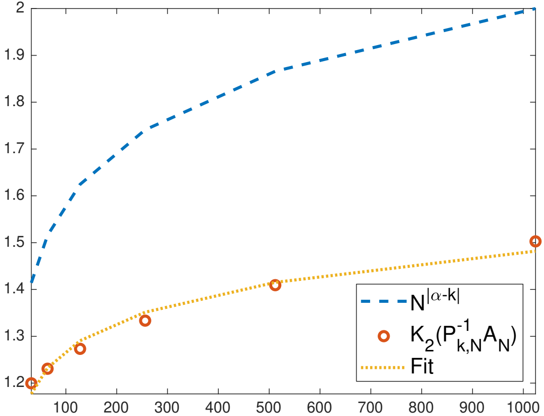
In our case

We expect the condition number of the preconditioned matrix to be $O(N^{|\alpha - k|})$, $k \in \{1,2\}$.

Preconditioning GLT with GLT

Let's numerically test our idea.

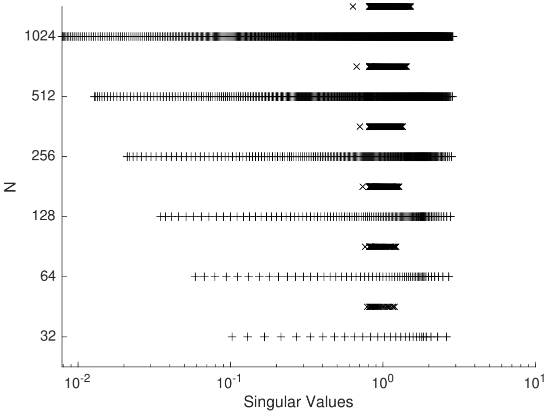




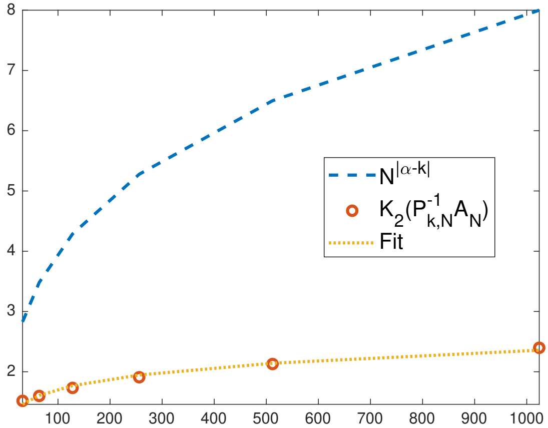
$\alpha = 1.9,\ k = 2$

Preconditioning GLT with GLT

Let's numerically test our idea.



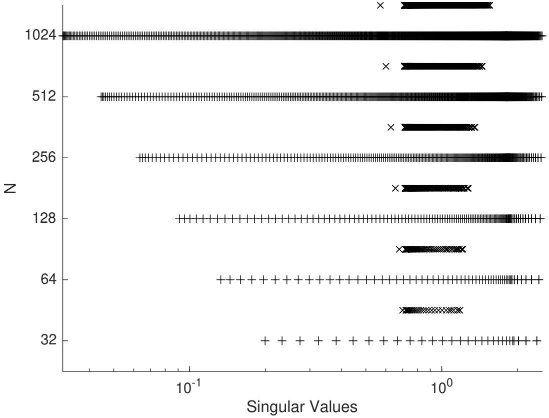
$\alpha = 1.7,\ k = 2$



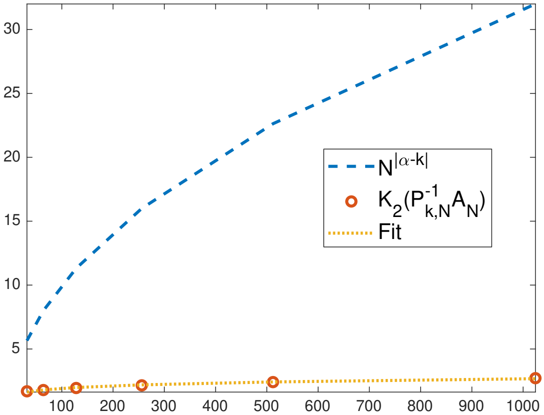
$\alpha = 1.7,\ k = 2$

Preconditioning GLT with GLT

Let's numerically test our idea.



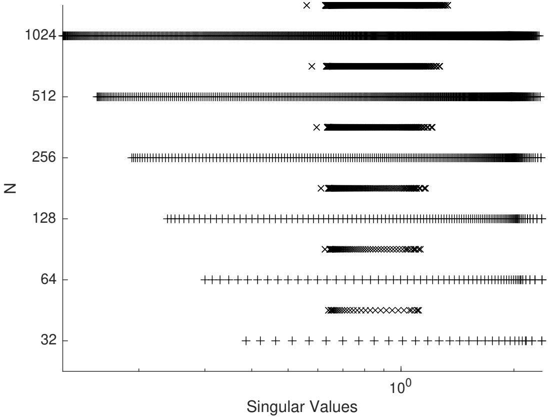
$\alpha = 1.5,\ k = 2$



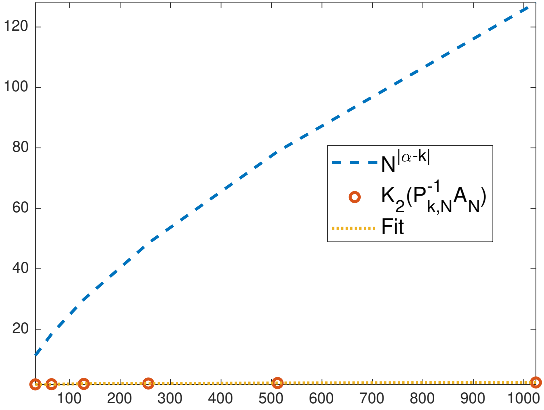
$\alpha = 1.5,\ k = 2$

Preconditioning GLT with GLT

Let's numerically test our idea.



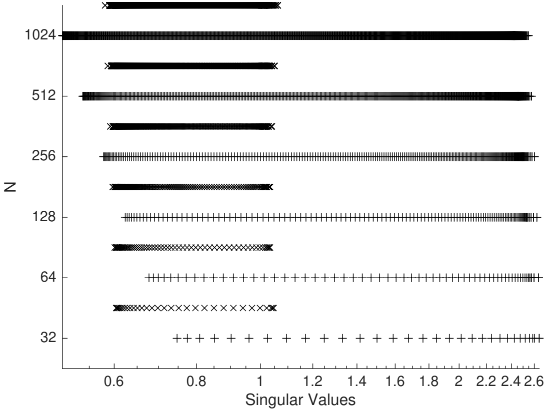
$\alpha = 1.3, \; k = 2$



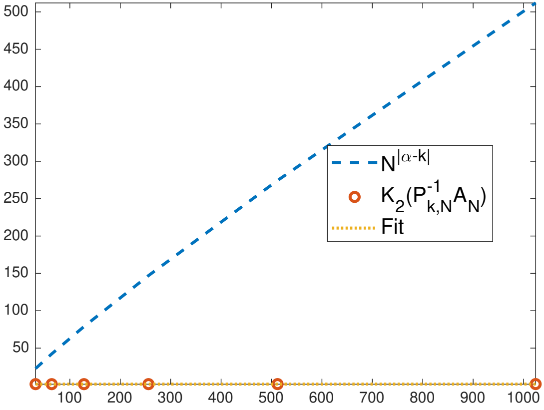
$\alpha = 1.3, \; k = 2$

Preconditioning GLT with GLT

Let's numerically test our idea.



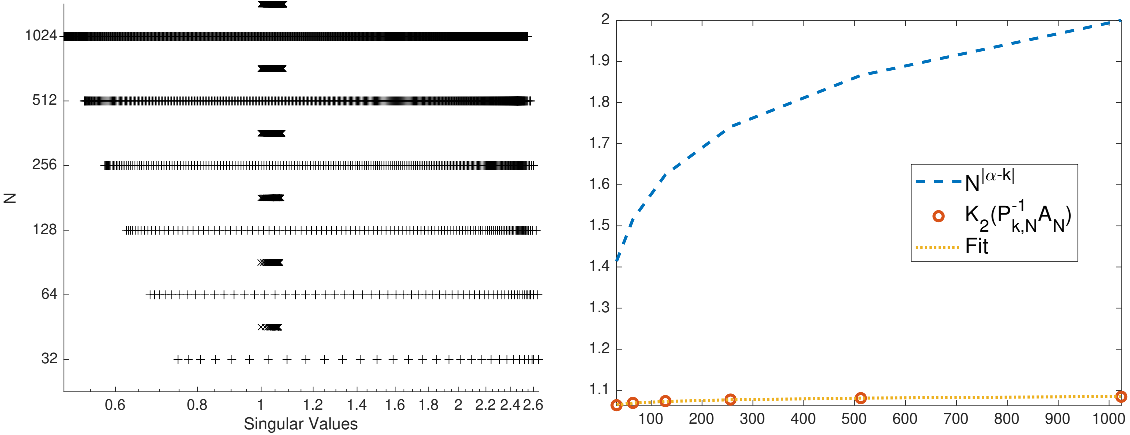
$\alpha = 1.1,\ k = 2$



$\alpha = 1.1,\ k = 2$

Preconditioning GLT with GLT

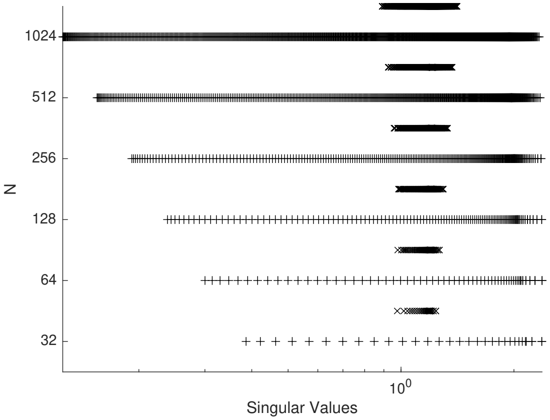
Let's numerically test our idea.



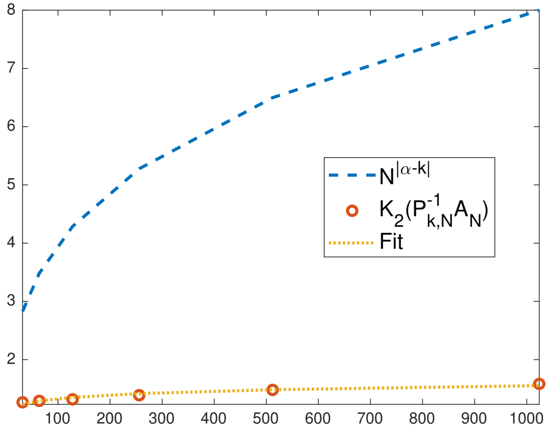
$\alpha = 1.1,\ k = 1$ $\qquad$ $\alpha = 1.1,\ k = 1$

Preconditioning GLT with GLT

Let's numerically test our idea.



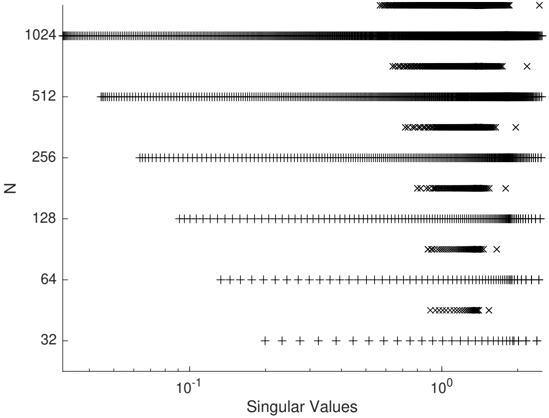
$\alpha = 1.3,\ k = 1$



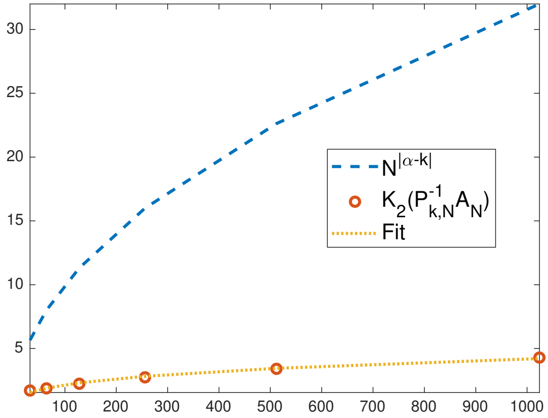
$\alpha = 1.3,\ k = 1$

Preconditioning GLT with GLT

Let's numerically test our idea.



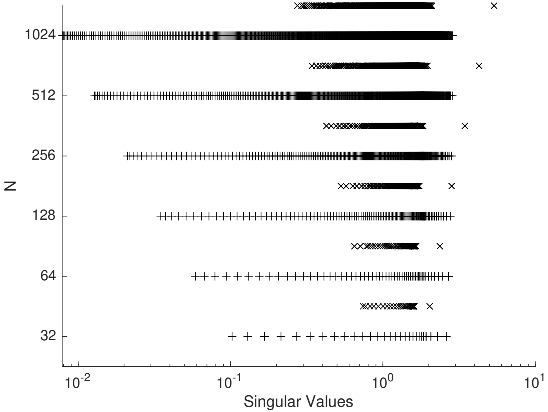
$\alpha = 1.5,\ k = 1$



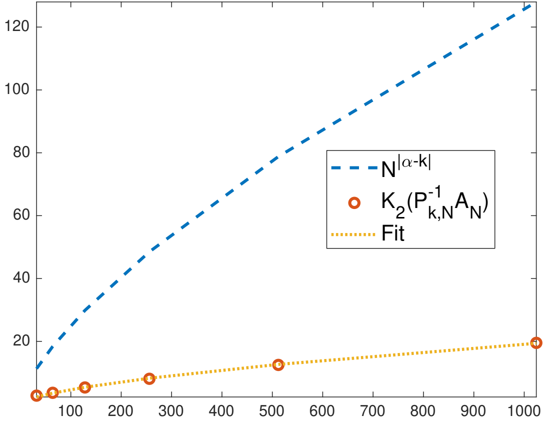
$\alpha = 1.5,\ k = 1$

Preconditioning GLT with GLT

Let's numerically test our idea.



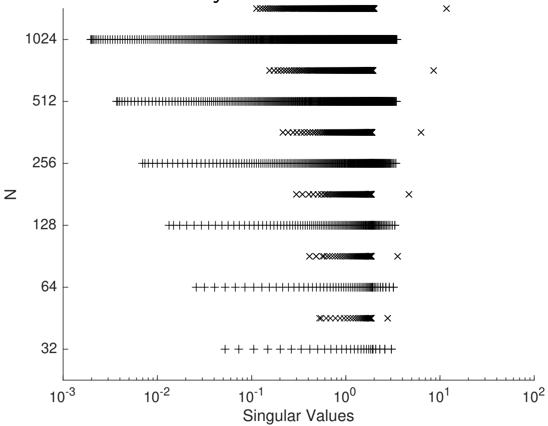
$\alpha = 1.7,\ k = 1$



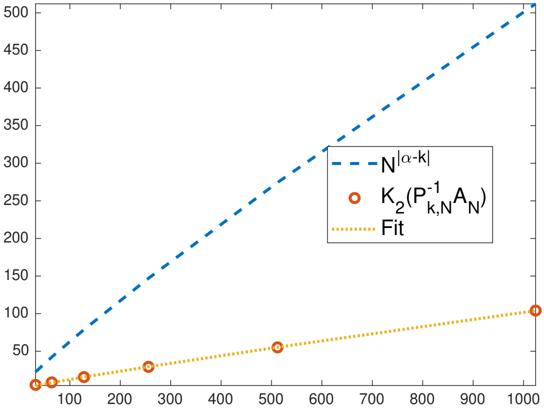
$\alpha = 1.7,\ k = 1$

Preconditioning GLT with GLT

Let's numerically test our idea.



$\alpha = 1.3,\ k = 1$



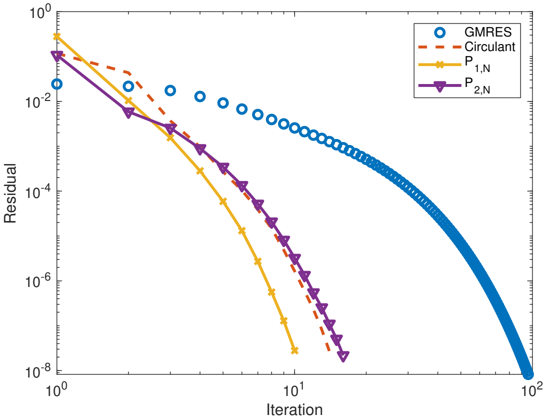
$\alpha = 1.3,\ k = 1$

Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.2	2^5	31	13	10	13
	2^6	50	14	11	15
	2^7	64	14	11	16
	2^8	75	15	11	16
	2^9	84	15	11	16
	2^{10}	91	14	10	16
	2^{11}	96	14	10	16

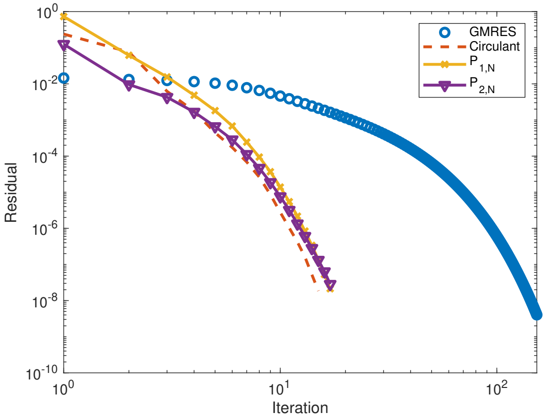


Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.3	2^5	31	13	13	14
	2^6	55	14	15	15
	2^7	79	15	16	16
	2^8	100	15	16	17
	2^9	119	15	16	17
	2^{10}	136	15	17	17
	2^{11}	153	15	17	17

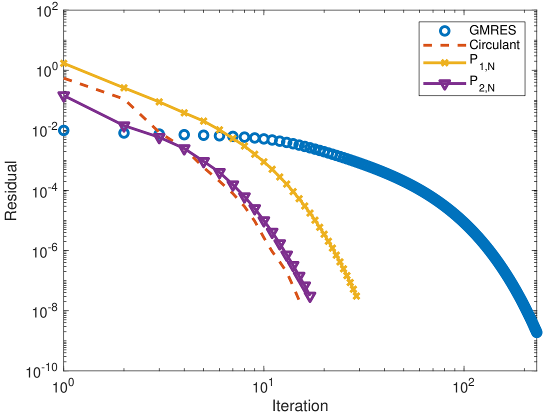


Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.4	2^5	31	13	16	13
	2^6	59	14	20	15
	2^7	92	15	23	16
	2^8	127	15	25	16
	2^9	161	15	26	17
	2^{10}	196	15	28	17
	2^{11}	231	15	29	17

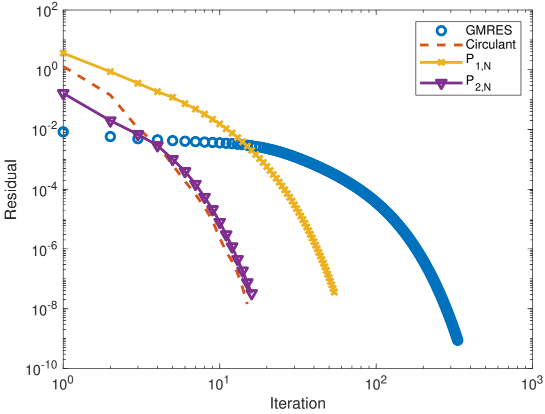


Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.5	2^5	32	13	19	12
	2^6	61	14	25	14
	2^7	104	15	32	15
	2^8	155	15	38	15
	2^9	209	15	43	16
	2^{10}	268	15	49	16
	2^{11}	332	15	54	16

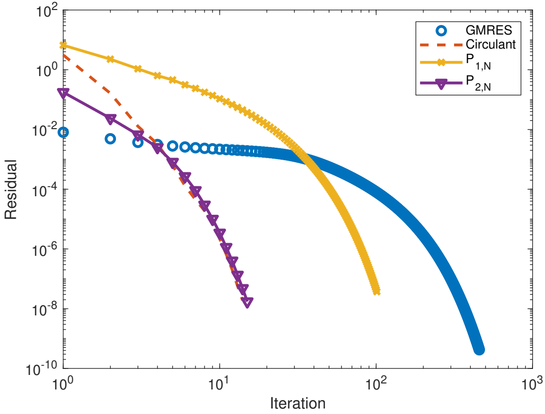


Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.6	2^5	32	13	22	11
	2^6	62	13	31	12
	2^7	112	14	42	13
	2^8	183	14	55	14
	2^9	262	14	69	14
	2^{10}	353	14	84	15
	2^{11}	456	14	101	15

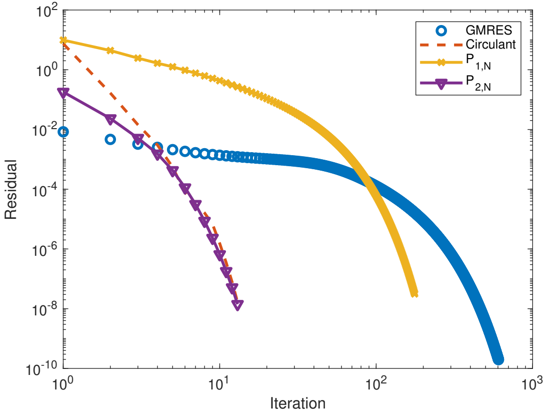


Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.7	2^5	32	12	25	10
	2^6	64	13	38	11
	2^7	118	13	55	12
	2^8	207	13	77	12
	2^9	319	13	104	12
	2^{10}	449	13	136	13
	2^{11}	605	13	176	13

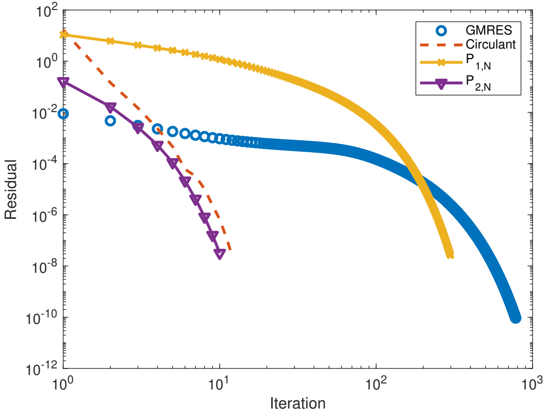


Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.8	2^5	32	12	27	9
	2^6	64	12	44	9
	2^7	126	13	71	10
	2^8	225	13	108	10
	2^9	378	13	157	10
	2^{10}	559	12	219	10
	2^{11}	779	12	298	10

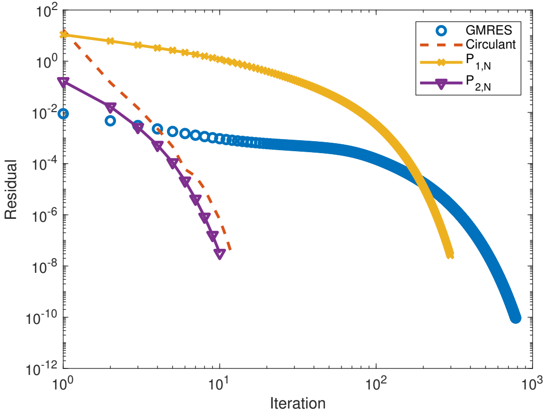


Preconditioning GLT with GLT

Test case is

$$d^{+}(x,t) = \Gamma(3-\alpha)x^{\alpha}, \qquad d^{-}(x,t) = \Gamma(3-\alpha)(2-x)^{\alpha}$$

α	N	GMRES	P	$P_{1,N}$	$P_{2,N}$
1.8	2^5	32	12	27	9
	2^6	64	12	44	9
	2^7	126	13	71	10
	2^8	225	13	108	10
	2^9	378	13	157	10
	2^{10}	559	12	219	10
	2^{11}	779	12	298	10



To **do better** we need to move towards Multigrid methods.

Circulant matrices at any cost

Despite the **clear negative results** concerning the impossibility of obtaining a cluster using circulant matrices in the space-dependent case, the *literature contains several attempts* in this direction.

Circulant matrices at any cost

Despite the **clear negative results** concerning the impossibility of obtaining a cluster using circulant matrices in the space-dependent case, the *literature contains several attempts* in this direction.

One of the most reused idea originates from (Pan et al. 2014), and goes as follows

1. We want to solve a "diagonal times Toeplitz" linear system, i.e.,

$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T\right),$$

Circulant matrices at any cost

Despite the **clear negative results** concerning the impossibility of obtaining a cluster using circulant matrices in the space-dependent case, the *literature contains several attempts* in this direction.

One of the most reused idea originates from (Pan et al. 2014), and goes as follows

1. We want to solve a "diagonal times Toeplitz" linear system, i.e.,
$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$
2. Call $d_i^+ = d^+(x_i)$ and $d_i^- = d(x_i)$, $i = 1, 2, \ldots, N$,

Circulant matrices at any cost

Despite the **clear negative results** concerning the impossibility of obtaining a cluster using circulant matrices in the space-dependent case, the *literature contains several attempts* in this direction.

One of the most reused idea originates from (Pan et al. 2014), and goes as follows

1. We want to solve a "diagonal times Toeplitz" linear system, i.e.,
$$A_N = \nu I_N - \left(D_N^{+} G_N + D_N^{-} G_N^T \right),$$
2. Call $d_i^{+} = d^{+}(x_i)$ and $d_i^{-} = d(x_i)$, $i = 1, 2, \ldots, N$,
3. Define the Toeplitz matrices
$$K_i = \nu I_n - \left(d_i^{+} G_N + d_i^{-} G_N^T \right), \qquad i = 1, 2, \ldots, N.$$

Circulant matrices at any cost

Despite the **clear negative results** concerning the impossibility of obtaining a cluster using circulant matrices in the space-dependent case, the *literature contains several attempts* in this direction.

One of the most reused idea originates from (Pan et al. 2014), and goes as follows

1. We want to solve a "diagonal times Toeplitz" linear system, i.e.,
$$A_N = \nu I_N - \left(D_N^+ G_N + D_N^- G_N^T \right),$$
2. Call $d_i^+ = d^+(x_i)$ and $d_i^- = d(x_i)$, $i = 1, 2, \ldots, N$,
3. Define the Toeplitz matrices
$$K_i = \nu I_n - \left(d_i^+ G_N + d_i^- G_N^T \right), \qquad i = 1, 2, \ldots, N.$$
4. Since $\mathbf{e}_i^T A_N = e_i^T K_i$, approximate
$$\mathbf{e}_i^T A^{-1} \approx \mathbf{e}_i^T K_i^{-1}.$$

Circulant matrices at any cost

But how do we approximate the inversion?

Circulant matrices at any cost

But how do we approximate the inversion?

- Build $P_1 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T K_i^{-1}$

Circulant matrices at any cost

But how do we approximate the inversion?

- Build $P_1 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T K_i^{-1}$
  - it costs too much! N Toeplitz solve per iteration.

Circulant matrices at any cost

But how do we approximate the inversion?

- Build $P_1 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T K_i^{-1}$
  - it costs too much! N Toeplitz solve per iteration.
- Build $P_2 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T C_i^{-1}$ with $C_i = s(K_i)$ (Strang preconditioner)

Circulant matrices at any cost

But how do we approximate the inversion?

- Build $P_1 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T K_i^{-1}$
  - it costs too much! N Toeplitz solve per iteration.
- Build $P_2 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T C_i^{-1}$ with $C_i = s(K_i)$ (Strang preconditioner)
  - it still cost too much! $O(N^2 \log(N))$ per iteration.

Circulant matrices at any cost

But how do we approximate the inversion?

- Build $P_1 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T K_i^{-1}$
  - it costs too much! N Toeplitz solve per iteration.
- Build $P_2 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T C_i^{-1}$ with $C_i = s(K_i)$ (Strang preconditioner)
  - it still cost too much! $O(N^2 \log(N))$ per iteration.
- Build $P_3 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T \sum_{j=1}^{\ell} \phi_j(x_i) C_j^{-1}$
  - where for $\ell \ll N$ values $\{x_{i_j}\}_{j=1}^{\ell} \subset \{x_i\}_{i=1}^{N}$ $\phi_j(x)$ are the basis of the piecewise linear interpolation of

$$q_\lambda(x) = \frac{1}{\nu + \lambda d^+(x) + \bar{\lambda} d^-(x)}, \qquad \lambda \in \mathbb{C}.$$

Circulant matrices at any cost

But how do we approximate the inversion?

- Build $P_1 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T K_i^{-1}$
  - it costs too much! N Toeplitz solve per iteration.
- Build $P_2 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T C_i^{-1}$ with $C_i = s(K_i)$ (Strang preconditioner)
  - it still cost too much! $O(N^2 \log(N))$ per iteration.
- Build $P_3 = \sum_{i=1}^{N} \mathbf{e}_i \mathbf{e}_i^T \sum_{j=1}^{\ell} \phi_j(x_i) C_j^{-1}$ The cost is now $O(\ell N \log N)$ operations.
  - where for $\ell \ll N$ values $\{x_{i_j}\}_{j=1}^{\ell} \subset \{x_i\}_{i=1}^{N}$ $\phi_j(x)$ are the basis of the piecewise linear interpolation of

$$q_\lambda(x) = \frac{1}{\nu + \lambda d^+(x) + \bar{\lambda} d^-(x)}, \qquad \lambda \in \mathbb{C}.$$

Circulant matrices at any cost

The analysis of the 😐 P_3 preconditioner is quite involved, furthermore

- the iteration number dependence on the selection of the interpolation nodes and the value of λ is unclear,
- the **resulting preconditioner** is **always a circulant matrix**, thus the general theory tells us that there is no hope of getting a cluster of any sort.

Circulant matrices at any cost

The analysis of the 😐 P_3 preconditioner is quite involved, furthermore

- the iteration number dependence on the selection of the interpolation nodes and the value of λ is unclear,
- the **resulting preconditioner** is **always a circulant matrix**, thus the general theory tells us that there is no hope of getting a cluster of any sort.

⚠ The extension of this preconditioners to the multi-dimensional settings is even more challenging: interpolation of surfaces, and higher dimensional objects is a tough problem!

Circulant matrices at any cost

The analysis of the 😐 P_3 preconditioner is quite involved, furthermore

- the iteration number dependence on the selection of the interpolation nodes and the value of λ is unclear,
- the **resulting preconditioner** is **always a circulant matrix**, thus the general theory tells us that there is no hope of getting a cluster of any sort.

⚠ The extension of this preconditioners to the multi-dimensional settings is even more challenging: interpolation of surfaces, and higher dimensional objects is a tough problem!

✗ For these reasons we will not pursue further these results, if you are interested start from (Pan et al. 2014), and look to the next episodes.

Multidimensional cases

What happens if our equation becomes

$$\begin{cases} \frac{\partial W}{\partial t} = \left(\theta\, {}^{RL}D^{\alpha}_{[0,x]} \cdot + (1-\theta)\, {}^{RL}D^{\alpha}_{[x,1]} \cdot\right) W(x,y,t) + & \theta \in [0,1], \\ \qquad \left(\theta\, {}^{RL}D^{\alpha}_{[0,y]} \cdot + (1-\theta)\, {}^{RL}D^{\alpha}_{[y,1]} \cdot\right) W(x,y,t) & \\ W(0,t) = W(1,t) = 0, \qquad W(x,t) = W_0(x). & \end{cases}$$

- If we repeat the discretization procedure we have used in the 1D case we end up with a **block-Toeplitz-with-Toeplitz-blocks** matrix,
- then we could attempt solution by using a block-circulant-with-circulant-blocks preconditioner! In the 1D case (either symmetric or not) the procedure was working, maybe we are lucky...

Multidimensional cases

What happens if our equation becomes

$$\begin{cases} \frac{\partial W}{\partial t} = \left(d_x^+(x,t)\, {}^{RL}D^{\alpha}_{[0,x]} \cdot + 1 - \theta) d_x^-(x,t) {}^{RL}D^{\alpha}_{[x,1]} \cdot \right) W(x,y,t) +, \\ \qquad\qquad \left(d_y^+(x,y,t)\, {}^{RL}D^{\alpha}_{[0,y]} \cdot + 1 - \theta) d_y^-(x,y,t) {}^{RL}D^{\alpha}_{[y,1]} \cdot \right) W(x,y,t) \\ W(0,t) = W(1,t) = 0, \qquad W(x,t) = W_0(x). \end{cases}$$

- It should not be difficult to imagine, but in this case we should end up again with a **matrix sequence of GLT type**,
- we can attempt the solution by doing something similar to what we have done in the 1D case: using a Toeplitz preconditioner...

A negative result

In the constant coefficient case we have a **general negative result**:

> *"Any Circulant-Like Preconditioner for Multilevel Matrices Is Not Superlinear" – Serra Capizzano and Tyrtyshnikov 1999*

Theorem (Serra Capizzano and Tyrtyshnikov 1999, Theorem 4.1)

For $I_n + A_n$, $A_n = A_n(f)$ a p-level Toeplitz matrix, any preconditioner for the form $I_n + C_n$, where p_n is a p-level circulant matrix, is not superlinear.

A negative result

In the constant coefficient case we have a **general negative result**:

> *"Any Circulant-Like Preconditioner for Multilevel Matrices Is Not Superlinear" – Serra Capizzano and Tyrtyshnikov 1999*

Theorem (Serra Capizzano and Tyrtyshnikov 1999, Theorem 4.1)

For $I_n + A_n$, $A_n = A_n(f)$ a p-level Toeplitz matrix, any preconditioner for the form $I_n + C_n$, where p_n is a p-level circulant matrix, is not superlinear.

- The **number of iterations** for the preconditioned system **will always depend on the size of the system**!

A negative result

In the constant coefficient case we have a **general negative result**:

> *"Any Circulant-Like Preconditioner for Multilevel Matrices Is Not Superlinear" – Serra Capizzano and Tyrtyshnikov 1999*

Theorem (Serra Capizzano and Tyrtyshnikov 1999, Theorem 4.1)

For $I_n + A_n$, $A_n = A_n(f)$ a p-level Toeplitz matrix, any preconditioner for the form $I_n + C_n$, where p_n is a p-level circulant matrix, is not superlinear.

- The **number of iterations** for the preconditioned system **will always depend on the size of the system**!
- The dependence can still by milder than the one of the original system, thus there are cases in which this could be worthwhile (at least for a while).

A negative result

In the constant coefficient case we have a **general negative result**:

> *"Any Circulant-Like Preconditioner for Multilevel Matrices Is Not Superlinear" – Serra Capizzano and Tyrtyshnikov 1999*

Theorem (Serra Capizzano and Tyrtyshnikov 1999, Theorem 4.1)

For $I_n + A_n$, $A_n = A_n(f)$ a p-level Toeplitz matrix, any preconditioner for the form $I_n + C_n$, where p_n is a p-level circulant matrix, is not superlinear.

- The **number of iterations** for the preconditioned system **will always depend on the size of the system**!
- The dependence can still by milder than the one of the original system, thus there are cases in which this could be worthwhile (at least for a while).

It is a difficult world

Already the case with constant coefficient is difficult to treat. Maybe we can find a way to *reduce the number of dimensions*.

Another negative result and a proposal

- The result we have obtained by means of GLT theory for the variable coefficient case remains valid also in two dimensions: *no circulant preconditioner can have a strong cluster*!

Another negative result and a proposal

- The result we have obtained by means of GLT theory for the variable coefficient case remains valid also in two dimensions: *no circulant preconditioner can have a strong cluster*!
- We could attempt generalizing the $P_{1,N}$ and $P_{2,N}$ preconditioners to the new setting.

Another negative result and a proposal

- The result we have obtained by means of GLT theory for the variable coefficient case remains valid also in two dimensions: *no circulant preconditioner can have a strong cluster*!
- We could attempt generalizing the $P_{1,N}$ and $P_{2,N}$ preconditioners to the new setting.
- The matrix of the system in 2D has now the form

$$A_{\mathbf{N}} = \nu I_{\mathbf{N}} - \left(D_{\mathbf{N}}^{+}(G_{N_x} \otimes I_{N_y}) + D_{\mathbf{N}}^{-}(I_{N_x} \otimes G_{N_y}) \right), \qquad \mathbf{N} = (N_x, N_y).$$

  - If the **diffusion coefficients** are **constants**, this a BTTB matrix,
  - If the **diffusion coefficients** are **space variant**, we can show (following the same road as before) that the resulting matrix sequence is a GLT sequence.

Another negative result and a proposal

- The result we have obtained by means of GLT theory for the variable coefficient case remains valid also in two dimensions: *no circulant preconditioner can have a strong cluster*!
- We could attempt generalizing the $P_{1,N}$ and $P_{2,N}$ preconditioners to the new setting.
- The matrix of the system in 2D has now the form

$$A_{\mathbf{N}} = \nu I_{\mathbf{N}} - \left(D_{\mathbf{N}}^{+}(G_{N_x} \otimes I_{N_y}) + D_{\mathbf{N}}^{-}(I_{N_x} \otimes G_{N_y})\right), \qquad \mathbf{N} = (N_x, N_y).$$

  - If the **diffusion coefficients** are **constants**, this a BTTB matrix,
  - If the **diffusion coefficients** are **space variant**, we can show (following the same road as before) that the resulting matrix sequence is a GLT sequence.

- $P_{1,\mathbf{N}} = \nu I_{\mathbf{N}} - \left(D_{\mathbf{N}}^{+}(T_{N_x}(1-e^{-i\theta_1}) \otimes I_{N_y}) + D_{\mathbf{N}}^{-}(I_{N_x} \otimes T_{N_y}(1-e^{-i\theta_2}))\right)$;
- $P_{2,\mathbf{N}} = \nu I_{\mathbf{N}} - \left(D_{\mathbf{N}}^{+}(T_{N_x}(2-2\cos(\theta_1)) \otimes I_{N_y}) + D_{\mathbf{N}}^{-}(I_{N_x} \otimes T_{N_y}(2-2\cos(\theta_2)))\right)$.

The structure preserving preconditioners

To apply both $P_{1,\mathbf{N}}$ and $P_{2,\mathbf{N}}$ we now need to **solve an auxiliary sparse linear system** related to the **discretization of a 2D problem**.

Using a sparse direct solver is not going to scale well with $\mathbf{N} = (N_x, N_y)$,

The structure preserving preconditioners

To apply both $P_{1,\mathbf{N}}$ and $P_{2,\mathbf{N}}$ we now need to **solve an auxiliary sparse linear system** related to the **discretization of a 2D problem**.

- Using a sparse direct solver is not going to scale well with $\mathbf{N} = (N_x, N_y)$,
- We need to **employ an iterative technique** to do the **preconditioner application**!

The structure preserving preconditioners

To apply both $P_{1,\mathbf{N}}$ and $P_{2,\mathbf{N}}$ we now need to **solve an auxiliary sparse linear system** related to the **discretization of a 2D problem**.

- Using a sparse direct solver is not going to scale well with $\mathbf{N} = (N_x, N_y)$,
- We need to **employ an iterative technique** to do the **preconditioner application**!
- Methods of this type are usually called **multi-iterative methods**
  - $\Rightarrow$ If we apply $P_{1,\mathbf{N}}$ or $P_{2,\mathbf{N}}$ using a fixed number of iterations of a fixed point technique, then we can still use GMRES,
  - $\Rightarrow$ If we apply $P_{1,\mathbf{N}}$ or $P_{2,\mathbf{N}}$ using a variable number of iterations of a fixed point technique or a *nonstationary solver*, then we have to use the Flexible-GMRES.

The structure preserving preconditioners

To apply both $P_{1,\mathbf{N}}$ and $P_{2,\mathbf{N}}$ we now need to **solve an auxiliary sparse linear system** related to the **discretization of a 2D problem**.

- Using a sparse direct solver is not going to scale well with $\mathbf{N} = (N_x, N_y)$,
- We need to **employ an iterative technique** to do the **preconditioner application**!
- Methods of this type are usually called **multi-iterative methods**
  - $\Rightarrow$ If we apply $P_{1,\mathbf{N}}$ or $P_{2,\mathbf{N}}$ using a fixed number of iterations of a fixed point technique, then we can still use GMRES,
  - $\Rightarrow$ If we apply $P_{1,\mathbf{N}}$ or $P_{2,\mathbf{N}}$ using a variable number of iterations of a fixed point technique or a *nonstationary solver*, then we have to use the Flexible-GMRES.

What is the right combination?

The right combination of iterative schemes to use does really depend on the machine we have under our hands!

Flexible-GMRES (Saad 1993)

The **Flexible variant of GMRES** is built from the *right-preconditioned* GMRES algorithm.

Input: $A \in \mathbb{R}^{n \times n}$, m, $\mathbf{x}^{(0)}$, $M \in \mathbb{R}^{n \times n}$

1 $\mathbf{r}^{(0)} \leftarrow b - A\mathbf{x}^{(0)}$; `/* Arnoldi process */`
2 $\beta \leftarrow \|\mathbf{r}^{(0)}\|_2$, $\mathbf{v}^{(1)} \leftarrow \mathbf{r}^{(0)}/\beta$;
3 **for** $j = 1, \dots, m$ **do**
4     $\mathbf{z}^{(j)} \leftarrow P^{-1}\mathbf{v}^{(j)}$;
5     $\mathbf{w} \leftarrow A\mathbf{z}^{(j)}$;
6   **for** $i = 1, \dots, j$ **do**
7         $h_{i,j} \leftarrow < \mathbf{w}, \mathbf{v}^{(i)} >$;
8         $\mathbf{w} \leftarrow \mathbf{w} - h_{i,j}\mathbf{v}^{(i)}$;
9   **end**
10     $h_{j+1,j} \leftarrow \|\mathbf{w}\|_2$;
11     $\mathbf{v}^{(j+1)} \leftarrow \mathbf{w}/h_{j+1,j}$;
12 **end**
13 $V_m \leftarrow [\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(m)}]$;
`// Build the Krylov subspace basis`
14 $\mathbf{y}^{(m)} \leftarrow \arg\min_{\mathbf{y}} \|\beta \mathbf{e}_1 - \overline{H}_m \mathbf{y}\|_2$;
15 $\mathbf{x}^{(m)} \leftarrow \mathbf{x}^{(0)} + P^{-1} V_m \mathbf{y}^{(m)}$;
`// Conv. check, possibly a restart`
16 **if** *Stopping criteria satisfied* **then**
17   **Return:** $\tilde{\mathbf{x}} = \mathbf{x}^{(m)}$;
18 **else**
19     $\mathbf{x}^{(0)} \leftarrow \mathbf{x}^{(m)}$; `/* Restart */`
20   **goto** 1;
21 **end**

Same preconditioner

Line 15 forms the approximate solution of the linear system as $\mathbf{x}^{(0)} + P^{-1} V_m \mathbf{y}^{(m)}$.

Flexible-GMRES (Saad 1993)

The **Flexible variant of GMRES** is built from the *right-preconditioned* GMRES algorithm.

Input: $A \in \mathbb{R}^{n\times n}$, m, $\mathbf{x}^{(0)}$, $M \in \mathbb{R}^{n\times n}$

1 $\mathbf{r}^{(0)} \leftarrow b - A\mathbf{x}^{(0)}$; `/* Arnoldi process */`

2 $\beta \leftarrow \|\mathbf{r}^{(0)}\|_2$, $\mathbf{v}^{(1)} \leftarrow \mathbf{r}^{(0)}/\beta$;

3 **for** $j = 1, \dots, m$ **do**

4     $\mathbf{z}^{(j)} \leftarrow P^{-1}\mathbf{v}^{(j)}$;

5     $\mathbf{w} \leftarrow A\mathbf{z}^{(j)}$;

6   **for** $i = 1, \dots, j$ **do**

7         $h_{i,j} \leftarrow < \mathbf{w}, \mathbf{v}^{(i)} >$;

8         $\mathbf{w} \leftarrow \mathbf{w} - h_{i,j}\mathbf{v}^{(i)}$;

9   **end**

10     $h_{j+1,j} \leftarrow \|\mathbf{w}\|_2$;

11     $\mathbf{v}^{(j+1)} \leftarrow \mathbf{w}/h_{j+1,j}$;

12 **end**

13 $Z_m \leftarrow [\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}]$;

`// Build the Krylov subspace basis`

14 $\mathbf{y}^{(m)} \leftarrow \arg\min_{\mathbf{y}} \|\beta\mathbf{e}_1 - \overline{H}_m\mathbf{y}\|_2$;

15 $\mathbf{x}^{(m)} \leftarrow \mathbf{x}^{(0)} + Z_m\mathbf{y}^{(m)}$;

`// Conv. check, possibly a restart`

16 **if** *Stopping criteria satisfied* **then**

17   **Return:** $\tilde{\mathbf{x}} = \mathbf{x}^{(m)}$;

18 **else**

19     $\mathbf{x}^{(0)} \leftarrow \mathbf{x}^{(m)}$; `/* Restart */`

20   **goto** 1;

21 **end**

Changing preconditioner

Line 15 forms the approximate solution of the linear system as $\mathbf{x}^{(0)} + Z_m\mathbf{y}^{(m)}$.

Flexible-GMRES (Saad 1993)

With this variant of the GMRES we are solving

$$AP^{-1}\mathbf{y} = \mathbf{b}, \text{ with } P\mathbf{x} = \mathbf{y},$$

with a preconditioner P whose action depends on the vector to which it is applied,

- in **terms of memory** we have to store two basis instead of one,
- we use the **true residual** instead of the preconditioned one: *the results are more reliable!*

Flexible-GMRES (Saad 1993)

With this variant of the GMRES we are solving

$$AP^{-1}\mathbf{y} = \mathbf{b}, \text{ with } P\mathbf{x} = \mathbf{y},$$

with a preconditioner P whose action depends on the vector to which it is applied,

- in **terms of memory** we have to store two basis instead of one,
- we use the **true residual** instead of the preconditioned one: *the results are more reliable!*

Some **usual choices** of multi-iterative schemes are

- Inner/Outer GMRES method: we fix a preconditioner P, solve the systems

  $$\mathbf{z}^{(j)} \leftarrow P^{-1}\mathbf{v}^{(j)},$$

  by a recursive call to GMRES;
- A Multigrid algorithm in which some *smoother* or *coarse solver* is non stationary;
- Non stationary polynomial preconditioners.

Exploiting the Kronecker structure

The **multidimensional case** has a **new structure** we can exploit: **Kronecker sums**!

$$A_{\mathbf{N}} = \nu I_{\mathbf{N}} - \left(D^{+}_{\mathbf{N}} (G_{N_x} \otimes I_{N_y}) + D^{-}_{\mathbf{N}} (I_{N_x} \otimes G_{N_y}) \right) , \qquad \mathbf{N} = (N_x , N_y).$$

Exploiting the Kronecker structure

The **multidimensional case** has a **new structure** we can exploit: **Kronecker sums**!

$$A_{\mathbf{N}} = \nu I_{N_x} \otimes I_{N_y} - \left(D^{+}_{\mathbf{N}} (G_{N_x} \otimes I_{N_y}) + D^{-}_{\mathbf{N}} (I_{N_x} \otimes G_{N_y}) \right), \qquad \mathbf{N} = (N_x, N_y).$$

Exploiting the Kronecker structure

The **multidimensional case** has a **new structure** we can exploit: **Kronecker sums**!

$$A_{\mathbf{N}} = \nu I_{N_x} \otimes I_{N_y} - \left(D_{\mathbf{N}}^{+}(G_{N_x} \otimes I_{N_y}) + D_{\mathbf{N}}^{-}(I_{N_x} \otimes G_{N_y})\right), \qquad \mathbf{N} = (N_x, N_y).$$

If we assume **separable coefficients**, i.e.,

$$d^{+}(x,y) = d_1^{+}(x) d_2^{+}(y), \quad d^{-}(x,y) = d_1^{-}(x) d_2^{-}(y).$$

Exploiting the Kronecker structure

The **multidimensional case** has a **new structure** we can exploit: **Kronecker sums**!

$$A_{\mathbf{N}} = \nu I_{N_x} \otimes I_{N_y} - \left((D^{+}_{\mathbf{1},\mathbf{N_x}} \otimes D^{+}_{2,N_y})(G_{N_x} \otimes I_{N_y}) + (D^{-}_{\mathbf{1},\mathbf{N_x}} \otimes D^{-}_{2,N_y})(I_{N_x} \otimes G_{N_y}) \right)$$

- If we assume **separable coefficients**, i.e.,

$$d^{+}(x,y) = d^{+}_{1}(x) d^{+}_{2}(y), \quad d^{-}(x,y) = d^{-}_{1}(x) d^{-}_{2}(y).$$

- We write the solution vector $\mathbf{x}$ as a matrix X such that $\mathbf{x} = \mathrm{vec}(X)$, where $\mathrm{vec}(\cdot)$ is the operation that stacks the columns of X, and the right-hand side $\mathbf{b}$ as B with $\mathbf{b} = \mathrm{vec}(B)$.

Exploiting the Kronecker structure

The **multidimensional case** has a **new structure** we can exploit: **Kronecker sums**!

$$\text{Find } X \text{ s.t. } \nu X - D^{+}_{2,N_y} X G^{T}_{N_x} D^{+}_{1,N_x} - D^{-}_{2,N_y} G_{N_y} X D^{-}_{1,N_x} = B$$

- If we assume **separable coefficients**, i.e.,

$$d^{+}(x,y) = d^{+}_{1}(x) d^{+}_{2}(y), \quad d^{-}(x,y) = d^{-}_{1}(x) d^{-}_{2}(y).$$

- We write the solution vector $\mathbf{x}$ as a matrix X such that $\mathbf{x} = \mathrm{vec}(X)$, where $\mathrm{vec}(\cdot)$ is the operation that stacks the columns of X, and the right-hand side $\mathbf{b}$ as B with $\mathbf{b} = \mathrm{vec}(B)$.
- We got ourselves a **matrix equation** involving objects of "smaller size".

Conclusion and summary

- We have characterized the **spectral properties** of the involved matrix sequences,
- We investigated several preconditioning strategies that made use of the **structure** of the **underlying matrices**,
- We started investigating multi-iterative schemes and looking for ways of reducing the dimensionality of the involved problems.

Next up

- How and when do we solve the **matrix equation** formulation,
- What do we do when we have more than two dimensions?
- All-at-once formulations.

Bibliography I

Chan, R. H. and K.-P. Ng (1993). “Toeplitz preconditioners for Hermitian Toeplitz systems”. In: *Linear Algebra Appl.* 190, pp. 181–208. ISSN: 0024-3795. DOI: 10.1016/0024-3795(93)90226-E. URL: https://doi.org/10.1016/0024-3795(93)90226-E.

Donatelli, M., M. Mazza, and S. Serra-Capizzano (2016). “Spectral analysis and structure preserving preconditioners for fractional diffusion equations”. In: *J. Comput. Phys.* 307, pp. 262–279. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2015.11.061. URL: https://doi.org/10.1016/j.jcp.2015.11.061.

Garoni, C. and S. Serra-Capizzano (2017). *Generalized locally Toeplitz sequences: theory and applications. Vol. I*. Springer, Cham, pp. xi+312. ISBN: 978-3-319-53678-1; 978-3-319-53679-8. DOI: 10.1007/978-3-319-53679-8. URL: https://doi.org/10.1007/978-3-319-53679-8.

— (2018). *Generalized locally Toeplitz sequences: theory and applications. Vol. II*. Springer, Cham, pp. xi+194. ISBN: 978-3-030-02232-7; 978-3-030-02233-4. DOI: 10.1007/978-3-030-02233-4. URL: https://doi.org/10.1007/978-3-030-02233-4.

Bibliography II

- Okoudjou, K. A., L. G. Rogers, and R. S. Strichartz (2010). "Szegö limit theorems on the Sierpiński gasket". In: *J. Fourier Anal. Appl.* 16.3, pp. 434–447. ISSN: 1069-5869. DOI: 10.1007/s00041-009-9102-0. URL: https://doi.org/10.1007/s00041-009-9102-0.
- Pan, J. et al. (2014). "Preconditioning techniques for diagonal-times-Toeplitz matrices in fractional diffusion equations". In: *SIAM J. Sci. Comput.* 36.6, A2698–A2719. ISSN: 1064-8275. DOI: 10.1137/130931795. URL: https://doi.org/10.1137/130931795.
- Saad, Y. (1993). "A flexible inner-outer preconditioned GMRES algorithm". In: *SIAM J. Sci. Comput.* 14.2, pp. 461–469. ISSN: 1064-8275. DOI: 10.1137/0914028. URL: https://doi.org/10.1137/0914028.
- Serra, S. (1995). "New PCG based algorithms for the solution of Hermitian Toeplitz systems". In: *Calcolo* 32.3-4, 153–176 (1997). ISSN: 0008-0624. DOI: 10.1007/BF02575833. URL: https://doi.org/10.1007/BF02575833.

Bibliography III

Serra Capizzano, S. and E. Tyrtyshnikov (1999). "Any circulant-like preconditioner for multilevel matrices is not superlinear". In: *SIAM J. Matrix Anal. Appl.* 21.2, pp. 431–439. ISSN: 0895-4798. DOI: 10.1137/S0895479897331941. URL: https://doi.org/10.1137/S0895479897331941.

Tilli, P. (1998). "Locally Toeplitz sequences: spectral properties and applications". In: *Linear Algebra Appl.* 278.1-3, pp. 91–120. ISSN: 0024-3795. DOI: 10.1016/S0024-3795(97)10079-9. URL: https://doi.org/10.1016/S0024-3795(97)10079-9.