Compressive Fourier collocation methods for high-dimensional diffusion equations with periodic boundary conditions

Weiqi Wang (Speaker), Simone Brugiapaglia

Concordia University

CRM - Applied Mathematics Seminars
February 13th, 2022
Outline

- Introduction: High-dimensional diffusion equation
- Algorithm: Compressive Fourier collocation methods
- Convergence theorems
- Numerical results
- Current research: PINN and adaptive method
- Future research

Check out our work on: https://arxiv.org/abs/2206.01255
High dimensional PDEs

- Black-Scholes equation (mathematical finance)
- Hamilton-Jacobi-Bellman equation in high dimensional optimal control
- Fokker-Planck equation
- Many-electron Schrödinger equation in computational chemistry

More high-dimensional PDEs examples, see http://deeppde.org/intro/.
Curse of dimensionality

Dimensionally cursed phenomena occur in domains such as numerical analysis, sampling, combinatorics, machine learning, data mining and databases. The common theme of these problems is that when the dimensionality increases, the volume of the space increases so fast that the available data become sparse. In order to obtain a reliable result, the amount of data needed often grows exponentially with the dimensionality. (from Wikipedia)

We focus on reducing the curse of dimensionality when solving high-dimensional PDEs.
Model problem: High-dimensional diffusion equation

\[
\begin{cases}
-\nabla \cdot (a \nabla u) = F & \text{in } \Omega = [0, 1]^d \\
u \text{ periodic} & \text{on } \partial \Omega,
\end{cases}
\]

where \( a(x) > a_{\text{min}} > 0 \) defined in \( \Omega \), \( u = u(x) \), \( x = (x_1, x_2, \ldots, x_d) \), \( d \) is the number of dimensions, \( F \) is the forcing term.

We add an average constraint to guarantee the uniqueness of the solution.

\[
\int_{\Omega} u \, dx = 0
\]

For convenience, we define the operator \( \mathcal{L}[u] = -\nabla \cdot (a \nabla u) \)
Spectral collocation method I

Step 1: Assume the solution is in the form of

$$u(x) = \sum_{\nu \in \Lambda} c_{\nu} \psi_{\nu}(x),$$

where $\nu = (\nu_1, \nu_2, \ldots, \nu_d)$ is the index, $\Lambda$ is the set of the indices, $c_{\nu}$ are unknown coefficients, and $\psi_{\nu}(x) = \prod_{i=1}^{d} \psi_{\nu_i}(x_i)$ are Fourier basis functions in high dimension. Define the cardinality of the index set $N = |\Lambda|$. How to choose $\Lambda$?

For example, if $d = 3$, $\nu = (1, 2, 3)$, then

$$\psi_{\nu}(x) = \exp^{2\pi x_1 i} \exp^{4\pi x_2 i} \exp^{6\pi x_3 i}$$

Step 2: Choose collocation points $x_i \in \Omega$, $i = 1, \ldots, m$, using Monte Carlo sampling.
Spectral collocation method II

**Step 3:** Set the residues at collocation points equal to 0.

\[
\begin{bmatrix}
L[\psi_{v_1}](x_1) & L[\psi_{v_2}](x_1) & \ldots & L[\psi_{v_N}](x_1) \\
L[\psi_{v_1}](x_2) & L[\psi_{v_2}](x_2) & \ldots & L[\psi_{v_N}](x_2) \\
\vdots & \vdots & \ddots & \vdots \\
L[\psi_{v_1}](x_m) & L[\psi_{v_2}](x_m) & \ldots & L[\psi_{v_N}](x_m)
\end{bmatrix}
\begin{bmatrix}
c_{v_1} \\
c_{v_2} \\
\vdots \\
c_{v_N}
\end{bmatrix}
= 
\begin{bmatrix}
F(x_1) \\
F(x_2) \\
\vdots \\
F(x_m)
\end{bmatrix}
\]

and solve the linear system.

**Step 4:** Recover the solution \( u(x) \) from the coefficients \( c_v \).

\[
u(x) = \sum_{\nu \in \Lambda} c_{\nu} \psi_{\nu}(x).
\]
Compressive Sensing in numerical methods

- Compressive spectral collocation method [Brugiapaglia, 2020]
- Petrov-Galerkin discretizations [Jokar et al., 2010]
- Sublinear-Time Algorithms, sparse Fourier transform [Gross et al., 2022]
- Sparse spectral method for homogenization multiscale problems. [Daubechies et al., 2007]

Other Compressive Sensing resources

- A Mathematical Introduction to Compressive [Foucart and Rauhut, 2013]
- Sparse Polynomial Approximation of High-Dimensional Functions [Adcock et al., 2022]
Compressive Sensing in spectral collocation method

\[ u(x) = \sum_{k=1}^{N} c_{\nu_k} \psi_{\nu_k}(x) \]

If \( \mathbf{c} \) is a compressible vector, we can approximate \( u(x) \) using the follow \( s \)-sparse \( u_s \):

\[ u_s(x) = \sum_{\nu \in \Lambda_s} c_{\nu} \psi_{\nu}(x), \]

where \( \Lambda_s \) is an index set with cardinality \( s \), i.e. \( |\Lambda_s| = s \)

\[ \| \text{Error} \|_2 = \| u_{\text{exact}}(x) - u_s(x) \|_2 \]

\[ = \left\| \sum_{\nu \notin \Lambda_s} c_{\nu} \psi_{\nu}(x) \right\|_2 = \sqrt{\sum_{\nu \notin \Lambda_s} |c_{\nu}|^2} \]

Fast decay of the coefficients when the exact solution is smooth enough, e.g. mixed Sobolev regularity.

An example of \( |c_{\nu}| \) versus \( \nu \) in 1D, \( u(x) = \exp(\sin 2\pi x) \)
Compressive Sensing

Consider collocation points as sample points in compressive sensing. We can find \( s \)-sparse solution using sparse recovery methods.

\[
\begin{bmatrix}
L[\psi_{v_1}](x_1) & L[\psi_{v_2}](x_1) & L[\psi_{v_3}](x_1) & \ldots & L[\psi_{v_N}](x_1) \\
L[\psi_{v_1}](x_2) & L[\psi_{v_2}](x_2) & L[\psi_{v_3}](x_2) & \ldots & L[\psi_{v_N}](x_2) \\
L[\psi_{v_1}](x_3) & L[\psi_{v_2}](x_3) & L[\psi_{v_3}](x_3) & \ldots & L[\psi_{v_N}](x_3) \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
L[\psi_{v_1}](x_m) & L[\psi_{v_2}](x_m) & L[\psi_{v_3}](x_m) & \ldots & L[\psi_{v_N}](x_m)
\end{bmatrix}
\begin{bmatrix}
c_{v_1} \\
c_{v_2} \\
c_{v_3} \\
\vdots \\
c_{v_N}
\end{bmatrix}
= \begin{bmatrix}
F(x_1) \\
F(x_2) \\
F(x_3) \\
\vdots \\
F(x_m)
\end{bmatrix}
\]
Index set $\Lambda$

We use a hyperbolic cross of $\mathbb{Z}^d$ as index set $\Lambda$ i.e.

$$\Lambda = \left\{ \nu \in \mathbb{Z}^d : \prod_{i=1}^{d} (|\nu_i| + 1) \leq n \right\} \setminus \{0\}$$

Cardinality bounds [Chernov and Đũng, 2016]:

$$N = |\Lambda| \leq \min \left\{ 4n^5 16^d, e^{2n^2 + \log_2(d)} \right\} \ll n^d$$
Sparse recovery strategies

---

**Algorithm** Orthogonal Matching Pursuit (OMP)

**Input:** \( A \in \mathbb{C}^{m \times N}, \ b \in \mathbb{C}^{m}, \ K \in \mathbb{N} \);

**Output:** A \( K \)-sparse vector \( \hat{c} \);

\[
d_j \leftarrow \frac{1}{\sqrt{\sum_{i=1}^{m} |A_{ij}|^2}}, \text{ Normalize columns of } A, \text{ i.e. } A_{ij} \leftarrow A_{ij}d_j;
\]

**for** \( n = 0, \ldots, K - 1 \) **do**

\[
j_{n+1} \leftarrow \arg \max_{j \in [N]} \{ |(A^*(b - Az_n))_j| \};
\]

\[
S_{n+1} \leftarrow S_n \cup \{j_{n+1}\};
\]

\[
z_{n+1} \leftarrow \arg \min_{z \in \mathbb{C}^N} \{ \|b - Az\|_2, \text{ supp}(z) \subseteq S_{n+1} \};
\]

**end for**

\[
D \leftarrow \text{diag}(d), \ \hat{c} \leftarrow Dz_K;
\]

Alternatively, we solve the following QCBP (Quadratically-Constrained Basis Pursuit) problem using convex optimization.

\[
\hat{c} \in \arg \min_{z \in \mathbb{C}^N} \|z\|_1 \text{ such that } \|Az - b\|_2 \leq \eta,
\]
Compressive Fourier collocation methods summary

**Algorithm** Compressive Fourier collocation method

**Input:** \(a(x), F(x), d, n, m\)

**Output:** Solution \(\hat{u}(x)\)

Generate hyperbolic cross truncation set \(\Lambda = \{\nu_1, \ldots, \nu_m\}\);
Randomly generate \(m\) independent points \(x_1, \ldots, x_m\) in \(T^d\);

**for** \(i \leftarrow 1\) **to** \(m\) **do**

**for** \(j \leftarrow 1\) **to** \(N = |\Lambda|\) **do**

\[A_{ij} \leftarrow \frac{1}{\sqrt{m}} [-\nabla \cdot (a \nabla \Psi_{\nu_j})](x_i),\] where \(\Psi_{\nu} = \frac{1}{4\pi^2 \|\nu\|^2} \psi_{\nu};\)

**end for**

\[b_i \leftarrow \frac{1}{\sqrt{m}} f(x_i);\]

**end for**

Find an \(s\)-sparse approximate \(\hat{c} \approx \arg \min_{z \in \mathbb{C}^N, \|z\|_0 \leq s} \|Az - b\|_2^2\) via OMP;

\[\hat{u}(x) \leftarrow \sum_{\nu \in \Lambda} \hat{c}_\nu \Psi_{\nu}(x);\]
Riesz system and Riesz constants

**Definition (Riesz system)**

Let \((H, \langle \cdot, \cdot \rangle_H)\) denote a complex Hilbert space. A sequence \(\Phi = \{\Psi_v\}_{v \in \mathbb{N}}\) with \(\Psi_v \in H\) is called a Riesz system if there exist constants \(0 < b_\Psi < B_\Psi < \infty\) such that for every \(f = (f_n)_{n \in \mathbb{N}} \in \ell^2(\mathbb{C})\),

\[
    b_\Psi \|f\|_{\ell^2(\mathbb{C})}^2 \leq \left\| \sum_{j \in \mathbb{N}} f_j \Psi_j \right\|_H^2 \leq B_\Psi \|f\|_{\ell^2(\mathbb{C})}^2
\]

and define the bound of the system \(K_\Psi = \max_v \|\Psi_j\|_{L^\infty}\)

For example \(\Phi = \{\sin(2\pi j x)\}, \ j = 0, 1, \ldots n\) on \([-1, 1]\) is an Riesz system with \(b_\Phi = B_\Phi = 1\).

The idea of using Riesz system to prove convergence is from [Brugiapaglia et al., 2021].
**Constants in convergence theorems**

Consider a diffusion coefficient $a$ of the form

$$a = a_t + a^*, \quad \text{where} \quad a_t = e_0 + \sum_{\nu \in T} e_\nu \psi_\nu, \quad |T| \leq t,$$

assume that

$$\beta := \sqrt{t \left( \|a_t\|_{H^1}^2 - |e_0|^2 \right)} < (\sqrt{2} - 1)|e_0|,$$

$$\gamma := \frac{\sqrt{N}}{2\pi} \left| a^* \right|_{H^1} + \|a^*\|_{L^2} \leq \sqrt{|e_0|^2 - 2|e_0|\beta - \beta^2},$$

Then, $\Phi = \{\mathcal{L}(\Psi_\nu)\}_{\nu \in \Lambda}$ is a Riesz system with Riesz constants:

$$b_\Phi = \left( \sqrt{|e_0|^2 - 2|e_0|\beta - \beta^2 - \gamma} \right)^2 > 0,$$

$$B_\Phi = \left( \sqrt{\|a_t\|_{H^1}^2 + 2|e_0|\beta + \beta^2 + \gamma} \right)^2,$$

$$K_\Phi = |e_0| + \beta + \|a^*\|_{L^\infty} + \sum_{l=1}^{d} \left\| \frac{\partial a^*}{\partial x_l} \right\|_{L^\infty}.$$
Convergence theorem I

**Theorem (Accurate and stable recovery)**

There exist universal constants $c, C_1, C_2 > 0$ such that approximations obtained via compressive Fourier collocation and computed via OMP or QCBP satisfy the following recovery guarantees. Assume that the Riesz constants $b_\Phi$ and $B_\Phi$ satisfy the sufficient condition

$$
\frac{b_\Phi}{B_\Phi} > \begin{cases} 
1 - \frac{0.98}{13} & \approx 0.9246, \\
1 - \frac{0.98}{\sqrt{2}} & \approx 0.3070.
\end{cases} \quad (\text{OMP})
$$

$$
\frac{b_\Phi}{B_\Phi} > \begin{cases} 
1 - \frac{0.98}{13} & \approx 0.9246, \\
1 - \frac{0.98}{\sqrt{2}} & \approx 0.3070.
\end{cases} \quad (\text{QCBP})
$$

Let $s \in \mathbb{N}$ and assume that the number of collocation points satisfies

$$
m \geq c \max\{B_\Phi^{-2}, 1\} K_\Phi^2 s L,
$$

where

$$
L = \log^2 (sK_\Phi^2 \max\{B_\Phi^{-2}, 1\}) \min\{\log(n) + d, \log(2n) \log(2d)\} + \log(2/\epsilon),
$$
**Convergence theorem II**

**Theorem (Accurate and stable recovery, continued)**

Let \( \hat{c} \in \mathbb{C}^N \) be either the OMP solution computed via \( K = 24s \) iterations or any QCBP solution. Then, the corresponding compressive Fourier collocation approximation \( \hat{u} \) to \( u \) satisfies the following error bounds for all \( f \in L^2(\mathbb{T}^d) \) with probability at least \( 1 - \epsilon \):

\[
\| \Delta (u - \hat{u}) \|_{L^2} \leq \| \Delta (u - u_\Lambda) \|_{L^2} + C_1 \frac{\sigma_s(c_\Lambda)}{\sqrt{s}} + C_2 \| L [u - u_\Lambda] \|_{L^\infty},
\]

\[
\| u - \hat{u} \|_{L^2} \leq \| u - u_\Lambda \|_{L^2} + \frac{1}{4\pi^2} \left( C_1 \frac{\sigma_s(c_\Lambda)}{\sqrt{s}} + C_2 \| L [u - u_\Lambda] \|_{L^\infty} \right),
\]

where

\[
\sigma_s(c)_1 = \inf_{z \in \mathbb{C}^N, \|z\|_0 \leq s} \| c - z \|_1
\]
Diffusion coefficients and Exact solutions

**Diffusion coefficients.** In all experiments, we consider three types of diffusion coefficient $\in \mathbb{T}^d = [0, 1]^d$:

$$a_1(x) = 1,$$
$$a_2(x) = 1 + 0.25 \sin (2\pi x_1) \sin (2\pi x_2) + 0.25 \sin (4\pi x_1),$$
$$a_3(x) = 1 + 0.2 \exp (\sin (2\pi x_1) \sin (2\pi x_2)).$$

**Exact solutions.** In the two-dimensional case, we test the results for sparse and non-sparse exact solution

$$u_1(x) = \sum_{k=1}^{10} d_k \sin (2\pi m_k x_1) \sin (2\pi n_k x_2), \quad \forall x \in \mathbb{T}^d,$$
$$u_2(x) = \exp (\sin (2\pi x_1) + \sin (2\pi x_2)) - c, \quad \forall x \in \mathbb{T}^d.$$
Exact solutions in 2D

Figure: The sparse and non-sparse exact solutions $u_1$ and $u_2$. 
Non-sparse exact solution $u_2$, $d = 2$

Figure: ($d = 2$, non-sparse solution) Relative $L^2$-error versus number of collocation points $m$. The exact solution is $u_2$. 
Sparse exact solution $u_1$, $d = 8$

**Figure:** ($d = 8$, sparse solution) Relative $L^2$-error versus number of collocation points $m$. The exact solution is $u_1$. 
Non-sparse exact solution $u_2$, $d = 8$

Figure: ($d = 8$, non-sparse solution) Relative $L^2$-error versus number of collocation points $m$. The exact solution is $u_2$. 
Impact of the index set $\Lambda$, $d = 8$

Figure: ($d = 8$, impact of $\Lambda$) Relative $L^2$-error versus number of collocation points $m$. The exact solution, $u_2$, and the diffusion coefficient, $a_3$, are non-sparse. $\Lambda$ are hyperbolic cross set with $n = 7, 11, 15$
Impact of the dimensionality

Figure: (Impact of the dimensionality) Relative $L^2$-error versus number of collocation points $m$. The exact solution, $u_2$, and the diffusion coefficient, $a_3$, are non-sparse. The multi-index set $\Lambda$ is the largest hyperbolic cross set such that $|\Lambda| < 2600$. 
Current and future research

- Compare Compressive Fourier collocation method with other methods (e.g., deep neural network) in solving high-dimensional PDE. (with Nick Dexter)
- Adaptive methods.
- Push the dimension higher. Further improve the performance.
- Time-dependent problems.
- Solve different types of PDEs.
- Non-homogeneous boundary condition. Use polynomial basis instead Fourier basis for non-periodic problem.
Settings (PINN and adaptive method)

\[
\begin{align*}
-\nabla \cdot (a \nabla u) + \nu u &= F \quad \text{in } \Omega = [0, 1]^d \\
u & \text{ periodic on } \partial \Omega,
\end{align*}
\]

Diffusion: \( a(x) = 1 + \frac{1}{4} \sin 2\pi x_1 \sin 2\pi x_2 \)

The exact solutions (examples):

\[
u_1(x) = \sin(4\pi x_1) \sin(2\pi x_2) \]

\[
u_2(x) = \exp(\sin 2\pi x_1 + \sin 2\pi x_2) \quad \text{smooth function}
\]

\[
u_3(x) = \sum_{k=1}^{p} c_k \sin 2\pi m_{1k} x_{d_{1k}} \sin 2\pi m_{2k} x_{d_{2k}} \quad p, m \text{ are integers}
\]

\[
u_4(x) = \exp \left( 2 \sum_{k=1}^{d} \frac{1}{k+1} \sin 2\pi x_k \right) \quad d \text{ active variables}
\]
NN (Neural Networks)
PINNs (Physics-Informed Neural Networks)

Assume the output of the neural network is $NN(x)$. PINNs [Lagaris et al., 1998] solve the differential equation

$$\mathcal{L}[u](x) = 0 \quad x \in \Omega$$

by minimizing the loss function

$$\text{Loss}(NN) = \sum_{i=1}^{m} (\mathcal{L}[NN](x_i) - F(x_i))^2,$$

where $x_i \in \Omega$ are sample points. We add a layer with nodes

$$v_{ij}^{(0)} = \sigma(A \cos(\omega x_i + \varphi) + c), \quad \omega = \frac{2\pi}{L}$$

after the input layer to ensure the periodic boundary conditions [Dong and Ni, 2021], where $A$, $\varphi$, $c$ are trainable parameters.
NN to enforce periodic boundary conditions \((d = 2)\)
A sample PINN 2D results

Diffusion: \( a(x) = 1 + \frac{1}{4} \sin 2\pi x_1 \sin 2\pi x_2 \)

The exact solution: \( u(x) = \sin(4\pi x_1) \sin(2\pi x_2) \),

**Figure:** PINN error: (a) Error of original function. (b) Error of the gradient. (c) Error of the divergence.
PINN high-dimensional results \((d = 6)\)

**Example 1**

![Graph of Example 1](image1.png)

**Example 2**

![Graph of Example 2](image2.png)

**Figure:** (Impact of the number of sample points) Relative \(L^2\)-error versus number of epochs. The exact solution are (a) \(u_1\), (b) \(u_2\).
PINN high-dimensional results \((d = 6)\)

**Example 3 (3 terms)**

**Example 4**

*Figure:* (Impact of the number of sample points) Relative \(L^2\)-error versus number of epochs. The exact solution are (a) \(u_3\), (b) \(u_4\).
PINN high-dimensional results (dimensionality)

Figure: (Impact of the dimensionality) Relative $L^2$-error versus number of epochs (exact solution is $u_1$). (a) 3000 sample points. (b) 10000 sample points. With adequate sample points, neural networks converge to the exact solution, regardless the dimensionality.
PINN high-dimensional results (dimensionality)

Figure: (Impact of the dimensionality to the number of sample points) Relative $L^2$-error versus number of sample points for $d = 6, 10, 20$ (after 30000 epochs). (a) $u_1$ (b) $u_2$. 
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]

The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \not\in \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)) \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{\mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda\} \]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin(2\pi x_1) \sin(2\pi x_2) \]

The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)) \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)) , \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_\mu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]

The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)) \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \not\in \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \(|c_\nu|\)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)) \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1\ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]

The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[
\mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \}
\]

Absolute value of Fourier coefficients \(|c_\nu|\)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[
\mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \\
\exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \}
\]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]
The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1\ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_\nu| \)

An illustration of the adaptive selection process.
Adaptive selection [Cohen et al., 2017]

Diffusion:
\[ a(x) = 1 + 0.25 \sin 2\pi x_1 \sin 2\pi x_2 \]

The anisotropic exact solution:
\[ u(x) = \exp(\sin(2\pi x_1) + 0.25 \sin(2\pi x_2)), \]

Margin (red circles) of an index set \( \Lambda \) (blue dots):
\[ \mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d, |\mu| - e_i \in \Lambda \} \]

Absolute value of Fourier coefficients \( |c_{\nu}| \)

An illustration of the adaptive selection process.
Margin or reduced margin

Figure: Margin vs. Reduced margin. Margin or reduced margin (red circles) of an index set $\Lambda$ (blue dots).

Margin: $\mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \exists i \in 1 \ldots d : |\mu| - e_i \in \Lambda \}$.
Reduced margin:
$\mathcal{R}(\Lambda) = \{ \mu \in \mathbb{Z}^d, \mu \notin \Lambda, \forall i \in 1 \ldots d : |\mu| - e_i \in \Lambda \}$. 

Adaptive lower OMP

**Algorithm** Adaptive lower OMP

**Input:** Number of iterations $K \in \mathbb{N}$.

**Output:** A $K$-sparse vector $\hat{c} \in \mathbb{C}^N$.

\[
\Lambda_0 = \mathcal{R}(0), \; z_0 = 0, \; S_0 \leftarrow \emptyset
\]

**for** $n = 0, \ldots, K - 1$ **do**

- Compute $A$ on $\Lambda_n \cup \mathcal{R}(\Lambda_n)$
- $d_j \leftarrow 1 / \sqrt{\sum_{i=1}^{m} |A_{ij}|^2}$, Normalize $A$, i.e. $A_{ij} \leftarrow A_{ij}d_j$
- $\nu \leftarrow \text{argmax}_{v_j \in \mathcal{R}(\Lambda_n)} \{ |(A^*(b - Az_n))_j| \}$
- $\nu_j$ represent the index corresponds to $j$-th column of $A$
- $\Lambda_{n+1} \leftarrow \Lambda_n \cup \{v\}$
- $S_{n+1} \leftarrow \text{Entry set corresponds to the index set } \Lambda_{n+1}$
- $z_{n+1} \leftarrow \text{arg min}_{z \in \mathbb{C}^N} \{ \| b - Az \|_2, \text{supp}(z) \subseteq S_{n+1} \}$
- $D \leftarrow \text{diag}(d)$, $z_{n+1} \leftarrow Dz_{n+1}$

**end for**

$\hat{c} \leftarrow z_{n+1}$
Adaptive OMP results ($d = 6$)

**Figure**: Relative $L^2$-error versus size of the index set. The number of sample points $m = 3000$. The exact solution are (a) $u_1$, (b) $u_2$. 
Adaptive OMP results \((d = 6)\)

**Figure:** Relative \(L^2\)-error versus size of the index set. The number of sample points \(m = 3000\). The exact solution are (a) \(u_3\), (b) \(u_4\).
Thanks for your attention!
Sparse Polynomial Approximation of High-Dimensional Functions, volume 25.
SIAM.

A compressive spectral collocation method for the diffusion equation under the restricted isometry property.
In Quantification of Uncertainty: Improving Efficiency and Technology, pages 15–40. Springer.

Sparse recovery in bounded riesz systems with applications to numerical methods for PDEs.

New explicit-in-dimension estimates for the cardinality of high-dimensional hyperbolic crosses and approximation of functions having mixed smoothness.

Discrete least-squares approximations over optimized downward closed polynomial spaces in arbitrary dimension.

A sparse spectral method for homogenization multiscale problems.


