Fast Sparse Spectral Methods for Higher Dimensional PDEs

Jie Shen

Purdue University

Collaborators: Li-Lian Wang, Haijun Yu and Alexander Alekseenko

Research supported by AFOSR and NSF

ICERM workshop, June 3-7, 2013

Outlines

- Approximation by hyperbolic cross
- Sparse grids for bounded domains
- Sparse grids for unbounded domains
- Application to electronic Schrödinger equation
- Adaptive sparse tensor products with multi-wavelets for the Boltzmann collision operator
- Concluding remarks

• Electronic Schrödinger equation: find the eigenvalues and eigenfunctions of the Hamilton operator

$$H = -\frac{1}{2}\sum_{i=1}^{N}\Delta_i - \sum_{i=1}^{N}\sum_{\nu=1}^{K}\frac{Z_\nu}{|x_i - a_\nu|} + \sum_{i=1}^{N}\sum_{j>i}^{N}\frac{1}{|x_i - x_j|},$$

where $x_1, \cdots, x_N \in \mathbb{R}^3$ are coordinates of N electrons.

• Boltzmann Equation:

$$\frac{\partial f}{\partial t} + \frac{\mathbf{p}}{m}\cdot\nabla_{\mathbf{x}} f = \mathbf{G}(f, f)$$

where $\mathbf{x}, \mathbf{p}$ are position and momentum of particle in $\mathbb{R}^3$, m is the particle mass and f is the density distribution function.

• Fokker-Planck Navier-Stokes equations for FENE dumbbell model:

$$\begin{cases} \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla_{\mathbf{x}})\mathbf{u} + \nabla_{\mathbf{x}} p = \frac{\nu}{\mathrm{Re}} \Delta_x \mathbf{u} + \frac{1-\nu}{\mathrm{Re}\,\mathrm{De}} \nabla_{\mathbf{x}} \cdot \tau_p, \\ \nabla_{\mathbf{x}} \cdot \mathbf{u} = 0, \\ \partial_t f + \nabla_{\mathbf{x}} \cdot (\mathbf{u} f) + \nabla_{\mathbf{q}} \cdot (\nabla_{\mathbf{x}} \mathbf{u} \cdot \mathbf{q} f) = \frac{1}{\mathrm{De}} \nabla_{\mathbf{q}} \cdot (\mathbf{F}(\mathbf{q}) f) + \frac{1}{\mathrm{De}} \Delta_{\mathbf{q}} f, \\ \tau_p(\mathbf{x}, t) = \int_W (\mathbf{F}(\mathbf{q}) \otimes \mathbf{q})\, f(\mathbf{x}, \mathbf{q}, t) \mathrm{d}\mathbf{q}. \end{cases}$$

$x \in \Omega \subset \mathbb{R}^3$, Ω is the fluid space; $\mathbf{q} \in W \subset \mathbb{R}^3$, W is the configuration space; $\mathbf{F}(\mathbf{q})$ is the configuration force.

A model problem

Consider the model elliptic equation:

$$\alpha u - \Delta u = f \ \text{ in } \ \Omega = (-1,1)^d; \quad u|_{\partial\Omega} = 0.$$

Given an approximation space V_n and an interpolation operator I_n, the (weighted) spectral-Galerkin method is to find $u_n \in V_n$ such that

$$(\nabla u_n, \nabla(v_n \omega)) = (I_n f, v_n)_\omega, \ \forall v_n \in V_n.$$

A model problem

Consider the model elliptic equation:

$$\alpha u - \Delta u = f \ \text{ in } \ \Omega = (-1,1)^d; \quad u|_{\partial\Omega} = 0.$$

Given an approximation space V_n and an interpolation operator I_n, the (weighted) spectral-Galerkin method is to find $u_n \in V_n$ such that

$$(\nabla u_n, \nabla(v_n\omega)) = (I_n f, v_n)_\omega, \ \forall v_n \in V_n.$$

Error estimate:

$$\|\nabla(u-u_n)\|_{L^2_\omega} \lesssim \inf_{v_n \in V_n} \|\nabla(u-v_n)\|_{L^2_\omega} + \|f - I_n f\|_{L^2_\omega}.$$

approximation error interpolation error

Usual spectral method:

Let $V_n = P_{n+1} \cap H_0^1(\Omega)$ where P_{n+1} is the space of polynomials of degree $\leq n+1$ in EACH variable.

- $N = \dim(V_n) = n^d$;
- $\inf_{v_n \in V_n} \|\nabla(u - v_n)\|_{L^2_\omega} \lesssim n^{1-s}\|u\|_{H^s_\omega} \lesssim N^{(1-s)/d}\|u\|_{H^s_\omega}$.

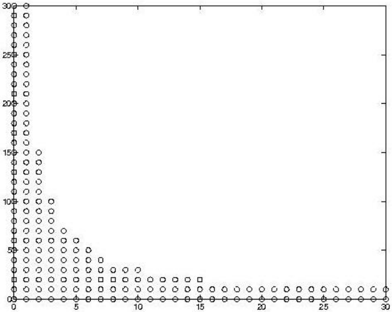
The # of degree of freedom increases exponentially fast with d — the convergence rate deteriorates rapidly as d increases.
This is the so called **Curse of dimensionality!**

Hyperbolic cross (Korobov, Babenko ’57)

The hyperbolic cross is defined as

$$\mathcal{K}(n) = \left\{ \mathbf{k} \in \mathbb{Z}_+^d : \prod_{i=1}^{d} \max(k_i, 1) \leq n \right\}$$

The cardinality of $\mathcal{K}(n)$ is of order $n(\log n)^{d-1}$.

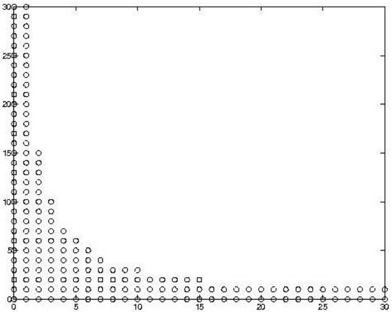


Hyperbolic cross (Korobov, Babenko '57)

The hyperbolic cross is defined as

$$\mathcal{K}(n) = \left\{ \mathbf{k} \in \mathbb{Z}_+^d : \prod_{i=1}^{d} \max(k_i, 1) \leq n \right\}$$

The cardinality of $\mathcal{K}(n)$ is of order $n(\log n)^{d-1}$.



$$\text{Set} \quad V_n = \text{span}\{\prod_{i=1}^{d} \phi_{k_i}(x_i) : \prod_{i=1}^{d} \max(k_i, 1) \leq n\},$$

in what functional spaces does V_n provide good approximations?

Table : Comparison of grids and hyperbolic cross

n	d=2			d=3			d=4	
	Grids	$\lvert\mathcal{K}(n)\rvert$	$\frac{\lvert\mathcal{K}(n)\rvert}{nlog(n)}$	Grids	$\lvert\mathcal{K}(n)\rvert$	$\frac{\lvert\mathcal{K}(n)\rvert}{nlog(n)^2}$	Grids	$\lvert\mathcal{K}(n)\rvert$
9	100	42	2.1	1000	141	3.2	10000	424
13	196	64	1.9	2744	228	2.7	38416	720
17	324	87	1.8	5832	321	2.4	104976	1041
21	484	113	1.8	10648	435	2.2	234256	1457
25	676	138	1.7	17576	546	2.1	456976	1877
29	900	162	1.7	27000	646	2.0	810000	2229
33	1156	190	1.6	39304	778	1.9	1336336	2745
37	1444	217	1.6	54872	904	1.9	2085136	3239
41	1764	243	1.6	74088	1021	1.8	3111696	3683
45	2116	273	1.6	97336	1165	1.8	4477456	4243

Korobov spaces for periodic functions

Given $f \in L_p^2(0, 2\pi)^d$ and its Fourier series

$$f(\mathbf{x}) = \sum_{\mathbf{k}\in\mathbb{Z}^d} \hat{f}(\mathbf{k}) e^{i\mathbf{k}\cdot\mathbf{x}} \quad \text{with } \hat{f}(\mathbf{k}) = \frac{1}{|\Omega|}\int_\Omega f(\mathbf{x}) e^{-i\mathbf{k}\cdot\mathbf{x}} d\mathbf{x}.$$

Let us define $r(\mathbf{k}) := \prod_{j=1}^d \max(|k_j|, 1), \quad \mathbf{k} = (k_1, \ldots, k_d)$, then, the Korobov space of order $\alpha \in \mathbb{R}$ is

$$\begin{aligned} K_p^\alpha &:= \left\{ f \in L_p^2(\Omega) : \quad \sum_{\mathbf{k}\in\mathbb{Z}^d} |\hat{f}(\mathbf{k})|^2 r(\mathbf{k})^{2\alpha} < \infty \right\} \\ &= \left\{ \frac{\partial^{q_1+\ldots+q_d}}{\partial x_1^{q_1} \ldots \partial x_d^{q_d}} f \in L_p^2(\Omega) : 0 \le q_j \le \alpha,\ 1 \le j \le d \right\}. \end{aligned}$$

Setting $X_n := \left\{ f : f(\mathbf{x}) = \sum_{r(\mathbf{k})\leq n} \hat{f}(\mathbf{k}) e^{i\mathbf{k}\cdot\mathbf{x}} \right\}$. Then, one can prove

$$\inf_{v_n \in X_n} \|v - v_n\|_{K_p^r} \lesssim n^{r-s} \|v\|_{K_p^s}. \quad \text{(e.g. Griebel \& Hamaekers 2007)}$$

Setting $X_n := \left\{ f : f(\mathbf{x}) = \sum_{r(\mathbf{k}) \le n} \hat{f}(\mathbf{k}) e^{i\mathbf{k}\cdot\mathbf{x}} \right\}$. Then, one can prove

$$\inf_{v_n \in X_n} \|v - v_n\|_{K_p^r} \lesssim n^{r-s} \|v\|_{K_p^s}. \quad \text{(e.g. Griebel \& Hamaekers 2007)}$$

Remarks:

- The estimate still depends on d since $N = O(n(\log n)^{d-1})$.
- It is also possible to remove the dependence on d by using the optimized (energy based) hyperbolic cross space (Bungartz & Griebel '04):

$$\mathcal{K}_\sigma(n) = \left\{ \mathbf{k} \in \mathbb{Z}_+^d : \prod_{i=1}^{d} \max(k_i, 1) |\, |\mathbf{k}|_\infty^{-\sigma} \le n^{1-\sigma} \right\}.$$

Then, for $\sigma \in (0,1)$, $\text{Card}(K_\sigma(n)) = O(n)$. In this case, the estimate holds only for $\sigma \le \frac{r}{s}$.

Setting $X_n := \left\{f : f(\mathbf{x}) = \sum_{r(\mathbf{k})\le n} \hat{f}(\mathbf{k})e^{i\mathbf{k}\cdot\mathbf{x}}\right\}$. Then, one can prove

$$\inf_{v_n\in X_n} \|v - v_n\|_{K_p^r} \lesssim n^{r-s}\|v\|_{K_p^s}.$$ (e.g. Griebel & Hamaekers 2007)

Remarks:

- The estimate still depends on d since $N = O(n(\log n)^{d-1})$.
- It is also possible to remove the dependence on d by using the optimized (energy based) hyperbolic cross space (Bungartz & Griebel '04):

$$\mathcal{K}_\sigma(n) = \left\{\mathbf{k} \in \mathbb{Z}_+^d : \prod_{i=1}^{d} \max(k_i, 1)| \, |\mathbf{k}|_\infty^{-\sigma} \le n^{1-\sigma}\right\}.$$

  Then, for $\sigma \in (0, 1)$, $\mathrm{Card}(K_\sigma(n)) = O(n)$. In this case, the estimate holds only for $\sigma \le \frac{r}{s}$.
- These estimates were extended to the non-periodic case and unbounded domains in S. & Wang (SINUM '10).

Korobov type spaces for non-periodic functions

Let $\omega^{\alpha,\beta} = (1-x)^{\alpha}(1+x)^{\beta}$ and $J_n^{(\alpha,\beta)}$ be the Jacobi weight and Jacobi polynomial with index $\alpha, \beta > -1$. We define the multi-dimensional Jacobi weights and Jacobi polynomials by

$$\omega^{\bar{\alpha},\bar{\beta}}(\bar{x}) = \Pi_{i=1}^{d} \omega^{\alpha_i,\beta_i}(x_i), \ \Phi_{\bar{k}}^{\bar{\alpha},\bar{\beta}}(\bar{x}) = \Pi_{i=1}^{d} J_{k_i}^{\alpha_i,\beta_i}(x_i).$$

We then define (for any integer $t \geq 0$)

$$K_{\bar{\alpha},\bar{\beta}}^{t} := \{u : \|u\|_{K_{\bar{\alpha},\bar{\beta}}^{t}}^{2} := \sum_{|\bar{r}|_{\infty} \leq t} \|\partial^{\bar{r}} u\|_{\omega^{\bar{\alpha}+\bar{r},\bar{\beta}+\bar{r}}}^{2} < \infty\}.$$

Denote $P_n^{\bar\alpha,\bar\beta} = \text{span}\{\Phi_{\bar k}^{\bar\alpha,\bar\beta}(\bar x) : \Pi_{i=1}^d \max(k_i, 1) \le n\}$, and define $\Pi_n^{\bar\alpha,\bar\beta} : L^2_{\omega^{\bar\alpha,\bar\beta}} \to P_n^{\bar\alpha,\bar\beta}$ be the orthogonal projector defined by

$$(u - \Pi_n^{\bar\alpha,\bar\beta} u, v_K)_{\omega^{\bar\alpha,\bar\beta}} = 0, \ \forall v_n \in P_n^{\bar\alpha,\bar\beta}.$$

Theorem. (S. & Wang '10) For $u \in K^s_{\bar\alpha,\bar\beta}$ and $0 \le t < s$, we have

$$\|u - \Pi_n^{\bar\alpha,\bar\beta} u\|_{K^t_{\bar\alpha,\bar\beta}} \lesssim n^{t-s} |u|_{K^s_{\bar\alpha,\bar\beta}},$$

where

$$|u|^2_{K^s_{\bar\alpha,\bar\beta}} := \sum_{|\bar r|_\infty = s} \|\partial^{\bar r} u\|^2_{\omega^{\bar\alpha+\bar r,\bar\beta+\bar r}}.$$

Projection errors in the multi-D case

Denote $P_n^{\bar{\alpha},\bar{\beta}} = \text{span}\{\Phi_{\bar{k}}^{\bar{\alpha},\bar{\beta}}(\bar{x}) : \Pi_{i=1}^d \max(k_i, 1) \le n\}$, and define $\Pi_n^{\bar{\alpha},\bar{\beta}} : L^2_{\omega^{\bar{\alpha},\bar{\beta}}} \to P_n^{\bar{\alpha},\bar{\beta}}$ be the orthogonal projector defined by

$$(u - \Pi_n^{\bar{\alpha},\bar{\beta}} u, v_K)_{\omega^{\bar{\alpha},\bar{\beta}}} = 0, \ \forall v_n \in P_n^{\bar{\alpha},\bar{\beta}}.$$

Theorem. (S. & Wang '10) For $u \in K^s_{\bar{\alpha},\bar{\beta}}$ and $0 \le t < s$, we have

$$\|u - \Pi_n^{\bar{\alpha},\bar{\beta}} u\|_{K^t_{\bar{\alpha},\bar{\beta}}} \lesssim n^{t-s} |u|_{K^s_{\bar{\alpha},\bar{\beta}}},$$

where

$$|u|^2_{K^s_{\bar{\alpha},\bar{\beta}}} := \sum_{|\bar{r}|_\infty = s} \|\partial^{\bar{r}} u\|^2_{\omega^{\bar{\alpha}+\bar{r},\bar{\beta}+\bar{r}}}.$$

Approximation results in usual H^1-norm and for elliptic equations are also established in S. & Wang '10.

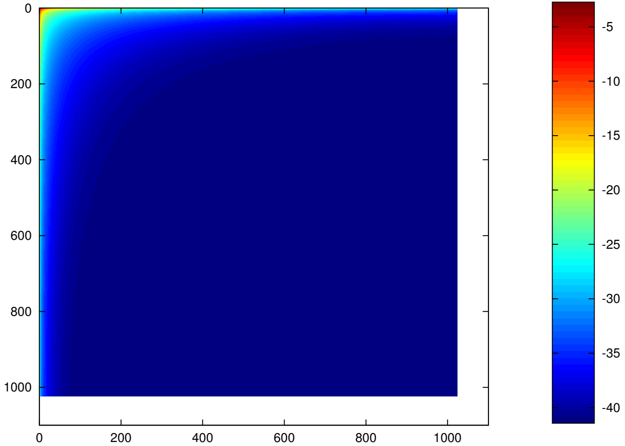


Figure : Spectrum of a highly anisotropic exact solution

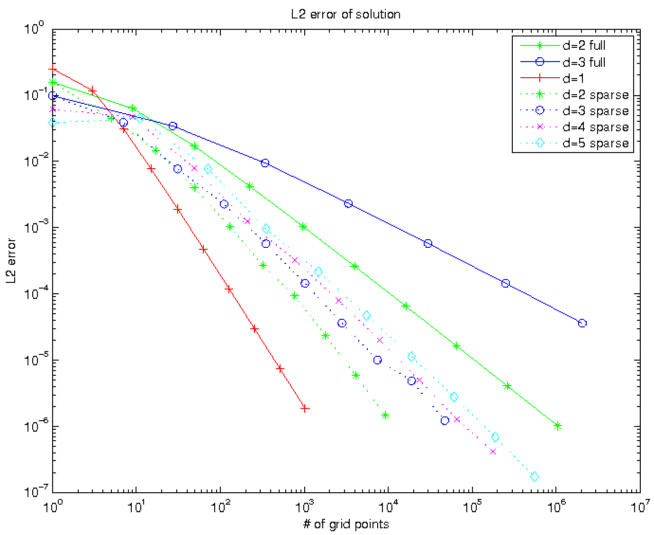


Figure : Convergence history: comparison with the full grid

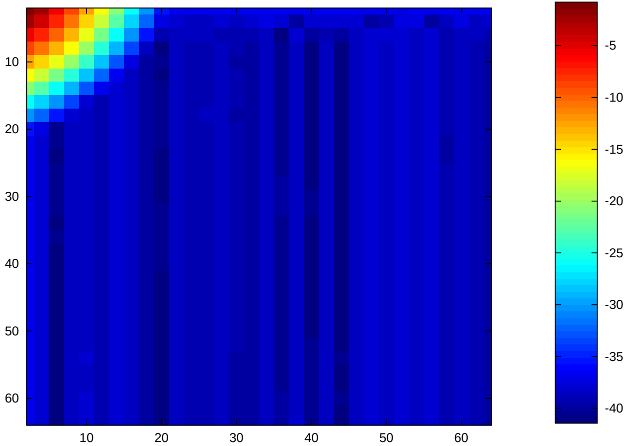


Figure : Spectrum of an isotropic exact solution

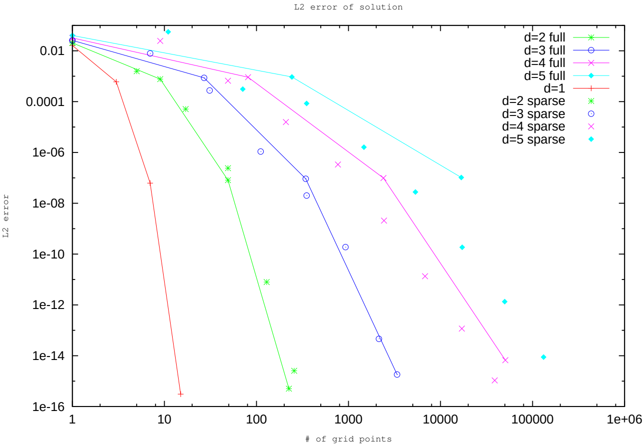


Figure : Convergence history: comparison with the full grid

Sparse grids

Sparse grid construction (Smolyak '63):
Start from a sequence of (preferably **nested**) univariate quadrature/interpolation formulas:

$$Q_i f := \sum_{j=1}^{n_i} f(x_j^i)\omega_j^i, \qquad \Delta_i = Q_i - Q_{i-1} \text{ with } Q_0 = 0.$$

Define a multidimensional quadrature rule by:

$$Q_l^{(d)} f := \sum_{d \le |\mathbf{i}| \le l} (\Delta_{i_1} \otimes \cdots \otimes \Delta_{i_d}) f.$$

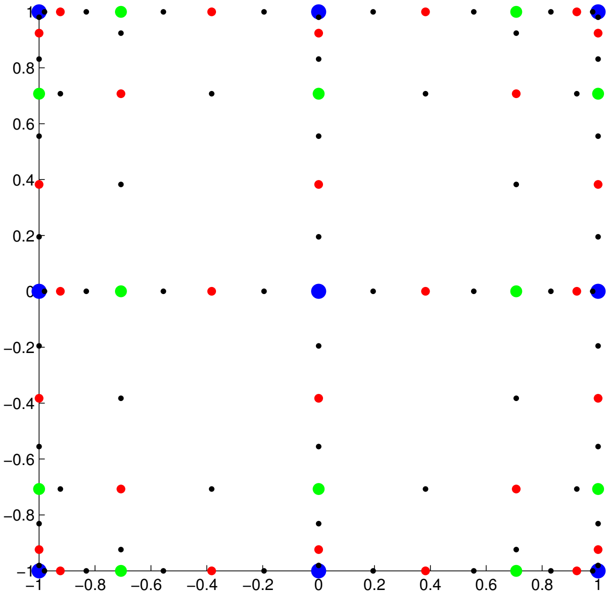


Figure : $Q_5^{(2)}$ based on Chebyshev-Gauss-Lobatto quadrature.

Sparse "grids" in frequency space

Similarly, we can define sparse "grids" in frequency space.
For example, by splitting the 1-D basis $\{P_0(x), P_1(x), P_2(x), ...\}$ into

$$\Delta_1 = \{P_0, P_1, P_2\},\ \Delta_2 = \{P_3, P_4\},\ \Delta_3 = \{P_5, P_6, P_7, P_8\},\ \cdots$$

we can define sparse "grids" in frequency space by

$$H_l^{(d)} := \sum_{d \le |\mathbf{i}| \le l} (\Delta_{i_1} \otimes \cdots \otimes \Delta_{i_d}).$$

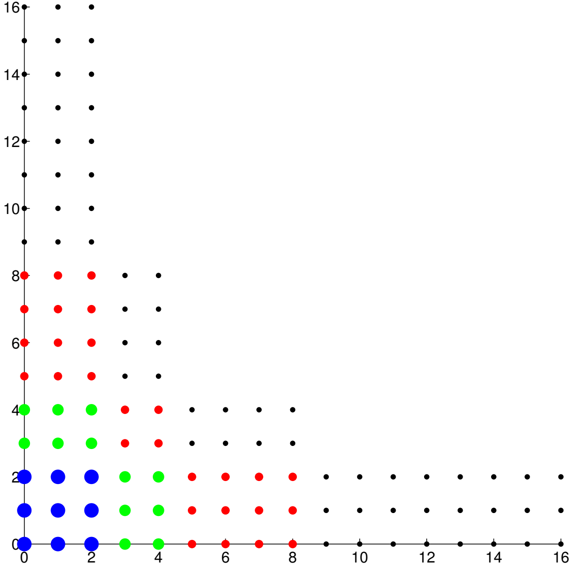
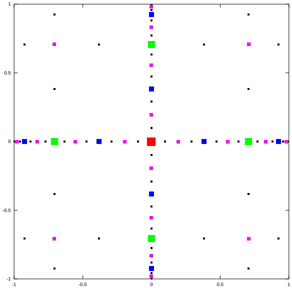
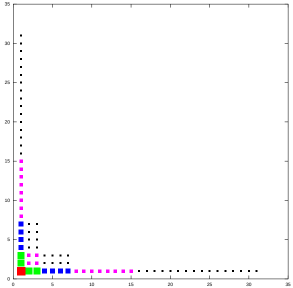


Figure : $H_5^{(2)}$ in frequency space.

Optimized hyperbolic cross/sparse grid





The key ingredient for constructing a fast transform between the space grids in physical and frequency spaces is the **Hierarchical basis**.

Hierarchical basis and interpolation on sparse grid

The key ingredient for constructing a fast transform between the space grids in physical and frequency spaces is the **Hierarchical basis**.

Given a sparse grid generated by nested 1-D quadratures with the index set $I_i = \{0, 1, \cdots, n_i - 1.\}$ A set of function $\{\phi_k\}$ form a **hierarchical bases** if

$$\phi_k(x_j^i) = 0, \ \forall j \quad \text{if } k \geq n_i.$$

Then, we can rearrange $\{x_j^i\}$ as $\{x_j\}$ and define an interpolation operator by

$$f(x_j) = \sum_{k \in I^i} b_k \phi_k(x_j), \quad \text{for any } j \in I^i, \quad i = 1, 2, \ldots$$

where $\{b_k\}$ do not depend on the level index i.

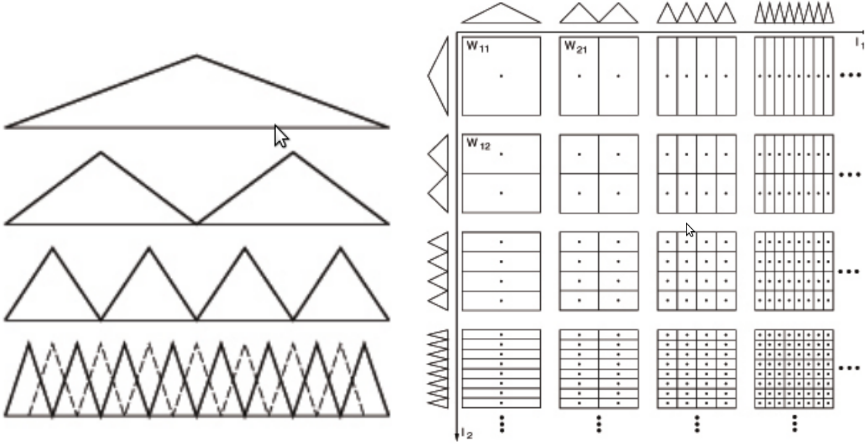


Figure : **Left:** The 1-d linear hierarchical finite element bases; **Right:** The 2-d construction of sparse grid (from Bungartz and Griebel 2004)

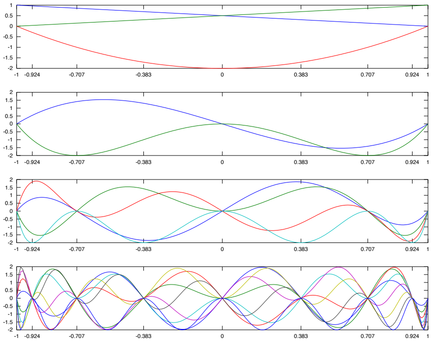


Figure : Hierarchical basis functions with Chebyshev-Gauss-Lobatto quadrature:
$\phi_k = T_k$, for $k \in I^1$; $\quad \phi_k = T_k - T_{2^i - k}$, for $k \in \tilde{I}^i = I^i \backslash I^{i-1}, i > 1$.

Multi-dimensional case:

Define the multi-D index set corresponding to the sparse grid

$$I_l^{(d)} := \sum_{d \le |\mathbf{i}| \le l} (\tilde{I}_{i_1} \otimes \cdots \otimes \tilde{I}_{i_d}),$$

and the multi-dimensional hierarchical bases by

$$\phi_{\mathbf{k}}(\mathbf{x}) = \Pi_{i=1}^{d} \phi_{k_i}(x_i).$$

Then, we can define the interpolation operator $U_l^{(d)}$ on the sparse grid by

$$(U_l^{(d)} f)(\mathbf{x_j}) = \sum_{\mathbf{k} \in I_l^{(d)}} b_{\mathbf{k}} \phi_{\mathbf{k}}(\mathbf{x_j}), \quad \forall \mathbf{j} \in I_l^{(d)}.$$

The transform between the values at the sparse grid and the coefficients of the hierarchical basis can be performed with quasi-optimal computational complexity!

Implementation of hyperbolic-cross/sparse-grid solver

$$\alpha u - \Delta u = f, \mathbf{x} \in \Omega = (-1,1)^d, \quad u|_{\partial\Omega} = 0.$$

- Sparse grid in frequency space:

$$\mathcal{I}_d^q = \left\{(k_1, k_2, \ldots, k_d) : 0 \leqslant k_s < 2^{i_s} - 1, i_s \geqslant 0, |\mathbf{i}|_1 = q\right\},$$

$$V_d^q = \left\{\tilde{\phi}_{\mathbf{k}}, k \in \mathcal{I}_d^q\right\};$$

- Sparse grid in physical space:

$$\mathcal{J}_d^q = \left\{(k_1, k_2, \ldots, k_d) : 0 \leqslant k_s < 2^{i_s} + 1, i_s \geqslant 0, |\mathbf{i}|_1 = q\right\},$$

$$\mathcal{X}_d^q = \left\{x_{\mathbf{j}}, \mathbf{j} \in \mathcal{J}_d^q\right\};$$

- $\mathcal{U}_d^q$ the interpolation on $\mathcal{X}_d^q$.
- Find $u_d^q \in V_d^q$ such that

$$\alpha(u_d^q, v)_\omega - (\Delta u_d^q, v)_\omega = (\mathcal{U}_d^q f, v)_\omega, \quad \forall v \in V_d^q.$$

Choice of basis functions for V_d^q

Remark: While the hierarchical basis plays a critical role in fast transform, it lacks good orthogonality property.

Choice of basis functions for V_d^q

Remark: While the hierarchical basis plays a critical role in fast transform, it lacks good orthogonality property.

- Chebyshev-Galerkin method:

$$\phi_k(x) = T_k(x) - T_{k+2}(x), \quad \omega(x) = (1 - x^2)^{-1/2}$$

1-d mass matrix: banded; 1-d stiff matrix: triangular;
the multi-D stiffness matrix is not sparse.

Choice of basis functions for V_d^q

Remark: While the hierarchical basis plays a critical role in fast transform, it lacks good orthogonality property.

- Chebyshev-Galerkin method:

$$\phi_k(x) = T_k(x) - T_{k+2}(x), \quad \omega(x) = (1 - x^2)^{-1/2}$$

1-d mass matrix: banded; 1-d stiff matrix: triangular;
the multi-D stiffness matrix is not sparse.

- Legendre-Galerkin method:

$$\phi_k(x) = L_k(x) - L_{k+2}(x), \quad \omega(x) = 1$$

1-d mass matrix: banded; 1-d stiff matrix: diagonal;
Legendre-Gauss type points not nested; lack of fast transform.

Choice of basis functions for V_d^q

Remark: While the hierarchical basis plays a critical role in fast transform, it lacks good orthogonality property.

- Chebyshev-Galerkin method:

$$\phi_k(x) = T_k(x) - T_{k+2}(x), \quad \omega(x) = (1 - x^2)^{-1/2}$$

1-d mass matrix: banded; 1-d stiff matrix: triangular;
the multi-D stiffness matrix is not sparse.

- Legendre-Galerkin method:

$$\phi_k(x) = L_k(x) - L_{k+2}(x), \quad \omega(x) = 1$$

1-d mass matrix: banded; 1-d stiff matrix: diagonal;
Legendre-Gauss type points not nested; lack of fast transform.

- Chebyshev-Legendre-Galerkin method (Don & Gottlieb '94, S. '96)

Linear algebraic system

Denote

$$\phi_{\mathbf{k}}(\mathbf{x}) = \phi_{k_1}(x_1)\cdots\phi_{k_d}(x_d), \quad \text{then} \quad V_d^q = \left\{\phi_{\mathbf{k}}, k \in \mathcal{I}_d^q\right\}.$$

The Galerkin method leads to

$$(\alpha M + S)\mathbf{u} = \mathbf{f},$$

where

$$M = \left(m_{k_1,j_1}\cdots m_{k_d,j_d}\right)_{\mathbf{k},\mathbf{j}\in\mathcal{I}_d^q},$$

$$S = \left(s_{k_1,j_1} m_{k_2,j_2}\cdots m_{k_d,j_d}\right)_{\mathbf{k},\mathbf{j}\in\mathcal{I}_d^q} + \cdots + \left(m_{k_1,j_1}\cdots m_{k_2,j_2} s_{k_d,j_d}\right)_{\mathbf{k},\mathbf{j}\in\mathcal{I}_d^q},$$

$$m_{k,j} = (\phi_k, \phi_j)_\omega, \quad s_{k,j} = (-\Delta\phi_k, \phi_j)_\omega$$

$$\mathbf{f} = (f_{\mathbf{k}})_{\mathbf{k}\in\mathcal{I}_d^q}, \quad f_{\mathbf{k}} = (\mathcal{U}_d^q f, \phi_{\mathbf{k}}).$$

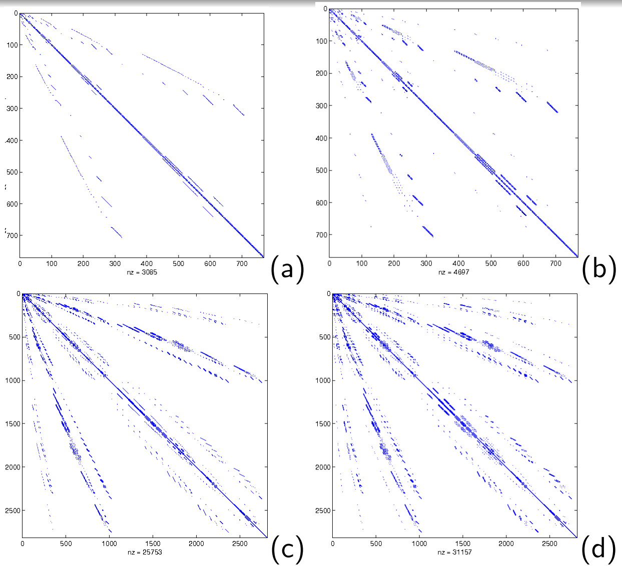


Figure : Sparsity of Legendre-Galerkin method: (a) stiff $\mathcal{X}_2^8$; (b) system $\mathcal{X}_2^8$; (c) stiff $\mathcal{X}_3^9$; (d) system $\mathcal{X}_3^9$.

d	q	n	nnz_0	nnz_1	$cond_1$	$pcond_1$
2	4	17	45	49	5.008	2.035
	5	49	157	193	13.248	2.819
	6	129	461	625	55.992	5.941
	7	321	1229	1777	156.295	8.670
	8	769	3085	4657	746.584	20.689
	9	1793	7437	11569	2125.186	30.454
4	5	9	29	29	1.938	1.836
	6	49	281	281	5.510	3.795
	7	209	1809	1809	33.153	7.297
	8	769	9009	9025	212.514	14.519
	9	2561	37601	37873	995.447	30.715

Table : nnz_0: number of non-zeros of stiff matrix; nnz_1: number of non-zeros of system matrix ($\alpha = 1$); $cond_1$: the condition number of system matrix ($\alpha = 1$); $pcond_1$: the condition number of system matrix ($\alpha = 1$) with diagonal pre-conditioner.

PCG with fast matrix-vector product algorithm

For the Chebyshev sparse Galerkin method:

- The evaluation of $\mathbf{f}$ can be performed using the fast transform algorithm.
- The stiff matrix S is not sparse but the matrix-vector product $M\mathbf{u}$, $S\mathbf{u}$ can be performed with quasi-optimal computational complexity.
- Solve the linear system by a preconditioned CG type method — How to construct an optimal pre-conditioner is still an open problem.

PCG with fast matrix-vector product algorithm

For the Chebyshev sparse Galerkin method:

- The evaluation of $\mathbf{f}$ can be performed using the fast transform algorithm.
- The stiff matrix S is not sparse but the matrix-vector product $M\mathbf{u}$, $S\mathbf{u}$ can be performed with quasi-optimal computational complexity.
- Solve the linear system by a preconditioned CG type method — How to construct an optimal pre-conditioner is still an open problem.

Not optimal computational complexity, but memory requirement is small since it does not need explicit formation of stiffness/mass matrices.

Direct sparse solvers

In order to use an efficient sparse solver, we need to form explicitly the system matrix — only possible with Chebyshev-Legendre sparse grid method

- The evaluation of **f** is done by the fast Chebyshev-Legendre transform (by FMM, Alpert & Rokhlin '91).
- The sparse mass and stiff matrices can be assembled with a fast algorithm.
- Solve the linear system by using an efficient sparse solver
  - UMFPack (T. Davis) — competitive with PCG in some cases
  - AMG (Y. Notay) — competitive with PCG in some cases
  - SuperMF (J. Xia et al.) — can lead to quasi-optimal computational complexity.

Direct sparse solvers

In order to use an efficient sparse solver, we need to form explicitly the system matrix — only possible with Chebyshev-Legendre sparse grid method

- The evaluation of **f** is done by the fast Chebyshev-Legendre transform (by FMM, Alpert & Rokhlin '91).
- The sparse mass and stiff matrices can be assembled with a fast algorithm.
- Solve the linear system by using an efficient sparse solver
  - UMFPack (T. Davis) — competitive with PCG in some cases
  - AMG (Y. Notay) — competitive with PCG in some cases
  - SuperMF (J. Xia et al.) — can lead to quasi-optimal computational complexity.

Quasi-optimal computational complexity, but memory requirement is large.

Sparse grid by Hermite Gauss quadrature

- "The natural choice": Construct Sparse grids from the Hermite-Gauss points and Laguerre-Gauss points.
- **The problem**: The Hermite-Gauss and Laguerre-Gauss points are not nested — leads to too many points in the sparse grid so it is not computationally effective.

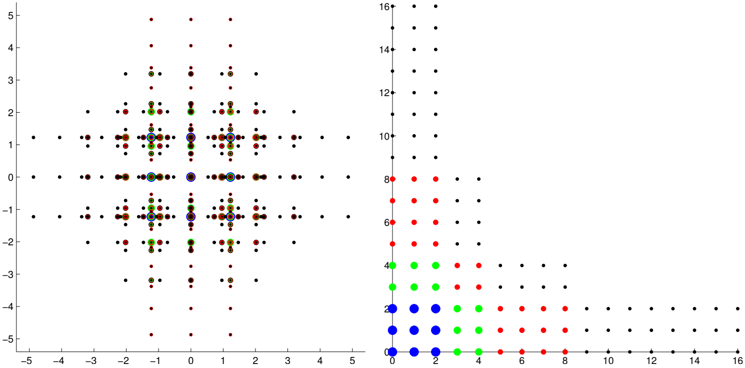


Figure : Left: Hermite sparse grid $\mathcal{X}_2^5$; Right: the corresponding index

Mapped Chebyshev method

- Given a mapping $x = x(\xi) : (-1, 1) \to \mathbb{R}$ and its inverse g $\xi = \xi(x) : \mathbb{R} \to (-1, 1)$. The mapped Chebyshev functions

$$\hat{T}_k(x) = T_k(\xi(x))\mu(\xi(x)),$$

$$\mu(\xi) = \sqrt{\omega(\xi)/x'(\xi)} \text{ with } \omega(\xi) = 1/\sqrt{1-\xi^2}$$

which satisfies

$$(\hat{T}_k, \hat{T}_j) = \int_{\mathbb{R}} \hat{T}_k(x)\, \hat{T}_j(x) dx = \int_{-1}^{1} T_k(\xi) T_j(\xi) \omega(\xi) d\xi = \delta_{kj}.$$

- The convergence rate will depend on the rate of decay at infinity
- A class of rational mapping: $(-1, 1) \to \mathbb{R}$:

$$x'(\xi) = \frac{L}{(1-\xi^2)^{1+r/2}}, \quad r \geqslant 0,$$

Two most useful cases: $r = 0, 1$:

$$x(\xi) = \begin{cases} \frac{L}{2}\log\frac{1+\xi}{1-\xi}, & r = 0 \\ \frac{L\xi}{\sqrt{1-\xi^2}}, & r = 1 \end{cases}; \qquad \xi(x) = \begin{cases} \tanh(\frac{1}{L}x) & r = 0 \\ \frac{x}{\sqrt{x^2+L^2}}, & r = 1 \end{cases}.$$

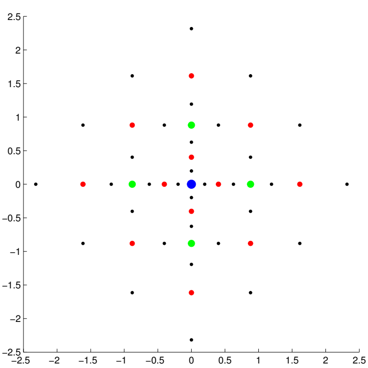


Figure : Mapped Chebyshev sparse grids. Left: $\mathcal{X}_2^5$ with $r = 0$; Right: $\mathcal{X}_2^5$ with $r = 1$.

Let us define our hyperbolic approximation space by

$$\mathbf{X}_N^d := \operatorname{span}\left\{\widehat{\mathbf{T}}_{\mathbf{k}} \ : \ \mathbf{k} \in \Upsilon_N^H\right\},$$

with

$$\Upsilon_N^H = \left\{\mathbf{k} \in \mathbb{N}_0^d : 1 \le |\mathbf{k}|_{\mathrm{mix}} := \prod_{j=1}^{d} \max\{1, k_j\} \le N\right\}.$$

We define the orthogonal projection $\boldsymbol{\pi}_N^d : L^2(\mathbb{R}^d) \to \mathbf{X}_N^d$, and the mapped derivative

$$D_x u := a(x)\frac{d\hat{u}}{dx} \ \text{ with } \ a(x) = \frac{dx}{d\xi}, \quad \hat{u}(x) = \frac{u(x)}{\mu(\xi(x))},$$

$$D_{\mathbf{x}}^{\mathbf{k}} u = D_{x_1}^{k_1} \cdots D_{x_d}^{k_d} u, \quad \varpi^{(1+r)/2+\mathbf{k}} = \prod_{j=1}^{d} (1-\xi_j^2)^{(1+r)/2+k_j}.$$

Theorem. For $r \geq 0$ and any $u \in \mathbf{K}^m(\mathbb{R}^d)$, we have

$$\|\pi_N^d u - u\|_{L^2(\mathbb{R}^d)} \leq CN^{-m}|u|_{\mathbf{K}^m(\mathbb{R}^d)}, \quad m \geq 0,$$

and

$$\|\nabla(\pi_N^d u - u)\|_{L^2(\mathbb{R}^d)} \leq CN^{1-m}|u|_{\mathbf{K}^m(\mathbb{R}^d)}, \quad m \geq 1.$$

where $\mathbf{K}^m(\mathbb{R}^d)$ is the Korobov-type space

$$\mathbf{K}^m(\mathbb{R}^d) := \Big\{u \,:\, D_{\mathbf{x}}^{\mathbf{k}} u \in L^2_{\varpi^{(1+r)/2+\mathbf{k}}}(\mathbb{R}^d),\ 0 \leq |\mathbf{k}|_\infty \leq m\Big\}, \quad \forall m \in \mathbb{N}_0,$$

and

$$|u|_{\mathbf{K}^m(\mathbb{R}^d)} = \Big(\sum_{|\mathbf{k}|_\infty = m} \big\|D_{\mathbf{x}}^{\mathbf{k}} u\big\|_{L^2_{\varpi^{(1+r)/2+\mathbf{k}}}(\mathbb{R}^d)}\Big)^{\frac{1}{2}}.$$

- Mass matrix $M = I$; Stiff matrix S: banded — leads to system matrix with much less non-zero entries.

$$\mathbb{S} = S \otimes M \otimes \cdots \otimes M + M \otimes S \otimes M \cdots \otimes M + M \otimes \cdots \otimes M \otimes S$$

- Fast transform available on mapped Chebyshev sparse grids.

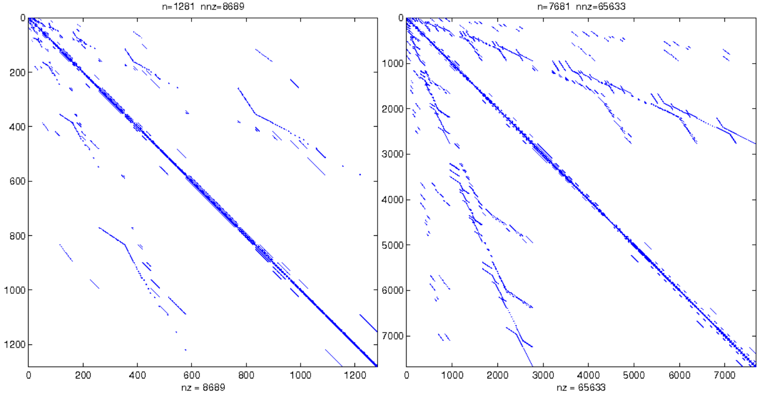


Figure : Sparsity of the mapped Chebyshev method: Left: $d = 2, q = 8$;

The linear system for Poisson type equations can be solved efficiently by using the sparse fast transform and PCG or a sparse solver such as CHOLMOD.

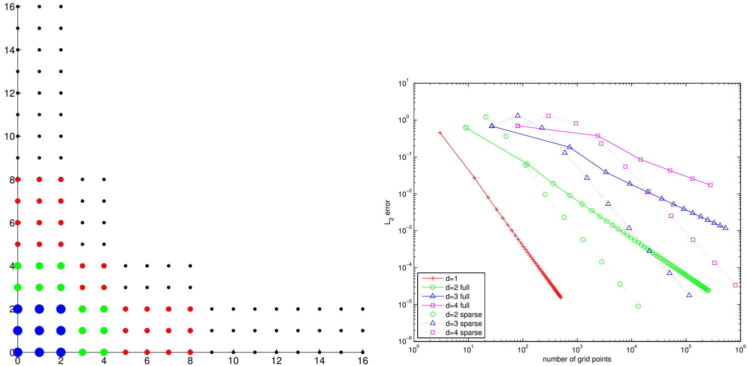


Figure : Comparison between the MCFG method and MCSG method (with $r = 1$) for solving Poisson-type equation with an anisotropic solution with algebraic delay.

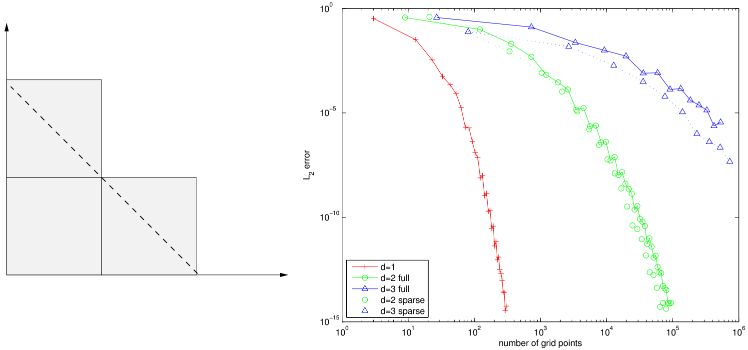
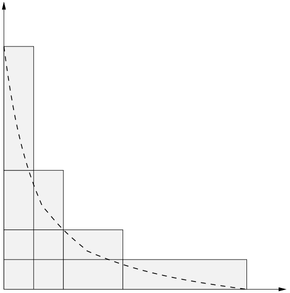


Figure : Left: the optimal frequency index set for tensor product of 1-D function with geometric convergence; Right: Comparison between the MCFG method and MCSG method (with $r = 1$).



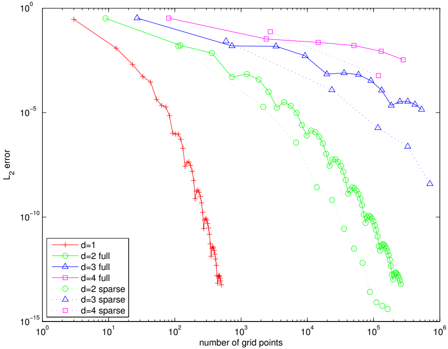


Figure : Left: the optimal frequency index set for tensor product of 1-D function with subgeometric convergence; Right: Comparison between the MCFG method and MCSG method (with $r = 1$).

Problems with variable coefficients

- Use a constant coefficient problem as preconditioner
- While the system matrix is full, but the matrix-vector product can be performed in quasi-optimal complexity

		Ex. 1: $r = 0$		Ex. 1: $r = 1$		Ex. 2: $r = 0$	
d	q	iter#	CPU	iter#	CPU	iter#	CPU
1	7	13	3.000(-3)	13	4.000(-3)	6	1.000(-3)
	8	13	7.000(-3)	13	6.000(-3)	5	2.000(-3))
	9	15	1.000(-2)	13	1.000(-2)	5	4.000(-3)
2	8	12	4.400(-2)	13	5.200(-2)	6	2.600(-2)
	9	13	1.040(-1)	14	1.150(-1)	5	4.200(-2)
	10	13	2.270(-1)	14	2.410(-1)	5	8.700(-2)
3	9	14	6.560(-1)	13	7.780(-1)	6	2.920(-1)
	10	14	1.510(-0)	13	1.911(-0)	6	6.830(-1)
	11	14	4.071(-0)	13	4.749(-0)	6	1.857(-0)

Table : The numbers of iterations and CPU time of using BICGSTAB solving equations with non-constant coefficients

Application to electronic Schrödinger equation

- Much effort have been devoted to this problem using density function theory, but not much work on solving the equation directly;
- It has been shown (Yserentant '04) that its solution lives in the Korobov space;
- The problem is high-dimensional and set in unbounded domains; previous work (Gribel & Hamaekers '07) was based on domain truncation and Fourier method — the effect of domain truncation is not easy to quantify;
- Our fast sparse spectral method in unbounded domain is well suited for this problem.

Consider the 1-D electronic Schrödinger equation:

$$H\psi = \lambda\psi$$

with

$$H = T + V := -\frac{1}{2}\sum_{i=1}^{N}\Delta_i + \Big(\sum_{i=1}^{N} Z|x_i| - \sum_{i=1}^{N}\sum_{j>i}^{N}|x_i - x_j|\Big).$$

We look for the ground state energy which is the smallest eigenvalue of H.

Consider the 1-D electronic Schrödinger equation:

$$H\psi = \lambda\psi$$

with

$$H = T + V := -\frac{1}{2}\sum_{i=1}^{N}\Delta_i + \Big(\sum_{i=1}^{N} Z|x_i| - \sum_{i=1}^{N}\sum_{j>i}^{N}|x_i - x_j|\Big).$$

We look for the ground state energy which is the smallest eigenvalue of H.

Sparse spectral-Galerkin approximation: Find $\psi \in V_N^q,\ \lambda \in \mathbb{R}$ such that

$$(T\psi, \phi) + < V\psi, \phi >_{disc} = \lambda(\psi, \phi), \quad \forall \phi \in V_N^q,$$

where V_N^q consists of mapped Chebyshev polynomials.

Using the mapped Chebyshev polynomials, the above reduces to:

$$(\mathcal{S} + \mathcal{V})E = \lambda \mathcal{M} E$$

where

- the mass matrix $\mathcal{M}$ is diagonal,
- the stiffness matrix $\mathcal{S}$ is sparse,
- the matrix $\mathcal{V}$ is full, but the matrix-vector product $\mathcal{V}\bar{x}$ can be performed in quasi-optimal complexity thanks to the fast Chebyshev transform.

How to find the smallest eigenvalue?

- The direct application of Arnoldi method for computing the smallest eigenvalue converges very slowly;
- To accelerate the Arnoldi method, we employed a shifted-inverse technique which requires solving

$$(\mathcal{S} + \mathcal{V} - \delta I)\bar{x} = \bar{f};$$

- We used $\mathcal{S} - \delta I$ to precondition the above system.

The above approach appears to be very efficient and robust. For example, six-dimensional problem with $q = 12$ can be solved on a laptop using single core within 3 hours.

q	dof	$L = 0.5$	$L = 0.6$	$L = 0.75$	$L = 1$
6	257	2.0139266	1.9089328	1.8596860	1.8463230
7	577	1.9411353	1.8801002	1.8579695	1.8511776
8	1281	1.9053250	1.8713405	1.8624749	1.8607346
9	2817	1.8838480	1.8657901	1.8622006	1.8610317
10	6145	1.8734093	1.8646394	1.8634796	1.8632008
11	13313	1.8678381	1.8638844	1.8634728	1.8632195
12	28673	1.8655343	1.8639148	1.8638087	1.8637460
13	61441	1.8644735	1.8638533	1.8638094	1.8637472
14	106497	1.8640681	1.8639032	1.8638934	1.8638778
15	188417	1.8639412	1.8639003	1.8638934	1.8638778
16	335873	1.8639259	1.8639160	1.8639144	1.8639104

Table : Two electrons: the smallest eigenvalue should lie in [1.8639144,1.8639259] with a reletive error of 3.1×10^{-6}.

q	dof	Memory	$L = 0.5$	$L = 0.6$	$L = 0.75$	$L = 1$
6	729		14.19322	10.62023	7.14268	4.13509
7	3645	1M	11.53182	8.98147	6.56414	4.57501
8	14337	7M	9.84225	8.20117	6.84163	6.28908
9	49761	30M	9.13927	8.08522	7.40965	7.55470
10	159489	158M	8.81094	8.15501	7.88803	8.19936
11	483201	879M	8.65394	8.26734	8.21052	8.36132
12	1403137	5163M	8.57456	8.35870	8.37154	8.44094
13	3940609	16689M	8.52448	8.42065	8.44336	8.46657

Table : Six electrons: the smallest eigenvalue should lie in [8.46657, 8.52448] with a reletive error of 3.4×10^{-3}.

q	dof	memory	$L = 0.5$	$L = 0.6$	$L = 0.75$	$L = 1$
8	6561	2M	19.011	14.176	9.525	5.514
9	41553	30M	16.487	12.652	9.076	6.059
10	193185	312M	14.387	11.764	9.591	8.622
11	768609	1621M	13.428	11.723	10.682	10.210
12	2772225	16587M	13.054	11.995	11.601	11.339

Table : Eight electrons: the smallest eigenvalue should lie in [11.339, 13.054] with a relative error of 7.6×10^{-2}.

N	q	DoF	δ	n_1	n_2	N	q	DoF	δ	n_1	n_2
1						4					
	4	17	0.79	7	10		7	2769	3.84	9	41
	5	33	0.80	7	12		8	7681	4.27	9	44
	6	65	0.80	7	13		9	20481	4.58	7	51
	7	129	0.80	7	14		10	52933	4.64	7	56
	8	257	0.80	7	15		11	133889	4.70	5	69
	9	513	0.80	7	16		12	331777	4.70	5	74
	10	1025	0.80	7	17		13	808961	4.70	5	87
2						6					
	5	113	1.72	9	31		7	3645	4.13	9	37
	6	257	1.79	9	31		8	14337	4.57	21	49
	7	577	1.84	7	33		9	49761	6.28	23	80
	8	1281	1.84	7	36		10	159489	7.55	13	110
	9	2817	1.84	7	36		11	483201	8.19	9	128
	10	6145	1.84	9	37		12	1403137	8.36	7	141
	11	13313	1.84	9	37		13	3940609	8.42	7	148

Table : Numbers of iterations used in ARPack and BICGSTAB.

Adaptive sparse tensor products using multi-wavelets

- The one domain approach is not suitable for problems with local features, such as those in many challenging applications. For this type of problems, an efficient adaptive procedure, e.g., wavelets, is needed.

Adaptive sparse tensor products using multi-wavelets

- The one domain approach is not suitable for problems with local features, such as those in many challenging applications. For this type of problems, an efficient adaptive procedure, e.g., wavelets, is needed.
- Standard wavelets are generated by a single function: the order of accuracy is low and fixed.

Adaptive sparse tensor products using multi-wavelets

- The one domain approach is not suitable for problems with local features, such as those in many challenging applications. For this type of problems, an efficient adaptive procedure, e.g., wavelets, is needed.
- Standard wavelets are generated by a single function: the order of accuracy is low and fixed.
- We propose to use the Legendre multi-wavelets (Alpert '93), which allow spectral accuracy, and we propose to use a sparse tensor product with these multi-wavelets so that it is feasible for higher-dimensional problems.

Legendre multi-wavelets in $L^2(-1,1)$

Let $\phi_j(x)$ be the Legendre polynomial of degree j and $V_0^k = \text{span}\{\phi_j(x)\} : 0 \leq j \leq k-1\}$. For $n \geq 1$, we define V_n^k as the space of piece-wise polynomials of degree less than k on a regular mesh with $h = 2^{1-n}$. Then, a set of basis function for V_n^k is given by

$$\phi_{jl}^n(x) = 2^{\frac{n}{2}}\phi_j(2^n x - l), \quad j = 0, \ldots, k-1, \quad l = 0, \ldots, 2^n - 1.$$

Legendre multi-wavelets in $L^2(-1,1)$

Let $\phi_j(x)$ be the Legendre polynomial of degree j and $V_0^k = \text{span}\{\phi_j(x)\} : 0 \le j \le k-1\}$. For $n \ge 1$, we define V_n^k as the space of piece-wise polynomials of degree less than k on a regular mesh with $h = 2^{1-n}$. Then, a set of basis function for V_n^k is given by

$$\phi_{jl}^n(x) = 2^{\frac{n}{2}}\phi_j(2^n x - l), \quad j = 0, \dots, k-1, \quad l = 0, \dots, 2^n - 1.$$

Let $W_0^k = V_0^k$ and define the multi-wavelet subspace W_n^k, $n = 1, 2, \dots$ as the L^2-orthogonal compliment of V_{n-1}^k in V_n^k, i.e.,

$$V_{n-1}^k \oplus W_n^k = V_k^n.$$

Then, we have

$$V_n^k = W_0^k \oplus W_1^k \oplus \dots \oplus W_n^k.$$

let $\psi_0, \ldots, \psi_{k-1}$ be the orthogonal basis for W_1^k. Then, W_n^k ($n \geq 1$) is spanned by

$$\psi_{jl}^n(x) = 2^{\frac{n}{2}} \psi_j(2^n x - l), \quad j = 0, \ldots, k-1, \quad l = 0, \ldots, 2^n - 1.$$

let $\psi_0, \ldots, \psi_{k-1}$ be the orthogonal basis for W_1^k. Then, W_n^k ($n \geq 1$) is spanned by

$$\psi_{jl}^n(x) = 2^{\frac{n}{2}} \psi_j(2^n x - l), \quad j = 0, \ldots, k-1, \quad l = 0, \ldots, 2^n - 1.$$

We have

$$\int_{-1}^{1} \psi_{il}^n(x) \psi_{i'l'}^{n'}(x)\, dx = \delta_{ii'} \delta_{ll'} \delta_{nn'},$$

and they form a complete orthonormal basis in $L^2(-1, 1)$.

Sparse tensor products in $(-1,1)^d$

Let $\Omega = (-1,1)^d$.

- The regular tensor product approximation space

$$\mathcal{V}_L^k = V_L^k \oplus \cdots \oplus V_L^k,$$

which can be written as

$$\mathcal{V}_L^k = \sum_{0 \le l_i \le L} W_{l_1}^k \oplus W_{l_2}^k \cdots \oplus W_{l_d}^k.$$

However, $\dim(\mathcal{V}_L^k) = O(2^{Ld})$ which is not feasible for d large.

Sparse tensor products in $(-1,1)^d$

Let $\Omega = (-1,1)^d$.

- The regular tensor product approximation space

$$\mathcal{V}_L^k = V_L^k \oplus \cdots \oplus V_L^k,$$

which can be written as

$$\mathcal{V}_L^k = \sum_{0 \leq l_i \leq L} W_{l_1}^k \oplus W_{l_2}^k \cdots \oplus W_{l_d}^k.$$

However, $\dim(\mathcal{V}_L^k) = O(2^{Ld})$ which is not feasible for d large.

- The sparse tensor product approximation space

$$\hat{\mathcal{V}}_L^k = \sum_{0 \leq l_1 + \cdots + l_d \leq L} W_{l_1}^k \oplus W_{l_2}^k \cdots \oplus W_{l_d}^k,$$

with $\dim(\hat{\mathcal{V}}_L^k) = O(L^{d-1}2^L)$.

Galerkin approximation of the collision operator with sparse multi-wavelets

Let $L = (l_1, \cdots, l_d)$ with $0 \leq l_1 + \cdots + l_d \leq L$. Then $\phi_L^k \in \hat{\mathcal{V}}_L^k$, and the inner product of the collision operator with $\hat{\mathcal{V}}_L^k$ is

$$
\begin{aligned}
I(\phi_L^k) &= \int_{R^3} \phi_L^k(v) dv \int_{R^3} \int_{S^2} [f(x,v')f(x,w') - f(x,v)f(x,w)]|v-w|dS \\
&= \int_{R^3} \int_{R^3} f(x,v)f(x,w)dvdw \\
&\quad \frac{|v-w|}{2} \int_{S^2} [\phi_L^k(v') + \phi_L^k(w') - \phi_L^k(v) - \phi_L^k(w)]dS \\
&= \int_{R^3} \int_{R^3} f(x,v)f(x,w)A(v,w;\phi_L^k)dvdw,
\end{aligned}
$$

where

$$
A(v,w;\phi_L^k) = \frac{|v-w|}{2} \int_{S^2} [\phi_L^k(v') + \phi_L^k(w') - \phi_L^k(v) - \phi_L^k(w)]dS.
$$

Proposed algorithm

- In practice, we can replace the sparse Legendre multi-wavelets by corresponding Lagrangian basis functions on a sparse grid.
- Given a sparse grid for v and w, we pre-compute $A(v, w; \phi_L^k)$ for all $\phi_L^k \in \hat{\mathcal{V}}_L^k$.
- The total # of entries in this trilinear tensor is of order $(L^{d-1}2^L)^3$. However, due to the local feature of ϕ_L^k, most of $A(v, w; \phi_L^k)$ are zeros.
- We are in the process of working out the detail, and hopefully this will be a feasible alternative approach for the collision operator.

Concluding remarks

- Fast sparse spectral algorithms for elliptic equations in high-dimensional bounded and unbounded domains
    - fast transforms between mapped Chebyshev sparse grids and corresponding “hyperbolic cross” space;
    - fast solvers for the resulting linear system.

Concluding remarks

- Fast sparse spectral algorithms for elliptic equations in high-dimensional bounded and unbounded domains
  - fast transforms between mapped Chebyshev sparse grids and corresponding "hyperbolic cross" space;
  - fast solvers for the resulting linear system.
- Preliminary results to electronic Schrödinger equation indicate that this approach is promising for solving moderately high-dimensional PDEs.

Concluding remarks

- Fast sparse spectral algorithms for elliptic equations in high-dimensional bounded and unbounded domains
  - fast transforms between mapped Chebyshev sparse grids and corresponding “hyperbolic cross” space;
  - fast solvers for the resulting linear system.
- Preliminary results to electronic Schrödinger equation indicate that this approach is promising for solving moderately high-dimensional PDEs.
- A Galerkin sparse tensor product approach with multi-wavelets is proposed for dealing with the collision operator.

Thank You!