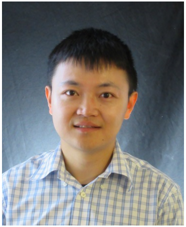
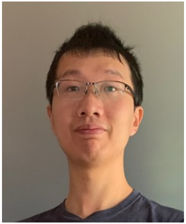
Sparse Grid Discontinuous Galerkin (DG) Methods for High Dimensional PDEs

Yingda Cheng

Virginia Tech

Dahlquist Prize Lecture, SciCADE 2024

Acknowledgement

- We would like to thank Wei Guo (Texas Tech), Juntao Huang (Texas Tech) for their significant contributions to this work.
- We also like thank: Zixuan Wang, Qi Tang, Zhanjing Tao, Yuan Liu, Yan Jiang *et al.*
- We thank NSF, DOE and Simons Foundation for partially funding this research.

Outline

Motivation

- We are interested in computing solutions to a class of **high dimensional transport** PDEs.
- Examples include: high dimensional kinetic transport problem (Vlasov, Boltzmann) in plasma, high dimensional Hamilton-Jacobi equations.
- Conventional mesh based numerical solvers runs into the *curse of dimensionality*. DOF scales like $O(N^d)$, fixed order error is $O(N^{-k})$, therefore error behaves like $O(DOF^{-k/d})$. **No storage, No accuracy!**
- In recent years, approaches have been developed, e.g. machine learning, tensor based approaches etc. extracting low dimensional underlying structure.
- This talk will focus on sparse grid compression techniques.

Sparse grid: a tool to break the curse of dimensionality

- Sparse grid method was introduced by Smolyak (63) for high dimensional quadrature, and is widely used for uncertainty quantification Xiu, Hesthaven (05...).
- Sparse grid PDE solver: Zenger (91), Griebel (91,98,05...), Bungartz, Griebel (04) . Most work focus on continuous FEM, and spectral methods.

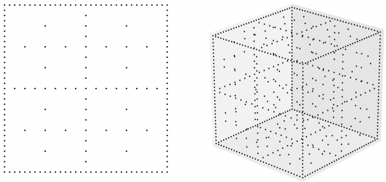


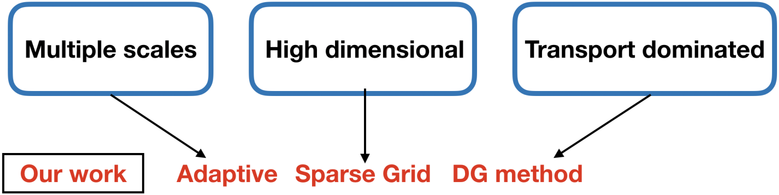
Fig. 5: Two-dimensional sparse grid (left) and three-dimensional sparse grid (right) of level $n = 5$.

Figure: From Garcke, SG in a nutshell

Discontinuous Galerkin (DG) FEM

- DG is a type of finite element method using discontinuous polynomial as underlying approximating space.
- Invented by by Reed and Hill (73). Runge-Kutta discontinuous Galerkin (RKDG) method for hyperbolic conservation laws was developed in Cockburn and Shu (89, 90,...). Review paper see Arnold, Cockburn, Brezzi, Marini (02).
- For convection dominated problems, DG method can offer many advantages. In particular, its flexible framework is friendly to sparse grid.

Our approach



<u>Outline</u>

- **Sparse grid DG** method use multiwavelet basis and the DG weak form as building blocks.
- **Adaptive sparse grid DG** method perform thresholding based on hierarchical coefficients.
- **Nonlinear equations** use interpolatory multiwavelets with stabilization mechanisms.

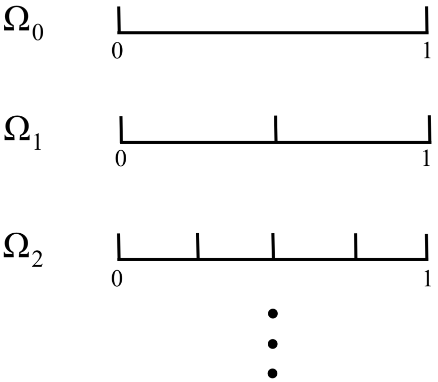
Outline

Outline

The mesh

Consider $\Omega = [0,1]$ and define n-th level grid

$$\Omega_n = \{I_n^j = (2^{-n}j, 2^{-n}(j+1)], j = 0, \ldots, 2^n - 1\}$$



Multiresolution analysis

Conventional approximation space on the n-th level grid Ω_n

$$V_n^k = \{v : v \in P^k(I_n^j), \forall j = 0, \ldots, 2^n - 1\}$$

$$dim(V_n^k) = 2^n(k+1)$$

Nested structure

$$V_0^k \subset V_1^k \subset V_2^k \subset V_3^k \subset \cdots$$

W_n^k: orthogonal complement of V_{n-1}^k in V_n^k, for $n > 1$, represents the finer level details when the mesh is refined, satisfying

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

$$W_n^k \perp V_{n-1}^k$$

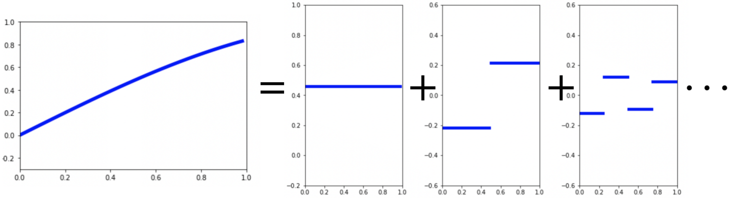
Let $W_0^k := V_0^k$, then

$$V_N^k = \bigoplus_{0 \leq n \leq N} W_n^k$$

Illustration

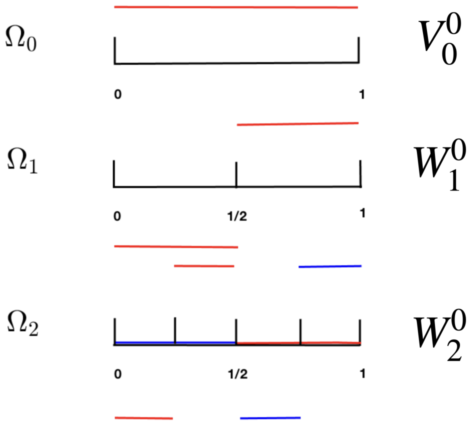
Let P_n^k denotes the L^2 projection on to mesh level n, then

$$P_N^k f = \underbrace{P_0^k f}_{V_0^k} + \underbrace{(P_1^k - P_0^k) f}_{W_1^k} + \underbrace{(P_2^k - P_1^k) f}_{W_2^k} + \cdots + \underbrace{(P_N^k - P_{N-1}^k) f}_{W_N^k}$$



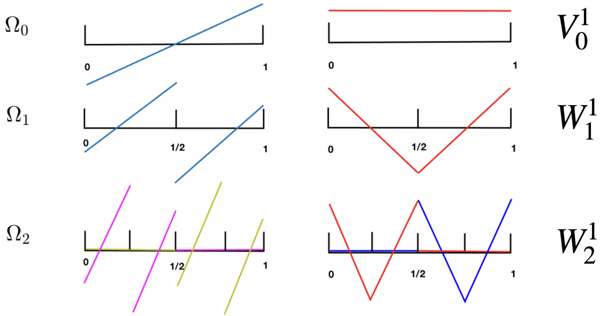
V_0^0 W_1^0 W_2^0

Bases on different levels for $k = 0$



Higher order

For $k = 1$



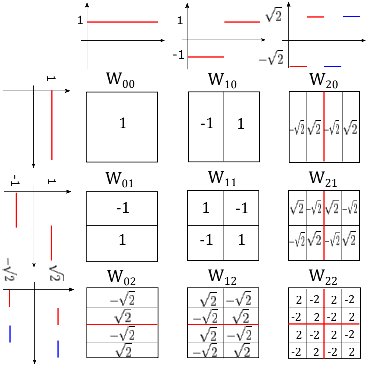
Arbitrary order k : we use L^2 orthogonal multiwavelets by Alpert (93).

Higher-D: full grid approximation space

Full grid space:

$$\mathbf{V}_N^k = \bigoplus_{|\mathbf{l}|_\infty \leq N} \mathbf{W}_\mathbf{l}^k$$

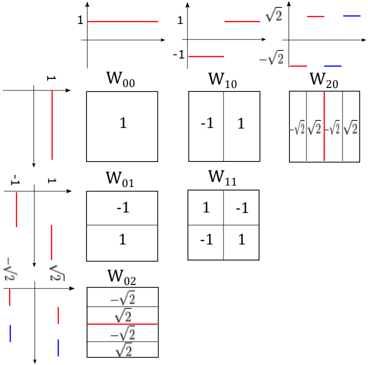
$d = 2, \ N = 2, \ k = 0$



$$dim(\mathbf{V}_N^k) = 2^{Nd}(k+1)^d \quad \text{or} \quad O(h^{-d})$$

Sparse grid approximation space

We consider the sparse grid space: $\hat{\mathbf{V}}_N^k := \bigoplus_{|\mathbf{l}|_1 \leq N} \mathbf{W}_\mathbf{l}^k$



$$dim(\hat{\mathbf{V}}_N^k) = O(2^N N^{d-1}(k+1)^d) \quad \text{or} \quad O(h^{-1}|\log_2 h|^{d-1})$$

Sparse grid DG

Consider the linear transport equation with variable coefficient

$$\begin{cases} u_t + \nabla \cdot (\boldsymbol{\alpha}(\mathbf{x}, t)\, u) = 0, & \mathbf{x} \in \Omega = [0,1]^d, \\ u(0, \mathbf{x}) = u_0(\mathbf{x}), \end{cases} \tag{1}$$

The semi-discrete sparse grid DG [1] formulation for (1) is defined as follows: find $u_h \in \hat{\mathbf{V}}_N^k$, such that

$$\int_\Omega (u_h)_t\, v_h\, d\mathbf{x} = \int_\Omega u_h \boldsymbol{\alpha} \cdot \nabla v_h\, d\mathbf{x} - \sum_{e \in \Gamma} \int_e \widehat{\boldsymbol{\alpha} u_h} \cdot [v_h]\, ds, \tag{2}$$

for $\forall\, v_h \in \hat{\mathbf{V}}_N^k$, where $\widehat{\boldsymbol{\alpha} u_h}$ defined on the element interface denotes a monotone numerical flux.

[1] Guo, Cheng, SISC, 2016

Stability (constant coefficient case)

Theorem (Guo, Cheng, SISC, 2016)

The DG scheme (2) for (1) is L^2 stable when $\boldsymbol{\alpha}$ is a constant vector, i.e.

$$\frac{d}{dt}\int_\Omega (u_h)^2\, d\mathbf{x} = -\sum_{e\in\Gamma}\int_e \frac{|\boldsymbol{\alpha}\cdot\mathbf{n}|}{2}|[u_h]|^2 ds \le 0. \qquad (3)$$

Error estimate (constant coefficient case)

Inspired by Schwab, Suli, Todor (08), we obtain convergence results.

Theorem (Guo, Cheng, SISC, 2016)

Let u be the exact solution, and u_h be the numerical solution to the semi-discrete scheme (2) with numerical initial condition $u_h(0) = \mathbf{P}u_0$. For $k \geq 1$, $u_0 \in \mathcal{H}^{p+1}(\Omega)$, $1 \leq q \leq \min\{p, k\}$, $N \geq 1$, $d \geq 2$, we have for all $t \geq 0$,

$$\|u_h - u\|_{L^2(\Omega_N)} \leq \left(2\sqrt{C_d\|\boldsymbol{\alpha}\|_2 t}\, C_\star(k,q,d,N) + (\bar{\bar{c}}_{k,0,q} + B_0(k,q,d)\kappa_0(k,q,N)^d)2^{-N/2}\right) 2^{-N(q+1/2)}|u_0|_{\mathcal{H}^{q+1}(\Omega)},$$

where C_d is a generic constant with dependence only on d,
$C_\star(k,q,d,N) = \max_{s=0,1}\left(\bar{\bar{c}}_{k,s,q} + B_s(k,q,d)\kappa_s(k,q,N)^d\right)$. *The constants $\bar{\bar{c}}_{k,s,q}$, $B_s(k,q,d)$, $\kappa_s(k,q,N)$ are defined in L^2 projection error estimates.*

Convergence rate $O((\log h)^d h^{k+1/2})$.

Numerical result

We consider the following linear advection problem

$$\begin{cases} u_t + \sum_{m=1}^{d} u_{x_m} = 0, \quad \mathbf{x} \in [0,1]^d, \\ u(0,\mathbf{x}) = \sin\left(2\pi \sum_{m=1}^{d} x_m\right), \end{cases} \tag{4}$$

subject to periodic boundary conditions.

Error and DOF

N	h_N	DOF	L^2 error	order	DOF	L^2 error	order
		$k=1,\ d=3$			$k=1,\ d=4$		
4	1/16	832	3.72E-01	–	3072	4.99E-01	–
5	1/32	2176	1.19E-01	1.64	8832	2.40E-01	1.06
6	1/64	5504	2.96E-02	2.01	24320	9.84E-02	1.28
7	1/128	13568	8.85E-03	1.74	64768	3.21E-02	1.62
		$k=2,\ d=3$			$k=2,\ d=4$		
4	1/16	2808	1.10E-02	–	15552	2.80E-02	–
5	1/32	7344	1.79E-03	2.63	44712	5.82E-03	2.27
6	1/64	18576	3.97E-04	2.17	123120	1.37E-03	2.09
7	1/128	45792	5.14E-05	2.95	327888	2.58E-04	2.41

Table: L^2 errors and orders of accuracy, DOF.

FG: 56Million **FG: 21Billion**

Outline

Adaptivity

- Sparse grid has poor resolution when function is not smooth.
- We need to use adaptivity. The idea is to threshold based on the hierarchical coefficients, like MRA for image processing.
- Note: when the solution is regular, adaptive sparse grid will fall back to standard sparse grid, retaining its advantage for high dimensional problems. This is different from traditional $h-$adaptive method.
- Closely related: adaptive wavelet methods Dahmen (97), Cohen (00), multiresolution finite difference/finite volume methods for hyperbolic PDEs. Harten (95).., Adaptive multiresolution DG schemes Calle et al. (2005), Archibald et al. (2011), Hovhannisyan et al. (2014), Gerhard et al. (2015)

Predict ⟶ **Refine** ⟶ **Evolve** ⟶ **Coarsen**

Refinement criteria

For a function $u(\mathbf{x}) \in \mathcal{H}^{p+1}(\Omega)$, we can show that $u(\mathbf{x}) = \sum_{\mathbf{l}\in\mathbb{N}_0^d} \sum_{\mathbf{j}\in B_{\mathbf{l}}, \mathbf{1}\le\mathbf{i}\le\mathbf{k}+\mathbf{1}} u_{\mathbf{i},\mathbf{l}}^{\mathbf{j}} v_{\mathbf{i},\mathbf{l}}^{\mathbf{j}}(\mathbf{x})$, where the hierarchical coefficient is $u_{\mathbf{i},\mathbf{l}}^{\mathbf{j}} = \int_\Omega u(\mathbf{x}) v_{\mathbf{i},\mathbf{l}}^{\mathbf{j}}(\mathbf{x}) d\mathbf{x}$.

An element $V_{\mathbf{l}}^{\mathbf{j}} := \{v_{\mathbf{i},\mathbf{l}}^{\mathbf{j}}, \mathbf{1} \le \mathbf{i} \le \mathbf{k}+\mathbf{1}\}$ is considered important if

$$\left(\sum_{\mathbf{1}\le\mathbf{i}\le\mathbf{k}+\mathbf{1}} |u_{\mathbf{i},\mathbf{l}}^{\mathbf{j}}|^2\right)^{\frac{1}{2}} > \varepsilon, \tag{5}$$

where ε is a prescribed error threshold.

A similar coarsening criteria can be defined.

Adaptive evolution algorithm [2]

Input: Hash table H and leaf table L at t^n, numerical solution $u_h^n \in \mathbf{V}_{N,H}^k$.
Parameters: Maximum level N, polynomial degree k, error constants ε, η, CFL constant.
Output: Hash table H and leaf table L at t^{n+1}, numerical solution $u_h^{n+1} \in \mathbf{V}_{N,H}^k$.

- **Prediction.** Given a hash table H that stores the numerical solution u_h at time step t^n, calculate Δt. Predict the solution by the DG scheme using space $\mathbf{V}_{N,H}^k$ and the forward Euler time stepping method. Generate the predicted solution $u_h^{(p)}$.

[2] Guo, Cheng, SISC, 2017

Adaptive evolution algorithm

- **Refinement.** Based on $u_h^{(p)}$, screen all elements. Add children of elements according to refinement criteria. This step generates the updated hash table $H^{(p)}$ and leaf table $L^{(p)}$.
- **Evolution.** Evolve the solution from t^n to t^{n+1} by the DG scheme using space $\mathbf{V}^k_{N,H^{(p)}}$ and the third order Runge-Kutta time stepping method. This step generates the pre-coarsened numerical solution $\tilde{u}_h^{n+1}$.
- **Coarsening.** Coarsen $\tilde{u}_h^{n+1}$ according to the coarsening criteria.

Example: Vlasov-Maxwell simulation

One-species Vlasov-Maxwell is a fundamental model in plasma

$$\partial_t f + \xi \cdot \nabla_{\mathbf{x}} f + (\mathbf{E} + \xi \times \mathbf{B}) \cdot \nabla_{\xi} f = 0 \ , \tag{6a}$$

$$\frac{\partial \mathbf{E}}{\partial t} = \nabla_{\mathbf{x}} \times \mathbf{B} - \mathbf{J}, \qquad \frac{\partial \mathbf{B}}{\partial t} = -\nabla_{\mathbf{x}} \times \mathbf{E} \ , \tag{6b}$$

$$\nabla_{\mathbf{x}} \cdot \mathbf{E} = \rho - \rho_i, \qquad \nabla_{\mathbf{x}} \cdot \mathbf{B} = 0 \ , \tag{6c}$$

with

$$\rho(\mathbf{x}, t) = \int_{\Omega_\xi} f(\mathbf{x}, \xi, t) d\xi, \qquad \mathbf{J}(\mathbf{x}, t) = \int_{\Omega_\xi} f(\mathbf{x}, \xi, t) \xi d\xi \ .$$

Numerical scheme

The semi-discrete DG methods for the VM system are: to find $f_h \in \hat{\mathcal{G}}_h^k$, $\mathbf{E}_h, \mathbf{B}_h \in \hat{\mathcal{U}}_h^k$, such that for any $g \in \hat{\mathcal{G}}_h^k$, $\mathbf{U}, \mathbf{V} \in \hat{\mathcal{U}}_h^k$, [3]

$$\begin{aligned}
&\int_\Omega \partial_t f_h g d\mathbf{x} d\xi - \int_\Omega f_h \xi \cdot \nabla_\mathbf{x} g d\mathbf{x} d\xi - \int_\Omega f_h (\mathbf{E}_h + \xi \times \mathbf{B}_h) \cdot \nabla_\xi g d\mathbf{x} d\xi \\
&\quad + \int_{\Omega_\xi} \int_{\mathcal{E}_x} \widehat{f_h \xi} \cdot [g]_x ds_x d\xi + \int_{\Omega_x} \int_{\mathcal{E}_\xi} \widehat{f_h (\mathbf{E}_h + \xi \times \mathbf{B}_h)} \cdot [g]_\xi ds_\xi dx = 0 \,, \qquad (7a)
\end{aligned}$$

$$\int_{\Omega_x} \partial_t \mathbf{E}_h \cdot \mathbf{U} dx = \int_{\Omega_x} \mathbf{B}_h \cdot \nabla_\mathbf{x} \times \mathbf{U} dx + \int_{\mathcal{E}_x} \widehat{\mathbf{B}_h} \cdot [\mathbf{U}]_\tau ds_x - \int_{\Omega_x} \mathbf{J}_h \cdot \mathbf{U} dx \,, \qquad (7b)$$

$$\int_{\Omega_x} \partial_t \mathbf{B}_h \cdot \mathbf{V} dx = - \int_{\Omega_x} \mathbf{E}_h \cdot \nabla_\mathbf{x} \times \mathbf{V} dx - \int_{\mathcal{E}_x} \widehat{\mathbf{E}_h} \cdot [\mathbf{V}]_\tau ds_x \,, \qquad (7c)$$

with

$$\mathbf{J}_h(\mathbf{x}, t) = \int_{\Omega_\xi} f_h(\mathbf{x}, \xi, t) \xi d\xi \ \in \hat{\mathcal{U}}_h^k.$$

For the Vlasov part, we adopt the global Lax-Friedrichs flux. For the Maxwell part, we use the upwind flux or the alternating flux.

[3] Cheng, Gamba, Li, Morrison, SINUM 2014, Tao, Guo, Cheng, JCP, 2019

Properties

Theorem (Mass conservation)

The numerical solution $f_h \in \hat{\mathcal{G}}_h^k$ with $k \geq 0$ satisfies

$$\frac{d}{dt}\int_\Omega f_h d\mathbf{x} d\xi + \Theta_{h,1}(t) = 0 \,, \tag{8}$$

where $\Theta_{h,1}(t) = \int_{\Omega_x}\int_{\mathcal{E}_\xi^b} f_h \max((\mathbf{E}_h + \xi \times \mathbf{B}_h)\cdot \mathbf{n}_\xi, 0) ds_\xi d\mathbf{x}$.

Theorem (Energy conservation)

For $k \geq 2$, the numerical solution $f_h \in \hat{\mathcal{G}}_h^k$, $\mathbf{E}_h, \mathbf{B}_h \in \hat{\mathcal{U}}_h^k$ with the upwind numerical fluxes for the Maxwell part satisfies

$$\frac{d}{dt}\left(\int_\Omega f_h|\xi|^2 d\mathbf{x} d\xi + \int_{\Omega_x}(|\mathbf{E}_h|^2 + |\mathbf{B}_h|^2) d\mathbf{x}\right) + \Theta_{h,2}(t) + \Theta_{h,3}(t) = 0 \,,$$

with

$$\Theta_{h,2}(t) = \int_{\mathcal{E}_x}\left(|[\mathbf{E}_h]_\tau|^2 + |[\mathbf{B}_h]_\tau|^2\right) ds_x \,, \qquad \Theta_{h,3}(t) = \int_{\Omega_x}\int_{\mathcal{E}_\xi^b} f_h|\xi|^2 \max((\mathbf{E}_h + \xi \times \mathbf{B}_h)\cdot \mathbf{n}_\xi, 0) ds_\xi d\mathbf{x} \,.$$

While for the scheme with alternating flux for the Maxwell part, we have

$$\frac{d}{dt}\left(\int_\Omega f_h|\xi|^2 d\mathbf{x} d\xi + \int_{\mathcal{T}_h^x}(|\mathbf{E}_h|^2 + |\mathbf{B}_h|^2) d\mathbf{x}\right) + \Theta_{h,3}(t) = 0 \,.$$

Theorem (L^2-stability of f_h)

For $k \geq 0$, the numerical solution $f_h \in \hat{\mathcal{G}}_h^k$ satisfies

$$\frac{d}{dt}\left(\int_\Omega |f_h|^2 d\mathbf{x} d\xi\right) \leq 0 \,.$$

Streaming Weibel instability

We consider 1D2V problem

$$f_t + \xi_2 f_{x_2} + (E_1 + \xi_2 B_3) f_{\xi_1} + (E_2 - \xi_1 B_3) f_{\xi_2} = 0 \ , \tag{9}$$

$$\frac{\partial B_3}{\partial t} = \frac{\partial E_1}{\partial x_2}, \quad \frac{\partial E_1}{\partial t} = \frac{\partial B_3}{\partial x_2} - j_1, \quad \frac{\partial E_2}{\partial t} = -j_2 \ , \tag{10}$$

The initial conditions are given by

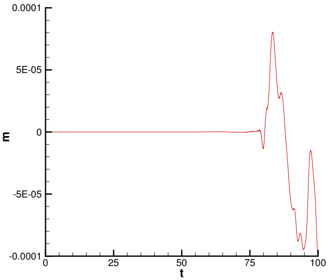
$$f(x_2, \xi_1, \xi_2, 0) = \frac{1}{\pi\beta} e^{-\xi_2^2/\beta} [\delta e^{-(\xi_1 - v_{0,1})^2/\beta} + (1-\delta) e^{-(\xi_1 + v_{0,2})^2/\beta}], \tag{11}$$

$$E_1(x_2, \xi_1, \xi_2, 0) = E_2(x_2, \xi_1, \xi_2, 0) = 0, \qquad B_3(x_2, \xi_1, \xi_2, 0) = b \sin(k_0 x_2) \ , \tag{12}$$

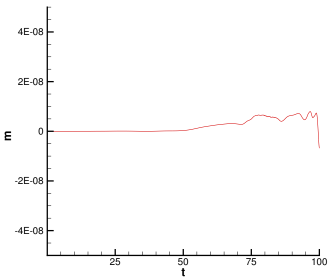
where $b = 0$ is an equilibrium state composed of counter-streaming beams propagating perpendicular to the direction of inhomogeneity, $\beta^{1/2}$ is the thermal velocity and δ is a parameter measuring the symmetry of the electron beams. $\beta = 0.01$, $b = 0.001$ Here, $\Omega_x = [0, L_y]$, where $L_y = 2\pi/k_0$, and we set $\Omega_\xi = [-1.2, 1.2]^2$. $\delta = 0.5$, $v_{0,1} = v_{0,2} = 0.3$, $k_0 = 0.2$.

Mass conservation

We compare the sparse grid (SG) DG ($N = 8, k = 3$) with adaptive sparse grid (ASG) DG scheme ($N = 6, k = 3, \epsilon = 2 \times 10^{-7}$).

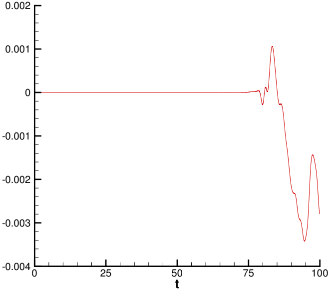


(a) Relative error in mass: SG

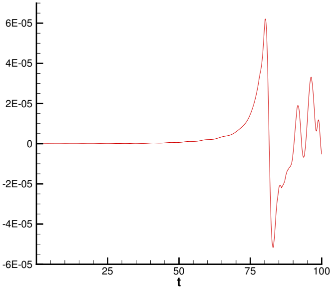


(b) Relative error in mass: ASG

Energy conservation

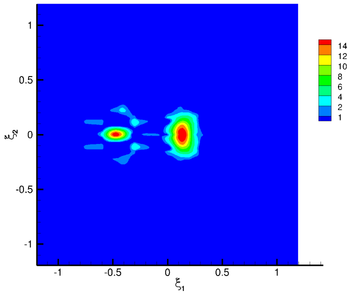


(c) Relative error in energy: SG

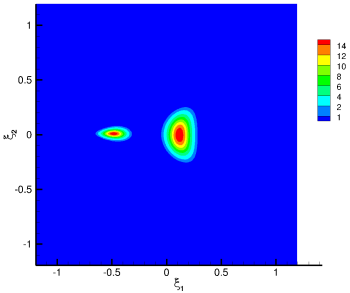


(d) Relative error in energy: ASG

Contour plots

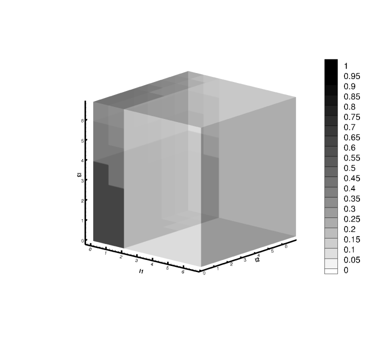


(e) Contour at $x_2 = 0.05\pi, t = 82$: SG

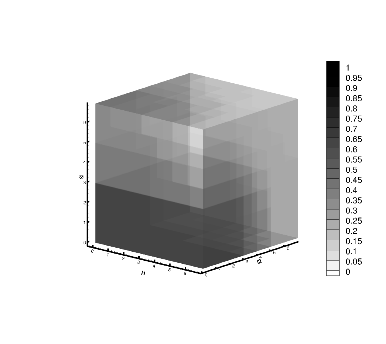


(f) Contour at $x_2 = 0.05\pi, t = 82$: ASG

Percent of active elements by ASG

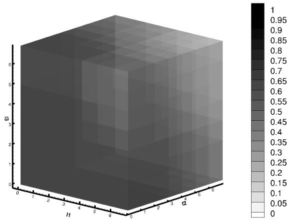


(g) $t = 0$. Active elements: 0.73%

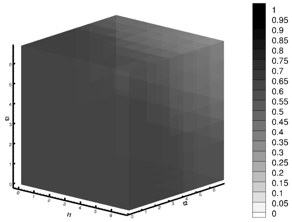


(h) $t = 55$. Active elements: 4.36%

Percent of active elements by ASG



(i) $t = 82$. Active elements: 26.55%



(j) $t = 100$. Active elements: 52.41%

Outline

Nonlinear problems

- Example: nonlinear source term $f(u)$ requires evaluating terms

$$\int_{\Omega} f(u_h) v_h dx = \sum_K \int_K f(u_h) v_h dx,$$

  where u_h is represented by multiwavelet basis functions.
- We cannot afford to sum up on all elementary cells K as this requires $O(h^{-d})$ operations.

Nonlinear problems

- Example: nonlinear source term $f(u)$ requires evaluating terms

$$\int_\Omega f(u_h)v_h dx = \sum_K \int_K f(u_h)v_h dx,$$

where u_h is represented by multiwavelet basis functions.

- We cannot afford to sum up on all elementary cells K as this requires $O(h^{-d})$ operations.
- The idea is to switch to nodal basis and evaluate

$$\int_\Omega \mathcal{I} f(u_h)v_h dx,$$

where the interpolation operator $\mathcal{I}$ is consistent with the adaptive structure of the solution. We will demonstrate the 1D construction.

1D: nested points

Consider the domain $I = [0, 1]$, we use the same notation.In addition, we define $k + 1$ distinct points on each cell

$$x^j_{i,n} = 2^{-n} j + 2^{-n} \alpha_i \tag{13}$$

with $\alpha_i \in [0, 1]$, $i = 1, \dots, k + 1$.
In particular, the collection of those points $X^k_n = \{x^j_{i,n}\}$ is called *nested points*, if

$$X^k_0 \subset X^k_1 \subset X^k_2 \subset \cdots . \tag{14}$$

1D

Since $\{X_n^k\}$ are nested, the points can be rearranged in such a way that

$$X_n^k = X_0^k \cup \widetilde{X}_1^k \cup \cdots \cup \widetilde{X}_n^k, \quad \text{with } \widetilde{X}_n^k = X_n^k / X_{n-1}^k. \tag{15}$$

Moreover, we can now define the subspace $\widetilde{W}_n^k$, $n \geq 1$, as the complement of V_{n-1}^k in V_n^k, in which the piecewise polynomials vanish at all points in X_{n-1}^k,

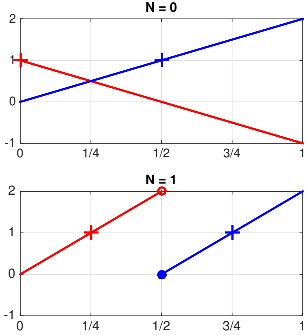
$$V_n^k = V_{n-1}^k \oplus \widetilde{W}_n^k. \tag{16}$$

This corresponds to

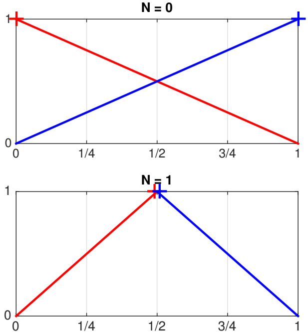
$$I_N^k f = \underbrace{I_0^k f}_{V_0^k} + \underbrace{(I_1^k - I_0^k) f}_{\widetilde{W}_1^k} + \underbrace{(I_2^k - I_1^k) f}_{\widetilde{W}_2^k} + \cdots + \underbrace{(I_N^k - I_{N-1}^k) f}_{\widetilde{W}_N^k}$$

[3] Tao, Jiang, Cheng JCP 2021

1D-Example



(k) P^1: $x_0 = 0$, $x_1 = 1/2$



(l) P^1: $x_0 = 0$, $x_1 = 1$

Figure: Interpolation points and multiwavelets: P^1.

[3] Tao, Jiang, Cheng JCP 2021

Fast methods

- Fast methods are critical to an efficient implementation.
- In Shen, Yu (10, 12), they developed fast method associated with sparse grid.

Fact

$$f_{n_1,n_2} = \sum_{0\le n_1'+n_2'\le N} f'_{n_1',n_2'} t^{(1)}_{n_1',n_1} t^{(2)}_{n_2',n_2}, \quad 0 \le n_1 + n_2 \le N. \tag{17}$$

is equivalent to

$$g_{n_1,n_2'} = \sum_{0\le n_1'\le N-n_2'} f'_{n_1',n_2'} t^{(1)}_{n_1',n_1}, \quad 0 \le n_1 + n_2' \le N, \tag{18}$$

with

$$f_{n_1,n_2} = \sum_{0\le n_2'\le N-n_1} g_{n_1,n_2'} t^{(2)}_{n_2',n_2}, \quad 0 \le n_1 + n_2 \le N. \tag{19}$$

if $T^{(1)}$ is lower triangular or $T^{(2)}$ is upper triangular.

Fast methods

- Similar conclusion holds true for *adaptive sparse grid* when we replace the index set $n_1 + n_2 \leq N$ by a downward closed index set.
- This enables a fast implementation of linear operators in the solutions by splitting the transformation operator into lower and upper triangular parts.

Nonlinear conservation law

We consider nonlinear conservation law

$$u_t + \nabla \cdot f(u) = 0, \tag{20}$$

The semi-discrete DG formulation is

$$\int_\Omega (u_h)_t v_h dx - \int_\Omega \mathcal{I} f(u_h) \cdot \nabla v_h dx + \int_\Omega \mathcal{I} \hat{f}(u_h) \cdot n_K v_h ds = 0 \tag{21}$$

$\mathcal{I}$ is the Hermite interpolation operator with one order higher degree.

One higher order: for accuracy, Hermite: for stability.

Artificial viscosity

For capturing shock, we add artificial viscosity

$$\int_\Omega (u_h)_t v_h d\mathbf{x} - \int_\Omega \mathcal{I} f(u_h) \cdot \nabla v_h d\mathbf{x} + \int_\Omega \mathcal{I}\hat{f}(u_h) \cdot n_K v_h ds - \int_\Omega \nu(u_h) \nabla u_h \cdot \nabla v_h d\mathbf{x} = 0. \quad (22)$$

where $\nu = \nu(u_h) > 0$ is artificial viscosity depending on u_h. The artificial viscosity is only imposed in the leaf element and is determined in the following approach:

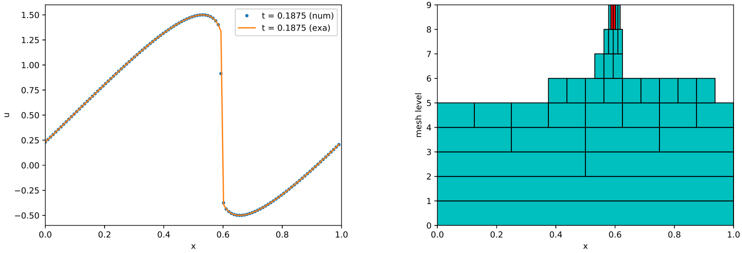
$$\nu = \begin{cases} 0, & \text{if} \quad s_e \leq s_0 + \kappa, \\ \nu_0 h, & \text{otherwise.} \end{cases}$$

where $\nu_0 > 0$ and κ are constants chosen empirically. In the computation, we typically take $\nu_0 = 1$ and $\kappa = 0$. The parameters s_e and s_0 are given as

$$s_e = \log_{10} \left(\sum_{\mathbf{1} \leq \mathbf{i} \leq \mathbf{k}+\mathbf{1}} |u^{\mathbf{j}}_{\mathbf{i},\mathbf{l}}|^2 \right)^{\frac{1}{2}}, \quad s_0 = \log_{10}(2^{-(k+1)|\mathbf{l}|_1}). \quad (23)$$

For smooth regions, s_e should be the same order as s_0. In the discontinuous regions, s_e should be much larger than s_0.

Numerical results: 1D Burgers' equation [4]



(a) solution (b) elements

Figure: $t = 0.1875$. $N = 8$ and $\epsilon = 10^{-4}$. $N = 9, k = 2, P^3$ Hermite interpolation. red: elements with artificial viscosity

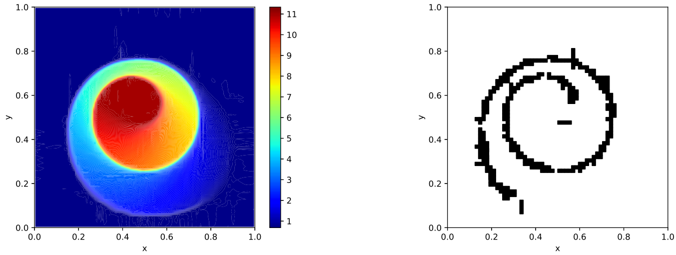
[4] Huang, Cheng, SISC, 2020

Numerical results: 2D KPP rotating wave problem [5]

$$u_t + \sin(u)_x + \cos(u)_y = 0.$$

The initial condition is

$$u_0(x,y) = \begin{cases} 3.5\pi, & (x-1/2)^2 + (y-1/2)^2 \le \frac{1}{16}, \\ 0.25\pi, & \text{otherwise.} \end{cases}$$



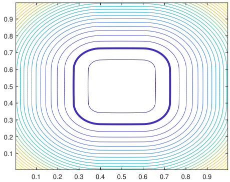
(a) solution

(b) elements with artificial viscosity

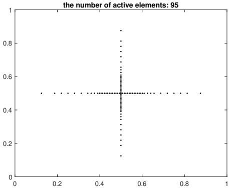
Hamilton-Jacobi equations [6]

We use local DG method by Yan, Osher JCP, 2011

$$\phi_t + \sum_{m=1}^{d} |\phi_{x_m}| = 0, \quad \mathbf{x} \in [0,1]^d, \quad d = 2, 3, 4$$



(a)

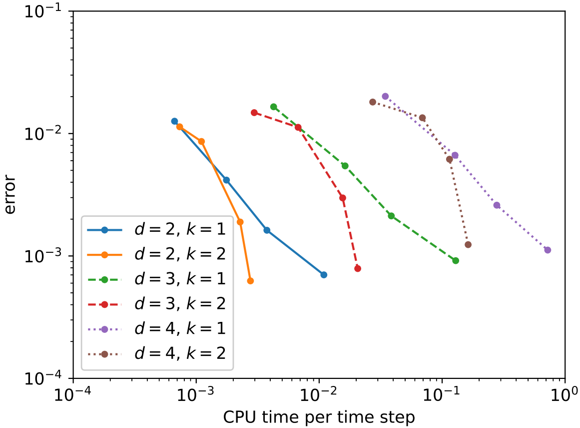


(b)

Figure: Example HJB. $T = 0.1$. $k = 2$, $M = 4$. $N = 7$. $\epsilon{=}10^{-7}$. (a) Contour plot of the numerical solution. (b) Active elements.

[6] Guo, Huang, Tao, Cheng, JCP, 2021

Error vs CPU: HJB



Outline

Conclusions

- **Main techniques**: DG weak form + adaptive sparse grid/multiwavelet compression+fast method.
- **Pros**: Inherit nice properties of DG, convergence estimate with high order (linear problem), natural adaptivity, can deal with transport dominated problem, nonlinear problem and boundary condition.

Conclusions

- **Main techniques**: DG weak form + adaptive sparse grid/multiwavelet compression+fast method.
- **Pros**: Inherit nice properties of DG, convergence estimate with high order (linear problem), natural adaptivity, can deal with transport dominated problem, nonlinear problem and boundary condition.
- **Cons**: Limitation in approximation (solution can be compressed in this format), unstructured mesh. Technicality in implementation.
- **Todo:** implicit solve, better time stepping, parallel.

Conclusions

- **Main techniques**: DG weak form + adaptive sparse grid/multiwavelet compression+fast method.
- **Pros**: Inherit nice properties of DG, convergence estimate with high order (linear problem), natural adaptivity, can deal with transport dominated problem, nonlinear problem and boundary condition.
- **Cons**: Limitation in approximation (solution can be compressed in this format), unstructured mesh. Technicality in implementation.
- **Todo:** implicit solve, better time stepping, parallel.
- **Source code** `https://github.com/JuntaoHuang/adaptive-multiresolution-DG`
- **Review** J. Huang, W. Guo and Y. Cheng, Adaptive sparse grid discontinuous Galerkin method: review and software implementation, Communications on Applied Mathematics and Computation, 2023.

The END!

Thank You!