Tuning Spectral Element Preconditioners for Parallel Scalability on GPUs

Malachi Phillips[1] Stefan Kerkemeier[2] Paul Fischer[1,2,3]

[1]Department of Computer Science, University of Illinois at Urbana-Champaign

[2]Mathematics and Computer Science, Argonne National Laboratory, Lemont, IL 60439

[3]Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign

Introduction

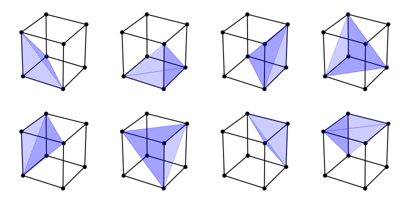
- Poisson solve for pressure arises in Navier-Stokes simulations.
- This solve encompasses the majority of the solution time.
- Spectral element (SE): E elements with polynomial degree p, $n \approx Ep^3$ unknowns and $\mathcal{O}(Ep^6)$ nonzeros.
  - Exploit tensor-product-sum factorization, $\mathcal{O}(Ep^4)$ cost to apply matrix-vector product [Deville et al., 2002].
  - Fast SE-based flow simulations require iterative solvers.
  - Poor conditioning of system: $\kappa(A) \sim \mathcal{O}(h^{-p})$ requires preconditioning.

Poisson Preconditioners

- Jacobi
- Additive Schwarz method (ASM)
- Low-order operator preconditioning (SEMFEM)
- Geometric p-Multigrid (pMG), requiring a smoother:
    - (Damped) Jacobi (again)
    - ASM (again)
    - Chebyshev polynomial smoothing

SEMFEM

- Precondition high-order system with low-order discretizations with coinciding nodes
- [Orszag, 1979] demonstrated $\kappa(\mathrm{M}^{-1}\mathrm{A}) \sim \pi^2/4$ scaling for second-order Dirichlet problems.
- [Bello-Maldonado and Fischer, 2019] proposed using one-per-vertex scheme.
  - Same approach, but solve resultant system using algebraic multigrid (AMG) implemented in CUDA in AmgX [Naumov et al., 2015] with single pass V-cycle, damped Jacobi relaxation ($\omega = 0.9$).



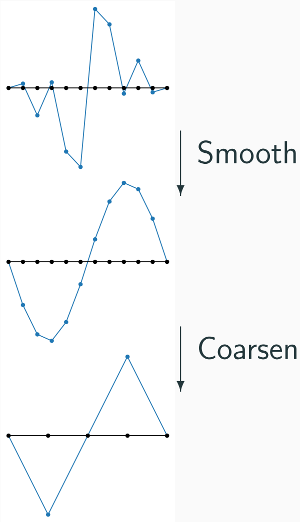
Multigrid

Algorithm 1 Multigrid V-cycle

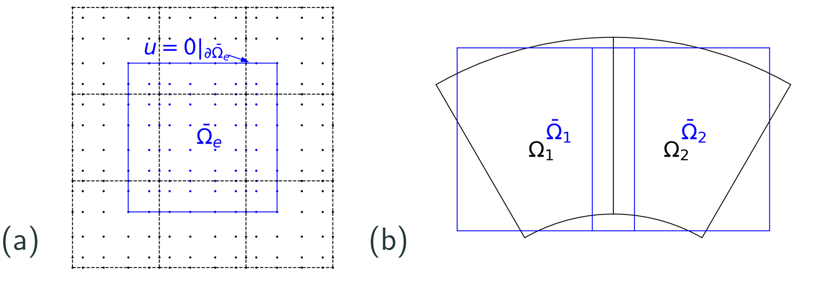
$x = x + M^{-1}(b - Ax)$ // smooth
$r = b - Ax$ // re-evaluate residual
$r_C = P^T r$ // coarsen
$e_C = A_C^{-1} r_C$ // solve/re-apply V-cycle
$e = P e_C$ // prolongate
$x = x + e$ // update solution
$x = x + M^{-1}(b - Ax)$ // post smoothing

Smoother choices:

- Damped Jacobi
- Schwarz
- Chebyshev Jacobi smoothing
- *Chebyshev Schwarz smoothing*



Additive Schwarz (ASM), Restrictive Additive Schwarz (RAS)



- $\mathrm{M}^{-1} := \sum_{e=1}^{E} \mathrm{W}_e \mathrm{R}_e^T \bar{\mathrm{A}}_e^{-1} \mathrm{R}_e$, subdomain (a) [Lottes and Fischer, 2005, Loisel et al., 2008].
- How to form $\bar{\mathrm{A}}_e^{-1}$?
  - Galerkin: $\bar{\mathrm{A}}_e = \mathrm{R}_e \mathrm{A} \mathrm{R}_e^T$, ruins $\mathcal{O}(p^3)$ storage, $\mathcal{O}(p^4)$ work per element.
  - Box-like approximation (b): recover $\mathcal{O}(p^3)$ storage, $\mathcal{O}(p^4)$ work per element using fast diagonalization method (FDM).

FDM

- 3D Poisson-in-a-box:

$$\bar{\mathsf{A}} = \mathsf{B}_z \otimes \mathsf{B}_y \otimes \mathsf{A}_x + \mathsf{B}_z \otimes \mathsf{A}_y \otimes \mathsf{B}_x + \mathsf{A}_z \otimes \mathsf{B}_y \otimes \mathsf{B}_x,$$

- Directly invert with FDM with $\mathcal{O}(p^3)$ storage, $\mathcal{O}(p^4)$ work:

$$\bar{A}^{-1} = (S_z \otimes S_y \otimes S_x) D^{-1} (S_z^T \otimes S_y^T \otimes S_x^T),$$

where

$$D = I \otimes I \otimes \Lambda_x + I \otimes \Lambda_y \otimes I + \Lambda_z \otimes I \otimes I$$

and each S_*, Λ_* are from the generalized eigenvalue problem:

$$\mathsf{A}_* \mathsf{s}_i = \lambda_i \mathsf{B}_* \mathsf{s}_i$$

where B_*, A_* are 1D mass-stiffness matrices.

Chebyshev Smoothing

Algorithm 2 Chebyshev smoother

$\theta = \frac{1}{2}(\lambda_{max} + \lambda_{min})$, $\delta = \frac{1}{2}(\lambda_{max} - \lambda_{min})$, $\sigma = \frac{\theta}{\delta}$, $\rho_1 = \frac{1}{\sigma}$

$r = \mathsf{S}(\mathsf{b} - \mathsf{A}\mathsf{x})$, $\mathsf{d}_1 = \frac{1}{\theta} r$, $\mathsf{x}_1 = 0$

for $k = 1, \ldots,$ `chebyshevOrder` **do**

  $\mathsf{x}_{k+1} = \mathsf{x}_k + \mathsf{d}_k$

  $\mathsf{r}_{k+1} = \mathsf{r}_k - \mathsf{S}\mathsf{A}\mathsf{d}_k$, $\rho_{k+1} = \frac{1}{2\sigma - \rho_k}$

  $\mathsf{d}_{k+1} = \rho_{k+1}\rho_k \mathsf{d}_k + \frac{2\rho_{k+1}}{\delta} \mathsf{r}_{k+1}$

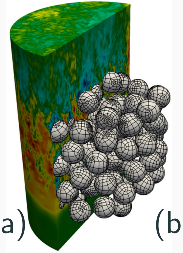
end for

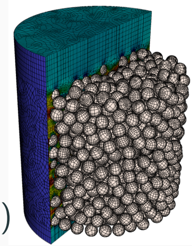
$\mathsf{x}_{k+1} = \mathsf{x}_k + \mathsf{d}_k$

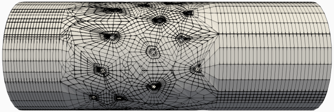
return x_{k+1}

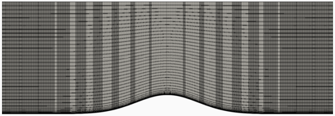
Max eigenvalue estimate $\tilde{\lambda}$ obtained with 10 Arnoldi iterations, $(\lambda_{max}, \lambda_{min}) = (1.1, 0.1)\tilde{\lambda}$. See [Adams et al., 2003, Phillips et al.,] for sensitivity to coefficients. Jacobi based S from [Adams et al., 2003, Sundar et al., 2015, Kronbichler and Ljungkvist, 2019].

Navier-Stokes

(a)


(b)


(c)


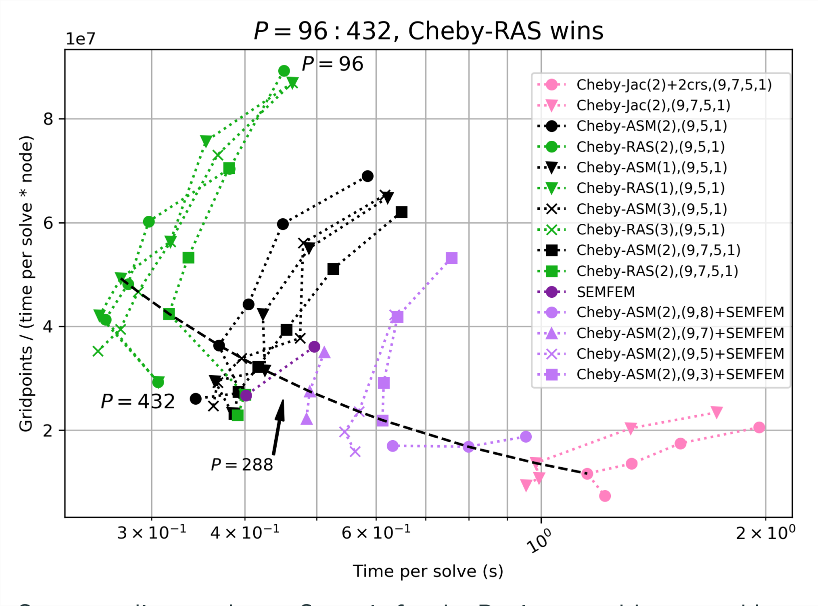
(d)


Case Name	E	p	n
146 pebble (a)	62K	7	21M
1568 pebble (b)	524K	7	180M
67 pebble (c)	122K	7	42M
Speed bump (d)	885K	9	645M
	CFL	Δt	$T_{restart}$
(a) [Lan et al., 2021]	4	2×10^{-3}	10
(b) [Lan et al., 2021]	4	5×10^{-4}	20
(c) [Yuan et al., 2020]	4	5×10^{-5}	10.6
(d) [Shur et al., 2021]	0.8	2×10^{-3}	5.6
	Re	Tol	Steps
(a)	5000	10^{-4}	2000
(b)	5000	10^{-4}	2000
(c)	1460	10^{-4}	2000
(d)	10^6	10^{-5}	2000

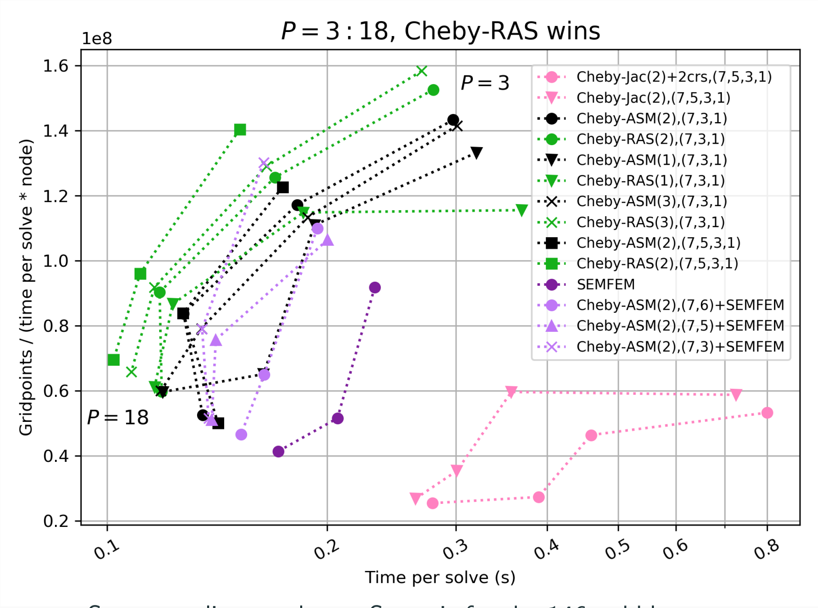
Problem discretization, timestepping parameters. Pebble cases use a two stage subcycling scheme. All cases use a 2nd-order timestepper with 10 prior solution vectors to generate the initial guess [Fischer, 1998].

Results

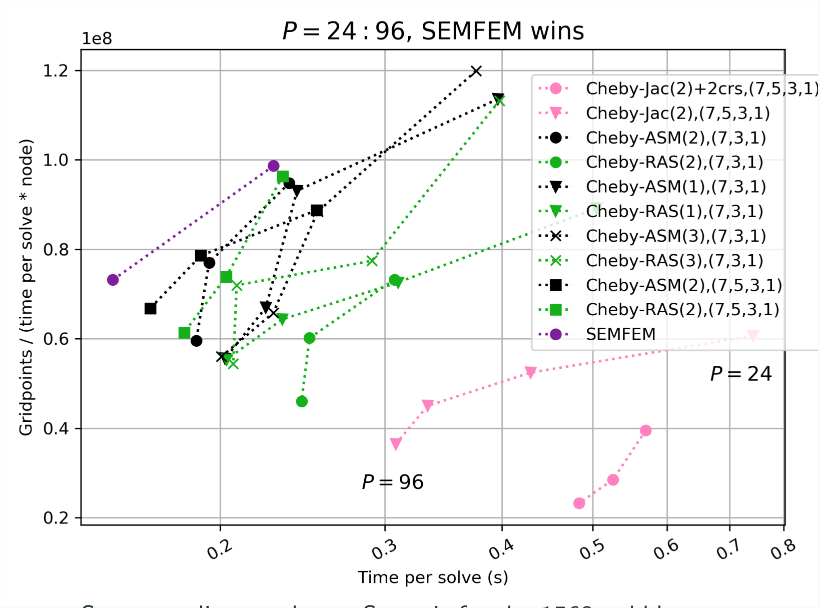
- All results on Summit (42 IBM Power9 CPUs, 6 NVIDIA V100 GPUs per node).
- Each of the P ranks are assigned one GPU, 6 GPUs per node (unless $P < 6$).
- At coarsest level, solve using one or two BoomerAMG V-cycles [Henson and Yang, 2002]
- pMG preconditioner with η-order Chebyshev-accelerated ξ smoother with a multigrid schedule Π is denoted Cheby-$\xi(\eta)$,Π.
    - Cheby-ASM(2),(7,3,1) denotes pMG preconditioning with a 2nd-order Chebyshev-accelerated ASM smoother with $p = 7$, $p = 3$, and $p = 1$ as multigrid levels.



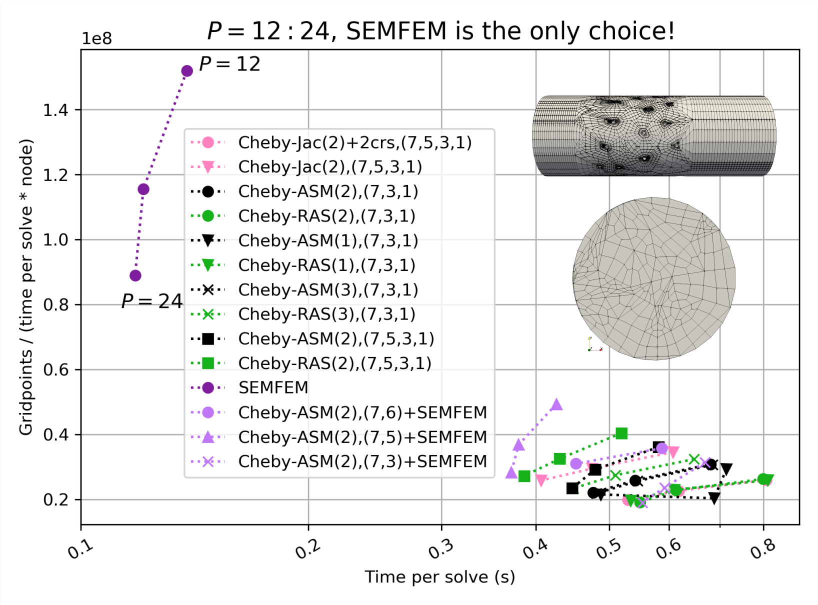
Strong scaling results on Summit for the Boeing speed bump problem.



Strong scaling results on Summit for the 146 pebble case.



Strong scaling results on Summit for the 1568 pebble case.



Strong scaling results on Summit for the 67 pebble case.

Tuning

- There is no single optimal preconditioning strategy.
- Tuning should be done by the software, not the user.
- In large-scale fluid mechanics applications, any tuning overhead is amortized over 10^4–10^5 Poisson solves.
- Auto-tuning of preconditioners considered by [Imakura et al., 2012, Yamada et al., 2018, Brown et al., 2021].
- A primitive exhaustive search over a few reasonable preconditioner setups recovers performance in the 67 pebble case.

Conclusion

- Introduce Chebyshev-accelerated Schwarz smoothers.
- Chebyshev-accelerated Schwarz smoothers with *p*-multigrid, and SEMFEM solved with AmgX, prove to be good SE Poisson preconditioners on OLCF's Summit.
- Introduce (primitive) tuner to select preconditioner strategy.
- Need additional understanding of Chebyshev-accelerated Schwarz smoothers through Local Fourier analysis [Thompson et al., 2021].
- nekRS: https://github.com/nek5000/nekrs

Acknowledgements

- This research is supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering and early testbed platforms, in support of the nation's exascale computing imperative.
- Oak Ridge Leadership Computing Facility at Oak Ridge National Laboratory, Office of Science of the U.S. Department of Energy under Contract DE-AC05-00OR22725.
- Meshing/visualization provided by YuHsiang Lan, Ramesh Balakrishnan, David Alan Reger, and Haomin Yuan
- The reviewers of this work for their insightful comments and suggestions.

References i

Adams, M., Brezina, M., Hu, J., and Tuminaro, R. (2003).
Parallel multigrid smoothing: polynomial versus Gauss–Seidel.
Journal of Computational Physics, 188(2):593–610.

Bello-Maldonado, P. D. and Fischer, P. F. (2019).
Scalable Low-Order Finite Element Preconditioners for High-Order Spectral Element Poisson Solvers.
SIAM Journal on Scientific Computing, 41(5):S2–S18.

Brown, J., He, Y., MacLachlan, S., Menickelly, M., and Wild, S. M. (2021).
Tuning Multigrid Methods with Robust Optimization and Local Fourier Analysis.
SIAM Journal on Scientific Computing, 43(1):A109–A138.

Deville, M. O., Fischer, P. F., Fischer, P. F., and Mund, E. (2002).
High-order methods for incompressible fluid flow, volume 9.
Cambridge university press.

Fischer, P. F. (1998).
Projection techniques for iterative solution of Ax= b with successive right-hand sides.
Computer methods in applied mechanics and engineering, 163(1-4):193–204.
Publisher: Elsevier.

Henson, V. E. and Yang, U. M. (2002).
BoomerAMG: A parallel algebraic multigrid solver and preconditioner.
Applied Numerical Mathematics, 41(1):155–177.

References iii

Imakura, A., Sakurai, T., Sumiyoshi, K., and Matsufuru, H. (2012).
An Auto-Tuning Technique of the Weighted Jacobi-Type Iteration Used for Preconditioners of Krylov Subspace Methods.
In *2012 IEEE 6th International Symposium on Embedded Multicore SoCs*, pages 183–190.

Kershaw, D. S. (1981).
Differencing of the diffusion equation in Lagrangian hydrodynamic codes.
Journal of Computational Physics, 39(2):375–395.
Publisher: Elsevier.

References iv

Kolev, T., Fischer, P., Austin, A. P., Barker, A. T., Beams, N., Brown, J., Camier, J.-S., Chalmers, N., Dobrev, V., Dudouit, Y., Ghaffari, L., Kerkemeier, S., Lan, Y.-H., Merzari, E., Min, M., Pazner, W., Ratnayaka, T., Shephard, M. S., Siboni, M. H., Smith, C. W., Thompson, J. L., Tomov, S., and Warburton, T. (2021).
CEED ECP Milestone Report: High-order algorithmic developments and optimizations for large-scale GPU-accelerated simulations.
Technical report, Zenodo.

Kronbichler, M. and Ljungkvist, K. (2019).
Multigrid for matrix-free high-order finite element computations on graphics processors.
ACM Transactions on Parallel Computing, 6(1):1–32.

References v

Lan, Y.-H., Fischer, P., Merzari, E., and Min, M. (2021).
All-Hex Meshing Strategies For Densely Packed Spheres.
Proceedings of the 29th International Meshing Roundtable, pages 293–305.

Loisel, S., Nabben, R., Szyld, D. B., Lottes, J., and Fischer, P. (2008).
On Hybrid Multigrid-Schwarz Algorithms.
J Sci Comput, page 11.

Lottes, J. W. and Fischer, P. F. (2005).
Hybrid Multigrid/Schwarz Algorithms for the Spectral Element Method.
Journal of Scientific Computing, 24(1):45–78.

References vi

Naumov, M., Arsaev, M., Castonguay, P., Cohen, J., Demouth, J., Eaton, J., Layton, S., Markovskiy, N., Reguly, I., Sakharnykh, N., Sellappan, V., and Strzodka, R. (2015).
AmgX: A Library for GPU Accelerated Algebraic Multigrid and Preconditioned Iterative Methods.
SIAM Journal on Scientific Computing, 37(5):S602–S626.
Publisher: Society for Industrial and Applied Mathematics.

Orszag, S. A. (1979).
Spectral Methods for Problems in Complex Geometrics.
In PARTER, S. V., editor, *Numerical Methods for Partial Differential Equations*, pages 273–305. Academic Press.

Phillips, M., Kerkemeier, S., and Fischer, P.
Tuning Spectral Element Preconditioners for Parallel Scalability on GPUs, pages 37–48.

Shur, M. L., Spalart, P. R., Strelets, M. K., and Travin, A. K. (2021).
Direct numerical simulation of the two-dimensional speed bump flow at increasing Reynolds numbers.
International Journal of Heat and Fluid Flow, 90:108840.
Publisher: Elsevier.

Sundar, H., Stadler, G., and Biros, G. (2015).
Comparison of multigrid algorithms for high-order continuous finite element discretizations.
Numerical Linear Algebra with Applications, 22(4):664–680.
_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nla.1979.

Thompson, J. L., Brown, J., and He, Y. (2021).
Local fourier analysis of p-multigrid for high-order finite element operators.
arXiv preprint arXiv:2108.01751.

Yamada, K., Katagiri, T., Takizawa, H., Minami, K., Yokokawa, M., Nagai, T., and Ogino, M. (2018).

Preconditioner auto-tuning using deep learning for sparse iterative algorithms.

In *2018 Sixth International Symposium on Computing and Networking Workshops (CANDARW)*, pages 257–262. IEEE.

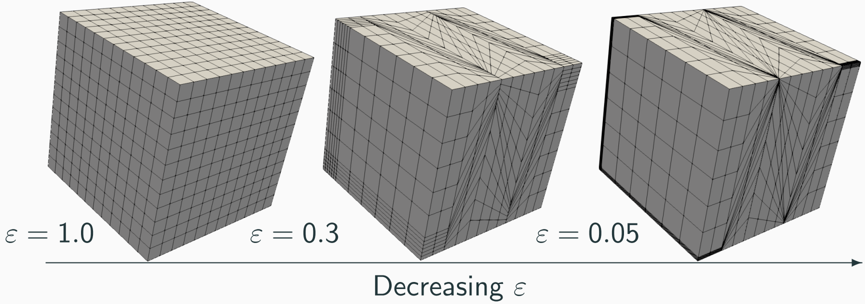
Yuan, H., Yildiz, M. A., Merzari, E., Yu, Y., Obabko, A., Botha, G., Busco, G., Hassan, Y. A., and Nguyen, D. T. (2020).

Spectral element applications in complex nuclear reactor geometries: Tet-to-hex meshing.

Nuclear Engineering and design, 357:110422.

Supplementary Material

Kershaw



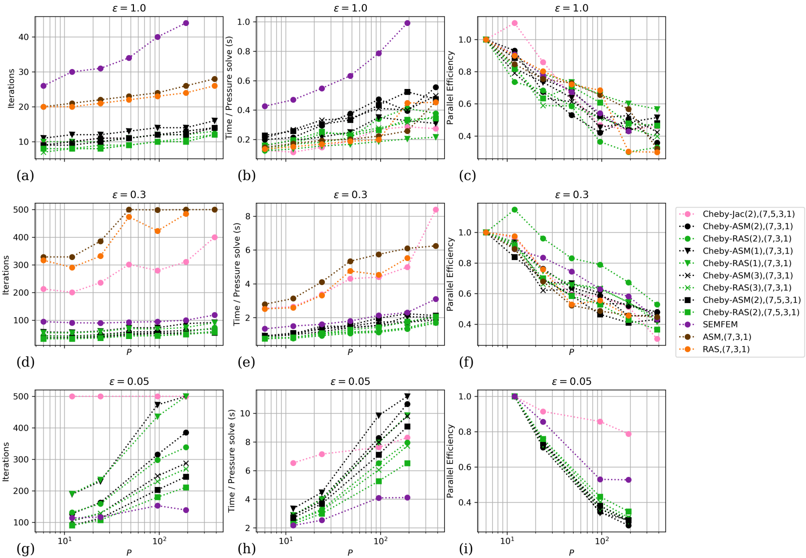
Kershaw family of meshes [Kolev et al., 2021, Kershaw, 1981] on $\Omega := [-1/2, 1/2]^3$, Dirichlet on $\partial\Omega$ with right hand side:

$$f(x, y, z) = 3\pi^2 \sin(\pi x) \sin(\pi y) \sin(\pi z) + g,$$

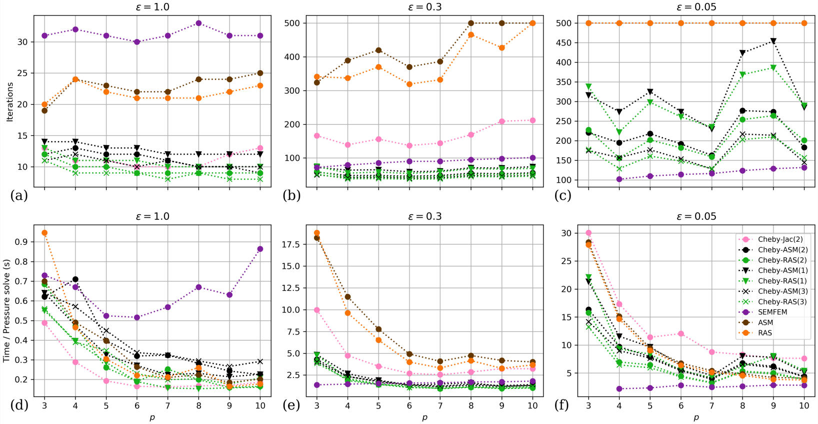
where $g(x, y, z)$ is a random, continous vector vanishing on $\partial\Omega$.

Vary (E, p) for weak scaling and order dependence studies.

Solution target: 10^{-8} relative residual reduction.



Kershaw weak scaling, GMRES(20). $n/P = 2.67M$.



Kershaw order dependence, GMRES(20). $n/P = 2.88M$, $P = 6$.

Spectral Element Poisson

- Poisson PDE:

$$-\nabla^2 u = f \text{ for } u, f \in \Omega \subset \mathbb{R}^3 \mapsto \mathbb{R}.$$

- Weak form: *Find $u \in X_0^p$ such that,*

$$(\nabla v, \nabla u)_p = (v, f)_p \ \ \forall v \in X_0^p.$$

- Results in linear system:

$$\mathrm{Au} = \mathrm{Bf},$$

with $\mathrm{B}_{ij} := (\phi_i, \phi_j)_p$, and $\mathrm{A}_{ij} := (\nabla\phi_i, \nabla\phi_j)_p$.

- Poor conditioning: $\kappa(A) \sim \mathcal{O}(h^{-p})$

Chebyshev Multiplier Sensitivity

$\lambda_{min} \backslash \lambda_{max}$	$0.9\tilde{\lambda}$	$0.95\tilde{\lambda}$	$1.0\tilde{\lambda}$	$1.1\tilde{\lambda}$	$1.2\tilde{\lambda}$	$1.3\tilde{\lambda}$
$0.0\tilde{\lambda}$	-	-	-	-	-	-
$0.025\tilde{\lambda}$	-	-	-	110	64	48
$0.05\tilde{\lambda}$	-	-	-	50	40	38
$0.1\tilde{\lambda}$	-	-	124	40	38	38
$0.2\tilde{\lambda}$	159	45	43	42	43	44
$0.25\tilde{\lambda}$	47	45	44	44	45	46

Iteration counts for the $(\lambda_{min}, \lambda_{max})$ sensitivity study. Omitted entries failed to converge in 1000 iterations.