Parallel-in-time solutions with ParaDiag

Josh Hope-Collins

January 10, 2023

Outline

Imperial College London

Outline

Parallel-in-time motivation

The ParaDiag idea

ParaDiag methods

Performance model

Linear shallow water equations

Nonlinear problems

Conclusions

Time-dependent O/PDEs

$$\boldsymbol{M}\partial_t u + \boldsymbol{K} u = b(t)$$

Time-dependent O/PDEs

$$\boldsymbol{M}\partial_t u + \boldsymbol{K} u = b(t)$$

$$\boldsymbol{M}\left(\frac{u^{n+1} - u^n}{\Delta t}\right) + \theta \boldsymbol{K} u^{n+1} + (1-\theta)\, \boldsymbol{K} u^n = b^{n+1}$$

Time-dependent O/PDEs

$$\boldsymbol{M}\partial_t u + \boldsymbol{K} u = b(t)$$

$$\boldsymbol{M}\left(\frac{u^{n+1} - u^n}{\Delta t}\right) + \theta \boldsymbol{K} u^{n+1} + (1-\theta)\,\boldsymbol{K} u^n = b^{n+1}$$

$$\left(\frac{\boldsymbol{M}}{\Delta t} + \theta \boldsymbol{K}\right) u^{n+1} = b^{n+1} - \left(\frac{-\boldsymbol{M}}{\Delta t} + (1-\theta)\,\boldsymbol{K}\right) u^n$$

Performance of serial method

$$\left(\frac{\boldsymbol{M}}{\Delta t} + \theta \boldsymbol{K}\right) u^{n+1} = \tilde{b}$$

Work: $W_s = k_s N_s^b N_x N_t \sim N_x N_t$

Processors: $P_s \sim N_x$

Time: $T_s = \dfrac{W_s}{P_s} = k_s N_s^b N_t \sim N_t$

Outline

Imperial College London

All-at-once system

$$\left(\frac{-\boldsymbol{M}}{\Delta t} + (1-\theta)\,\boldsymbol{K}\right)u^{n} + \left(\frac{\boldsymbol{M}}{\Delta t} + \theta\boldsymbol{K}\right)u^{n+1} = b^{n+1}$$

All-at-once system

$$\boldsymbol{A}_0 u^n + \boldsymbol{A}_1 u^{n+1} = b^{n+1}$$

All-at-once system

$$\boldsymbol{A}_0 u^n + \boldsymbol{A}_1 u^{n+1} = b^{n+1}$$

$$\boldsymbol{A}_0 u^0 + \quad \boldsymbol{A}_1 u^1 \qquad\qquad = b^1$$

All-at-once system

$$\boldsymbol{A}_0 u^n + \boldsymbol{A}_1 u^{n+1} = b^{n+1}$$

$$\begin{aligned}
\boldsymbol{A}_0 u^0 + \quad \boldsymbol{A}_1 u^1 \quad & & &= b^1 \\
\boldsymbol{A}_0 u^1 + \quad & \boldsymbol{A}_1 u^2 & &= b^2
\end{aligned}$$

All-at-once system

$$\boldsymbol{A}_0 u^n + \boldsymbol{A}_1 u^{n+1} = b^{n+1}$$

$$\begin{array}{lllll}
\boldsymbol{A}_0 u^0 + & \boldsymbol{A}_1 u^1 & & & = b^1 \\
 & \boldsymbol{A}_0 u^1 + & \boldsymbol{A}_1 u^2 & & = b^2 \\
 & & \boldsymbol{A}_0 u^2 + & \boldsymbol{A}_1 u^3 & = b^3
\end{array}$$

All-at-once system

$$\boldsymbol{A}_0 u^n + \boldsymbol{A}_1 u^{n+1} = b^{n+1}$$

$$\begin{array}{cccccc}
\boldsymbol{A}_0 u^0 + & \boldsymbol{A}_1 u^1 & & & & = b^1 \\
& \boldsymbol{A}_0 u^1 + & \boldsymbol{A}_1 u^2 & & & = b^2 \\
& & \boldsymbol{A}_0 u^2 + & \boldsymbol{A}_1 u^3 & & = b^3 \\
& & & \boldsymbol{A}_0 u^3 + & \boldsymbol{A}_1 u^4 & = b^4
\end{array}$$

All-at-once system

$$\boldsymbol{A}_0 u^n + \boldsymbol{A}_1 u^{n+1} = b^{n+1}$$

$$\begin{pmatrix} \boldsymbol{A}_1 & & & \\ \boldsymbol{A}_0 & \boldsymbol{A}_1 & & \\ & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \\ & & \boldsymbol{A}_0 & \boldsymbol{A}_1 \end{pmatrix} \begin{pmatrix} u^1 \\ u^2 \\ u^3 \\ u^4 \end{pmatrix} = \begin{pmatrix} b^1 - \boldsymbol{A}_0 u^0 \\ b^2 \\ b^3 \\ b^4 \end{pmatrix}$$

$$\boldsymbol{A}\boldsymbol{u} = \boldsymbol{b}$$

Diagonalising the all-at-once system

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}, \qquad \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}\boldsymbol{u} = \boldsymbol{b}$$

Diagonalising the all-at-once system

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}, \qquad \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}\boldsymbol{u} = \boldsymbol{b}$$

$$\boldsymbol{u} = \boldsymbol{V}\tilde{\boldsymbol{u}}, \quad \tilde{\boldsymbol{b}} = \boldsymbol{V}^{-1}\boldsymbol{b}$$

$$\boldsymbol{\Lambda}\tilde{\boldsymbol{u}} = \tilde{\boldsymbol{b}},$$

$$\begin{pmatrix} \boldsymbol{\Lambda}_1 & & & \\ & \boldsymbol{\Lambda}_2 & & \\ & & \boldsymbol{\Lambda}_3 & \\ & & & \boldsymbol{\Lambda}_4 \end{pmatrix} \begin{pmatrix} \tilde{u}^1 \\ \tilde{u}^2 \\ \tilde{u}^3 \\ \tilde{u}^4 \end{pmatrix} = \begin{pmatrix} \tilde{b}^1 \\ \tilde{b}^2 \\ \tilde{b}^3 \\ \tilde{b}^4 \end{pmatrix}$$

Requirements on the diagonalisation

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1},$$

1. Exists (not as trivial as it sounds)
2. $\boldsymbol{V}$ and $\boldsymbol{V}^{-1}$ are:
   - relatively cheap to compute
   - easily parallelisable in time
3. The cost to solve $\boldsymbol{\Lambda}_i$ is comparable to the serial problem

Outline

ParaDiag-I: A diagonalisable integrator

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{K}$$

$$\boldsymbol{A} = \boldsymbol{B}_1 \otimes \boldsymbol{M} + \boldsymbol{B}_2 \otimes \boldsymbol{K}$$

ParaDiag-I: A diagonalisable integrator

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{K}$$

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{B}_1 \otimes \boldsymbol{M} + \boldsymbol{B}_2 \otimes \boldsymbol{K} \\ &= (\boldsymbol{V} \otimes \boldsymbol{I}_x)\,(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right) \end{aligned}$$

- $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ must be *simultaneously* diagonalisable

ParaDiag-I: A diagonalisable integrator

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{K}$$

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{B}_1 \otimes \boldsymbol{M} + \boldsymbol{B}_2 \otimes \boldsymbol{K} \\ &= (\boldsymbol{V} \otimes \boldsymbol{I}_x)(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right) \end{aligned}$$

- $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ must be *simultaneously* diagonalisable
- $\boldsymbol{B}_1$ is diagonalisable if Δt_i are all different
- $\boldsymbol{B}_2$ is diagonal if $\theta = 1$

ParaDiag-II: Circulant all-at-once system

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \frac{-1}{\Delta t} \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & (1-\theta) \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{K}$$

$$\boldsymbol{P} = \boldsymbol{C}_1 \otimes \boldsymbol{M} + \boldsymbol{C}_2 \otimes \boldsymbol{K} \approx \boldsymbol{A}$$

ParaDiag-II: Circulant all-at-once system

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \frac{-1}{\Delta t} \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & (1-\theta) \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{K}$$

$$\boldsymbol{P} = \boldsymbol{C}_1 \otimes \boldsymbol{M} + \boldsymbol{C}_2 \otimes \boldsymbol{K} \approx \boldsymbol{A}$$

- $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are *simultaneously* diagonalisable using FFT

ParaDiag-II: Circulant all-at-once system

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \frac{-\alpha}{\Delta t} \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & \alpha\,(1-\theta) \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{K}$$

$$\boldsymbol{P}^{(\alpha)} = \boldsymbol{C}_1^{(\alpha)} \otimes \boldsymbol{M} + \boldsymbol{C}_2^{(\alpha)} \otimes \boldsymbol{K} \approx \boldsymbol{A}$$

- $\boldsymbol{C}_1^{(\alpha)}$ and $\boldsymbol{C}_2^{(\alpha)}$ are *simultaneously* diagonalisable using FFT

ParaDiag-II: Circulant all-at-once system

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \frac{-\alpha}{\Delta t} \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & \alpha(1-\theta) \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{K}$$

$$\boldsymbol{P}^{(\alpha)} = \boldsymbol{C}_1^{(\alpha)} \otimes \boldsymbol{M} + \boldsymbol{C}_2^{(\alpha)} \otimes \boldsymbol{K} \approx \boldsymbol{A}$$

- $\boldsymbol{C}_1^{(\alpha)}$ and $\boldsymbol{C}_2^{(\alpha)}$ are *simultaneously* diagonalisable using FFT
- $\alpha \in (0, 1]$, and in practice can be very small ($\approx 10^{-3}$)

Circulant diagonalisation

$$\boldsymbol{P}^{(\alpha)} = \boldsymbol{C}_1^{(\alpha)} \otimes \boldsymbol{M} + \boldsymbol{C}_2^{(\alpha)} \otimes \boldsymbol{K}, \qquad \boldsymbol{C}_j^{(\alpha)} = \boldsymbol{V}\boldsymbol{D}_j\boldsymbol{V}^{-1},$$

$$\boldsymbol{P}^{(\alpha)} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

Circulant diagonalisation

$$\boldsymbol{P}^{(\alpha)} = \boldsymbol{C}_1^{(\alpha)} \otimes \boldsymbol{M} + \boldsymbol{C}_2^{(\alpha)} \otimes \boldsymbol{K}, \qquad \boldsymbol{C}_j^{(\alpha)} = \boldsymbol{V}\boldsymbol{D}_j\boldsymbol{V}^{-1},$$

$$\boldsymbol{P}^{(\alpha)} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

$$\boldsymbol{\Lambda}_k = \left(\lambda_{1,k}\boldsymbol{M} + \lambda_{2,k}\boldsymbol{K}\right) \quad \forall k \in [1, N_t]$$

Circulant diagonalisation

$$\boldsymbol{P}^{(\alpha)} = \boldsymbol{C}_1^{(\alpha)} \otimes \boldsymbol{M} + \boldsymbol{C}_2^{(\alpha)} \otimes \boldsymbol{K}, \qquad \boldsymbol{C}_j^{(\alpha)} = \boldsymbol{V}\boldsymbol{D}_j\boldsymbol{V}^{-1},$$

$$\boldsymbol{P}^{(\alpha)} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

$$\boldsymbol{\Lambda}_k = \left(\lambda_{1,k}\boldsymbol{M} + \lambda_{2,k}\boldsymbol{K}\right) \quad \forall k \in [1, N_t]$$

$$\boldsymbol{V} = \boldsymbol{\Gamma}_\alpha^{-1}\boldsymbol{\mathcal{F}}^{-1}, \quad \boldsymbol{D}_k = \mathtt{diag}\left(\mathcal{F}\boldsymbol{\Gamma}_\alpha \boldsymbol{c}_j\right), \quad \boldsymbol{\Gamma}_\alpha = \mathtt{diag}\left(\alpha^{\frac{k-1}{N_t}}\right) \forall k \in [1, N_t]$$

Circulant linear solves

$$\boldsymbol{P}^{(\alpha)} = \left(\boldsymbol{V} \otimes \boldsymbol{I}_x\right)\left(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K}\right)\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

$$\boldsymbol{P}^{(\alpha)}\boldsymbol{x} = \boldsymbol{b}$$

Circulant linear solves

$$\boldsymbol{P}^{(\alpha)} = \left(\boldsymbol{V} \otimes \boldsymbol{I}_x\right)\left(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K}\right)\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

$$\boldsymbol{P}^{(\alpha)}\boldsymbol{x} = \boldsymbol{b}$$

- Step-(a) $\boldsymbol{y}_1 = \left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)\boldsymbol{b}$

Circulant linear solves

$$\boldsymbol{P}^{(\alpha)} = (\boldsymbol{V} \otimes \boldsymbol{I}_x) (\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K}) \left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

$$\boldsymbol{P}^{(\alpha)}\boldsymbol{x} = \boldsymbol{b}$$

- Step-(a) $\boldsymbol{y}_1 = \left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right) \boldsymbol{b}$
- Step-(b) $\left(\lambda_{1,j}\boldsymbol{M} + \lambda_{2,j}\boldsymbol{K}\right) \boldsymbol{y}_{2,n} = \boldsymbol{y}_{1,n} \quad \forall j \in [1, N_t]$

Circulant linear solves

$$\boldsymbol{P}^{(\alpha)} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

$$\boldsymbol{P}^{(\alpha)}\boldsymbol{x} = \boldsymbol{b}$$

- Step-(a) $\boldsymbol{y}_1 = \left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)\boldsymbol{b}$
- Step-(b) $\left(\lambda_{1,j}\boldsymbol{M} + \lambda_{2,j}\boldsymbol{K}\right)\boldsymbol{y}_{2,n} = \boldsymbol{y}_{1,n} \quad \forall j \in [1, N_t]$
- Step-(c) $\boldsymbol{x} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)\boldsymbol{y}_2$

Circulant linear solves

$$\boldsymbol{P}^{(\alpha)} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

$$\boldsymbol{P}^{(\alpha)}\boldsymbol{x} = \boldsymbol{b}$$

- Step-(a) $\boldsymbol{y}_1 = \left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)\boldsymbol{b}$
- Step-(b) $\left(\lambda_{1,j}\boldsymbol{M} + \lambda_{2,j}\boldsymbol{K}\right)\boldsymbol{y}_{2,n} = \boldsymbol{y}_{1,n} \quad \forall j \in [1, N_t]$
- Step-(c) $\boldsymbol{x} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)\,\boldsymbol{y}_2$

All-at-once solution strategies

- Preconditioned Krylov method: $\boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{x} = \boldsymbol{P}^{-1}\boldsymbol{b}$
  $\boldsymbol{P}^{-1}\boldsymbol{A}$ has N_x non-unit eigenvalues
- Richardson iteration: $\boldsymbol{P}\boldsymbol{x}^{k+1} = (\boldsymbol{P} - \boldsymbol{A})\,\boldsymbol{x}^{k} + \boldsymbol{b}$
  Convergence rate bounded by $\alpha/(1-\alpha)$

Outline

Performance of ParaDiag

$$\boldsymbol{P}^{(\alpha)} = (\boldsymbol{V} \otimes \boldsymbol{I}_x)(\boldsymbol{D}_1 \otimes \boldsymbol{M} + \boldsymbol{D}_2 \otimes \boldsymbol{K})\left(\boldsymbol{V}^{-1} \otimes \boldsymbol{I}_x\right)$$

Work: $W_p = 2k_p N_x N_p^b N_t \sim N_x N_x$

Processors: $P_p \sim N_x N_t$

Time: $T_p = \dfrac{W_p}{P_p} = k_p N_p^b \sim 1$

Speedup predictions

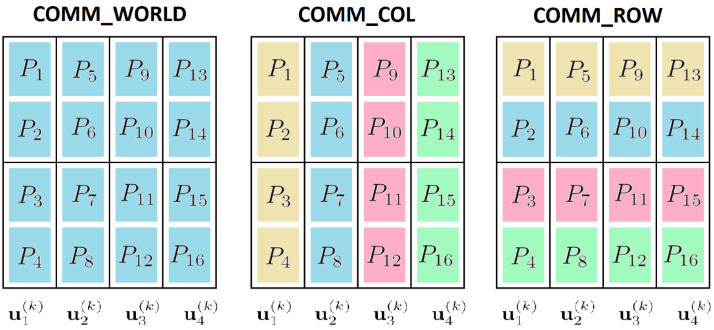
Speedup: $S = \dfrac{T_s}{T_p} = \dfrac{N_t}{\gamma\omega}$

"Difficulty" measures: $\gamma = \dfrac{k_p}{k_s}, \quad \omega = \dfrac{N_b^p}{N_b^s}$

Efficiency: $E = \dfrac{S}{P_p/P_s} = \dfrac{1}{2\gamma\omega}$

Imperial College London

Parallelisation



Caklovic, Speck & Frank 2021 arXiv:2103.12571

Outline

Linear shallow water equations

$$\partial_t \boldsymbol{u} + f\boldsymbol{u}^{\perp} = -g\nabla h$$
$$\partial_t h + H\nabla\cdot\boldsymbol{u} = 0$$

Horizontal velocity: $\boldsymbol{u}$

Fluid depth: h

Rotation: $\boldsymbol{u}^{\perp} = \hat{\boldsymbol{z}} \times \boldsymbol{u}$

Coriolis parameter: f

Gravity: g

Reference depth: H

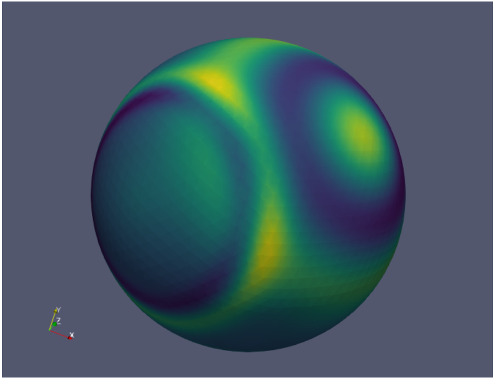
- Depth function space: Discontinuous Galerkin
- Velocity function space: Brezzi-Douglas-Marini
- Crank-Nicholson time integrator

Implicit SWE block solve

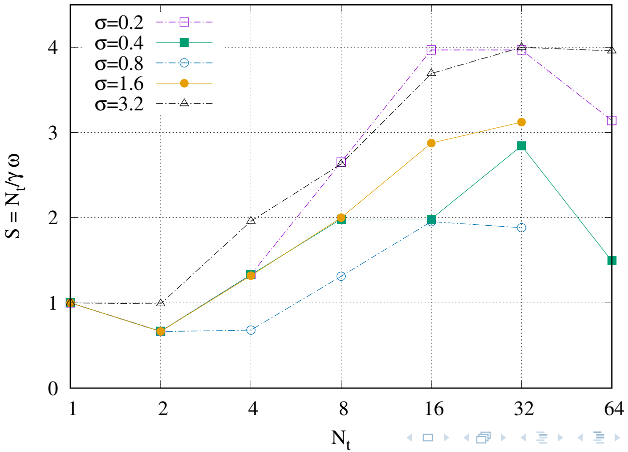
$$(\lambda_1^n \boldsymbol{M} + \lambda_2^n \boldsymbol{K})\boldsymbol{x} = \boldsymbol{b}$$

- Multigrid
  - Krylov solver (GMRES)
    - Preconditioner
      - Additive Vanka patch smoother
        - Direct LU patch solve
  - Direct LU coarse grid solve

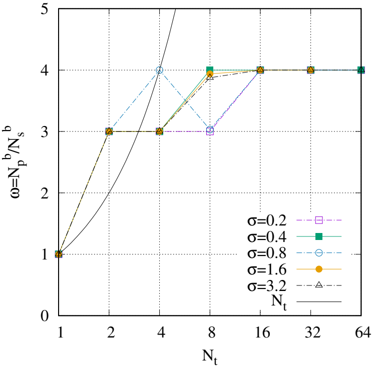
Gravity waves test case



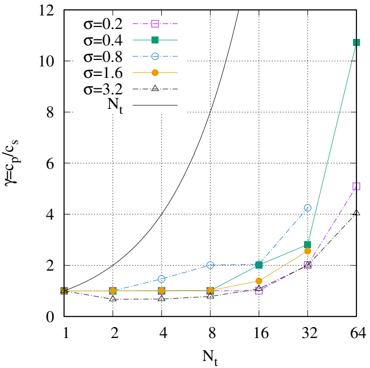
Ideal speedup



Iteration count scaling

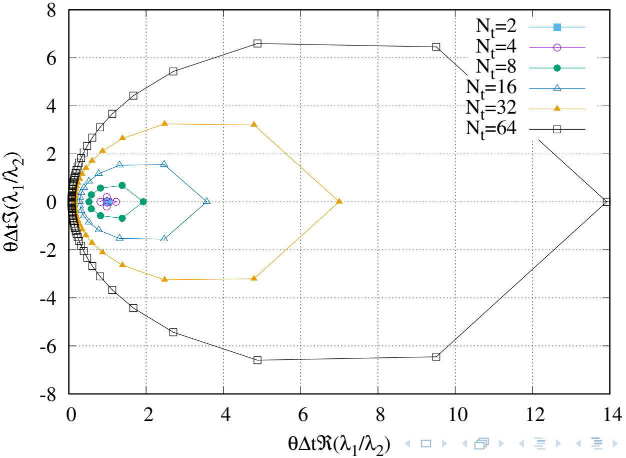


(a) $\omega = N_p^b / N_s^b$

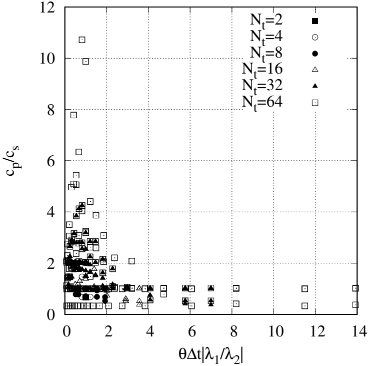


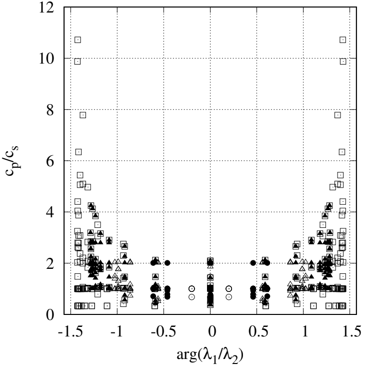
(b) $\gamma = c_p / c_s$

Eigenvalue distribution

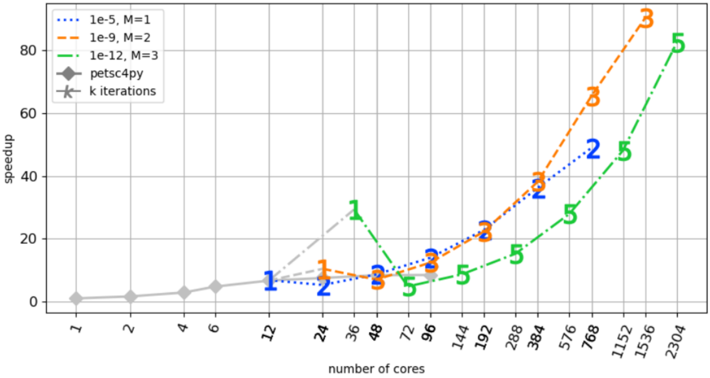


Block solve cost





Wallclock speedup at scale



Caklovic, Speck & Frank 2021 arXiv:2103.12571

Outline

Imperial College London

Nonlinear all-at-once system

$$\boldsymbol{M}\partial_t \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) = 0$$

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \left(\begin{pmatrix} \theta & & & \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \boldsymbol{I}_x \right) \begin{pmatrix} \boldsymbol{f}(\boldsymbol{u}^1) \\ \boldsymbol{f}(\boldsymbol{u}^2) \\ \boldsymbol{f}(\boldsymbol{u}^3) \\ \boldsymbol{f}(\boldsymbol{u}^4) \end{pmatrix}$$

Nonlinear all-at-once Jacobian

$$\boldsymbol{M}\partial_t \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) = 0$$

$$\begin{pmatrix} \theta\nabla\boldsymbol{f}(\boldsymbol{u}^1) & & & \\ (1-\theta)\nabla\boldsymbol{f}(\boldsymbol{u}^1) & \theta\nabla\boldsymbol{f}(\boldsymbol{u}^2) & & \\ & (1-\theta)\nabla\boldsymbol{f}(\boldsymbol{u}^2) & \theta\nabla\boldsymbol{f}(\boldsymbol{u}^3) & \\ & & (1-\theta)\nabla\boldsymbol{f}(\boldsymbol{u}^3) & \theta\nabla\boldsymbol{f}(\boldsymbol{u}^4) \end{pmatrix} \otimes \boldsymbol{I}_x$$

Nonlinear all-at-once Jacobian

$$\boldsymbol{M}\partial_t \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) = 0$$

$$\begin{pmatrix} \theta \nabla \overline{\boldsymbol{f}(\boldsymbol{u})} & & & \\ (1-\theta)\,\overline{\nabla \boldsymbol{f}(\boldsymbol{u})} & \theta \nabla \overline{\boldsymbol{f}(\boldsymbol{u})} & & \\ & (1-\theta)\,\overline{\nabla \boldsymbol{f}(\boldsymbol{u})} & \theta \nabla \overline{\boldsymbol{f}(\boldsymbol{u})} & \\ & & (1-\theta)\,\overline{\nabla \boldsymbol{f}(\boldsymbol{u})} & \theta \nabla \overline{\boldsymbol{f}(\boldsymbol{u})} \end{pmatrix} \otimes \boldsymbol{I}_x$$

Nonlinear all-at-once Jacobian

$$\boldsymbol{M}\partial_t \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) = 0$$

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \overline{\nabla \boldsymbol{f}(\boldsymbol{u})}$$

Nonlinear all-at-once Jacobian

$$\boldsymbol{M}\partial_t \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) = 0$$

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \overline{\nabla \boldsymbol{f}(\boldsymbol{u})}$$

$$\overline{\nabla \boldsymbol{f}(\boldsymbol{u})} = \sum_n^{N_t} \frac{\nabla \boldsymbol{f}(\boldsymbol{u}^n)}{N_t} \quad \text{or} \quad \overline{\nabla \boldsymbol{f}(\boldsymbol{u})} = \nabla \boldsymbol{f}\left(\sum_n^{N_t} \frac{\boldsymbol{u}^n}{N_t}\right)$$

Nonlinear all-at-once Jacobian

$$\boldsymbol{M}\partial_t \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) = 0$$

$$\begin{pmatrix} \frac{1}{\Delta t} & & & \frac{-1}{\Delta t} \\ \frac{-1}{\Delta t} & \frac{1}{\Delta t} & & \\ & \frac{-1}{\Delta t} & \frac{1}{\Delta t} & \\ & & \frac{-1}{\Delta t} & \frac{1}{\Delta t} \end{pmatrix} \otimes \boldsymbol{M} + \begin{pmatrix} \theta & & & (1-\theta) \\ (1-\theta) & \theta & & \\ & (1-\theta) & \theta & \\ & & (1-\theta) & \theta \end{pmatrix} \otimes \overline{\nabla \boldsymbol{f}(\boldsymbol{u})}$$

$$\overline{\nabla \boldsymbol{f}(\boldsymbol{u})} = \sum_n^{N_t} \frac{\nabla \boldsymbol{f}(\boldsymbol{u}^n)}{N_t} \quad \text{or} \quad \overline{\nabla \boldsymbol{f}(\boldsymbol{u})} = \nabla \boldsymbol{f}\left(\sum_n^{N_t} \frac{\boldsymbol{u}^n}{N_t}\right)$$

Nonlinear problems

Nonlinear system: $\boldsymbol{M}\partial_t u + \boldsymbol{f}(u)$

All-at-once system: $(\boldsymbol{B}_1 \otimes \boldsymbol{M})\,\boldsymbol{u} + (\boldsymbol{B}_2 \otimes \boldsymbol{I}_x)\,\boldsymbol{F}(\boldsymbol{u})$

All-at-once Jacobian: $(\boldsymbol{B}_1 \otimes \boldsymbol{M}) + (\boldsymbol{B}_2 \otimes \boldsymbol{I}_x)\,\nabla\boldsymbol{F}(\boldsymbol{u})$

ParaDiag Jacobian: $(\boldsymbol{C}_1 \otimes \boldsymbol{M}) + (\boldsymbol{C}_2 \otimes \nabla\boldsymbol{f}(\overline{\boldsymbol{u}}))$

Time average: $\overline{\boldsymbol{u}} = \sum_n^{N_t} \boldsymbol{u}^n / N_t$

Nonlinear shallow water equations

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + f \boldsymbol{u}^{\perp} = -g \nabla (h + b)$$
$$\partial_t h + \nabla \cdot (\boldsymbol{u} h) = 0$$

Horizontal velocity: $\boldsymbol{u}$

Fluid depth: h

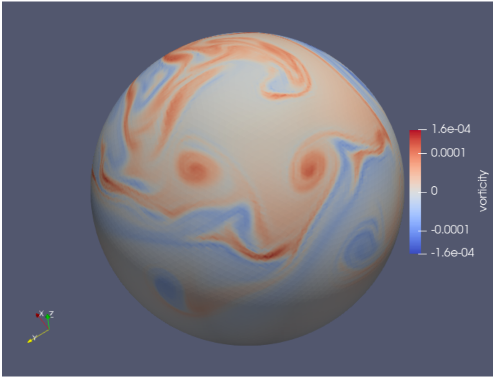
Gravity: g

Bathymetry: b

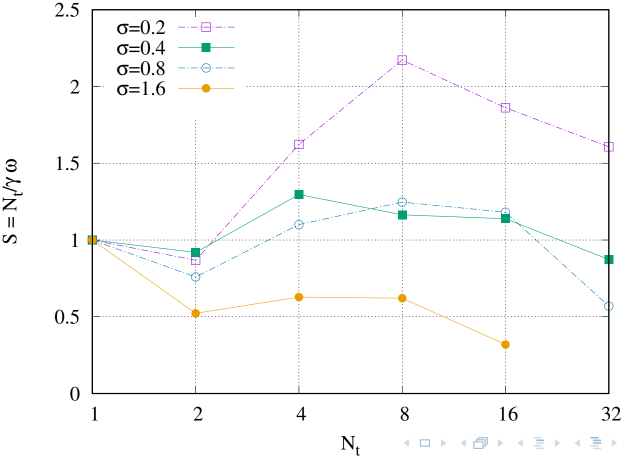
Coriolis parameter: f

Rotation: $\boldsymbol{u}^{\perp} = \hat{\boldsymbol{z}} \times \boldsymbol{u}$

Galewsky et al test case

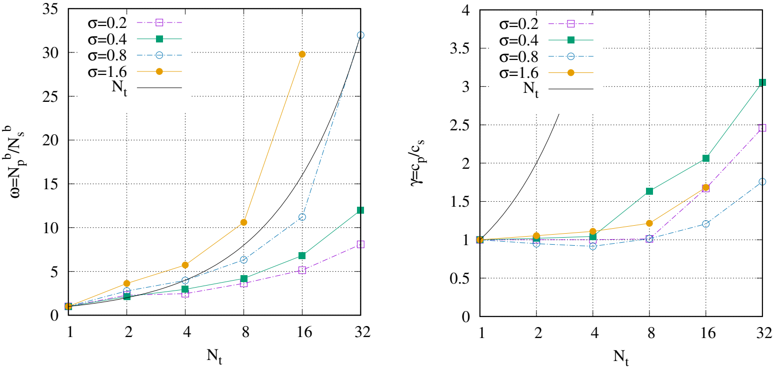


Ideal speedup



Imperial College London

Iteration count scaling



(a) $\omega = N_p^b / N_s^b$ (b) $\gamma = k_p / k_s$

Outline

Conclusions

Summary:

- Solve for entire all-at-once system
- Find suitable diagonalisation in time to expose parallelism
- Circulant preconditioner is diagonalisable with FFT
- Speedups are achievable, but nonlinear problems are hard!

Open questions:

- Better preconditioning for the block solves
- How to deal with non-linear systems when N_t is large?