ParaDiag: PinT Algorithms via Diagonalization Technique

PinT 2020: 9th Workshop on Parallel-in-Time Integration

Shu-Lin Wu

Email: wushulin84@hotmail.com

Co-authors: Martin Gander (UNIGE, Swiss)
Jun Liu (SIUE, USA), Xiaoqiang Yue (XTU, China)
Tao Zhou (CAS, China)

June 9, 2020

Catalog

ParaDiag-I: Direct Standalone Solver

Consider

$$\begin{cases} \dot{y} + Ay = 0, & t \in (0, T) \\ y(0) = y_0, & t = 0 \end{cases}$$

Suppose we use the Backward-Euler method with variable step-sizes $\{\Delta t_n\}$:

$$\begin{cases} \frac{y_1 - y_0}{\Delta t_1} + Ay_1 = 0 \\ \frac{y_2 - y_1}{\Delta t_2} + Ay_1 = 0 \\ \vdots \\ \frac{y_{N_t} - y_{N_t - 1}}{\Delta t_{N_t}} + Ay_{N_t} = 0 \end{cases} \Rightarrow \text{all-at-once system}$$

$$\left[\underbrace{\begin{pmatrix} \frac{I_x}{\Delta t_1} & & & \\ -\frac{I_x}{\Delta t_2} & \frac{I_x}{\Delta t_2} & & \\ & \ddots & \ddots & \\ & & -\frac{I_x}{\Delta t_{N_t}} & \frac{I_x}{\Delta t_{N_t}} \end{pmatrix}}_{:= B \otimes I_x} + \underbrace{\begin{pmatrix} A & & & \\ & A & & \\ & & \ddots & \\ & & & A \end{pmatrix}}_{:= I_t \otimes A} \right] \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N_t} \end{pmatrix}}_{:= \boldsymbol{y}} = \underbrace{\begin{pmatrix} \frac{y_0}{\Delta t_1} \\ 0 \\ \vdots \\ 0 \end{pmatrix}}_{:= \boldsymbol{b}}$$

ParaDiag-I: Direct Standalone Solver

$$\Leftrightarrow \boxed{(B \otimes I_x + I_t \otimes A)\boldsymbol{y} = \boldsymbol{b}}$$

$$B = \begin{pmatrix} \frac{1}{\Delta t_1} & & & \\ -\frac{1}{\Delta t_2} & \frac{1}{\Delta t_2} & & \\ & \ddots & \ddots & \\ & & -\frac{1}{\Delta t_{N_t}} & \frac{1}{\Delta t_{N_t}} \end{pmatrix} \in \mathbb{R}^{N_t \times N_t}$$

Diagonalize B as

$$B = VDV^{-1}, \ D = \begin{pmatrix} \frac{1}{\Delta t_1} & & & \\ & \frac{1}{\Delta t_2} & & \\ & & \ddots & \\ & & & \frac{1}{\Delta t_{N_t}} \end{pmatrix}$$

Diagonalization Technique: Original Idea

By the property of Kronecker product:

$$B \otimes I_x + I_t \otimes A = (V \otimes I_x)(D \otimes I_x + I_t \otimes A)(V^{-1} \otimes I_x)$$

Hence

$$(B \otimes I_x + I_t \otimes A)\boldsymbol{y} = \boldsymbol{b}$$

$$\Rightarrow \begin{cases} (V \otimes I_x)\boldsymbol{u} = \boldsymbol{b} & (a) \\ \left(\frac{I_x}{\Delta t_n} + A\right)\boldsymbol{w}_n = \boldsymbol{u}_n, \ n = 1, 2, \ldots, N_t & (b) \\ (V^{-1} \otimes I_x)\boldsymbol{y} = \boldsymbol{w} & (c) \end{cases}$$

- with V and V^{-1} given, only do **matrix-vector multiplications**:

$$\begin{cases} \boldsymbol{u} = (V^{-1} \otimes I_x)\boldsymbol{b} & (a) \\ \boldsymbol{y} = (V \otimes I_x)\boldsymbol{w} & (c) \end{cases}$$

- the N_t linear systems in Step-(b) are directly parallel

Maday-Rønquist. *Parallelization in time through tensor product space-time Solver*. CRM-2008

ParaDiag-I: Direct Standalone Solver

Proposition-A (how to fix $\{\Delta t_n\}$?)

Let $\tau > 1$ be a parameter and define the geometrically increasing step-sizes

$$\Delta t_n = \tau^{n-1}\Delta t_1,\ n = 1, 2, \ldots, N_t$$

Then B can be factorized with explicit formulas for eigenvector matrices

$$V = \begin{pmatrix} 1 & & & \\ p_1 & \ddots & & \\ \vdots & \ddots & \ddots & \\ p_{N_t-1} & \ldots & p_1 & 1 \end{pmatrix},\ V^{-1} = \begin{pmatrix} 1 & & & \\ q_1 & \ddots & & \\ \vdots & \ddots & \ddots & \\ q_{N_t-1} & \ldots & q_1 & 1 \end{pmatrix}$$

where $p_n := \frac{1}{\prod_{j=1}^{n}(1-\tau^j)}$ and $q_n := (-1)^n \tau^{\frac{n(n-1)}{2}} p_n$

Gander-Halpern-Rannou-Ryan. *A Direct Time Parallel Solver by Diagonalization for the Wave Equation.* SiSC-2019

ParaDiag-I: Direct Standalone Solver

Dual use of V and V^{-1}

$$\begin{cases} \boldsymbol{u} = (V^{-1} \otimes I_x)\boldsymbol{b} & (a) \\ \boldsymbol{y} = (V \otimes I_x)\boldsymbol{w} & (c) \end{cases}$$

Roundoff Error vs Discretization Error

- Roundoff error arises due to floating point operations
- Roundoff error is related to the **condition number of** V
- For given $\tau > 1$: $\Delta t_n = \tau^{n-1}\Delta t_1$, Cond($V$) increases fast as N_t grows
- The largest N_t is the one

  `roundoff error` $\approx$ `discretization error`

- N_t **can not** be large ($N_t = 20 \sim 30$)

ParaDiag-II

Waveform Relaxation (WR) *Variant*

ParaDiag-II: WR Variant

Consider initial-value ODE system:

$$\dot{y} + f(y) = 0,\ y(0) = y_0,\ t \in (0, T)$$

Do WR iterations with *head-tail coupled condition* :

$$\begin{cases} \dot{y}^k + f(y^k) = 0,\ t \in (0, T) \\ y^k(0) = \alpha\left[y^k(T) - y^{k-1}(T)\right] + y_0 \end{cases}$$

- $\alpha \in (0, 1)$ is a parameter
- The converged solution is the solution of $\dot{y} + f(y) = 0$ with $y(0) = y_0$

Gander-Wu. *Convergence analysis of a waveform relaxation method with head-tail coupled condition*. Numer Math-2019

ParaDiag-II: WR Variant

For linear case $\dot{y} + Ay = 0$, Backward-Euler method gives

$$\Bigg[\underbrace{\frac{1}{\Delta t}\begin{pmatrix} I_x & & & -\alpha I_x \\ -I_x & I_x & & \\ & \ddots & \ddots & \\ & & -I_x & I_x \end{pmatrix}}_{C_\alpha \otimes I_x} + \underbrace{\begin{pmatrix} A & & & \\ & A & & \\ & & \ddots & \\ & & & A \end{pmatrix}}_{I_t \otimes A}\Bigg] \underbrace{\begin{pmatrix} y_1^k \\ y_2^k \\ \vdots \\ y_{N_t}^k \end{pmatrix}}_{\boldsymbol{y}^k}$$

$$= \underbrace{\begin{pmatrix} \frac{-\alpha u_N^{k-1} + y_0}{\Delta t} \\ 0 \\ \vdots \\ 0 \end{pmatrix}}_{\boldsymbol{b}^{k-1}}, \quad C_\alpha := \frac{1}{\Delta t}\begin{pmatrix} 1 & & & -\alpha \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix} \in \mathbb{C}^{N_t \times N_t}$$

ParaDiag-II: WR Variant

Diagonalization of α-circulant matrix C_α

- *The $\alpha-$circulant matrix C_α can be diagonalized as $C_\alpha = V D_\alpha V^{-1}$ with*

$$\underbrace{\begin{aligned} & V = \Lambda_\alpha \mathbb{F}_{N_t}, \\ & \Lambda_\alpha = \operatorname{diag}\left(1, \alpha^{-\frac{1}{N_t}}, \ldots, \alpha^{-\frac{N_t-1}{N_t}}\right), \\ & \mathbb{F}_{N_t} = \frac{1}{\sqrt{N_t}} \begin{bmatrix} 1 & 1 & \ldots & 1 \\ 1 & \omega & \ldots & \omega^{N_t-1} \\ \vdots & \vdots & \ldots & \vdots \\ 1 & \omega^{N_t-1} & \ldots & \omega^{(N_t-1)(N_t-1)} \end{bmatrix} \end{aligned}}_{\textbf{Fourier matrix}}, \omega = e^{\frac{2\pi i}{N_t}}$$

- $\mathrm{Cond}_2(V) \leq \frac{1}{\alpha}$ (**Independent of N_t !**)

Bini-Latouche-Meini. *Numerical methods for structured Markov chains*. Oxford University Press: New York, 2005.

ParaDiag-II: WR Variant

So, we can solve $\boldsymbol{y}^k$ as follows

$$(C_\alpha \otimes I_x + I_t \otimes A)\boldsymbol{y}^k = \boldsymbol{b}^{k-1}$$

$$\Rightarrow \begin{cases} (\Lambda_\alpha \mathbb{F}_{N_t} \otimes I_x)\boldsymbol{u} = \boldsymbol{b}^{k-1} & (a) \\ (\lambda_n I_x + A)\,\boldsymbol{w}_n = \boldsymbol{u}_n, \ n = 1, 2, \ldots, N_t & (b) \\ (\mathbb{F}_{N_t}^{-1} \Lambda_\alpha^{-1} \otimes I_x)\boldsymbol{y}^k = \boldsymbol{w} & (c) \end{cases}$$

Parallelism

- **Step-(b): parallel-in-time**
- **Step-(a,c): FFT**

ParaDiag-II: WR Variant

Proposition-B

For $\dot{y} + Ay = 0$ with $y(0) = y_0$ and $\sigma(A) \subseteq \mathbb{C}^+$, the WR iteration

$$\begin{cases} \dot{y}^k + f(y^k) = 0, \ t \in (0, T) \\ y^k(0) = \alpha\left[y^k(T) - y^{k-1}(T)\right] + y_0 \end{cases}$$

1. $\rho \leq \begin{cases} \frac{\alpha e^{-T\lambda_{\min}}}{1-\alpha e^{-T\lambda_{\min}}}, & \text{Backward} - \text{Euler} \\ \frac{\alpha}{1-\alpha}, & \text{Trapezoidalrule} \end{cases}$
2. *Optimal* speedup=$\mathcal{O}(\frac{N_t}{k})$

Here $\lambda_{\min} \geq 0$ is the minimal real part of $\sigma(A)$

Gander-Wu. *Convergence analysis of a waveform relaxation method with head-tail coupled condition*. Numer Math-2019

ParaDiag-II: WR Variant

For nonlinear initial-value ODEs $\dot{y} + f(y) = 0$ with $y(0) = y_0$:

$$\dot{y}^k + f(y^k) = 0, \ y(0) = \alpha[y^k(T) - y^{k-1}(T)] + y_0$$

$$\hookrightarrow \left(\underbrace{\frac{1}{\Delta t}\begin{bmatrix} 1 & & & -\alpha \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}}_{:=C_\alpha \in \mathbb{R}^{N_t \times N_t}} \otimes I_x \right) \begin{bmatrix} y_1^k \\ y_2^k \\ \vdots \\ y_J^k \end{bmatrix} + \begin{bmatrix} f(y_1^k) \\ f(y_2^k) \\ \vdots \\ f(y_{N_t}^k) \end{bmatrix} = \begin{bmatrix} \frac{-\alpha y_{N_t}^{k-1} + y_0}{\Delta t} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\text{Jacobian Matrix} = C_\alpha \otimes I_x + \begin{bmatrix} \nabla f(y_1^{[l]}) & & & \\ & \nabla f(y_2^{[l]}) & & \\ & & \ddots & \\ & & & \nabla f(y_{N_t}^{[l]}) \end{bmatrix}$$

$$\approx C_\alpha \otimes I_x + I_t \otimes \left(\frac{1}{N_t} \sum_{n=1}^{N_t} \nabla f(y_n^{[l]}) \right)$$

ParaDiag-II: WR Variant

Proposition-C

For initial-value nonlinear ODEs $\dot{y}+f(y)=0$ with $y(0)=y_0$, assume

$$\langle f(y_1)-f(y_2), y_1-y_2\rangle \leq L\|y_1-y_2\|_2$$

Then, for the WR iteration at *continuous level*

$$\dot{y}^k+f(y^k)=0,\ y(0)=\alpha[y^k(T)-y^{k-1}(T)]+y_0$$

we have $\max_{t\in[0,T]}\|e^k\|_2 \leq \left(\frac{\alpha e^{-LT}}{1-\alpha e^{-LT}}\right)^k \|e^0(0)\|_2$

- Key idea: replace all $\left\{\nabla f(y_n^{[l]})\right\}$ by the averaged $\sum_{n=1}^{N_t}\frac{1}{N_t}\nabla f\left(y_n^{[l]}\right)$
- Simplified Newton method may diverge if $[0,T]$ is very large ! ($[0,T]=[0,T_1]\cup[T_1,T_2]\cup[T_{J-1},T]$)

Gander-Wu. *Convergence analysis of a waveform relaxation method with head-tail coupled condition.* Numer Math-2019

ParaDiag-II: WR Variant

Proposition-C

For initial-value nonlinear ODEs $\dot{y}+f(y)=0$ with $y(0)=y_0$, assume

$$\langle f(y_1)-f(y_2), y_1-y_2\rangle \leq L\|y_1-y_2\|_2$$

Then, for the WR iteration at *continuous level*

$$\dot{y}^k+f(y^k)=0,\ y(0)=\alpha[y^k(T)-y^{k-1}(T)]+y_0$$

we have $\max_{t\in[0,T]}\|e^k\|_2 \leq \left(\frac{\alpha e^{-LT}}{1-\alpha e^{-LT}}\right)^k \|e^0(0)\|_2$

- Key idea: replace all $\left\{\nabla f(y_n^{[l]})\right\}$ by the averaged $\sum_{n=1}^{N_t}\frac{1}{N_t}\nabla f\left(y_n^{[l]}\right)$
- Simplified Newton method may diverge if $[0,T]$ is very large ! $([0,T]=[0,T_1]\cup[T_1,T_2]\cup[T_{J-1},T])$

Gander-Wu. *Convergence analysis of a waveform relaxation method with head-tail coupled condition*. Numer Math-2019

ParaDiag-II: WR Variant

Example: Arenstorf Orbit

$$\begin{cases} \ddot{y}_1 = y_1 + 2\dot{y}_2 - a\frac{y_1 - b}{\phi_2(y_1,y_2)} - b\frac{y_1 + a}{\phi_1(y_1,y_2)} \\ \ddot{y}_2 = y_2 - 2\dot{y}_1 - a\frac{y_2}{\phi_2(y_1,y_2)} - b\frac{y_2}{\phi_1(y_1,y_2)} \end{cases}$$

$$y_1(0) = 0.994, \dot{y}_1(0) = 0, y_2(0) = 0, \dot{y}_2(0) = -2.00158510637908$$

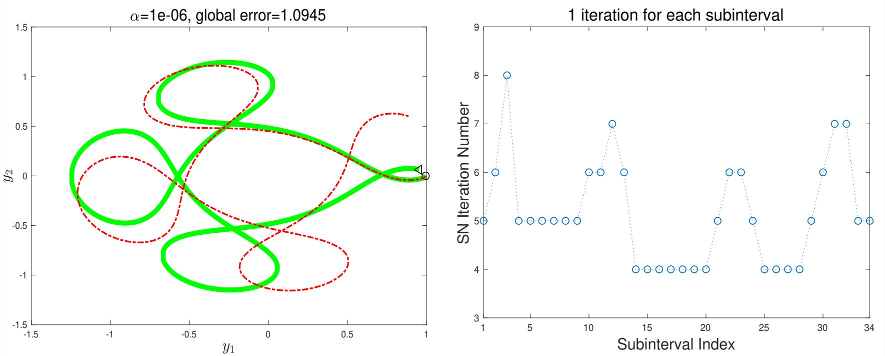
where $a = 0.012277471$, $b = 1 - a$ and

$$\phi_1 = \left((y_1 + a)^2 + y_2^2\right)^{\frac{3}{2}}, \ \phi_2 = \left((y_1 - b)^2 + y_2^2\right)^{\frac{3}{2}}$$

Numerical Setting

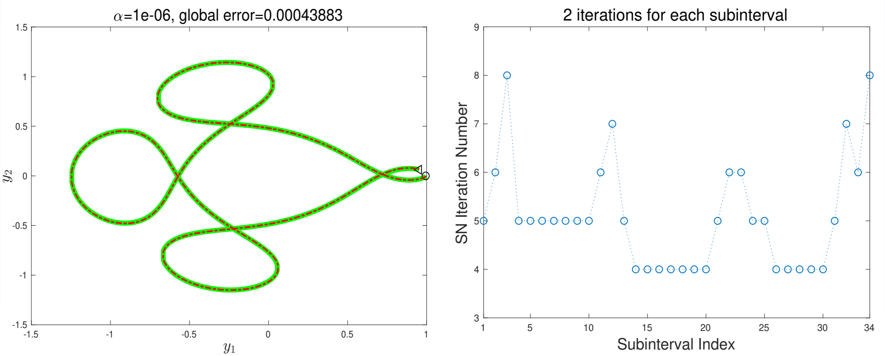
- $y_0^k = \alpha(y_{N_t}^k - y_{N_t}^{k-1}) + y_0$ with $y_{N_t}^0$ chosen randomly
- **divide** $[0, 17]$ **into 34 subintervals** $[0, \frac{1}{2}] \cup [\frac{1}{2}, 1] \cdots [\frac{33}{2}, 17]$
- $\Delta t = 2^{-12}$ for each subinterval

Diagonalization Technique: New Idea



- Do only **1** ParaDiag-II-WR iteration for each subinterval

Diagonalization Technique: New Idea



- Do **2** ParaDiag-II-WR iterations for each subinterval

ParaDiag-II

Parareal Variant

ParaDiag-II: Parareal Variant

For ODE problem

$$y'(t) = f(t, y(t)),\ y(0) = y_0,\ t \in (0, T)$$

The Parareal algorithm is:

$$y_{n+1}^{k+1} = \mathcal{G}\left(y_n^{k+1}, \Delta T\right) - \mathcal{G}\left(y_n^k, \Delta T\right) + \mathcal{F}^J\left(y_n^k, \Delta t\right)$$

i.e.

$$\underbrace{y_{n+1}^{k+1} = \mathcal{G}\left(y_n^{k+1}, \Delta T\right)}_{\mathbf{step-by-step\ !}} + \underbrace{\mathcal{F}^J\left(y_n^k, \Delta t\right) - \mathcal{G}\left(y_n^k, \Delta T\right)}_{:=\mathbf{b}_{n+1}^k}$$

- The coarse-grid-correction (CGC) is sequential in time
- CGC is often the bottleneck of speedup

ParaDiag-II: Parareal Variant

Suppose we need to solve initial-value ODEs:

$$\begin{cases} \dot{y} + Ay = 0 & t \in (0, T) \\ y(0) = y_0 & t = 0 \end{cases}$$

Idea:

Apply $\mathcal{F}$ using uniform Δt to $\begin{cases} \dot{y}^k + Ay^k = 0 & t \in (0, T) \\ y^k(0) = y_0 & t = 0 \end{cases}$

Apply $\mathcal{G}$ **using uniform** ΔT to $\dot{y}^{k+1} + Ay^{k+1} = 0$ with

$$y^{k+1}(0) = \alpha y^{k+1}(T) + y_0$$

ParaDiag-II: Parareal Variant

New CGC is

$$\Bigg[\underbrace{\frac{1}{\Delta T}\begin{pmatrix} I_x & & & -\alpha I_x \\ -I_x & I_x & & \\ & \ddots & \ddots & \\ & & -I_x & I_x \end{pmatrix}}_{C_\alpha \otimes I_x} + \underbrace{\begin{pmatrix} A & & & \\ & A & & \\ & & \ddots & \\ & & & A \end{pmatrix}}_{I_t \otimes A}\Bigg] \underbrace{\begin{pmatrix} y_1^{k+1} \\ y_2^{k+1} \\ \vdots \\ y_{N_t}^{k+1} \end{pmatrix}}_{\boldsymbol{y}^{k+1}} = \boldsymbol{b}^k$$

$$C_\alpha = \frac{1}{\Delta T}\begin{pmatrix} 1 & & & -\alpha \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix} \in \mathbb{C}^{N_t \times N_t}$$

ParaDiag-II: Parareal Variant

In each parareal iteration, the new CGC can be divided into 3 steps

$$(C_\alpha \otimes I_x + I_t \otimes A)\boldsymbol{y}^{k+1} = \boldsymbol{b}^k$$

$$\Rightarrow \begin{cases} (\Lambda_\alpha \mathbb{F}_{N_t} \otimes I_x)\boldsymbol{u} = \boldsymbol{b}^k & (a) \\ (\lambda_n I_x + A)\,\boldsymbol{w}_n = \boldsymbol{u}_n,\ n = 1, 2, \ldots, N_t & (b) \\ (\mathbb{F}_{N_t}^{-1}\Lambda_\alpha^{-1} \otimes I_x)\boldsymbol{y}^{k+1} = \boldsymbol{w} & (c) \end{cases}$$

Proposition-D

There is a threshold

$$\alpha^* = \frac{\rho_{\text{classical-parareal}}}{1 + \rho_{\text{classical-parareal}}}$$

such that $\rho_{\text{diag-parareal}}(\alpha) = \rho_{\text{classical-parareal}}$, *if* $\alpha \leq \alpha^*$

Wu. *Toward parallel CGC for the Parareal algorithm.* SiSC-2018

ParaDiag-II

Krylov Variant

ParaDiag-II: Krylov Variant

Optimal control of wave equation

$$\min_{u,\tilde{u}} \mathcal{L}(u,u) := \frac{1}{2}\|u-g\|^2_{L^2(\Omega\times(0,T))} + \frac{\gamma}{2}\|\tilde{u}\|^2_{L^2(\Omega\times(0,T))}$$

$$\text{s.t.} \begin{cases} u_{tt} - \Delta y = f + \tilde{u}, & \text{in } \Omega\times(0,T) \\ u = 0, & \text{on } \partial\Omega\times(0,T) \\ u(\cdot,0) = u_0, \quad u_t(\cdot,0) = u_1, & \text{in } \Omega \end{cases}$$

Implicit leap-frog time-discretization

$$\left(\underbrace{\begin{bmatrix} 1 & & & & \\ -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix}}_{:=B_1} \otimes I_x - \Delta t^2 \underbrace{\begin{bmatrix} 1 & & & & \\ 0 & 1 & & & \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \end{bmatrix}}_{:=B_2} \otimes \Delta_h \right) \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N_t} \end{bmatrix}}_{:=\boldsymbol{y}} = \boldsymbol{f} + \tilde{\boldsymbol{u}}$$

Li-Liu-Xiao. *A fast and stable preconditioned iterative method for optimal control problem of wave equations*. SiSC-2015.

ParaDiag-II: Krylov Variant

The saddle point matrix of the forward-backward coupled system

$$\boldsymbol{A} = \begin{bmatrix} B_1 & -\frac{\Delta t^2 \hat{I}_t}{\sqrt{\gamma}} \\ \frac{\Delta t^2 \check{I}_t}{\sqrt{\gamma}} & B_1^\top \end{bmatrix} \otimes I_x - \frac{\Delta t^2}{2} \begin{bmatrix} B_2 & \\ & B_2^\top \end{bmatrix} \otimes \Delta_h$$

where

$$\hat{I}_t = \begin{bmatrix} \frac{1}{2} & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}, \quad \check{I}_t = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & \frac{1}{2} \end{bmatrix}$$

Preconditioner

$$\boldsymbol{P} := \begin{bmatrix} C_1 & -\frac{\Delta t^2 I_t}{\sqrt{\gamma}} \\ \frac{\Delta t^2 I_t}{\sqrt{\gamma}} & C_1^\top \end{bmatrix} \otimes I_x - \frac{\Delta t^2}{2} \begin{bmatrix} C_2 & \\ & C_2^\top \end{bmatrix} \otimes \Delta_h$$

ParaDiag-II: Krylov Variant

$$B_1 = \begin{bmatrix} 1 & & & & \\ -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix}, \quad C_1 = \begin{bmatrix} 1 & & & {\color{red}1} & {\color{red}-2} \\ -2 & 1 & & & {\color{red}1} \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 1 & & & & \\ 0 & 1 & & & \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 1 & & & {\color{red}1} & {\color{red}0} \\ 0 & 1 & & & {\color{red}1} \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \end{bmatrix}$$

ParaDiag-II: Krylov Variant

- $\boldsymbol{P} := \begin{bmatrix} C_1 & -\frac{\Delta t^2 I_t}{\sqrt{\gamma}} \\ \frac{\Delta t^2 I_t}{\sqrt{\gamma}} & C_1^\top \end{bmatrix} \otimes I_x - \frac{\Delta t^2}{2} \boxed{\begin{bmatrix} C_2 & \\ & C_2^\top \end{bmatrix}} \otimes \Delta_h$
- $\boldsymbol{P} = \underbrace{\left(\begin{bmatrix} C_1 C_2^{-1} & -\frac{\Delta t^2 (C_2^{-1})^\top}{\sqrt{\gamma}} \\ \frac{\Delta t^2 C_2^{-1}}{\sqrt{\gamma}} & C_1^\top (C_2^{-1})^\top \end{bmatrix} \otimes I_x - \frac{\Delta t^2}{2} \begin{bmatrix} I_t & \\ & I_t \end{bmatrix} \otimes \Delta_h \right)}_{=:\widetilde{\boldsymbol{P}}} \left(\begin{bmatrix} C_2 & \\ & C_2^\top \end{bmatrix} \otimes I_x \right)$
- For any input vector $\boldsymbol{r}$, compute $\boldsymbol{s} = \boldsymbol{P}^{-1}\boldsymbol{r}$ via

$$\tilde{\boldsymbol{s}} := \begin{bmatrix} \tilde{\boldsymbol{s}}_1 \\ \tilde{\boldsymbol{s}}_2 \end{bmatrix} = \widetilde{\boldsymbol{P}}^{-1}\boldsymbol{r}, \ \boldsymbol{s} = \begin{bmatrix} (C_2^{-1} \otimes I_x)\tilde{\boldsymbol{s}}_1 \\ ((C_2^{-1})^\top \otimes I_x)\tilde{\boldsymbol{s}}_2 \end{bmatrix}$$

ParaDiag-II: Krylov Variant

Proposition-E

$$\widetilde{\boldsymbol{P}} = (V \otimes I_x)\left(\begin{bmatrix}\Sigma_1 & \\ & \Sigma_2\end{bmatrix} \otimes I_x - \frac{\Delta t^2}{2}\begin{bmatrix}I_t & \\ & I_t\end{bmatrix} \otimes \Delta_h\right)(V^{-1} \otimes I_x).$$

where Σ_1 and Σ_2 are diagonal matrices and

$$V = \begin{bmatrix}\mathbb{F}_{N_t} & \\ & \mathbb{F}_{N_t}\end{bmatrix}\begin{bmatrix}I_t & -\mathrm{i}D \\ \mathrm{i}D & I_t\end{bmatrix} \text{ (FFT is applicable)}$$

- The diagonalization of $\widetilde{\boldsymbol{P}}$ is not unique: for any invertible diagonal matrix $\widetilde{D}$ the factorization still holds by replacing V by $V\widetilde{D}$
- The above diagonalization is *Optimal* because $\mathrm{Cond}_2(V) = 1$ (proved)

Wu-Liu. *A parallel-in-time block-circulant preconditioner for optimal control of wave equations*. SiSC-2020

ParaDiag-II: Krylov Variant

Consider a 2D problem posed on $\Omega \times (0,T) = (0,1)^2 \times (0,2)$:

$$u_0(x,y) = \sin(\pi x)\sin(\pi y), \quad u_1(x,y) = \sin(\pi x)\sin(\pi y),$$

$$f(x,y,t) = (1+2\pi^2)e^t \sin(\pi x)\sin(\pi y) - \frac{1}{\gamma}(t-T)^2 \sin(\pi x)\sin(\pi y),$$

$$g(x,y,t) = (e^t + 2 + 2\pi^2(t-T)^2)\sin(\pi x)\sin(\pi y).$$

$\texttt{tol} = 10^{-7}$	$\gamma = 10^{-2}$		$\gamma = 10^{-4}$		$\gamma = 10^{-6}$		$\gamma = 10^{-8}$	
(N_x, N_x, N_t)	It_1	It_2	It_1	It_2	It_1	It_2	It_1	It_2
(16,16,17)	5	11	5	14	5	11	4	4
(32,32,33)	5	10	5	18	5	17	5	8
(64,64,65)	5	11	5	20	5	26	5	10
(128,128,129)	11	12	5	21	5	38	5	19
(256,256,257)	17	12	5	23	5	46	5	26

- It_1: GMRES iteration number using ParaDiag-II preconditioner $\boldsymbol{P}$
- It_2: GMRES iteration number using MSC preconditioner (matching Schur complement)
- γ is the regularization parameter

Many Thanks for Your Attention !

https://github.com/wushulin/ParaDiag
(Algorithms+Matlab Codes+Parallel Codes)