Fast and Accurate Simulation of Close-to-Touching Discs in 2D Stokes Flow

Dhairya Malhotra, Mariana Martinez Aguilar, Dan Fortunato

SciCADE 2024

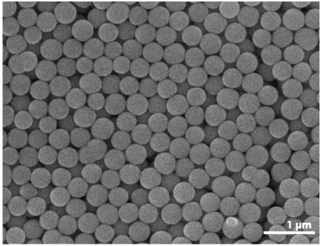
July 15, 2024

FLATIRON INSTITUTE

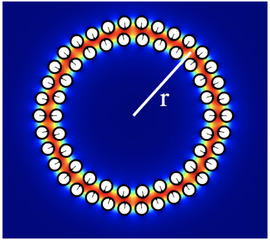
Particulate Flows

Applications:

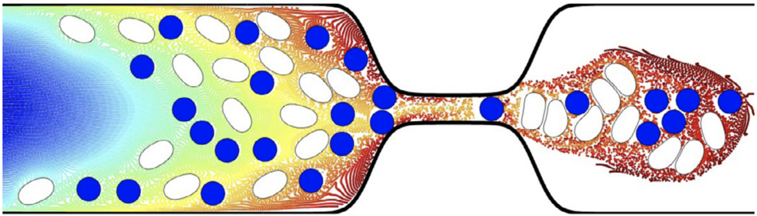
- Colloidal suspensions, emulsions
- Cellular biology, complex fluids
- Janus particles, clustering, self-assembly



Source: Wikipedia



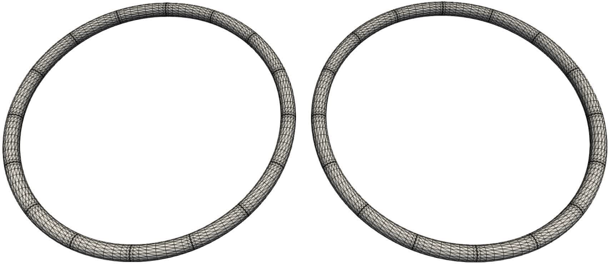
Fu, Quaife, Ryham, Young (2021)

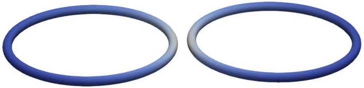


Lu, Rahimian, Zorin (JCP 2017)

Challenges

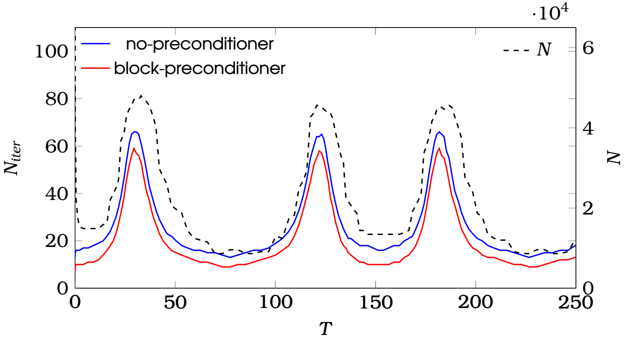
Example: Sedimenting rings in Stokes flow

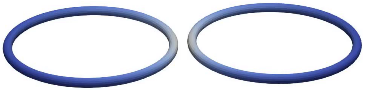




Challenges

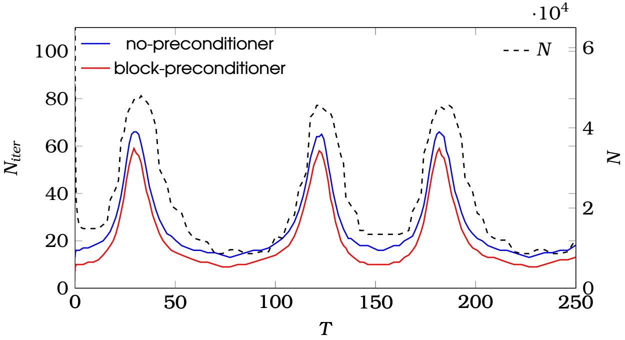
Example: Sedimenting rings in Stokes flow

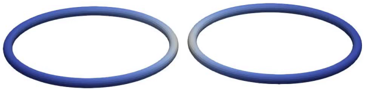




Challenges

Example: Sedimenting rings in Stokes flow





Close interactions:

$\sim 5\times$ **more unknowns,** $\sim 5\times$ **more GMRES iterations,** $\sim 5\times$ **smaller** ΔT

$> 100\times$ **slower!**

Problem Setup - Stokes Mobility

- n identical rigid discs $\Omega = \sum_{i=1}^{n} \Omega_i$

  given radius R, centers $\boldsymbol{x}_i^c$, forces $\boldsymbol{F}_i$, torques T_i,

  velocity $\boldsymbol{V}(\boldsymbol{x}) = \boldsymbol{v}_i + \boldsymbol{\omega}_i \times (\boldsymbol{x} - \boldsymbol{x}_i^c)$.

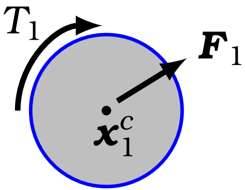
- Stokesian fluid in $\mathbb{R}^3 \setminus \Omega$

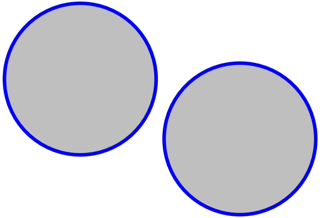
$$\Delta \boldsymbol{u} - \nabla p = 0, \quad \nabla \cdot \boldsymbol{u} = 0,$$

$$\boldsymbol{u} \to 0 \text{ as } \boldsymbol{x} \to \infty.$$

- Boundary conditions on $\partial\Omega$,

$$\boldsymbol{u} = \boldsymbol{V} + \boldsymbol{u}_s.$$



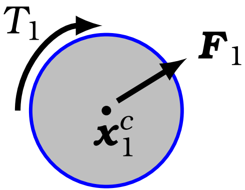


Problem Setup - Stokes Mobility

- n identical rigid discs $\Omega = \sum_{i=1}^{n} \Omega_i$

  given radius R, centers $\boldsymbol{x}_i^c$, forces $\boldsymbol{F}_i$, torques T_i,

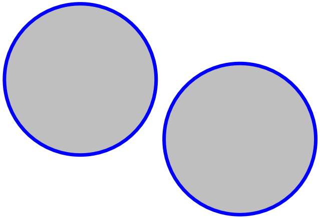
  velocity $\boldsymbol{V}(\boldsymbol{x}) = \boldsymbol{v}_i + \boldsymbol{\omega}_i \times (\boldsymbol{x} - \boldsymbol{x}_i^c)$.



- Stokesian fluid in $\mathbb{R}^3 \setminus \Omega$

$$\Delta \boldsymbol{u} - \nabla p = 0, \quad \nabla \cdot \boldsymbol{u} = 0,$$

$$\boldsymbol{u} \to 0 \ \text{as} \ \boldsymbol{x} \to \infty.$$



- Boundary conditions on $\partial\Omega$,

$$\boldsymbol{u} = \boldsymbol{V} + \boldsymbol{u}_s.$$

unknown: $\boldsymbol{V}(\boldsymbol{u}_i, \boldsymbol{\omega}_i)$

Boundary Integral Formulation

Represent fluid velocity: $\boldsymbol{u}(\boldsymbol{x}) = \int_{\partial\Omega} S(\boldsymbol{x}-\boldsymbol{y})\boldsymbol{\nu}(\boldsymbol{y}) + \int_{\partial\Omega} D(\boldsymbol{x}-\boldsymbol{y})\boldsymbol{\sigma}(\boldsymbol{y})$

Boundary Integral Formulation

Represent fluid velocity: $\boldsymbol{u} = \mathcal{S}[\boldsymbol{\nu}(\boldsymbol{F}_i, T_i)] + \mathcal{D}[\boldsymbol{\sigma}]$

Boundary Integral Formulation

Represent fluid velocity: $\boldsymbol{u} = \mathcal{S}[\boldsymbol{\nu}(\boldsymbol{F}_i, T_i)] + \mathcal{D}[\boldsymbol{\sigma}]$

and rigid body velocity: $\boldsymbol{V} = -\sum_{i=1}^{6n} \mathfrak{v}_i \mathfrak{v}_i^T \boldsymbol{\sigma}$

Boundary Integral Formulation

Represent fluid velocity: $\boldsymbol{u} = \mathcal{S}[\boldsymbol{\nu}(\boldsymbol{F}_i, T_i)] + \mathcal{D}[\boldsymbol{\sigma}]$

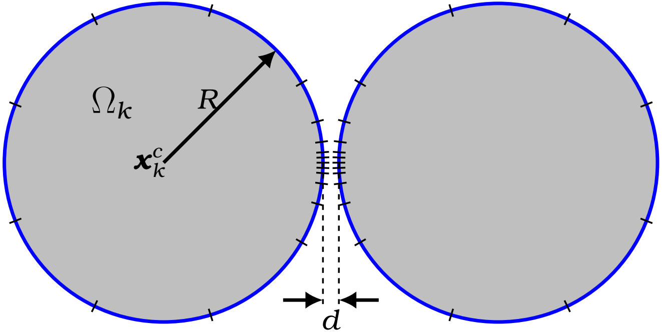
and rigid body velocity: $\boldsymbol{V} = -\sum_{i=1}^{6n} \mathfrak{v}_i \mathfrak{v}_i^T \boldsymbol{\sigma}$

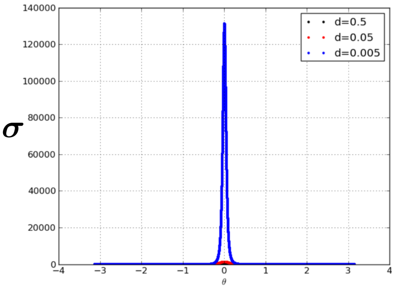
Applying boundary conditions $(\boldsymbol{u} = \boldsymbol{V} + \boldsymbol{u}_s \text{ on } \partial\Omega)$,

$$(I/2 + D)\,\boldsymbol{\sigma} + \sum_{i=1}^{6n} \mathfrak{v}_i \mathfrak{v}_i^T \boldsymbol{\sigma} = \boldsymbol{u}_s - S\,\boldsymbol{\nu}$$

(Pozrikidis - Boundary Integral and Singularity Methods for Linearized Viscous Flow)

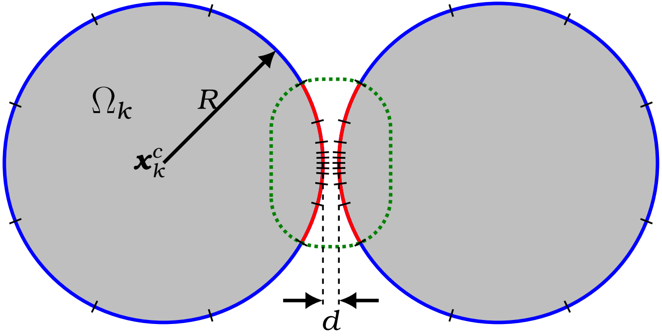
Nyström Discretization

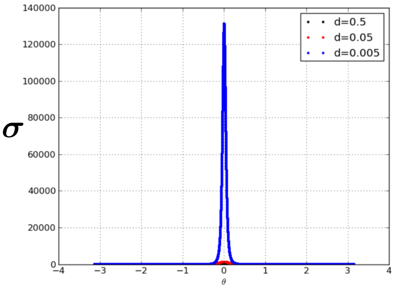




- Discretize $\partial\Omega$ into panels.
- Layer-potential operators:
  - adaptive quadrature for near integrals
  - special quadrature for singular integrals
- Solve BIE: $K\sigma = g$

Nyström Discretization

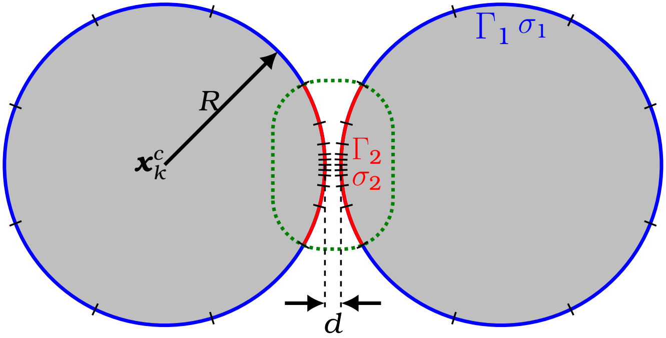




- Discretize $\partial\Omega$ into panels.
- Layer-potential operators:
  - adaptive quadrature for near integrals
  - special quadrature for singular integrals
- Solve BIE: $K\sigma = g$

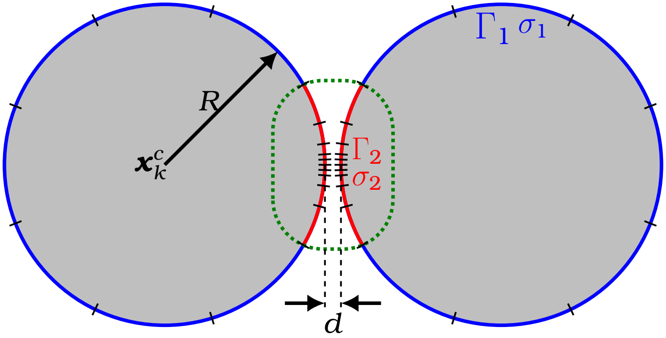
Compress close-interactions, and interpolate in d.

Compressing Close Interactions



$$\begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12} \\ \mathcal{K}_{21} & \mathcal{K}_{22} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

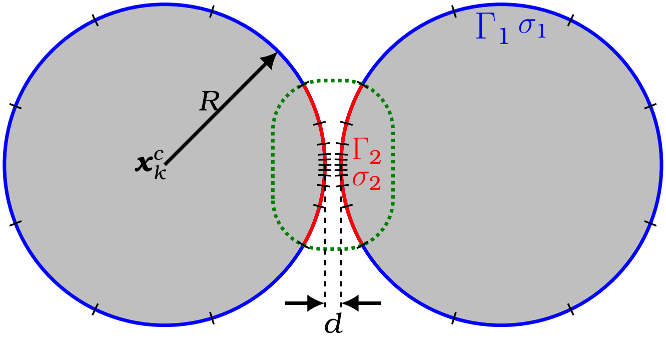
Compressing Close Interactions



$$\begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12} \\ \mathcal{K}_{21} & \mathcal{K}_{22} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

Build compression using RCIP method of Helsing

Compressing Close Interactions



$$\begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12} \\ \mathcal{K}_{21} & \mathcal{K}_{22} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

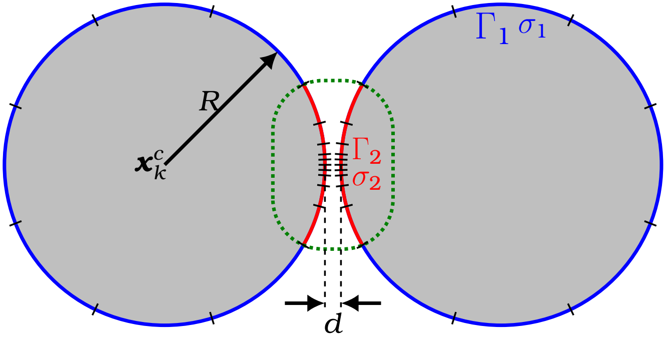
Build compression using RCIP method of Helsing

Right precondition with $\mathcal{K}_{22}^{-1}$:

$$\begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12}\mathcal{K}_{22}^{-1} \\ \mathcal{K}_{21} & I \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \overline{\sigma}_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

where $\overline{\sigma}_2 = \mathcal{K}_{22}\sigma_2$

Compressing Close Interactions



$$\begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12} \\ \mathcal{K}_{21} & \mathcal{K}_{22} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

Build compression using RCIP method of Helsing

Right precondition with $\mathcal{K}_{22}^{-1}$:

$$\begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12}\mathcal{K}_{22}^{-1} \\ \mathcal{K}_{21} & I \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \overline{\sigma}_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}$$

where $\overline{\sigma}_2 = \mathcal{K}_{22}\sigma_2$

$\Longrightarrow$ coarsen

$$\begin{pmatrix} K_{11} & K_{12}^c R \\ K_{21}^c & I \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \overline{\sigma}_2^c \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2^c \end{pmatrix}$$

where $R = W_c^{-1} P^T W_f K_{22}^{-1} P$.

Computing R_d On-the-Fly

Cost of computing R_d:

Direct: $\mathcal{O}((q \log d)^3)$

RCIP: $\mathcal{O}(q^3 \log d)$ [$\mathcal{O}(q^6 \log d)$ in 3D]

Computing R_d On-the-Fly

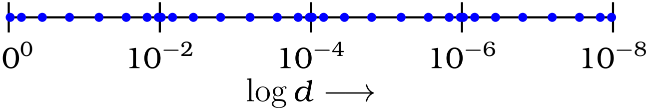
Cost of computing R_d:

Direct: $\mathcal{O}((q \log d)^3)$

RCIP: $\mathcal{O}(q^3 \log d)$ [$\mathcal{O}(q^6 \log d)$ in 3D]

Interpolating R_d: Interpolated Compressed Inverse Preconditioning (ICIP)

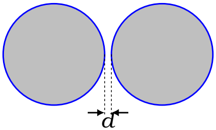
$$R_{ij}(d) = \sum_{k=0}^{p-1} \alpha_k T_k(\log d)$$



Interpolation cost: $\mathcal{O}(q^2 p)$ [$\mathcal{O}(q^4 p)$ in 3D]

Convergence Results

Errors (Stokes mobility with 2 discs):

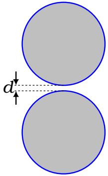


d	Adaptive Discretization	Interpolating R_d $p = 8$	$p = 16$	$p = 24$	$p = 32$
1e-1	7.6e-15	1.0e-4	2.9e-07	2.1e-09	9.1e-12
1e-3	4.4e-13	3.4e-5	5.6e-10	4.8e-14	
1e-5	9.0e-09	1.5e-5	2.1e-12		
1e-7	4.3e-07	1.7e-5	4.1e-11		
1e-8	5.3e-08	6.3e-4	3.9e-09		

p: interpolation order

GMRES Iterations

Iteration counts for ϵ_{GMRES}=1e-8



Adaptive discretization:

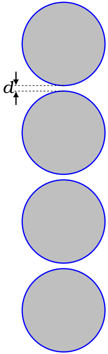
N_{disc}	$d =$1e-1	1e-2	1e-3	1e-4	1e-5	1e-6	1e-7
2	15	37	104	337	1283	1848	2344

Interpolated Compressed Inverse Preconditioning (ICIP):

N_{disc}	$d =$1e-1	1e-2	1e-3	1e-4	1e-5	1e-6	1e-7
2	18	20	21	21	21	21	21

GMRES Iterations

Iteration counts for ϵ_{GMRES}=1e-8



Adaptive discretization:

N_{disc}	$d =$1e-1	1e-2	1e-3	1e-4	1e-5	1e-6	1e-7
2	15	37	104	337	1283	1848	2344
4	25	75	271	1134	3770	5301	6620

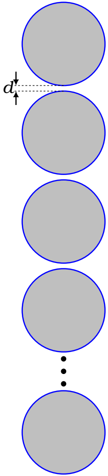
Interpolated Compressed Inverse Preconditioning (ICIP):

N_{disc}	$d =$1e-1	1e-2	1e-3	1e-4	1e-5	1e-6	1e-7
2	18	20	21	21	21	21	21
4	28	34	36	37	37	37	37

GMRES Iterations

Iteration counts

for ϵ_{GMRES}=1e-8



Adaptive discretization:

N_{disc}	$d =$1e-1	1e-2	1e-3	1e-4	1e-5	1e-6	1e-7
2	15	37	104	337	1283	1848	2344
4	25	75	271	1134	3770	5301	6620
16	35	147	629	2754	>8000		
64	36	148	683	3094			
256	37	149	683	3094			

Interpolated Compressed Inverse Preconditioning (ICIP):

N_{disc}	$d =$1e-1	1e-2	1e-3	1e-4	1e-5	1e-6	1e-7
2	18	20	21	21	21	21	21
4	28	34	36	37	37	37	37
16	46	71	74	80	86	87	88
64	49	96	108	131	186	237	251
256	49	98	110	134	220	371	608

Accelerating GMRES Solves

- Forward Euler: n-th time step
  - solve BIE using GMRES: $A_{y_n}\sigma_n = b_{y_n}$
  - advance to t_{n+1}: $y_{n+1} = y_n + h\, v(\sigma_n)$

Accelerating GMRES Solves

- Forward Euler: n-th time step
  - solve BIE using GMRES: $A_{y_n}\sigma_n = b_{y_n}$
  - advance to t_{n+1}: $y_{n+1} = y_n + h\,v(\sigma_n)$
- Use σ_{n-1} as initial guess to GMRES

Accelerating GMRES Solves

- Forward Euler: $n\text{-}th$ time step
  - solve BIE using GMRES: $A_{y_n}\sigma_n = b_{y_n}$
  - advance to t_{n+1}: $y_{n+1} = y_n + h\, v(\sigma_n)$
- Use σ_{n-1} as initial guess to GMRES : doesn't work well

Accelerating GMRES Solves

- Forward Euler: n-th time step
  - solve BIE using GMRES: $A_{y_n}\sigma_n = b_{y_n}$
  - advance to t_{n+1}: $y_{n+1} = y_n + h\, v(\sigma_n)$
- Use σ_{n-1} as initial guess to GMRES : doesn't work well

- Re-use Krylov subspace from previous time step?

Accelerating GMRES Solves

- Forward Euler: n-th time step
  - solve BIE using GMRES: $A_{y_n}\sigma_n = b_{y_n}$
  - advance to t_{n+1}: $y_{n+1} = y_n + h\, v(\sigma_n)$
- Use σ_{n-1} as initial guess to GMRES : doesn´t work well
- Re-use Krylov subspace from previous time step?
  - Krylov subspace: $X \leftarrow [b,\ Ab,\ \cdots,\ A^{k\text{-}1}b]$
  - Compute QR decomposition: $QR \leftarrow AX$
  - Preconditioner: $P := I - QQ^T + XR^{-1}Q^T$

$$P\,Ax = x \quad \text{for all} \quad x \in span(X)$$
$$P\,y = y \quad \text{for all} \quad y \perp span(X)$$

Accelerating GMRES Solves

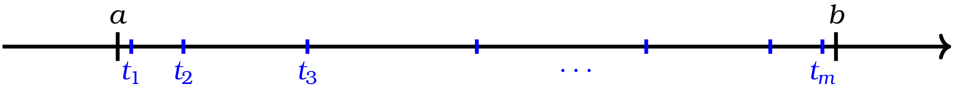
- Forward Euler: $n\text{-}th$ time step
  - solve BIE using GMRES: $A_{y_n}\sigma_n = b_{y_n}$
  - advance to t_{n+1}: $y_{n+1} = y_n + h\, v(\sigma_n)$
- Use σ_{n-1} as initial guess to GMRES : doesn't work well

- Re-use Krylov subspace from previous time step?
  - Krylov subspace: $X \leftarrow [b,\ Ab,\ \cdots,\ A^{k\text{-}1}b]$
  - Compute QR decomposition: $QR \leftarrow AX$
  - Preconditioner: $P := I - QQ^T + XR^{-1}Q^T$

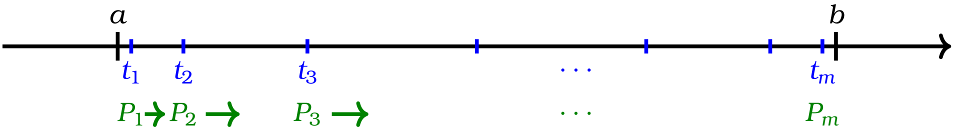
$$P\,Ax = x \quad \text{for all} \quad x \in span(X)$$
$$P\,y = y \quad \text{for all} \quad y \perp span(X)$$

Similar to subspace deflation

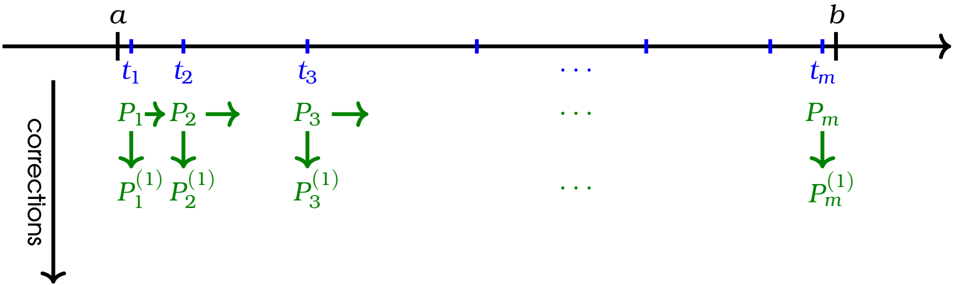
Krylov Preconditioning with SDC



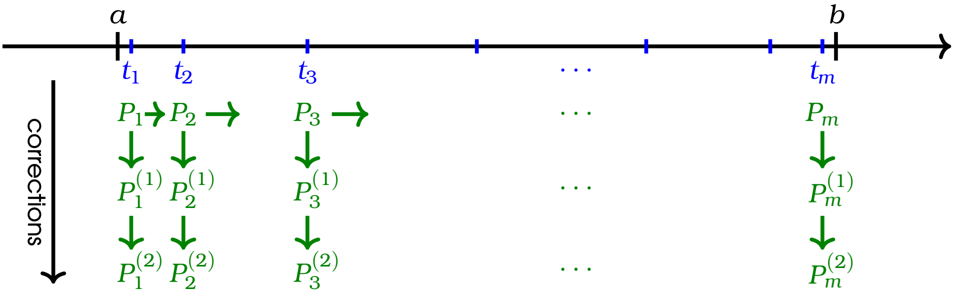
Krylov Preconditioning with SDC



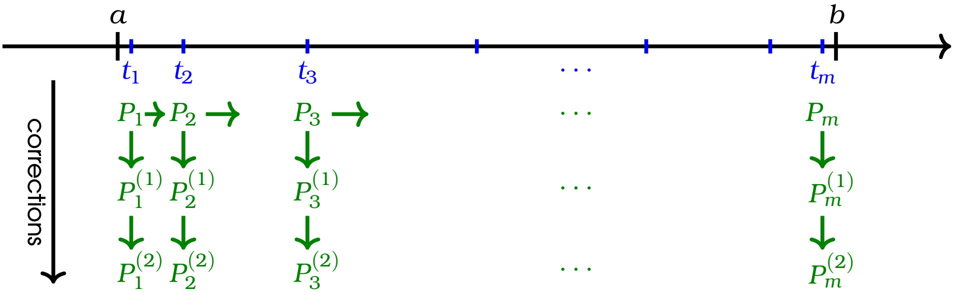
Krylov Preconditioning with SDC



Krylov Preconditioning with SDC



Krylov Preconditioning with SDC

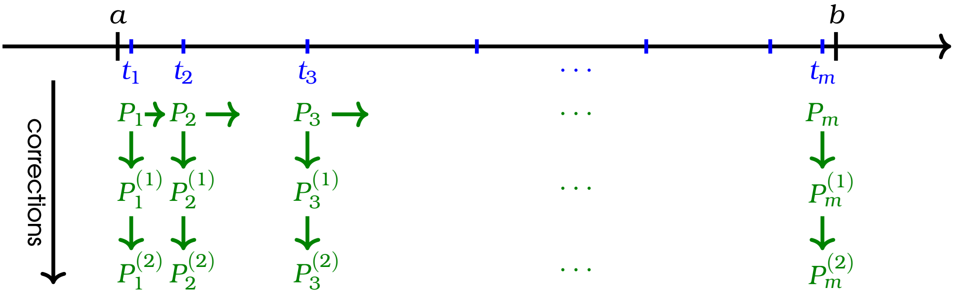


GMRES iter without preconditioner:

correction ↓ \ sub-step →				
66	66	66	66	66
66	66	66	66	66
66	66	66	66	66
66	66	66	66	66

Krylov Preconditioning with SDC

FLATIRON INSTITUTE



GMRES iter without preconditioner:

correction ↓ \ sub-step →					
	66	66	66	66	66
	66	66	66	66	66
	66	66	66	66	66
	66	66	66	66	66

GMRES iter with preconditioner:

correction ↓ \ sub-step →					
	66	30	22	45	30
	35	17	33	28	24
	8	4	14	5	12
	1	1	2	2	4

Numerical Results

Average iterations per GMRES solve: 30 (130 without Krylov preconditioner)

32 discs (36K unknowns), **minimum distance: 1e-4**, **10-th order SDC**

7-digits accuracy in quadratures, GMRES solve, and time-stepping.

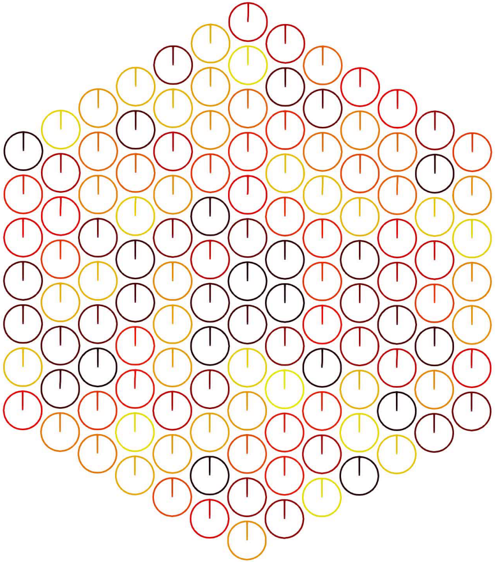
Numerical Results - Sedimentation Flow

200K unknowns
127 discs.

Minimum distance: 1e-4

10-th order adaptive SDC

5-digits accuracy in quadratures, GMRES solve, and time-stepping.



Conclusions

- Interpolated Compressed Inverse Preconditioning (ICIP)
  - fast pairwise preconditioner using interpolation of precomputed compressed interaction operators.
- Krylov subspace preconditioner
  - precondition using Krylov subspace constructed during the previous time steps.
  - accelerate time-stepping, particularly Picard iterations in spectral deferred correction.

Ongoing work:

- Flows in confined geometries.
- Extension to spheres in 3D.

Limitations:

- Restricted to simple geometries (disks, flat planes)