Employing HPC for Analyzing Nonlinear PDE Systems Beyond Simulation

Jonas Thies[*] Weiyan Song[*,†] Sven Baars[†] Fred Wubs[†]

[*] German Aerospace Center
Simulation and Software Technology

[†] University of Groningen
Department of Mathematics and Computer Science

SPPEXA

project ESSEX

DLR

Problem Setting: Nonlinear PDE System

2^{nd} order PDE after space discretization

- $\mathrm{M}\frac{\partial \Phi}{\partial t} = \mathrm{F}(\Phi, t)$
- with suitable boundary and initial conditions

Steady state; Φ as $t \to \infty$.
Standard technique: time stepping

- may take very long
- no information stability

physical difficulty: low frequency modes affect solution on very long time scales

Example: 3D Boussinesq equations

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= -\left((uu)_x + (vu)_y + (wu)_z\right) - p_x + \nu \nabla^2 u \\
\frac{\partial v}{\partial t} &= -\left((uv)_x + (vv)_y + (wv)_z\right) - p_y + \nu \nabla^2 v \\
\frac{\partial w}{\partial t} &= -\left((uw)_x + (vw)_y + (ww)_z\right) - p_z + \nu \nabla^2 w + g\alpha T \\
\frac{\partial T}{\partial t} &= -\left((uT)_x + (vT)_y + (wT)_z\right) + \kappa \nabla^2 T
\end{aligned}
$$

$$u_x + v_y + w_z = 0$$

with ν the *kinematic viscosity*, κ the *thermal diffusivity* and α the *heat expansion coefficient*.

Jacobian has saddle point structure:

$$\mathrm{J} = \begin{pmatrix} A & -\mathrm{Grad} \\ \mathrm{Div} & 0 \end{pmatrix}.$$

Rayleigh-Bénard Convection

- Cube-shaped domain
- heated from below
- Rayleigh-Number
  $\mathrm{Ra} = \frac{\alpha g \Delta T d^3}{\nu \kappa}$

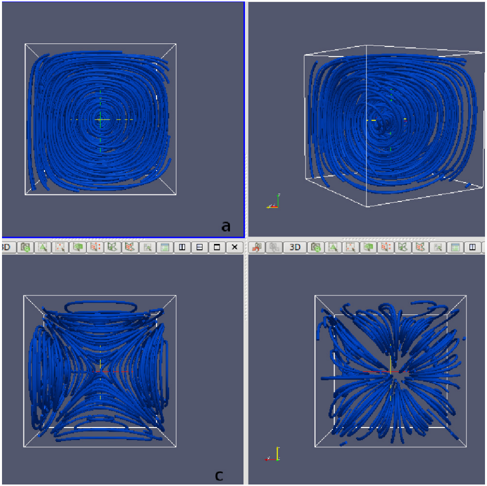
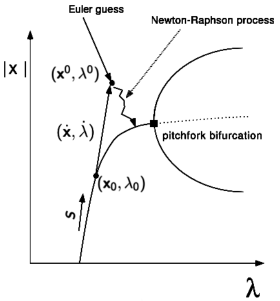


Figure: **Flow patterns near the first three primary bifurcations**
(a) x/y roll,
(b) diagonal roll,
(c) four rolls,
(d) toroidal roll

Pseudo-Arclength Continuation

- globalization strategy for Newton's: step through parameter space
- systematically compute 'hard-to-get' solutions



In order to e.g. get around turning points:

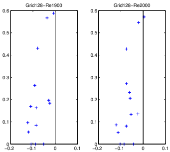
$$\begin{aligned} F(x, \lambda) &= 0 \\ \dot{x}_0^T(x - x_0) + \dot{\lambda}_0(\lambda - \lambda_0) &= \Delta s \end{aligned}$$

- Keller *et. al* 1977
- implementation: LOCA, `https://trilinos.org/`
- requires solving *bordered system* with $\hat{J} = \begin{pmatrix} F_x & F_\lambda \\ \dot{x}_0^T & \dot{\lambda} \end{pmatrix}$

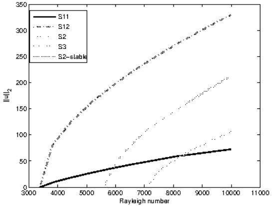
Linear Stability Analysis

- steady state is linearly stable if all eigenvalues of J are negative
- bifurcation points are locations in parameter space where the spectral properties of J change
- branch switching: force system in the direction of an unstable eigenvector at a bifurcation point

Figure: Hopf bifurcation in 3D lid-driven cavity



Bifurcation diagram for Rayleigh-Bénard



Jacobi-Davidson: Newton's as an Eigensolver

- Eigenvalue problem: solve $\mathrm{A}x - \lambda x = 0$ for (x, λ)
- Apply inexact Newton
- JDQR: subspace acceleration, locking and restart (Fokkema'99)

Jacobi-Davidson correction equation

- current approximation: $A\tilde{v} - \tilde{\lambda}\tilde{v} = r$,
- previously converged Schur vectors $\left(q_1, \ldots, q_k\right) = Q$
- solve approximately $(A - \tilde{\lambda}I)\Delta v = -r, \Delta v \perp \tilde{Q} = (Q, \tilde{v})$
- use some steps of preconditioned GMRES

Implementation: `https://bitbucket.org/essex/phist`

DLR

Projections and Bordering in JDQR

Inner solver: need to compute $t \perp \tilde{Q}$

Standard approach:

$$(I - \tilde{Q}\tilde{Q}^T)(A - \tilde{\lambda} I)$$

as operator, and

$$(I - \tilde{Q}_K(\tilde{Q}^T\tilde{Q}_K)^{-1}\tilde{Q}^T)K^{-1}$$

as (left) preconditioner, where K is a preconditioner for A or $A - \tilde{\lambda} I$ and $\tilde{Q}_K = K^{-1}\tilde{Q}$.

Equivalent formulation:

$$\begin{pmatrix} A - \tilde{\lambda} I & \tilde{Q} \\ \tilde{Q}^T & 0 \end{pmatrix}$$

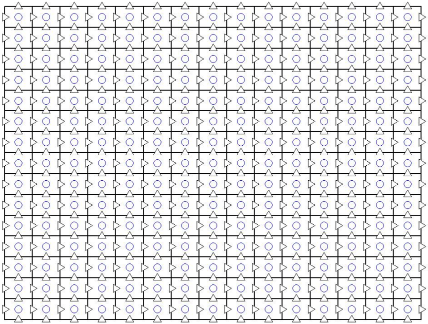
as operator, and

$$\begin{pmatrix} K & \tilde{Q}_K \\ \tilde{Q}^T & \tilde{Q}^T\tilde{Q}_K \end{pmatrix}^{-1}$$

as preconditioner

HYbrid Multi-Level Solver (HYMLS)

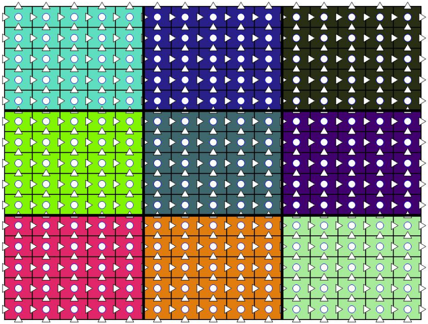
A special-purpose, semi-geometric ILU that recursively



HYbrid Multi-Level Solver (HYMLS)

A special-purpose, semi-geometric ILU that recursively

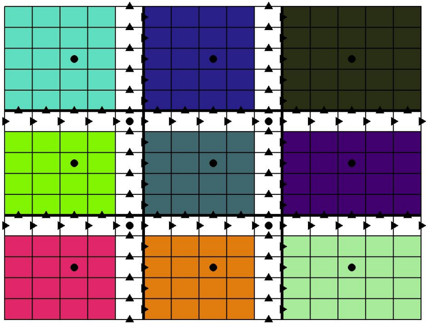
- partitions the domain



HYbrid Multi-Level Solver (HYMLS)

A special-purpose, semi-geometric ILU that recursively

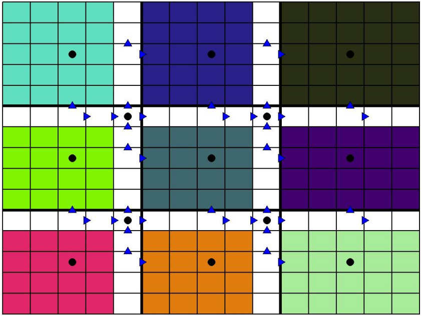
- partitions the domain
- eliminates the interior variables



HYbrid Multi-Level Solver (HYMLS)

A special-purpose, semi-geometric ILU that recursively

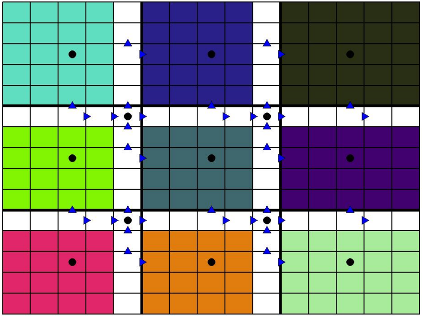
- partitions the domain
- eliminates the interior variables
- applies orth. transformations+dropping on the Schur complement



HYbrid Multi-Level Solver (HYMLS)

A special-purpose, semi-geometric ILU that recursively

- partitions the domain
- eliminates the interior variables
- applies orth. transformations+dropping on the Schur complement
- ... and ends up with a reduced matrix with similar structure and properties



HYMLS: Properties

Nice..

- dropping does not cause breakdowns
- properties of PDE preserved on coarser level
- grid independent converence rate
- no segregation, no nested iterations

But...

- only well understood for structured (Arakawa) grids
- $\mathcal{O}(N \log N)$ only achieved in 2D up to now
- have to form Schur complement (more expensive than MG)

Implementation: using Trilinos, available on request (`Jonas.Thies@DLR.de`)

HYMLS: Bordered Systems

Bordered Schur complement ILU

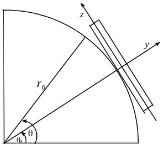
$$\begin{pmatrix} A_{11} & A_{12} & W_1 \\ A_{21} & A_{22} & W_2 \\ V_1^T & V_2^T & C \end{pmatrix} \approx \begin{pmatrix} L_{11} & 0 & 0 \\ A_{21} U_{11}^{-1} & \multicolumn{2}{c}{\multirow{2}{*}{$\hat{L}_{22}$}} \\ \hat{V}_1^T & & \end{pmatrix} \begin{pmatrix} U_{11} & L_{11}^{-1} A_{12} & \hat{W}_1 \\ 0 & \multicolumn{2}{c}{\multirow{2}{*}{$\hat{U}_{22}$}} \\ 0 & & \end{pmatrix}$$

where

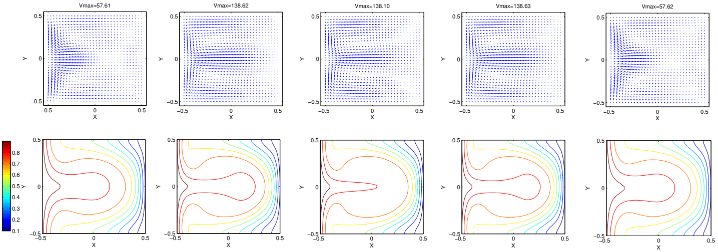
$$\hat{L}_{22} \hat{U}_{22} \approx \begin{pmatrix} A_{22} - A_{21} A_{11}^{-1} A_{12} & W_2 - A_{21} \hat{W}_1 \\ V_2^T - \hat{V}_1^T A_{12} & C - \hat{V}_1^T \hat{W}_1 \end{pmatrix}, \hat{W}_1 = L_{11}^{-1} W_1, \hat{V}_1^T = V_1^T U_{11}^{-1}.$$

- border already dense $\Rightarrow$ no dropping needed
- only need to re-factor last-level Schur complement
- preconditioner remains non-singular in turning points and Jacobi-Davidson

Rotating Differentially Heated Basin



- idealized 'thermally driven ocean'
- aspect ration 1:200
- rotating, driven by temperature difference north/south



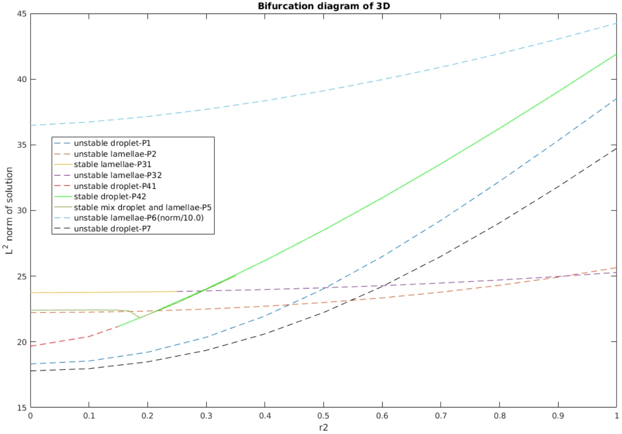
Turing Problem

Reaction-Diffusion problem

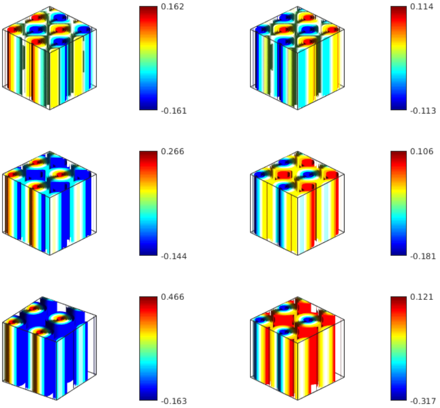
$$
\begin{aligned}
\frac{\partial u}{\partial t} &= D\delta\nabla^2 u + \alpha u(1 - r_1 v^2) + v(1 - r_2 u) \\
\frac{\partial v}{\partial t} &= \delta\nabla^2 v + v(\beta + \alpha r_1 uv) + u(\gamma + r_2 v)
\end{aligned}
\tag{1}
$$

- 2D: spot and stripe patterns
- can be solved using AMG
- TODO: JaDa + AMG has to be fixed for unsymmetric problems

3D Turing: many patterns and bifurcations

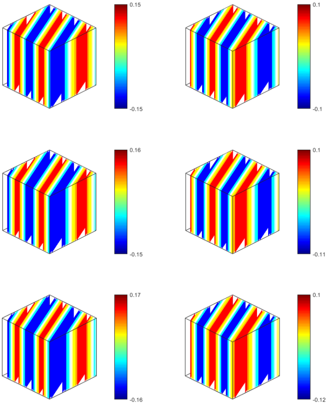


3D Turing: many patterns and bifurcations

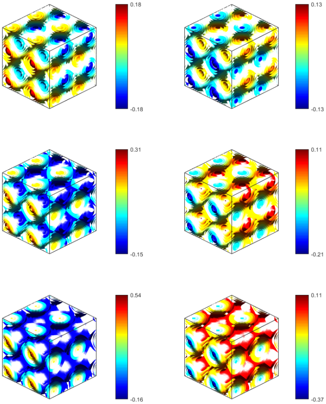


DLR

3D Turing: many patterns and bifurcations



3D Turing: many patterns and bifurcations



Summary and Outlook

Detailed insight in 3D dynamical systems by

- continuation as globalization strategy,
- Newton+Krylov+preconditioning, and
- linear stability analysis exploiting similar technology.

Future work

- HYMLS: $\mathcal{O}(N \log N)$ for 3D flow problems
- non-Hermitian Jacobi-Davidson with AMG preconditioning
- applications: full ocean model, stationary airplane flight, real biological problems