Lawrence Livermore National Laboratory

Improving Combustion Software to Solve Detailed Chemical Kinetics for HECC

Matthew McNenly (PI), Russell Whitesides, Daniel Flowers and Salvador Aceves

Project ID # ACE076

2012 DOE Vehicle Technologies Program
Annual Merit Review and Peer Evaluation Meeting
May 15, 2012 - Washington, DC

This presentation does not contain any proprietary or confidential information

LLNL-PRES-593472

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344

Overview

Timeline

- Ongoing project with yearly direction from DOE

Budget

- **FY11 funding: $300K**
- **FY12 funding: $340K**

Barriers

- Inadequate understanding of the fundamentals of HECC
- Predictive simulation of detailed fluid/chemistry coupling takes approx. a decade on a workstation
- Current combustion software doesn't take advantage of new CPU/GPUs

Partners

- Sandia, Oak Ridge, Los Alamos
- Ford, GM, Bosch, Volvo
- UC Berkeley, Univ. of Wisconsin, Univ. of Michigan, Lund Institute of Tech., Chalmers Univ. and UC Merced
- FACE working group, AEC MOU, SAE

Relevance to DOE objectives – ACE R&D subprogram

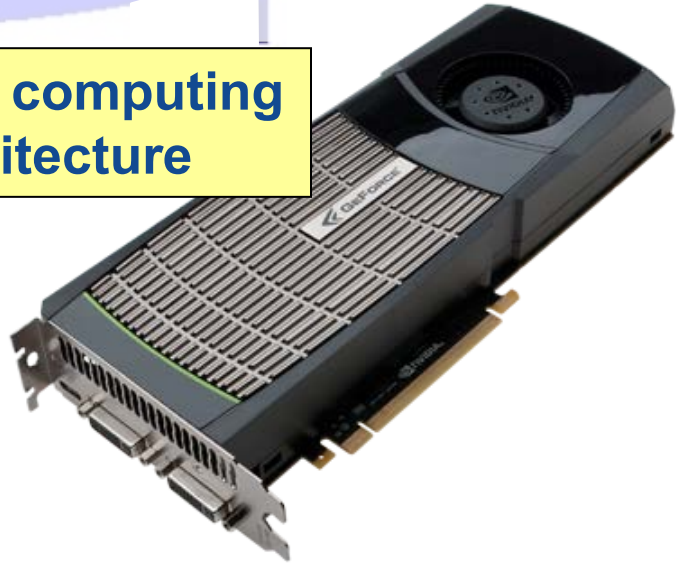
- Reducing the computational cost to solve detailed chemical kinetics provides more predictive power to all levels of HECC model fidelity (0-D to fully coupled CFD/detailed chemistry) and computing resources (workstation to supercomputer).
- Adapting combustion algorithms to massively parallel computing architecture ensures that the predictive power of our simulations can benefit from new technology growth.
- By providing fast combustion chemistry solvers to engine designers and researchers, we can accelerate progress on three major challenges identified in the VTP multi-year program plan:
  - A. *Lack of fundamental knowledge of advanced engine combustion regimes*
  - C. *Lack of modeling capability for combustion and emission control*
  - D. *Lack of effective engine controls*

Objective: Accelerate research in advanced combustion regimes by developing faster and more predictive engine models

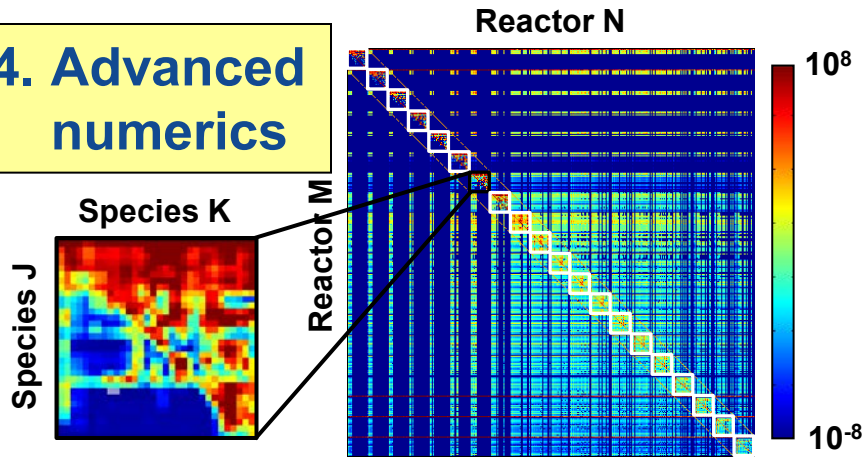
1. Low-cost models

2. Improving physical models

3. New computing architecture



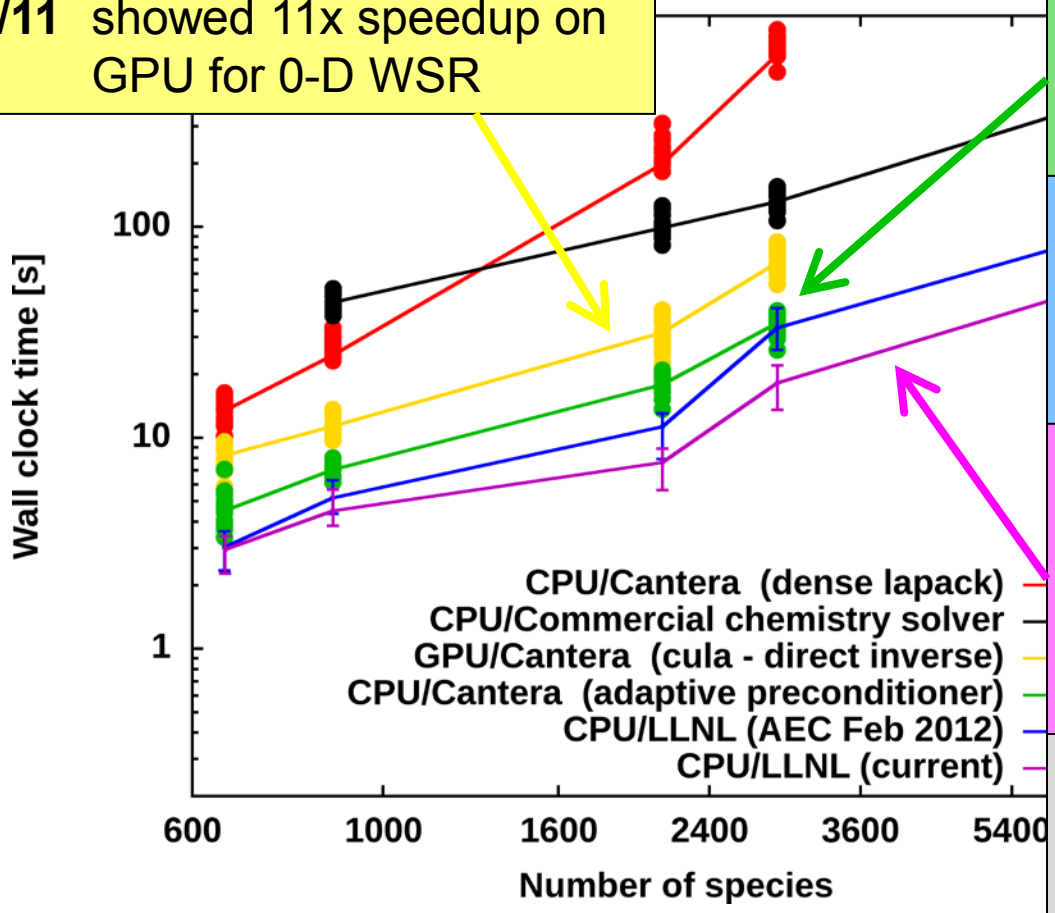
4. Advanced numerics



Milestones: new algorithms have demonstrated orders of magnitude speedup and been coupled to CFD

2011 Merit Review

2/11 showed 11x speedup on GPU for 0-D WSR



2012 Merit Review

8/11 showed 25x speedup on CPU for 0-D WSR using adaptive preconditioning

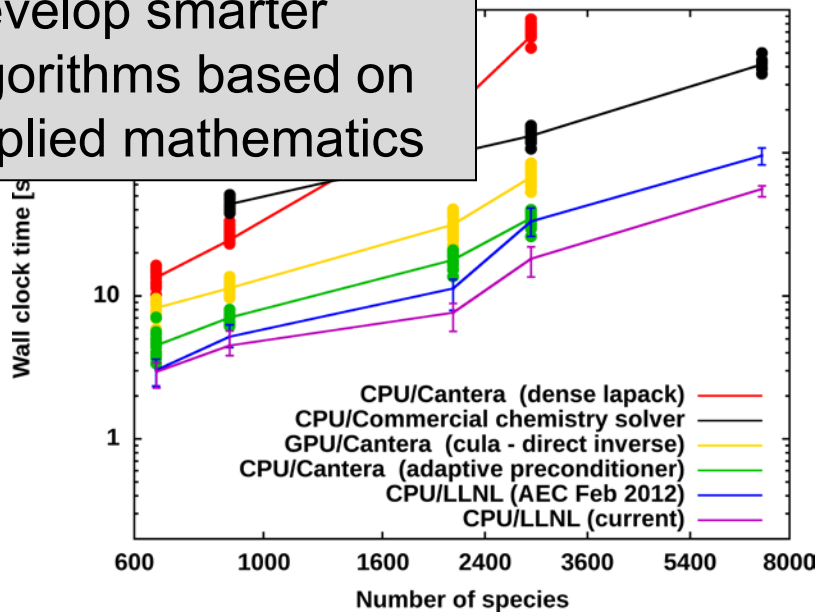
2/12 developed a *distributable* thermochemistry library that is faster than other open source codes

2/12 current algorithms orders of magnitude faster than and 8-10x faster than sophisticated commercial packages

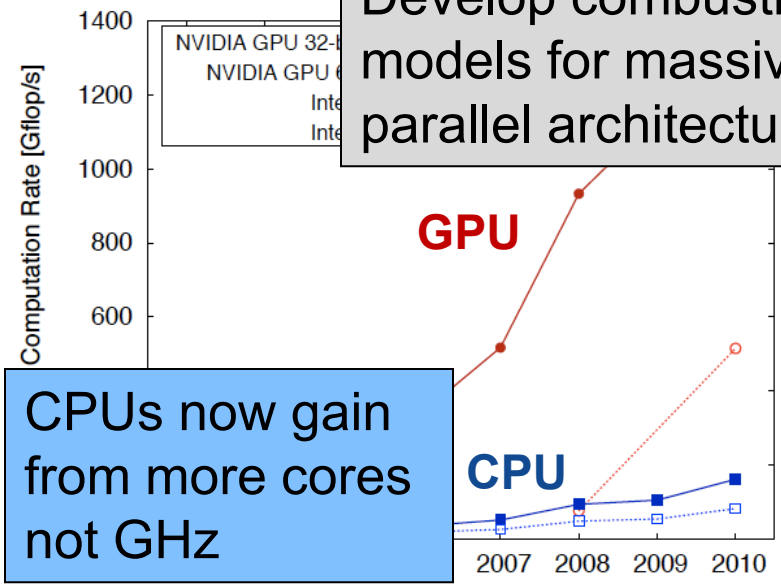
2/12 Adaptive preconditioning based chemistry solver implemented in fully-coupled CFD/LLNL multizone fluid/chemistry solver

Approach: bring high fidelity combustion simulations to the desktops of collaborators in industry, academia and national labs

Develop smarter algorithms based on applied mathematics



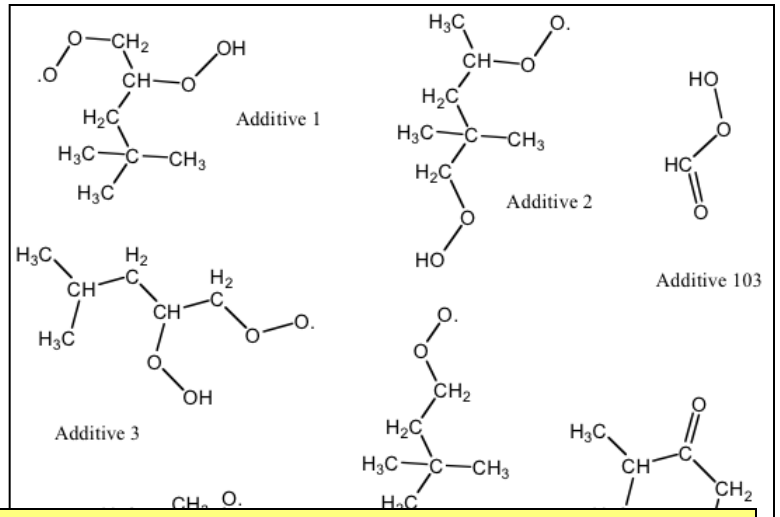
Develop combustion models for massively parallel architectures



CPUs now gain from more cores not GHz

- Gain fundamental and practical insight into High Efficiency Clean Combustion (HECC) regimes through numerical simulations and experiments
- Develop and apply numerical tools to simulate HECC by combining multidimensional fluid mechanics with chemical kinetics
- Reduce computational expense for HECC simulations
- Democratize simulation: bring computational tools to the desktop PC
  - *FY12 - release WSR and multizone (GT-Power linkable) models to MOU partners*

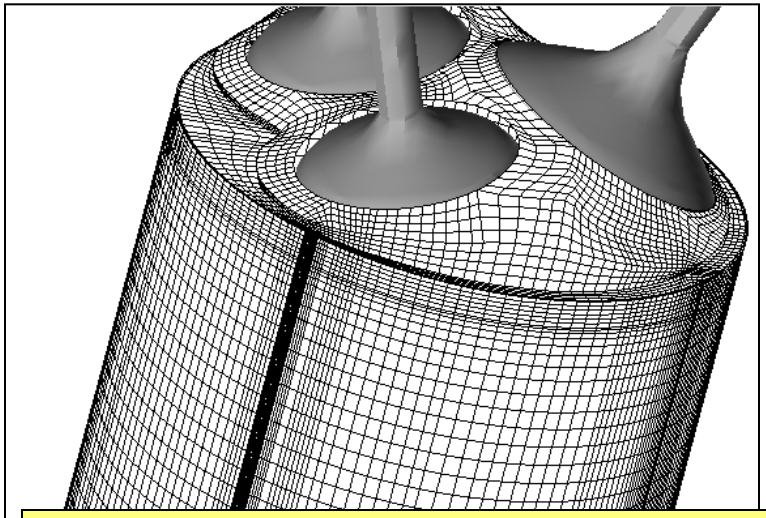
Challenge: Enhanced understanding of HECC requires computationally expensive models fully coupling detailed kinetics with CFD



Detailed Chemistry

Mechanisms for large fuel molecules contain +7000 species (*e.g.* LLNL's 2- and 3-methyl-alkanes Sarathy *et al.*)

+



3D Fluid dynamics

1M – 10 M fluid cells for well-resolved engineering mixing and turbulence model (Enaux *et al.*)

= 300,000 Pflop/s (chem-only), roughly a decade on current 12-core workstations = Not available for design

Objective: Bring the most physically accurate combustion models to engine designers and researchers

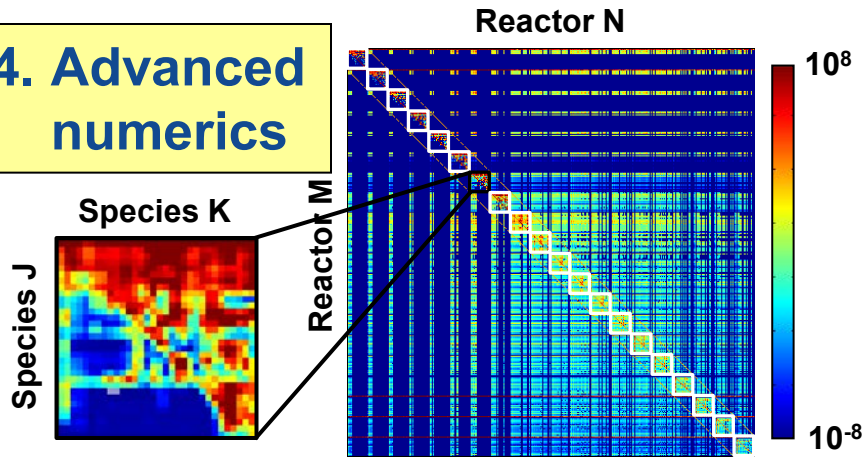
1. Low-cost models

2. Improving physical models

3. New computing architecture

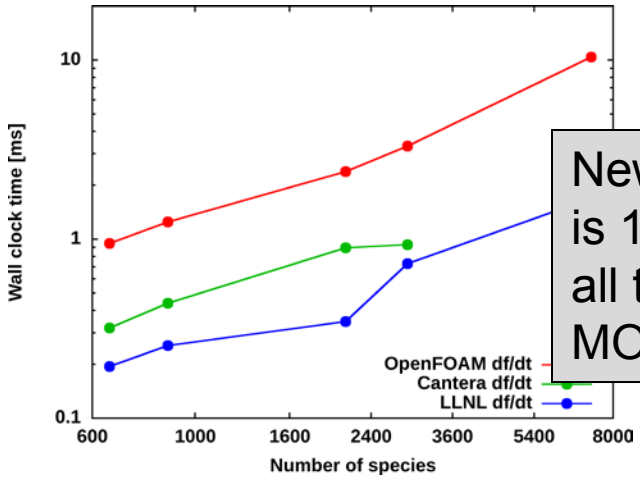
NVIDIA GTX480 image:
http://www.nvidia.com/object/product_geforce_gtx_480_us.html

4. Advanced numerics

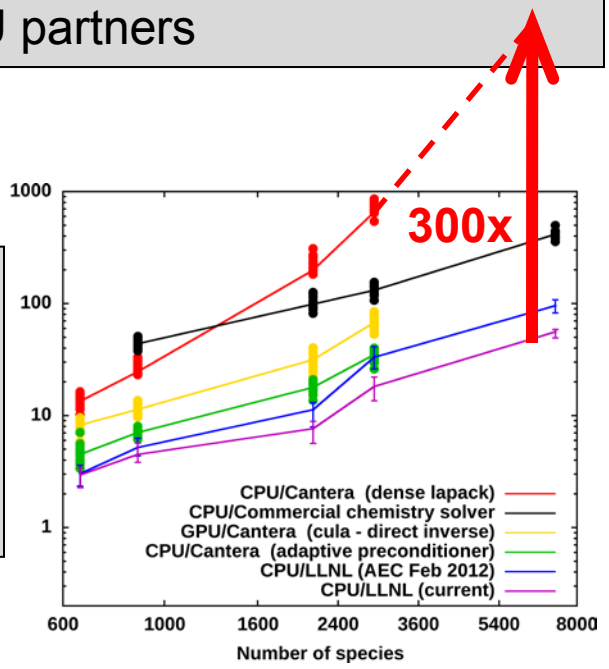


Technical Accomplishments: new algorithms have demonstrated orders of magnitude speedup and been coupled to CFD

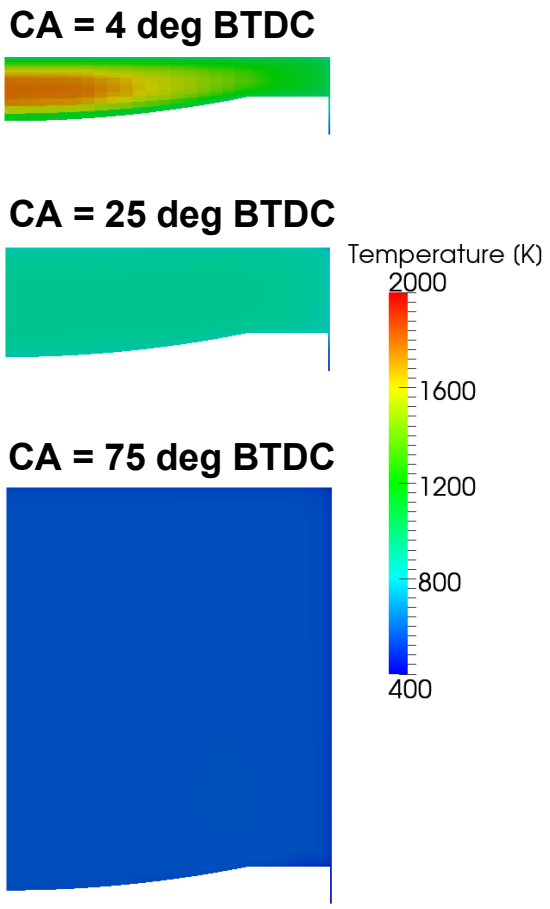
New solvers implemented in fully-coupled CFD/multizone model



New thermochemistry software is 1.5x – 5x faster and enables all tools to be distributed to MOU partners



New chemistry integrators developed from applied mathematics orders of magnitude faster than previous LLNL/CFD simulations



New thermochemistry software is needed to explore preconditioners and develop solvers for future computing architectures

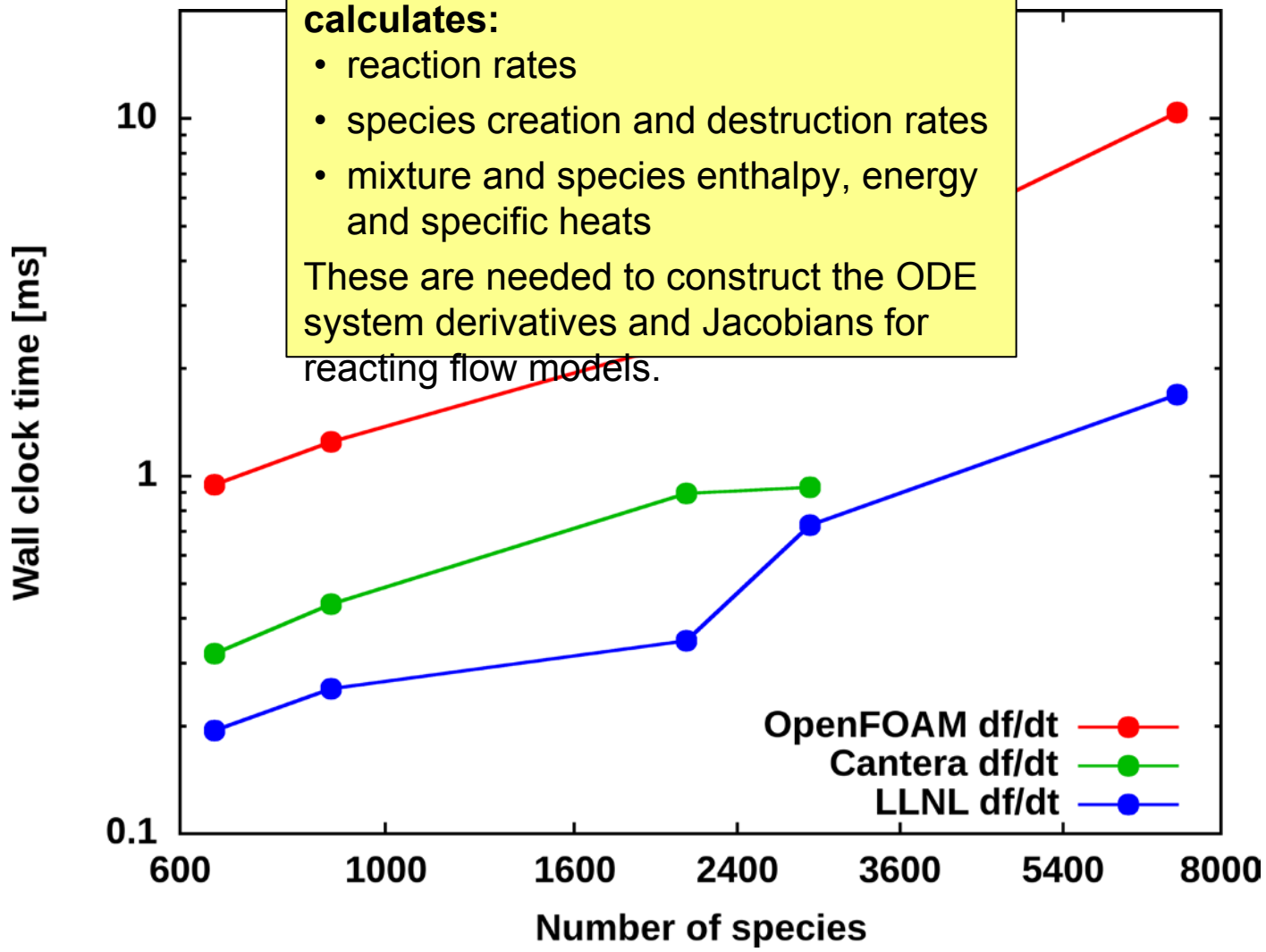
1. Easier access to the Jacobian information on-the-fly
   - Less time spent experimenting with new preconditioner and solver strategies
   - More robust kinetic mechanism diagnosis – important as the number of species and reactions continue to grow
2. Data structures optimized to current and future computing architectures
   - Multi-core CPU and GPU
   - Should still provide a high-level function interface for users that just want to model combustion
3. Need an open source thermochemistry software to combine with the improved solvers for the widest distribution

New LLNL thermochemistry software offers speedup over other open source codes for calculating ODE system derivatives

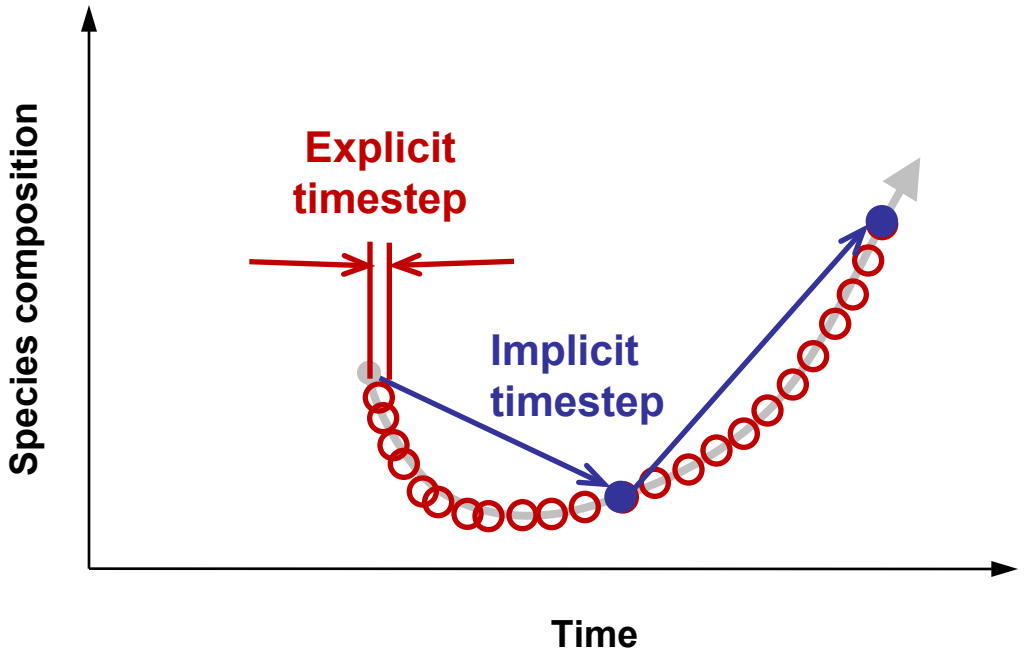
Thermochemistry software calculates:

- reaction rates
- species creation and destruction rates
- mixture and species enthalpy, energy and specific heats

These are needed to construct the ODE system derivatives and Jacobians for reacting flow models.



Implicit methods are necessary to integrate the chemical time scales over an engine cycle



Explicit Update
(lower cpu/step)

$$\frac{\partial x_1}{\partial t} = f_1(t, x_1, \ldots, x_N)$$
$$\frac{\partial x_2}{\partial t} = f_2(t, x_1, \ldots, x_N)$$
$$\vdots$$
$$\frac{\partial x_N}{\partial t} = f_N(t, x_1, \ldots, x_N).$$

Implicit Update
(more trajectory data)

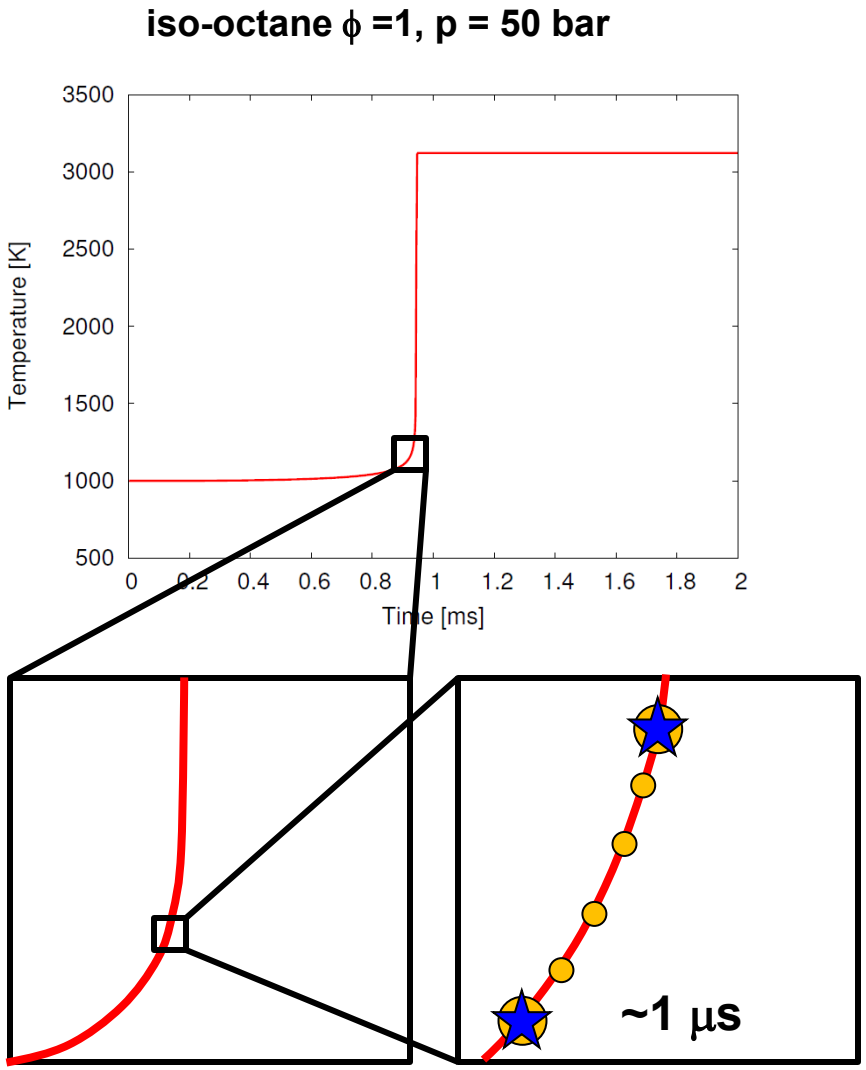
$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & & \frac{\partial f_2}{\partial x_N} \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_N}{\partial x_1} & \frac{\partial f_N}{\partial x_2} & \cdots & \frac{\partial f_N}{\partial x_N} \end{pmatrix}$$

During ignition:

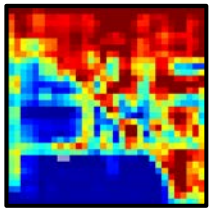
Δt (explicit) = 10^{-12} to 10^{-15} s

Δt (implicit) = 10^{-6} to 10^{-8} s

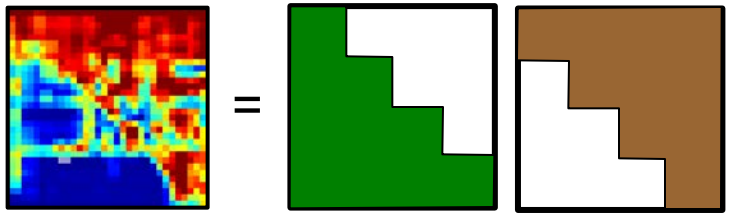
Jacobian matrix construction/solution is more than 95% of the simulation cost for large mechanisms – why?



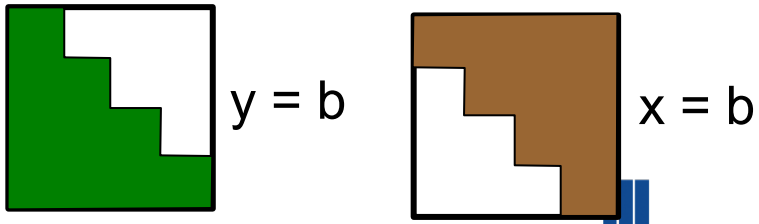
1. Construct J matrix from the thermodynamic state:

p, T, y_1, ..., y_N ⇒ J = 

2. Factor the J matrix into lower and upper triangle matrices



3. Iterate to the next step with backward solution to Jx = b



What is the physical meaning of the Jacobian?

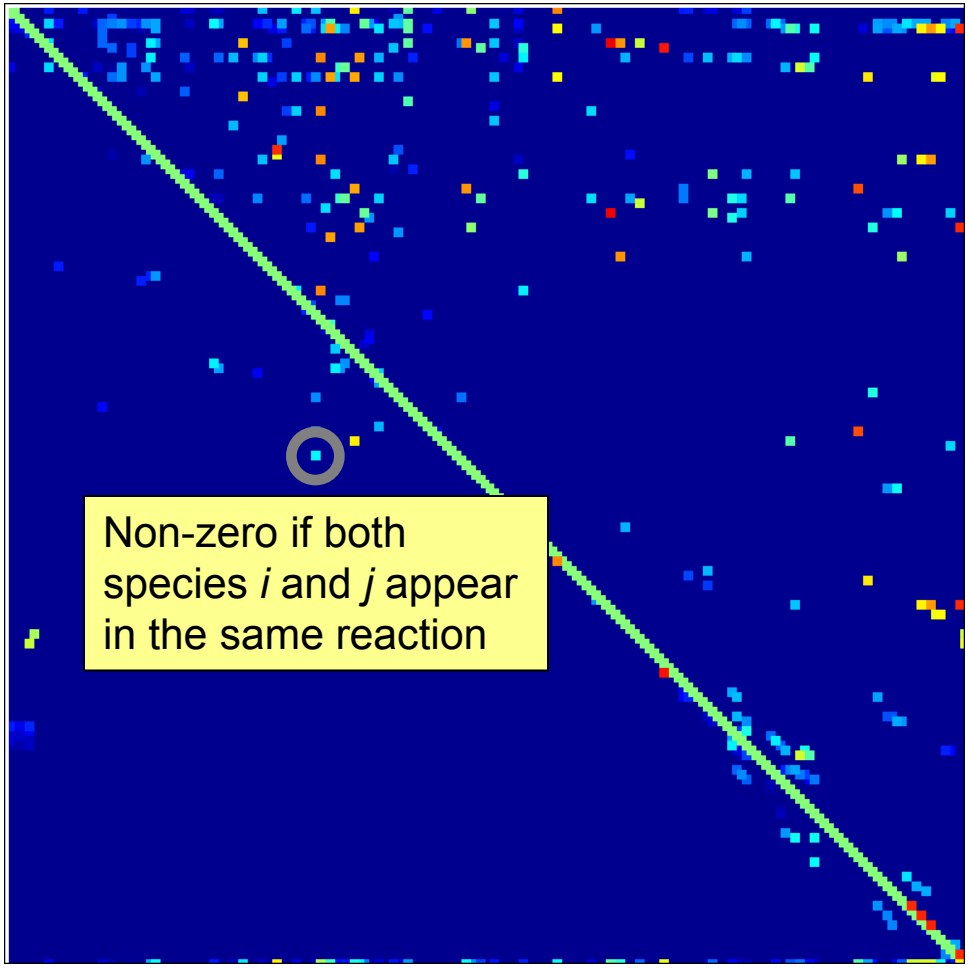
$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & & \frac{\partial f_2}{\partial x_N} \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_N}{\partial x_1} & \frac{\partial f_N}{\partial x_2} & \cdots & \frac{\partial f_N}{\partial x_N} \end{pmatrix}$$

Element: $J_{i,j} = \frac{dw_i}{dC_j}, \quad w_i = \frac{dC_i}{dt}$

Magnitude represents the characteristic frequency at which the two species are coupled

10^{-4} 10^{0} 10^{4}

Matrix element magnitude (relative to diagonal)



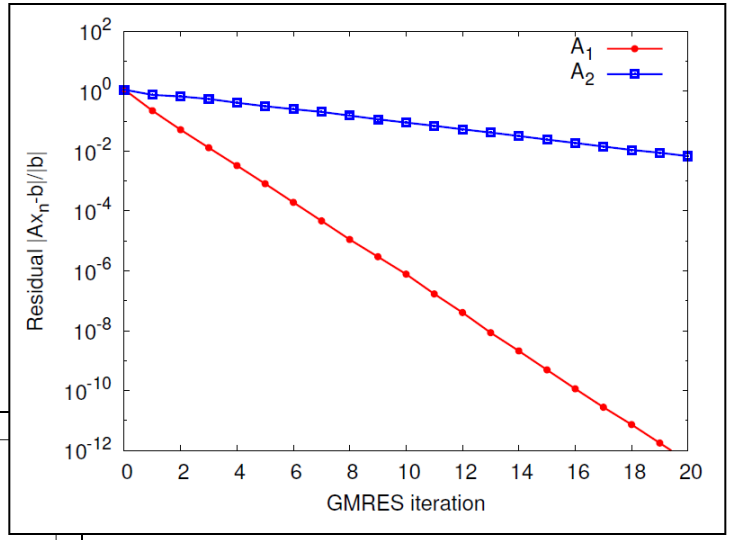
Approximate Jacobians can be used to precondition iterative linear system solvers like GMRES

Generalized **M**inimal **RES**iduals

$$E^{(n)} = \frac{\|Ax^{(n)} - b\|_2}{\|b\|_2} \leq \Lambda^n \mathrm{cond}(V)$$

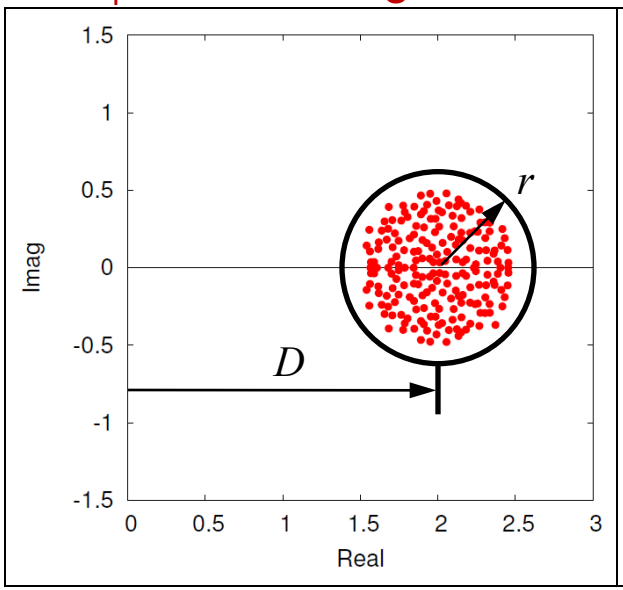
$$\Lambda \approx \frac{r}{D}$$

GMRES Error

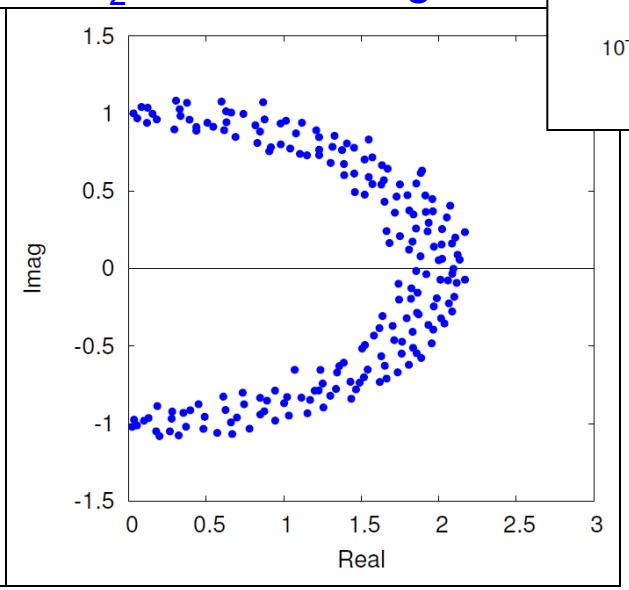


Eigenvalue Spectra (200 x 200)

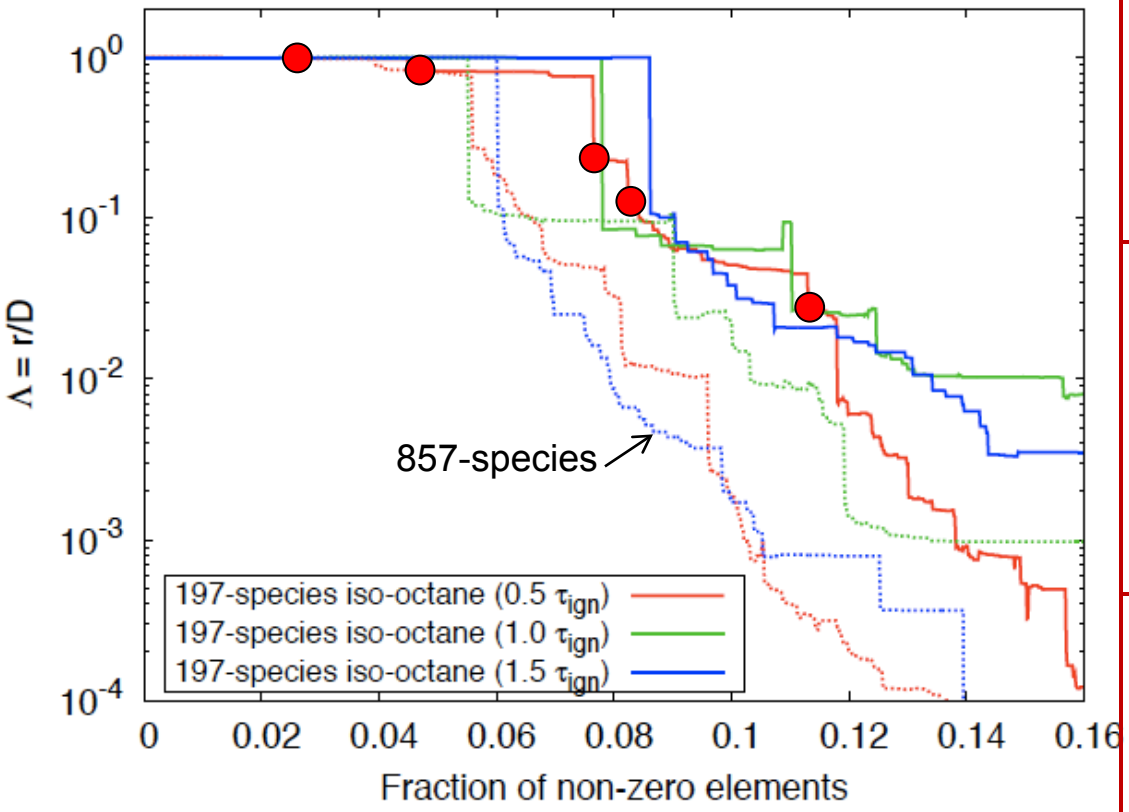
A_1: fast convergence

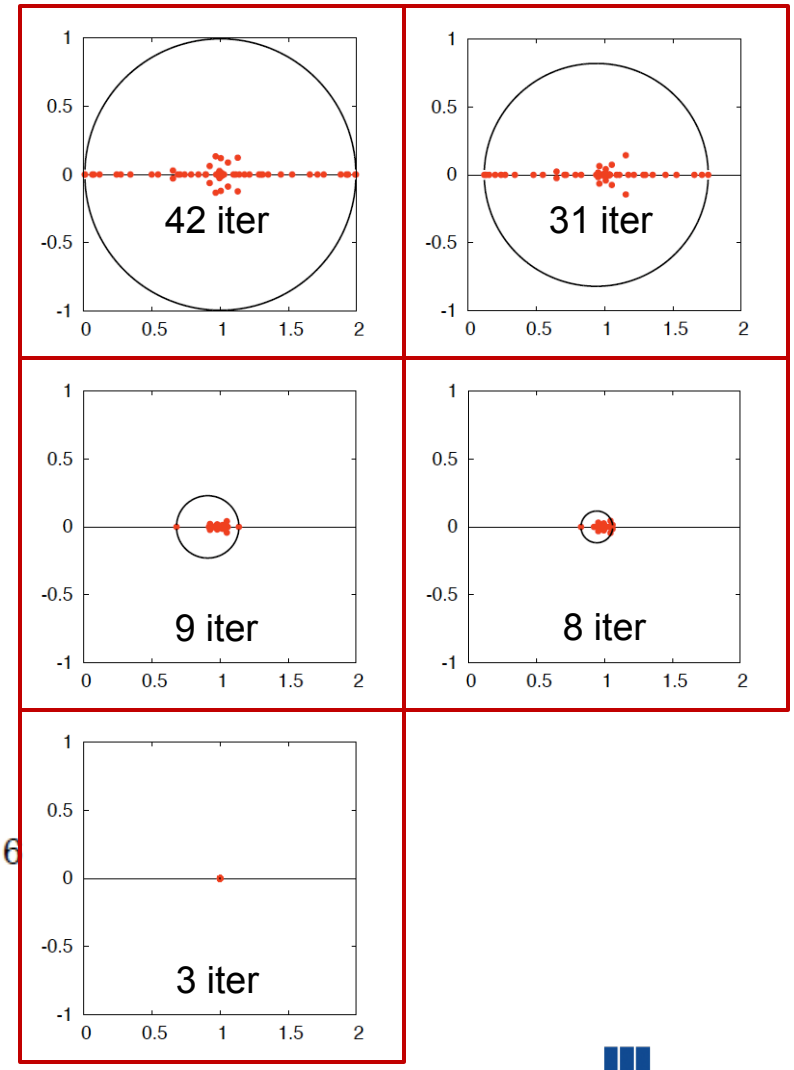


A_2: slow convergence



Direct reaction sorting shows promise to be a general low-cost preconditioner for the Jacobian

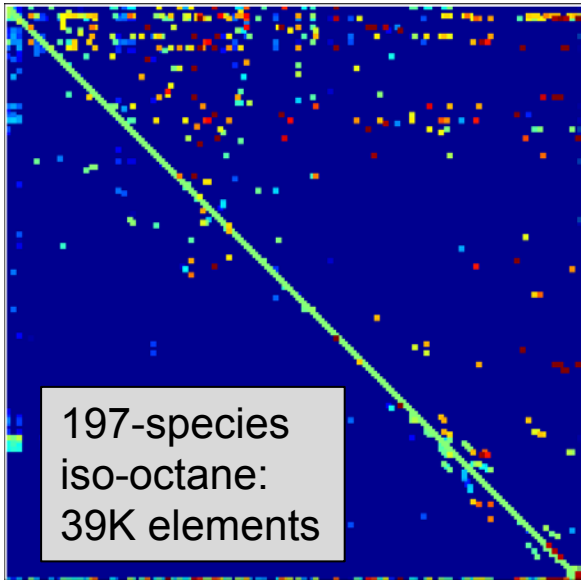




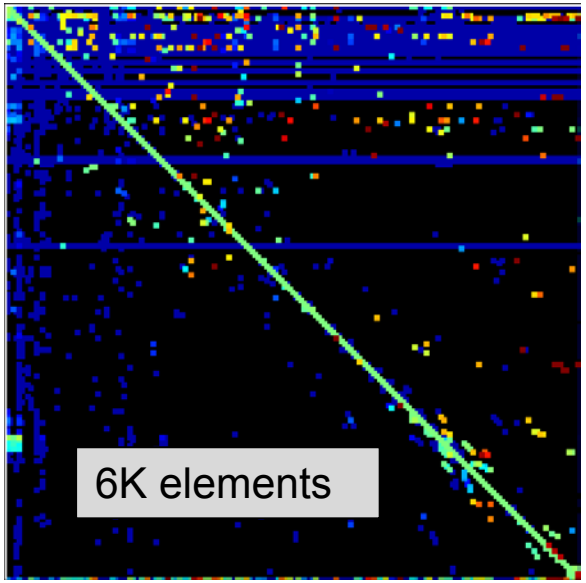
The dominant eigenstructure can be effectively captured with a preconditioner with less than 10% of the non-zero terms

Previous multizone/CFD chemistry integrators	Current Sophisticated Commercial Software	New LLNL approach

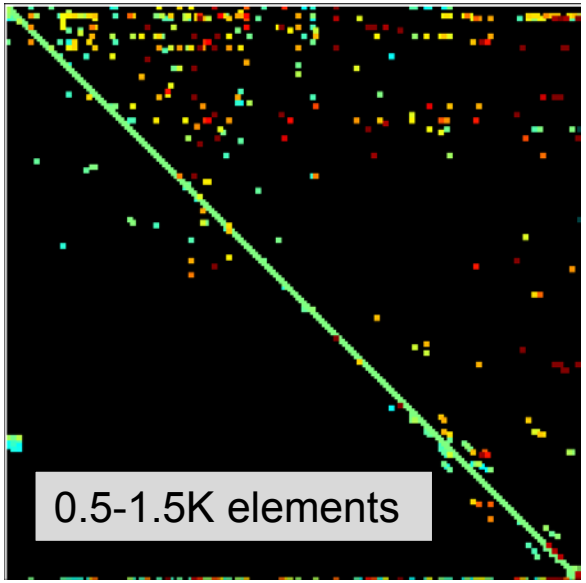
Dense Matrix



Sparse Matrix



Adaptive Preconditioner



10^{-4} 10^{0} 10^{4}

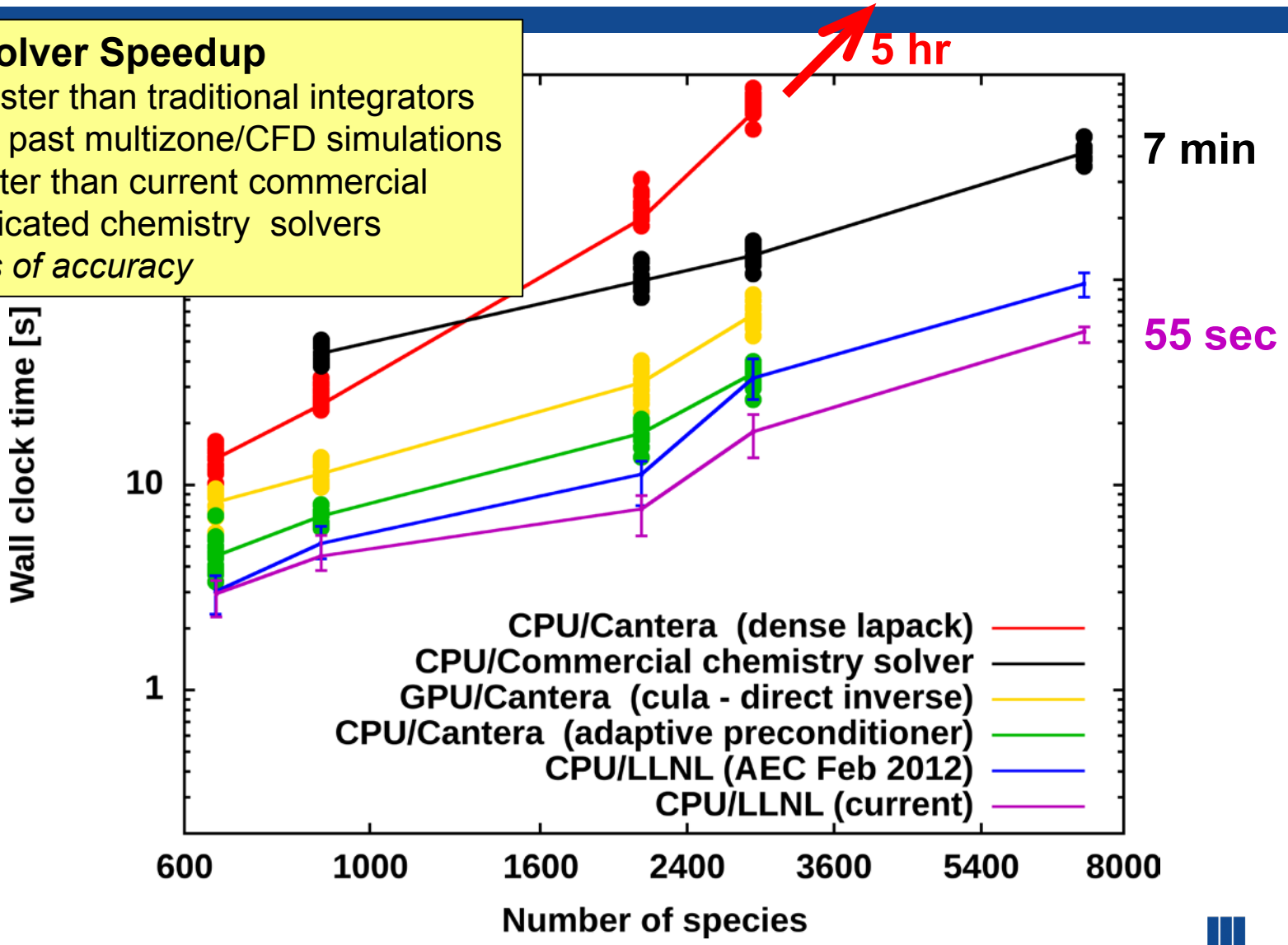
Matrix element magnitude
(relative to diagonal)

Black elements are zero and not stored by the sparse matrix solver

New thermochemistry software will enable future solver gains and already delivers and order of magnitude speedup

LLNL Solver Speedup

- 300x faster than traditional integrators used in past multizone/CFD simulations
- +8x faster than current commercial sophisticated chemistry solvers
- *No loss of accuracy*

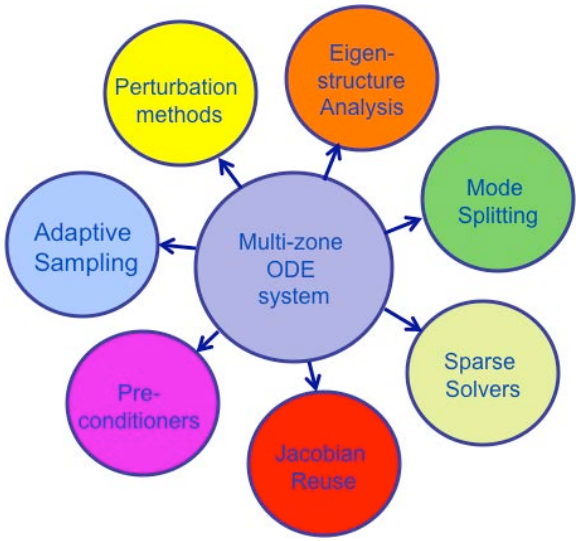


Collaboration: We have ongoing interactions Industry, National Labs, and Universities

- **Convergent Science Inc**; Multi-zone model development
- **Advanced Engine Combustion (AEC)** working group (Industry, National labs, Univ. of Wisc.); biannual presentations
- **Fuels for Advanced Combustion Engines (FACE)** working group
- **Sandia National Laboratory**; researchers on HCCI and PCCI, gaseous injection simulations
- **Oak Ridge National Laboratory**; SI-HCCI transition and ^{14}C exhaust analysis for HCCI and Diesel engines
- **Los Alamos National Laboratory**; Kiva4 development
- **Other Universities**: UC Berkeley, University of Wisconsin, University of Michigan, Lund Institute, Chalmers University, Tianjin University, and UC Merced
- **Ford**; gaseous direct injection
- **Delphi**; direct injection
- **Bosch**; High Performance Computing of HCCI/SI transition
- **GM**; Diesel engine
- **Volvo**; multi-zone cycle simulation

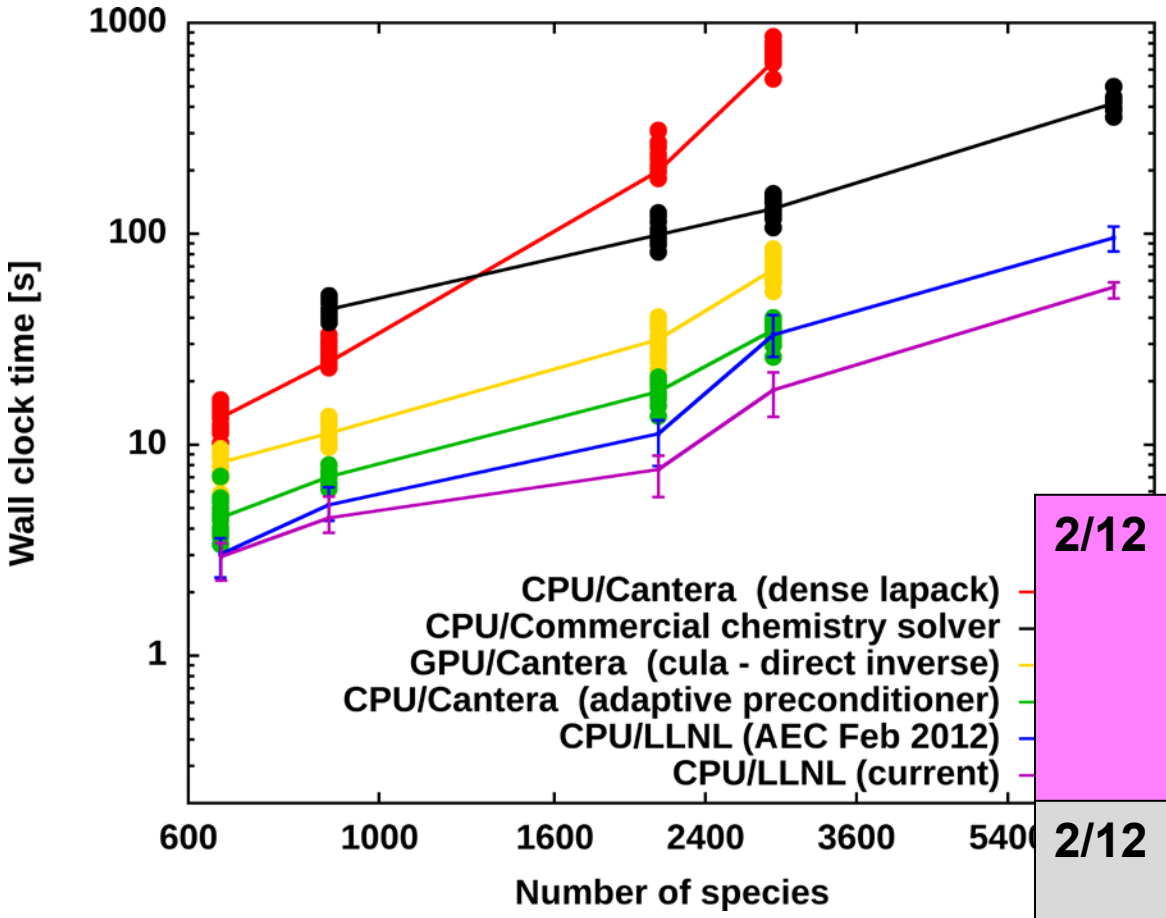
Future Work: We will explore strategies for improving efficiency of CFD and chemistry simulations

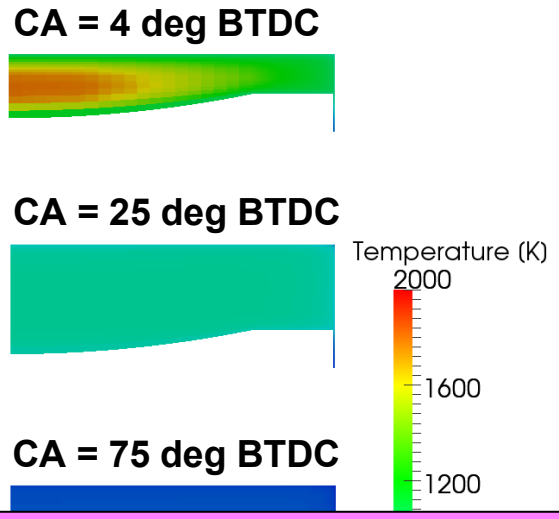
- ***Improved chemical reactor integration***
  - Sparse preconditioners (CPU & GPU)
  - Efficient data structures
  - Hybrid time-stepping solvers
  - New integration error control logic
  - New non-linear solvers



- ***Improved parallel engine CFD***
  - Multi-criteria multi-zone
  - More accurate multi-zone remap
  - Perturbed reactor integrator restart
  - Particle motion/collision algorithms (CPU & GPU)
- ***Making high fidelity combustion simulations available on a PC desktops***
  - FY12 - release WSR and multizone (GT-Power linkable) models to MOU partners
  - FY13 - release fully-coupled CFD/multizone solver to MOU partners and interested software companies (move to open source)

Summary: We will continue our research for better algorithms to accelerate the development and design of efficient engines





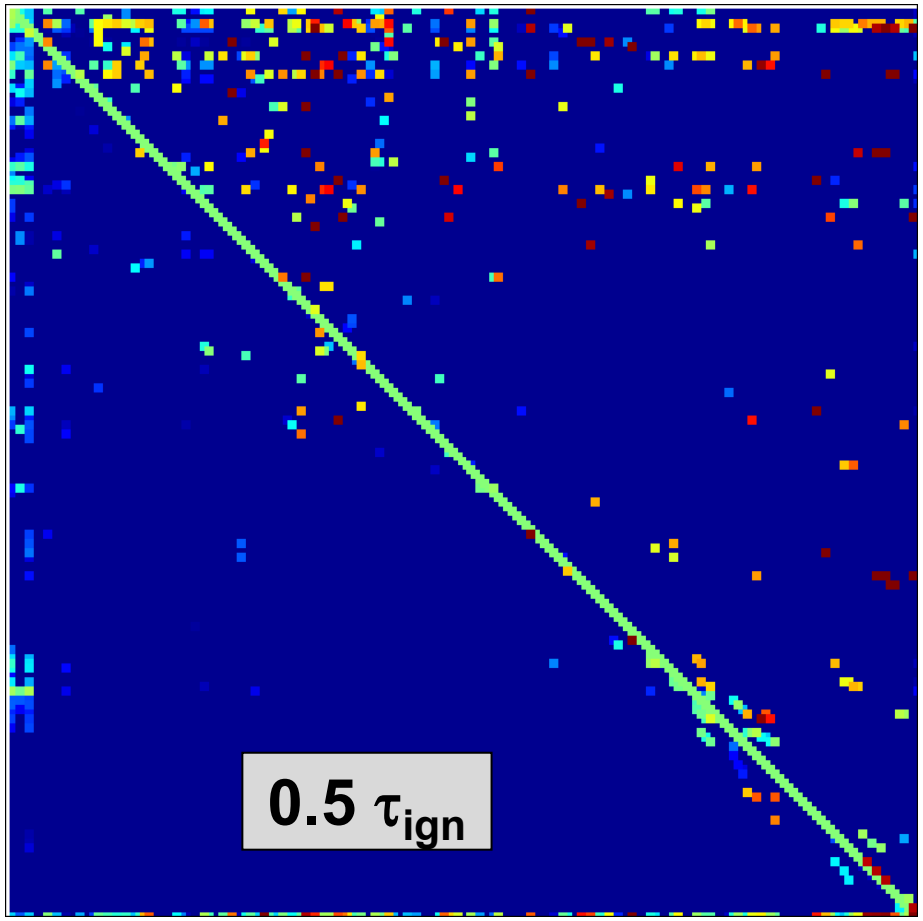
2/12 current algorithms orders of magnitude faster than simple implementations and 10x faster than sophisticated commercial packages

2/12 Adaptive preconditioning based chemistry solver implemented in fully-coupled CFD/LLNL multizone fluid/chemistry solver

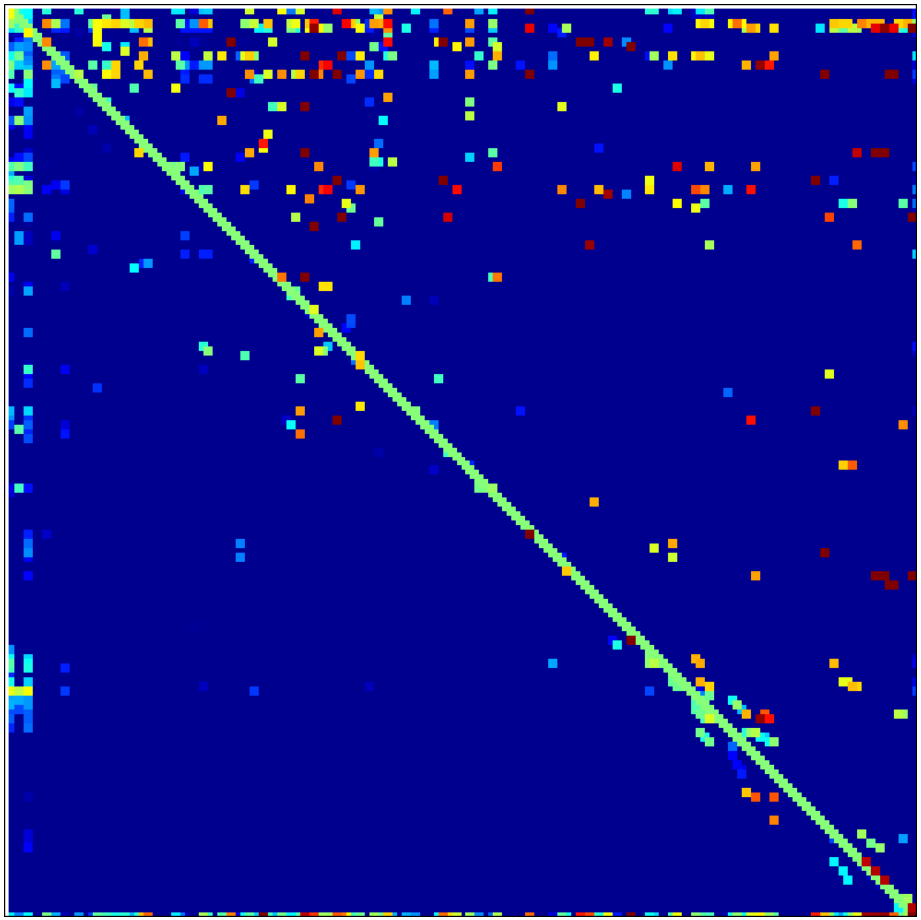
Technical Back-Up Slides

The evolution of the Jacobian is similar for different equivalence ratios except near equilibrium

ϕ = 0.25, p = 2 bar

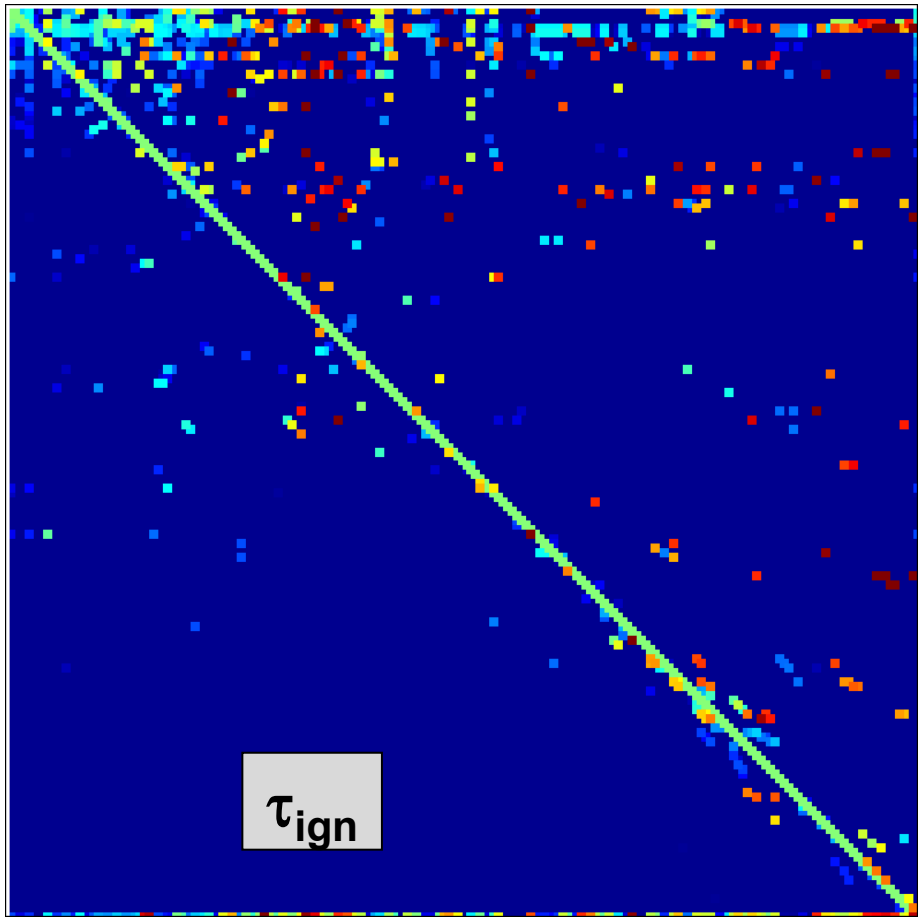


ϕ = 1, p = 2 bar

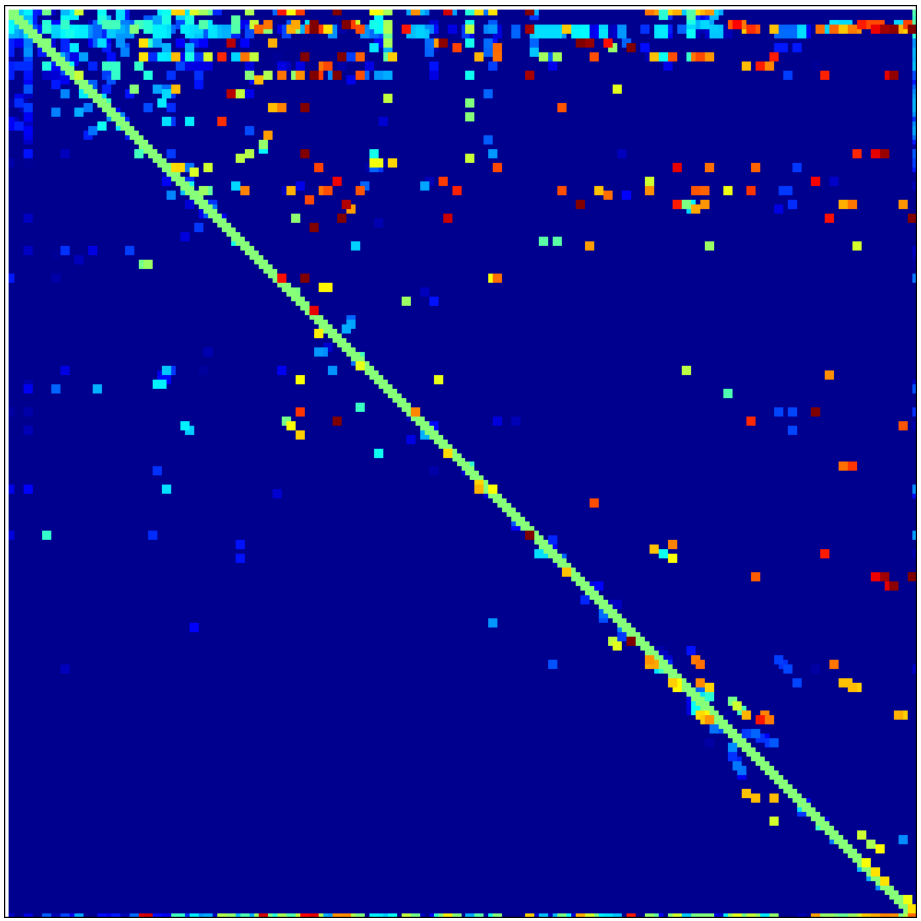


The evolution of the Jacobian is similar for different equivalence ratios except near equilibrium

ϕ = 0.25, p = 2 bar

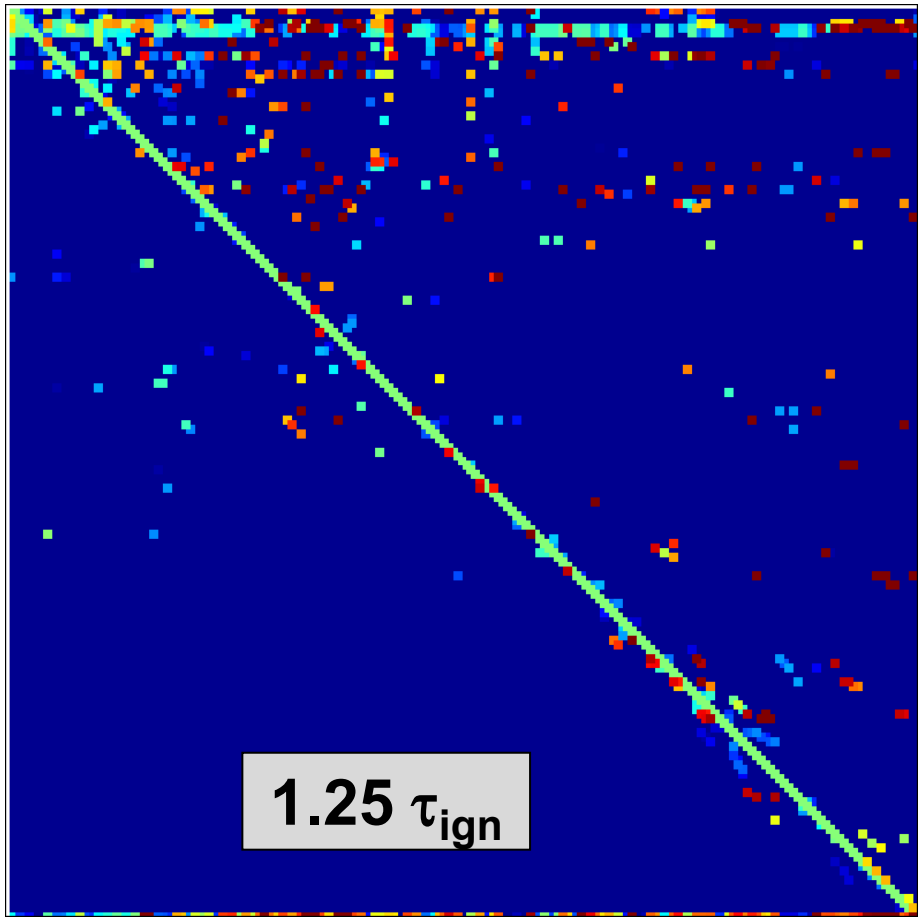


ϕ = 1, p = 2 bar

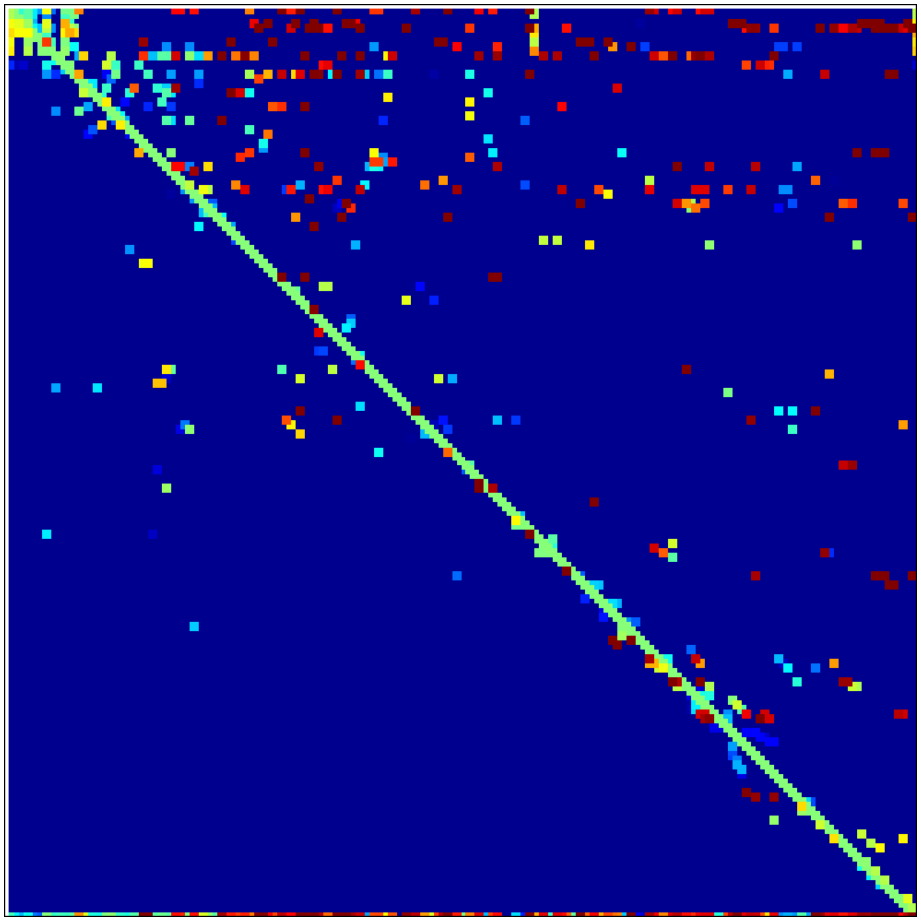


The evolution of the Jacobian is similar for different equivalence ratios except near equilibrium

ϕ = 0.25, p = 2 bar

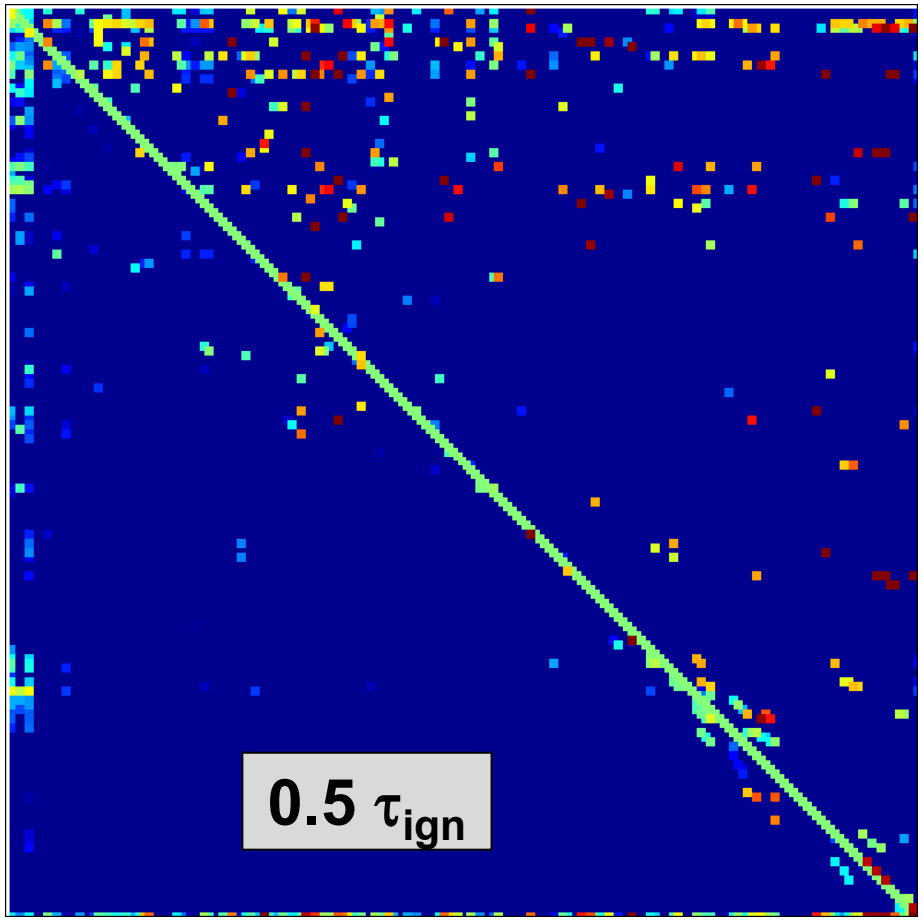


ϕ = 1, p = 2 bar

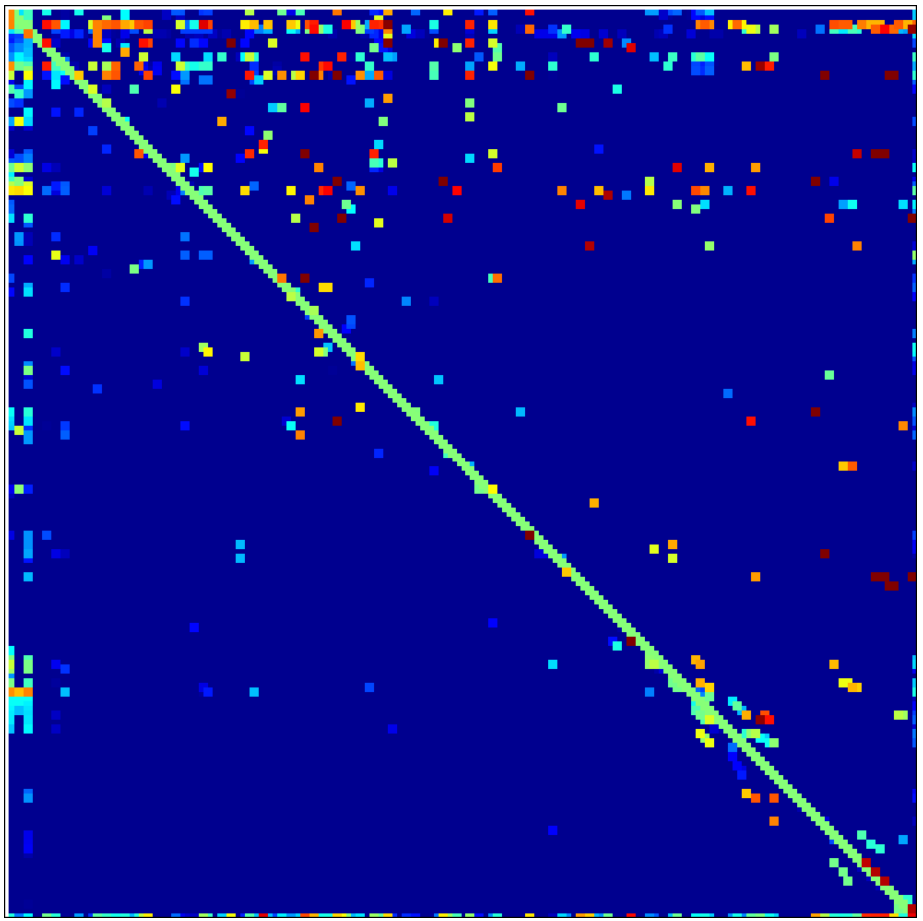


The evolution of the Jacobian is similar for different pressures except near equilibrium

$\phi = 1$, p = 2 bar

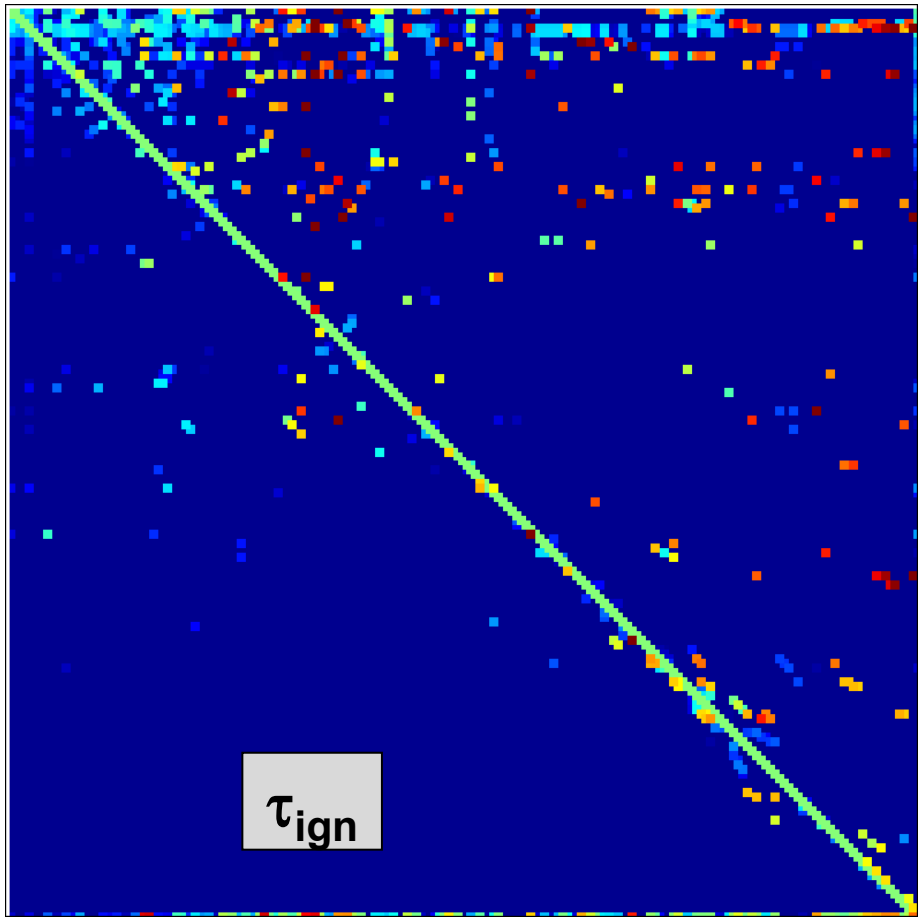


$\phi = 1$, p = 20 bar

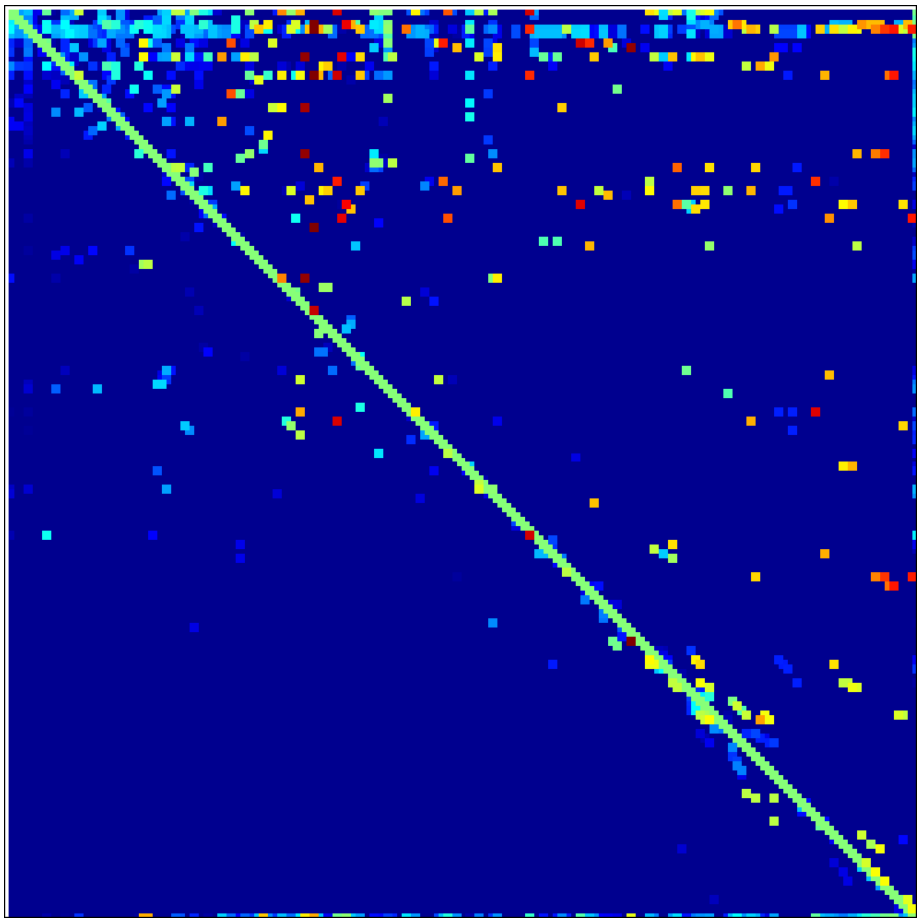


The evolution of the Jacobian is similar for different pressures except near equilibrium

$\phi = 1$, p = 2 bar

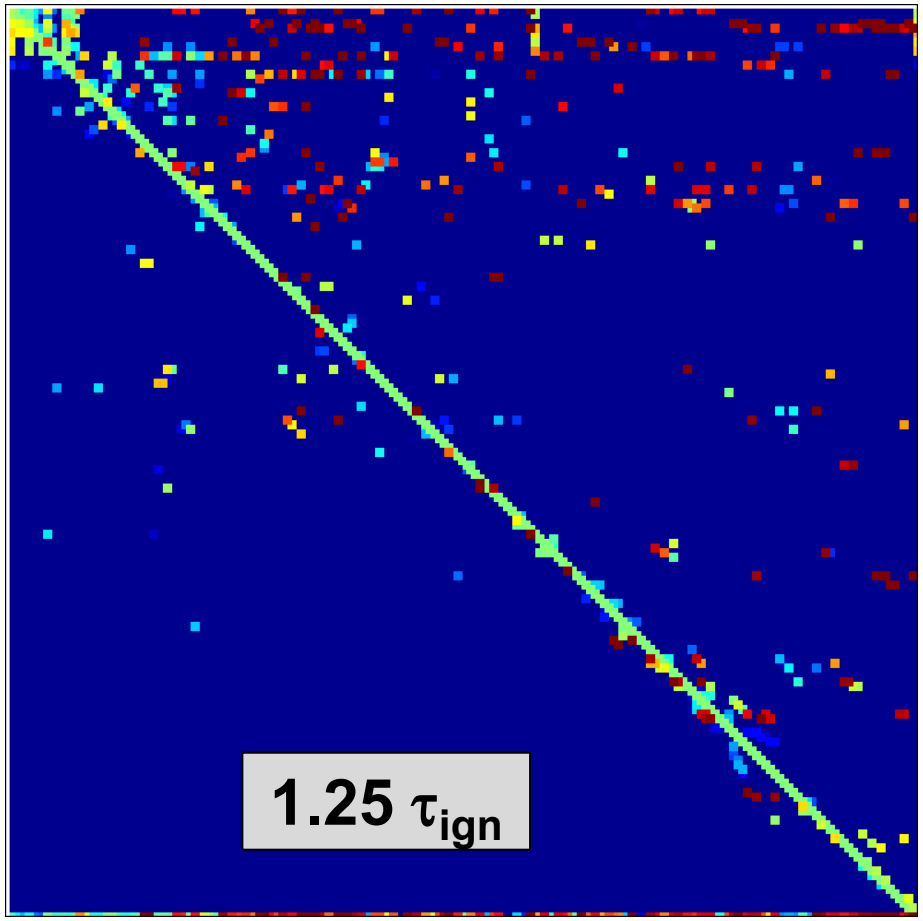


$\phi = 1$, p = 20 bar



The evolution of the Jacobian is similar for different pressures except near equilibrium

$\phi = 1$, p = 2 bar



$\phi = 1$, p = 20 bar