Linear Solver Performance Analysis of MFiX integrated with a Next Generation Computational Framework

V. Kotteda, PostDoc,
V. **Kumar**, Associate Professor, Mechanical
A Rodriguez, UG RA,
A Schiaffino, Grad. Student, Mechanical,
A Chattopadhyay, Grad. Student, Computational Science
The University of Texas at El Paso

W. **Spotz**, Senior Research Staff, Sandia National Labs – NM

NETL Workshop on Multiphase Flow Science
August 8-10, 2017
Morgantown, WV

Outline

- Goals and Objectives
- Technical approach
  - What is Trilinos?
  - Scalability and Portability
  - Sandia’s Next-Gen Computational Framework - MPI-X, UQ, etc
  - Trilinos-MFiX Framework
- Results & Discussion
- Concluding remarks
- Future work

Goal and Objectives

The technical goal of this project is to develop, validate and implement **advanced linear solvers** to replace **MFiX's existing linear solvers**. This goal will be achieved by integrating **Trilinos**, a publicly available open-source linear equation solver library developed by **Sandia** National Laboratory. The project will **demonstrate scalability** of the Trilinos- MFIX interface on various high-performance computing (HPC) facilities including the ones funded by the Department of Energy (DOE).

The expected results of the project will be **reduction of computational time** when solving complex gas-solid flow and reaction problems in MFIX, and reduction in time and cost of adding new algorithms and physics based models into MFIX

Objectives

- Create a framework to integrate MFIX with Trilinos linear solver packages
- Validate MFIX suites of problems on HPC systems with and without GPU acceleration
- Evaluate the performance

MFiX: Challenges and Opportunities

MFiX (developed by NETL)
- Model multiphase physics
- Widely used by the fossil fuel reactor communities and beyond
- can significantly reduce time & cost to design a reactor

Ax = b

$$\varepsilon_g + \sum_{m=1}^{M} \varepsilon_{sm} = 1$$

$$\frac{\partial}{\partial t}(\varepsilon_g \rho_g) + \nabla.(\varepsilon_g \rho_g \vec{U}_g) = R_g$$

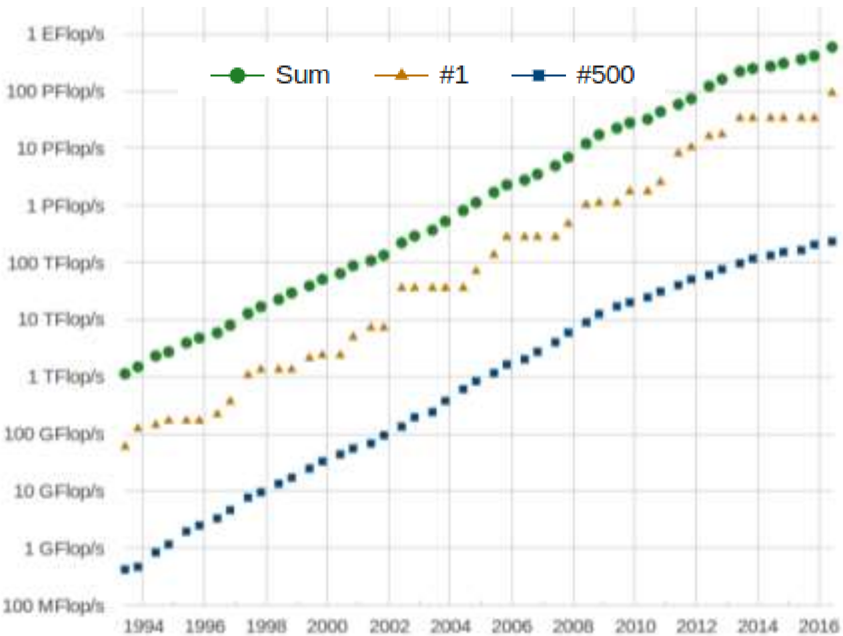
$$\frac{\partial}{\partial t}(\varepsilon_g \rho_g \vec{U}_g) + \nabla.(\varepsilon_g \rho_g \vec{U}_g \vec{U}_g) = -\varepsilon_g \nabla P_g + \nabla.\tau_g - \sum_{m=1}^{M} I_{gs_m} + \varepsilon_g \rho_g \vec{g}$$

$$\varepsilon_g \rho_g C_{pg} \left(\frac{\partial T_g}{\partial t} + \vec{U}_g.\nabla T_g \right) = \nabla \vec{q}_g - \sum_{m=1}^{M} H_{gs_m} - \Delta H_{rg} + H_{wall}(T_{wall} - T_g)$$

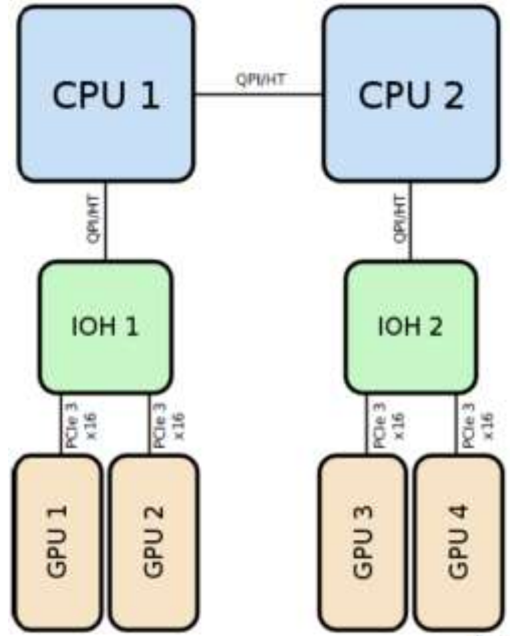
However
- **Computational expense** for most **practical applications** can make it **impractical**
- Limited software capabilities
  - Linear solver, MPI-X, UQ, etc.
- Can result in poor convergence especially in complex non-linear problems

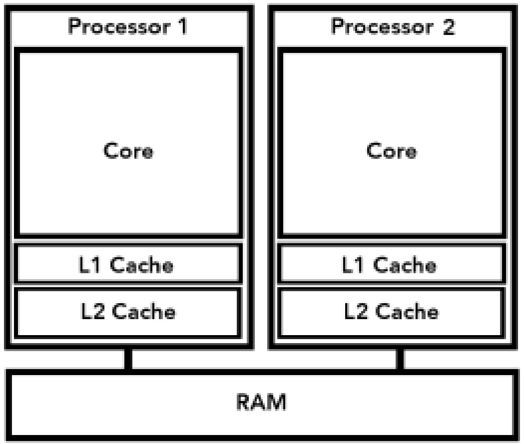
But, could be made more practical if we could significantly reduce time-to-solution by
- Effectively exploiting HPC systems (massively parallel computers, GPUs, multithreading..)
- **Leveraging** state-of-the-art **preconditions and linear solver libraries**
- Providing a long-term portable and scalable software development framework

Supercomputers



https://www.top500.org/





Cray X1
3 TF
2004

Cray XT3
Single-core
26 TF
2005

Cray XT3
Dual-core
54 TF
2006

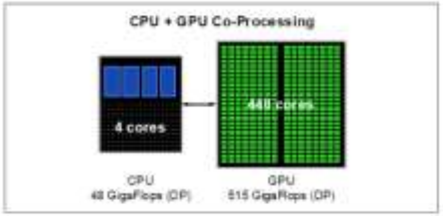
Cray XT4
119 TF
2007

Cray XT4
Quad-core
263 TF
2008

Cray "Baker"
6-core, dual-socket SMP
~1000 TF
100TB, 2.5PB
2009

CRAY
AMD Opteron
Nvidia Tesla
20 PF
2012

NUDT
Intel Phi
54 TF
2015



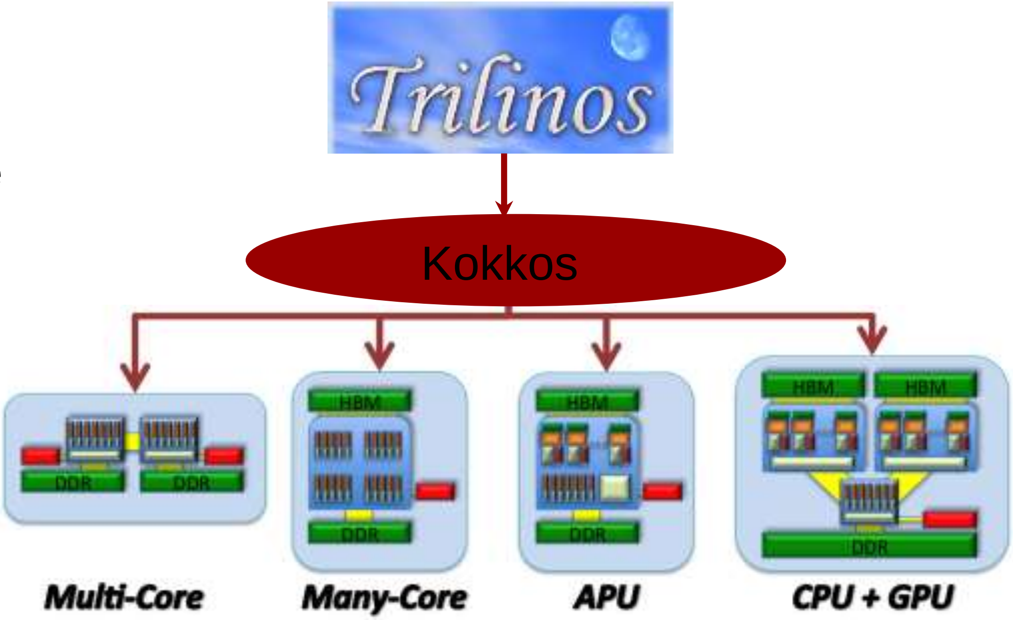
Sandia National Laboratories

What is Trilinos?

- Object-oriented software framework for…
- Solving big complex science & engineering problems
- More like LEGO™ bricks than Matlab™

Trilinos provides the state-of-the-art preconditions and **linear solver** libraries

- demonstrate **scalability** on **current HPC systems**
- illustrate plans for **continued maintenance**
- include **support for new hardware technologies**



Next-generation Computational Framework

- High fidelity modeling and simulation is a **critical enabling technology** to understand physical systems
- Exascale computing
  - provides potential for **high fidelity solutions to complex multiscale multiphysics problems**
  - requires exploiting **homogeneous or heterogeneous** mixtures CPUs and specialized processing units (modern day supercomputers!)
  - requires **next generation of computational framework and models** that are portable and scalable to 100,000s of processors

physics

L(u)=f
Math. model

$L_h(u_h)=f_h$
Numerical model

$u_h=L_h^{-1} \bullet f_h$
Algorithms

computation

Numerical math
Convert to models that can be solved on digital computers

Algorithms
Find faster and more efficient ways to solve numerical models

Trilinos

discretizations
Time domain
Space domain

methods
Automatic diff.
Domain dec.
Mortar methods

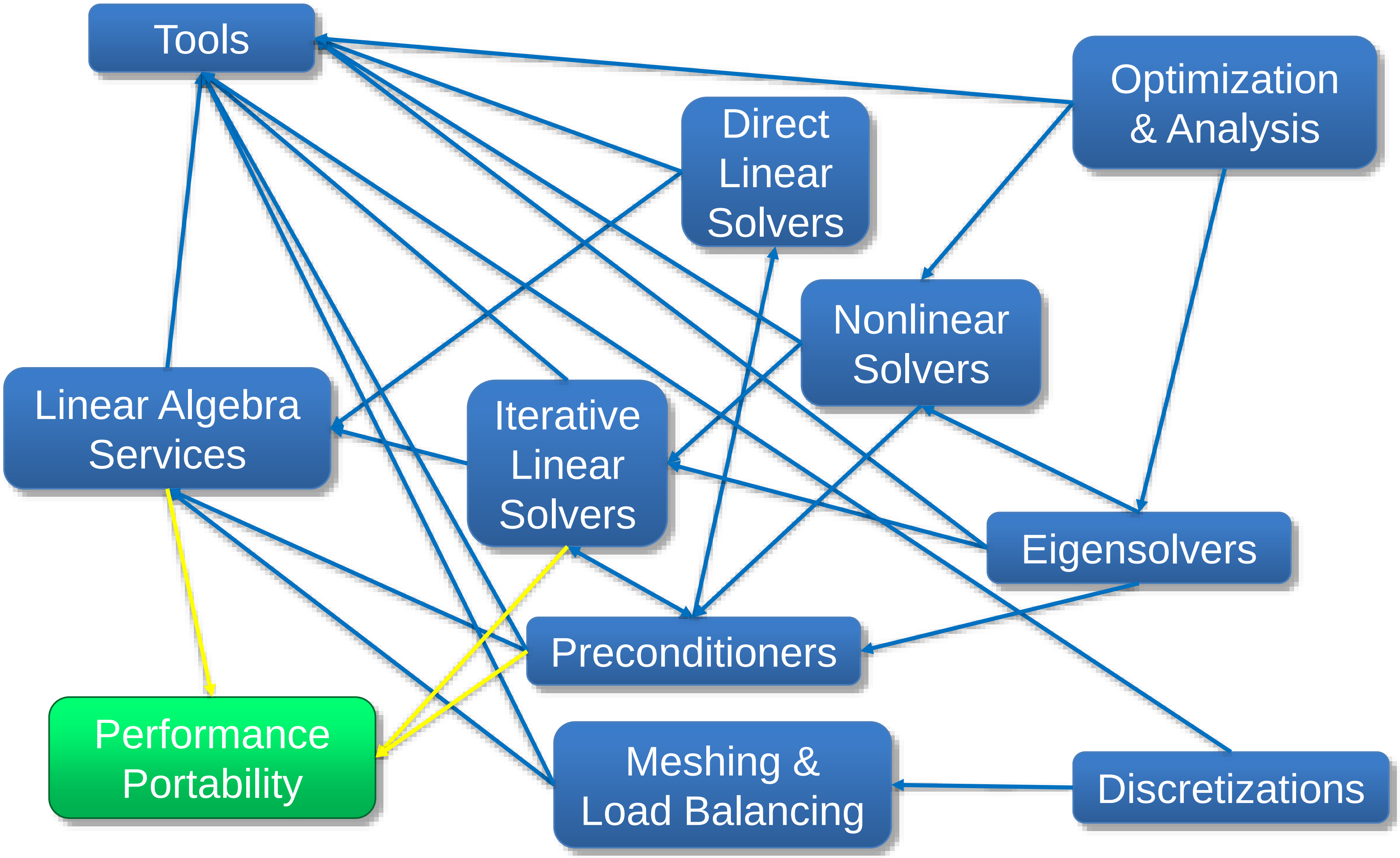
solvers
Linear
Nonlinear
Eigenvalues
Optimization

core
Petra
Utilities
Interfaces
Load Balancing

- Beyond a “solvers” framework
- Natural expansion of capabilities to satisfy application and research needs
- Discretization methods, Automatic Differentiation, Mortar methods, …
-

UTEP
Sandia National Laboratories

Trilinos Package Categories



Sandia National Laboratories

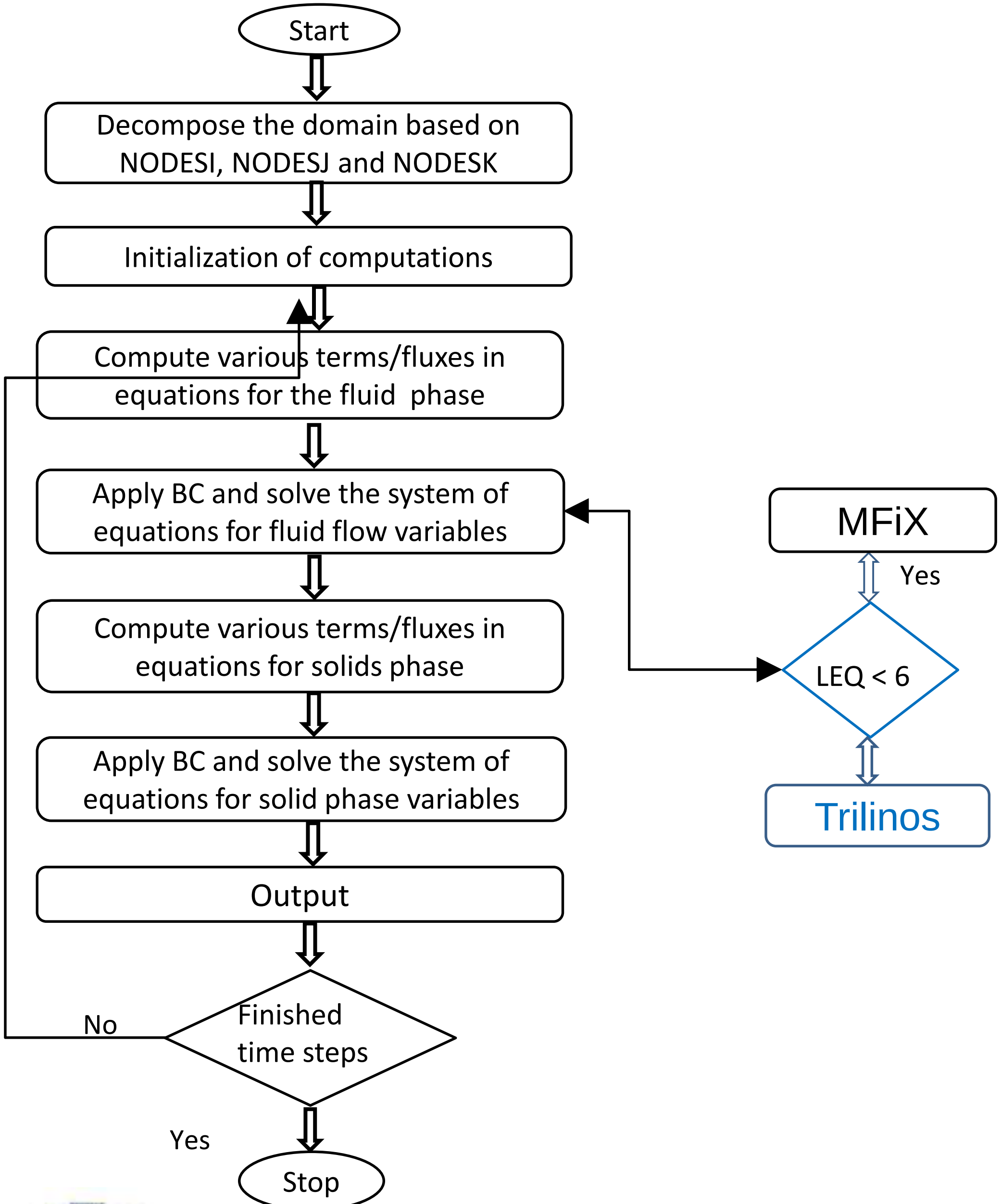
Trilinos Package Advancement

Category	1st Generation	2nd Generation	3rd Gen
Linear Algebra Services	**Epetra**, EpetraExt, Komplex	**Tpetra**, Xpetra, Domi, RTOp, Thyra	**Tpetra**, Xpetra
Tools	Teuchos, Triutils, Galeri, Optika, Trios	Teuchos, Sacado, Trios	
Direct Linear Solvers	**Amesos**, Pliris	**Amesos2**	
Iterative Linear Solvers	**AztecOO**	**Belos**, Stratimikos	**Belos**
Preconditioners	**IFPACK**, **ML**	**IFPACK2**, **MueLu**, ShyLU	**IFPACK2**, **MueLu**
Nonlinear Solvers	NOX, LOCA	NOX, LOCA	
Eigensolvers	Anasazi	Anasazi	
Optimization & Analysis	MOOCHO	MOOCHO, OptiPack, Phalanx, Piro, ROL	
Meshing & Load Balance	STK, Zoltan, Isorropia, Mesquite, Moertel	STK, Zoltan2, Pamgen	
Discretizations	Intrepid, Shards, Rythmos	Intrepid, Shards, Tempus	
Performance Portability			**Kokkos**

Trilinos + MFiX

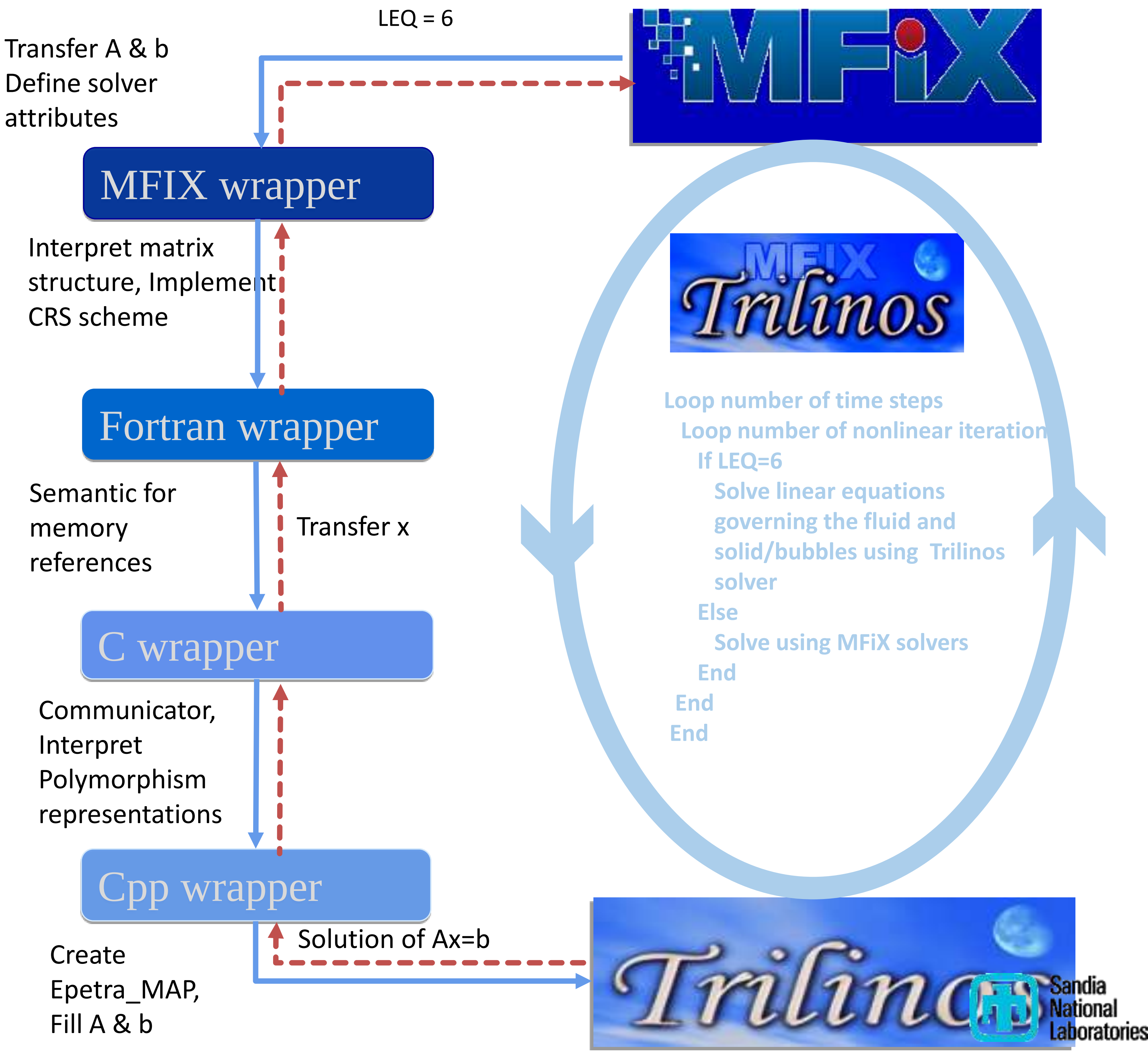
Using Trilinos Linear Solver with MFiX

Trilinos-MFiX Flow chart



LEQ	Method	Description
1	SOR	Point Successive Over Relaxation
2	BiCGSTAB	Bi-Conjugate Gradient STABilized method
3	GMRES	Generalized Minimal RESidual algorithm
4	BICGSTAB + GMRES	
5	CG	Conjugate Gradient
6	BiCGSTAB/GMRES/CG/Direct/….	Trilinos

A language independent interface to integrate legacy codes

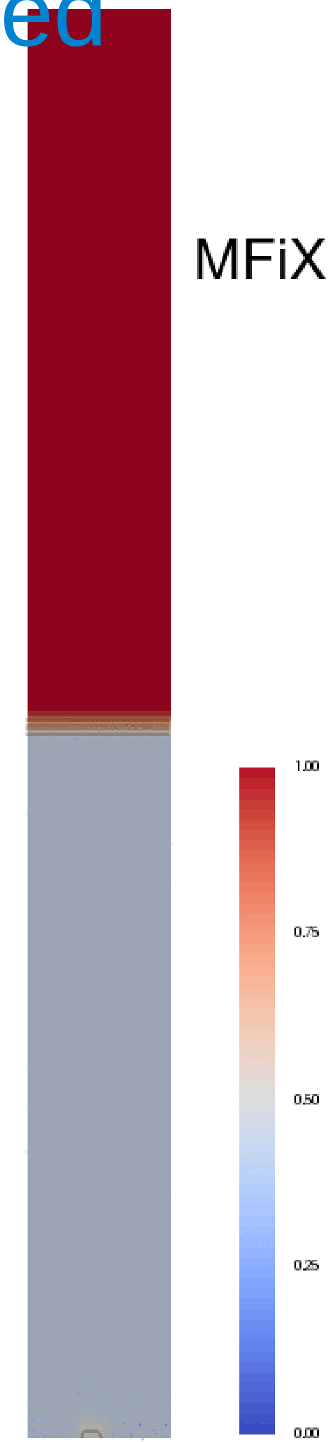


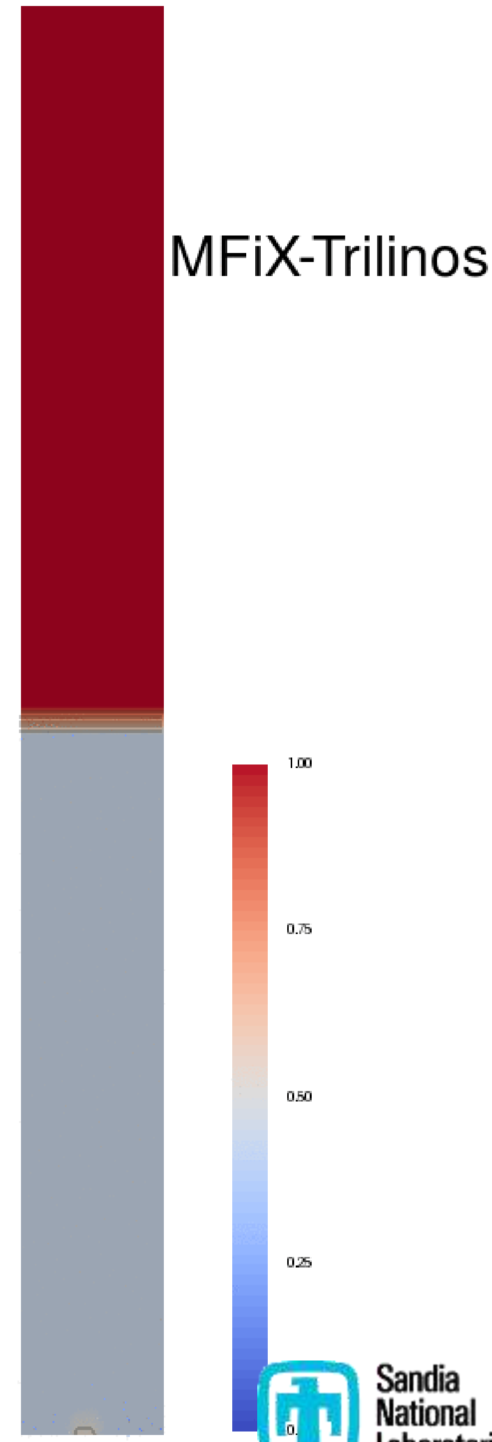
2D bubbling fluidized bed

Properties
particle diameter = 0.04cm
particle density = 2.0 g/cm^3
Restitution co-efficient = 0.80
Angle of internal friction = 30
Fluid viscosity = 0.00018 g/cm s
Fluid density = 0.0012 g/cm^3

Boundary conditions
Inlet: constant mass inflow
124.6 cm/s for 4.3<x<5.7; 25.9cm/s for 0<x<4.3, 5.7<x<10
Sidewalls: slip condition
Outlet : pressure outflow condition (p = 0)

Dimensions: 10cm X 100cm



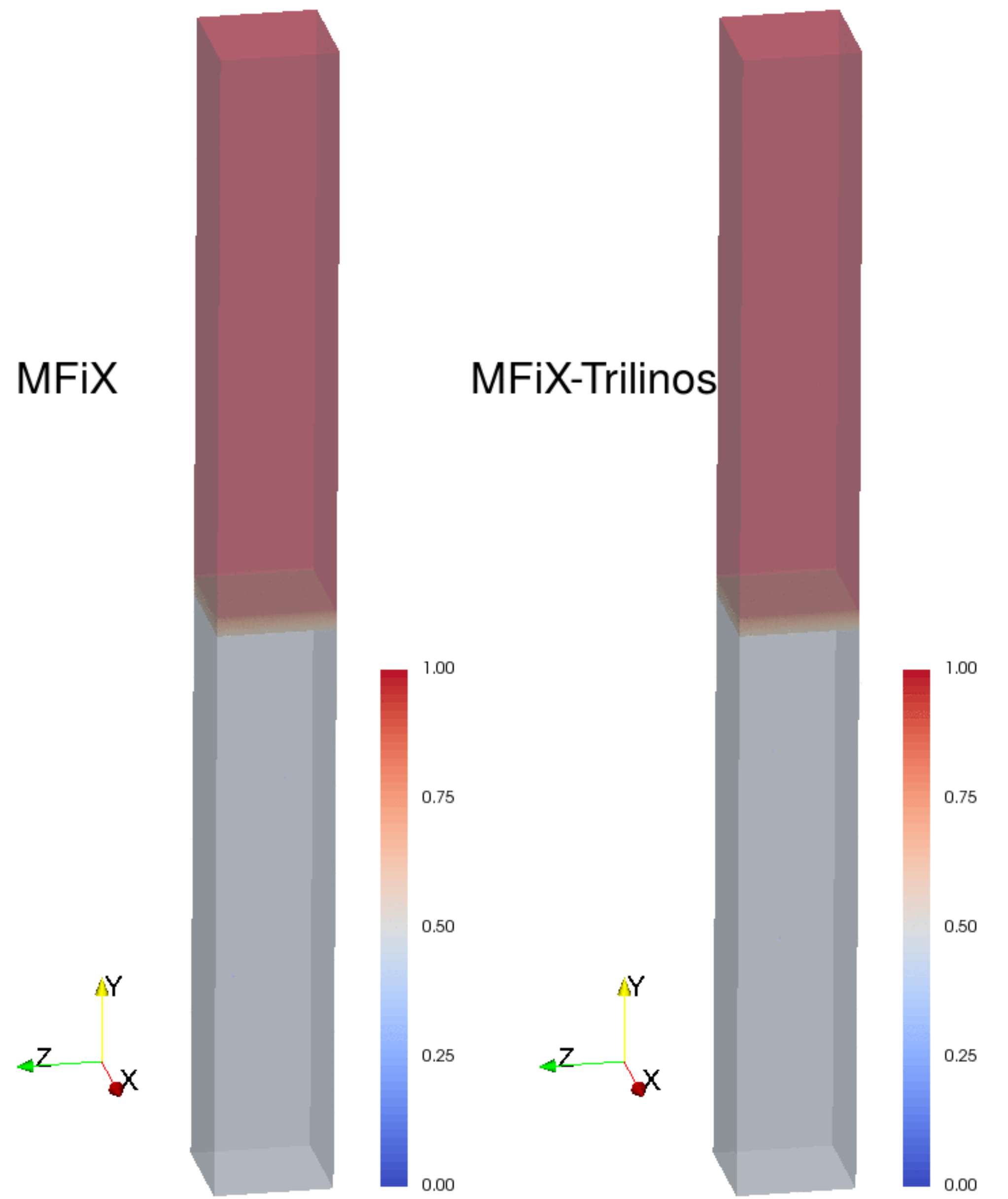


Flow in a fluidized bed (3D)

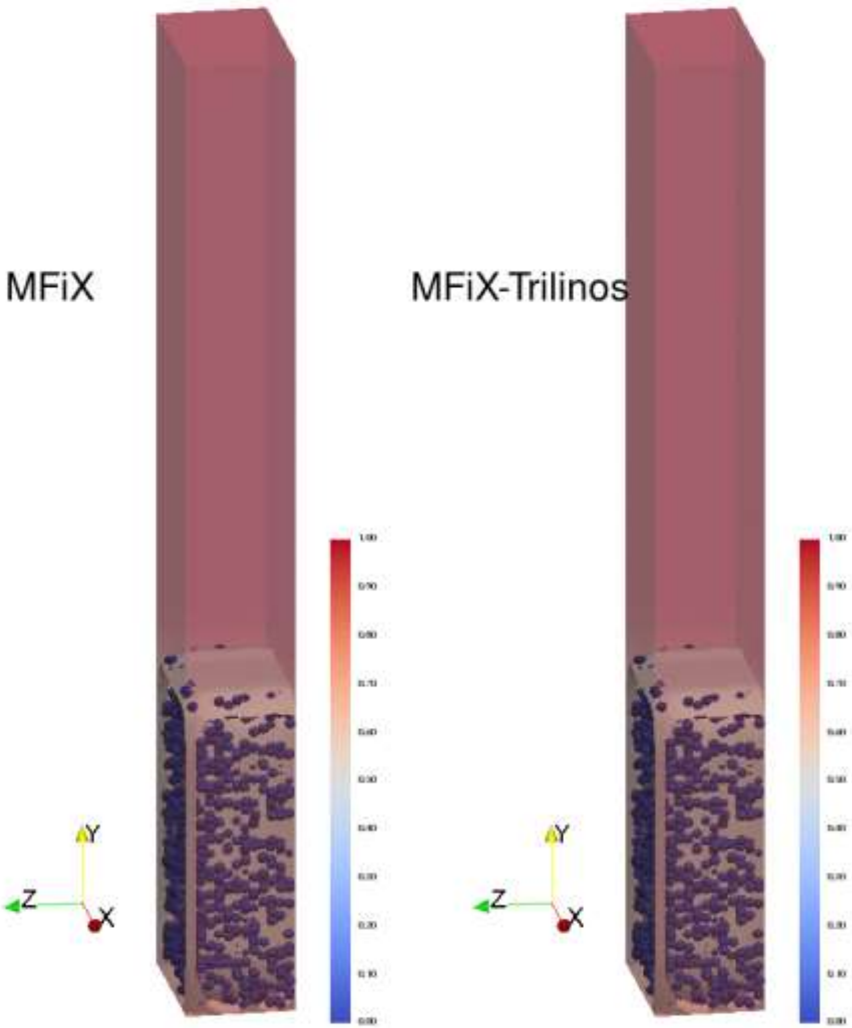
Bubbling bed
Dimensions : 10cm x 10cm x 100cm

Sand particle diameter	= 0.04cm
Sand particle density	= 2.0 g/cm^3
Restitution co-efficient	= 0.80
Angle of internal friction	= 30
Fluid (gas) viscosity	= 0.00018 g/cm s
Fluid density	= 0.0012 g/cm^3

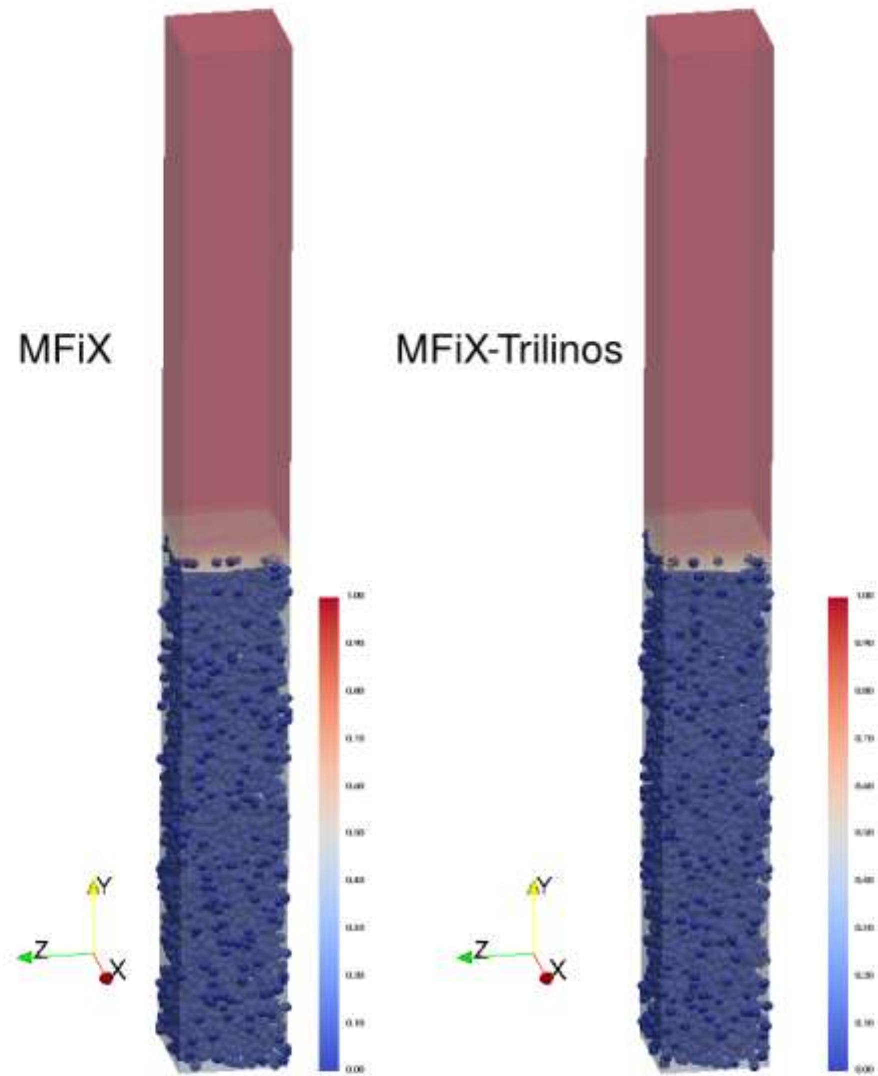
Boundary conditions
Inlet: constant mass inflow
(124.6 cm/s for $4.3<x<5.7$, $4.3<z<5.7$)
Sidewalls: slip condition
Outlet : pressure outflow condition



Flow in a fluidized bed



DEM



PIC

Sandia National Laboratories

Speedup - Trilinos MFiX vs MFiX

3D Bubbling flow Problem

Case 1: Mesh Size = 10M
Case 2: Mesh size = 200M

Computers:

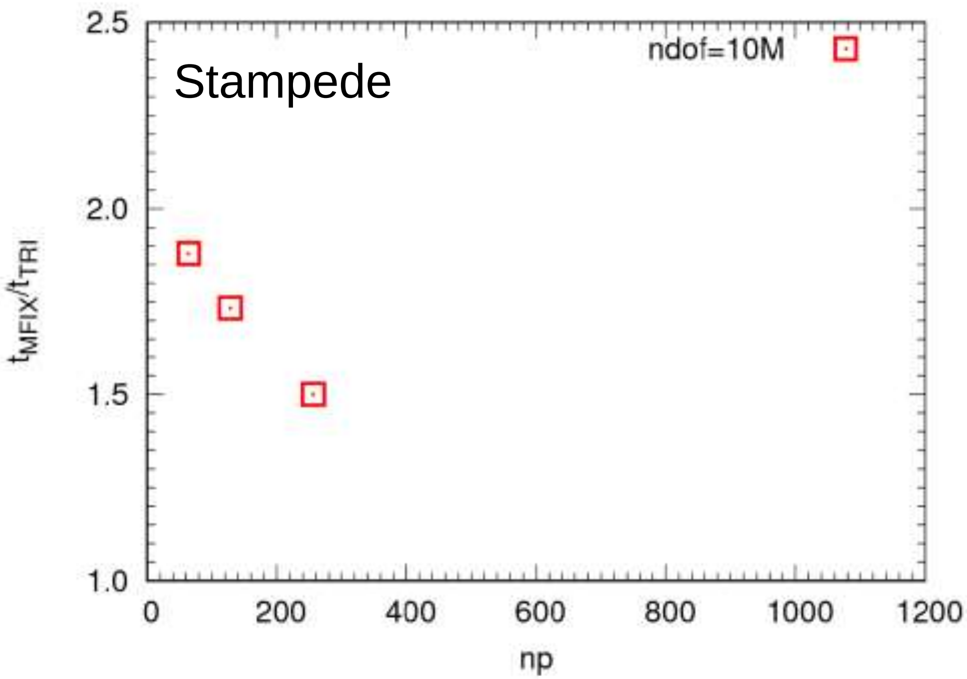
Stampede: Texas Advanced Computing Center (TACC)
Bridges: Pittsburgh Supercomputing Center (PSC)
Comet: San Diego Supercomputer Center (SDSC)

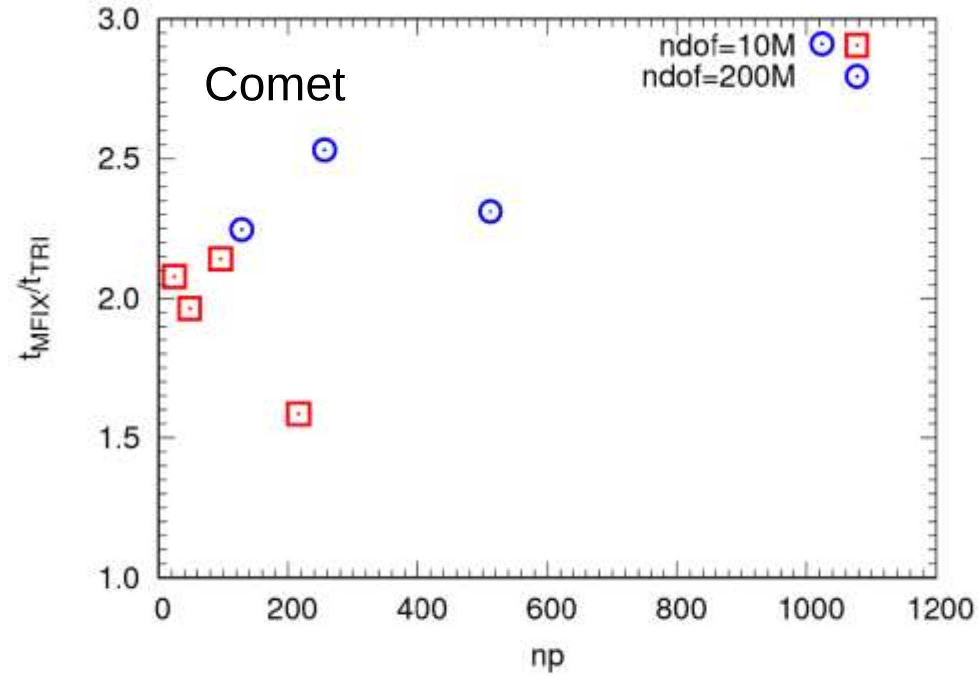
Various computer architectures
used for the performance analysis study

	Stampede (AS)	Comet (AC)	Bridges (AB)
Model (Intel Xeon)	E5-2680 2.7GHz	E5-2695 2.5GHz	E5-2695 2.30 GHz
Cores per socket	8	12	14
Sockets	2	2	2
L1 cache (KB)	32	32	32
L2 cache (KB)	256	256	256
L3 cache (KB)	20480	30720	35840
RAM (GB)	32	128	130

Sandia National Laboratories

Performance of MFiX-Trilinos

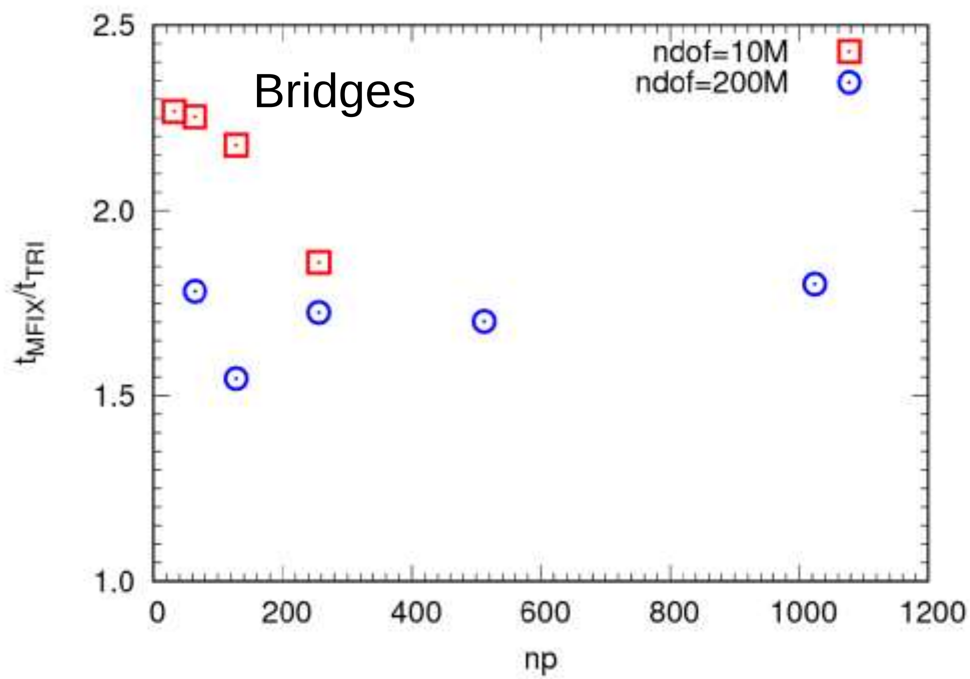




Trilinos

Solvers: BiCGStab
Packages: Tpetra (obj), Belos (solver), MueLu (pre)

PC: Jacobi



UTEP

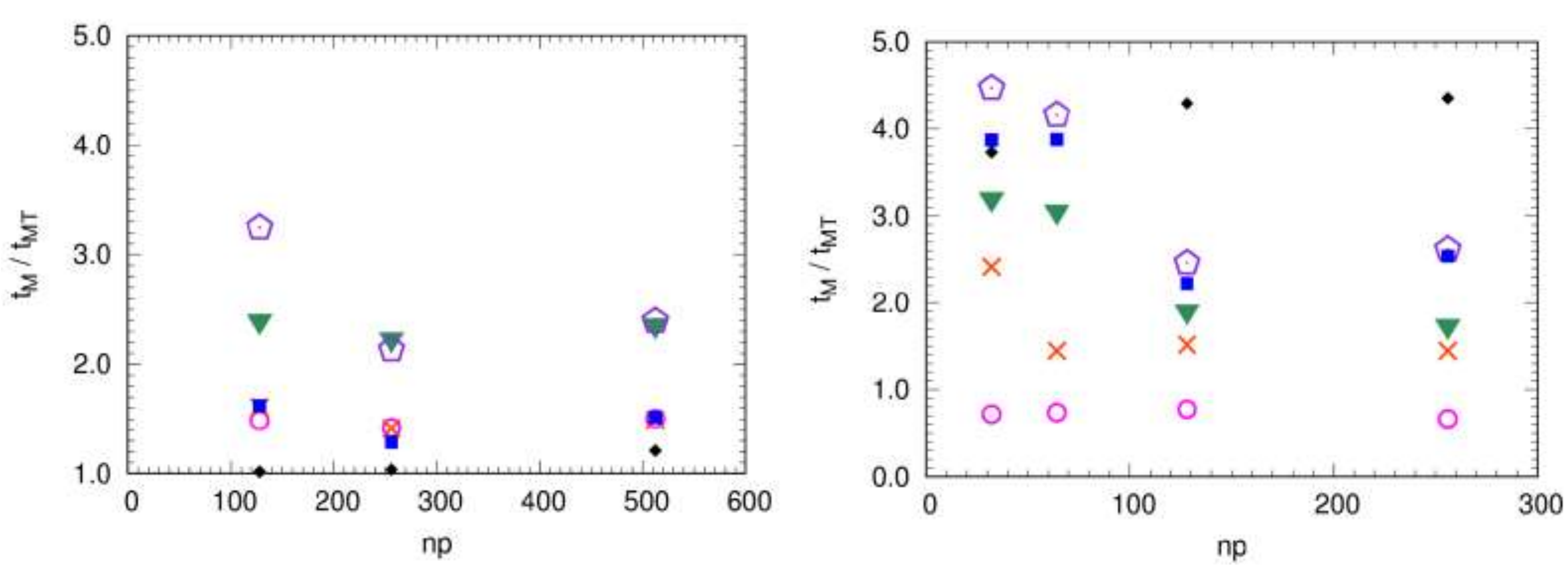
Sandia National Laboratories

Linear iterative solvers

Trilinos

Solvers: BiCGStab, TFQMR, Pseudoblock TFQMR
GMRES, Flexible GMRES, Recycling GMRES, Hybrid GMRES

Packages: Tpetra (obj), Belos (solver), MueLu (pre)

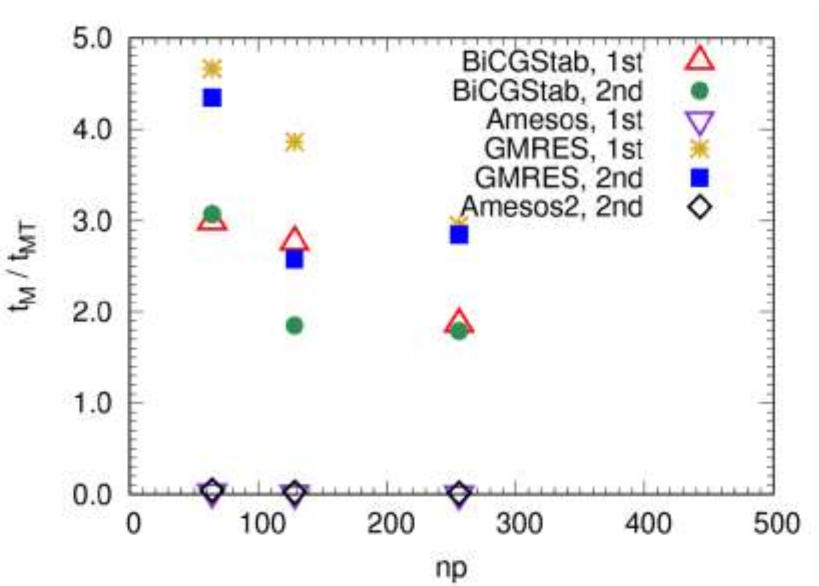


Mesh M1 (10M)

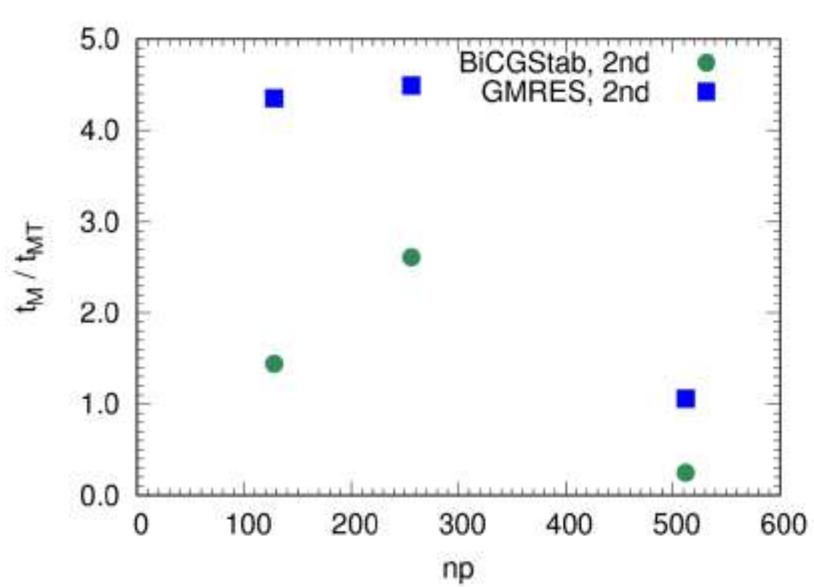
Mesh M2 (200M)

First and second generation solver stacks

First Generation
Iterative Solvers: BiCGStab, GMRES
Packages: Epetra (obj), Aztec(solver), ML
PC: Smoothed Aggregation



Second generation
Iterative Solvers: BiCGStab, GMRES
Packages: Tpetra (obj), Belos (solver), MueLu
PC: Smoothed Aggregation



Concluding remarks and future work

- Presented MFiX linear solver integration framework with Trilinos's Linear solver
- Analyzed solver performance for various problem sizes and HPC systems -- a speed of ~5 times was noticed

Future

- Portability – Kokkos
- Efficient memory and data transfer management

Thank you!

Q&A ?

Acknowledgments: DOE/NETL – UCR, XSEDE, Sandia, ME & Computational Depts