Science and Technology Facilities Council

Multiscale Universal Interface for Heterogeneous high-Performance Computing Systems

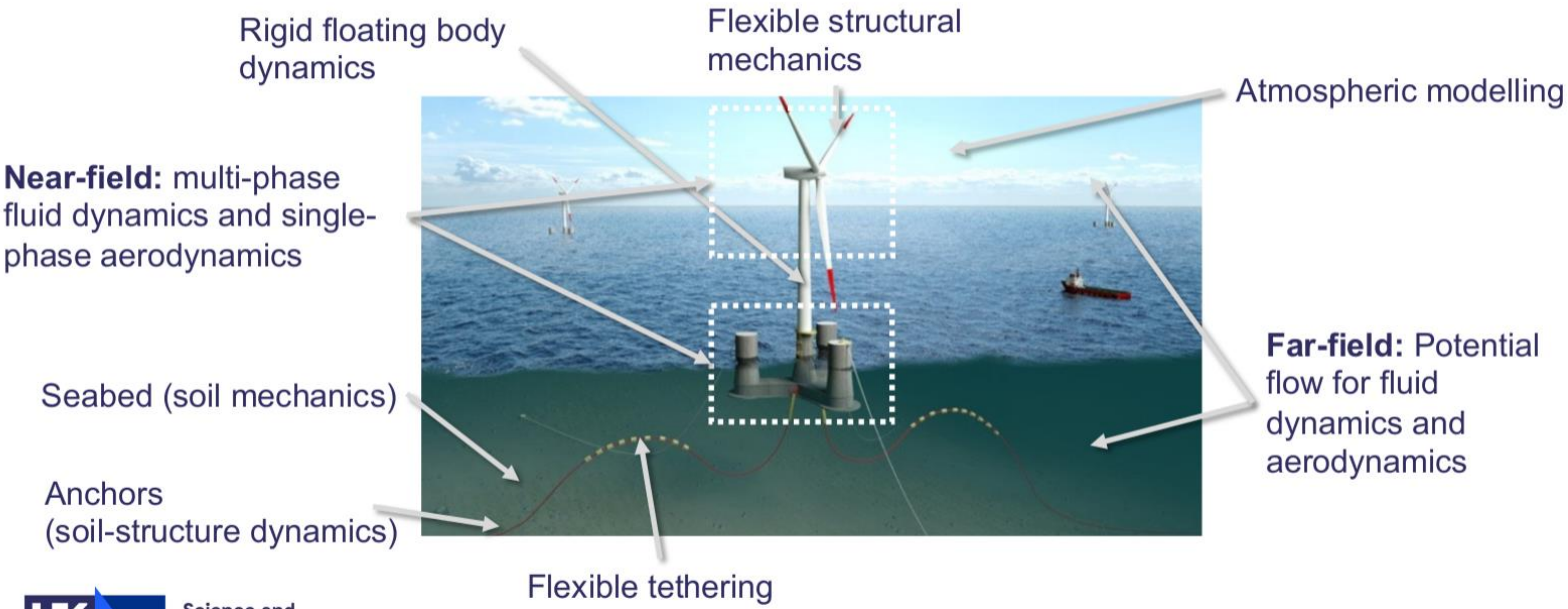
Omar Mahfoze, Mayank Kumar, Wendi Liu, Stephen M. Longshaw, David R. Emerson

UKRI Science & Technology Facilities Council, Daresbury Laboratory, UK

CoSeC
Computational Science Centre for Research Communities

Why do we need to couple codes?

Complex multiphysics/multiscale problems



Science and Technology Facilities Council
Scientific Computing

www.ccp-wsi.ac.uk/data_repository/test_cases/test_case_015

CCP-WSI a Collaborative Computational Project in Wave Structure Interaction
www.ccp-wsi.ac.uk

HEC-WSI a High End Computing Consortium for Wave Structure Interaction
www.hec-wsi.ac.uk

Why do we need to couple codes?

- **Multi-physics:** Coupling different methods or solvers to simulate a single problem involving multiple physical aspects (e.g. structural mechanics and fluid dynamics)
- **Multi-scale:** Coupling different methods or solvers to simulate a physical process while considering significantly different length or time scales
- Code coupling can be: **monolithic** or **partitioned**:
  - **Software:** single executable *or* multiple executables with inter-process communication
  - **Algorithmic:** single system of equations *or* multiple discrete systems joined through a connective term such as a boundary condition
- Partitioned solutions need complex inter-process communication and generalised capabilities like spatial interpolation

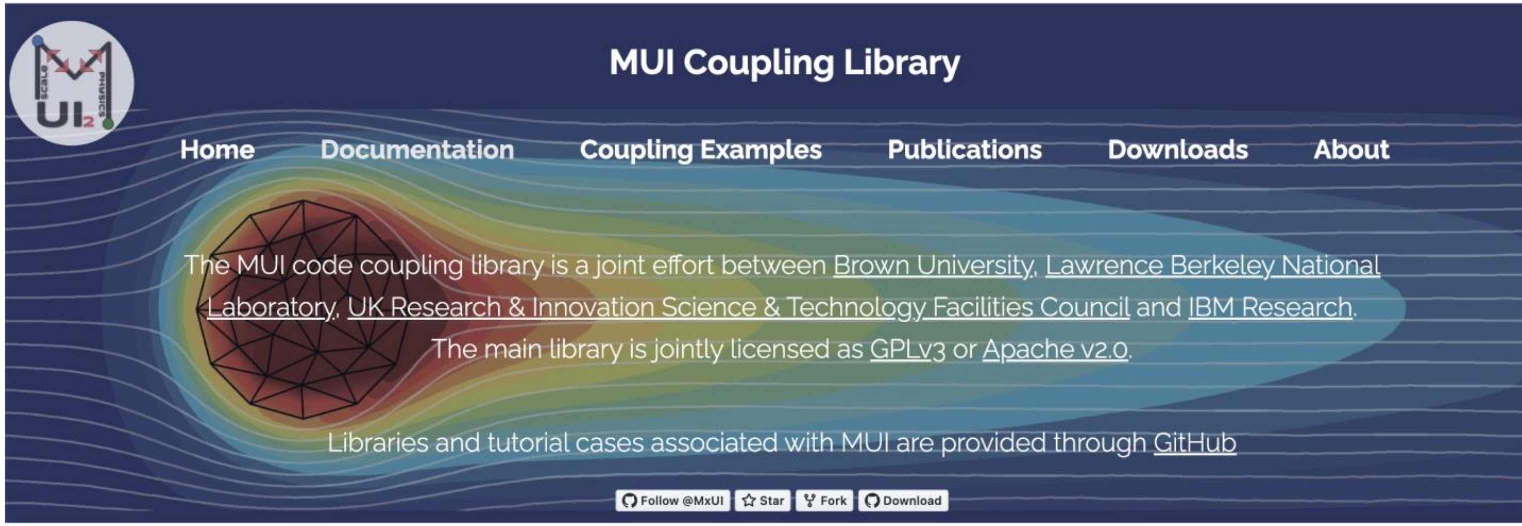
Coupling Library:
The Multiscale Universal Interface (MUI)

Science and Technology Facilities Council
Scientific Computing

MUI Overview

- Written in C++11 (with wrappers for C, Fortran and Python)
- Open-source, licensed at the user's choice as either GPLv3 or Apache 2.0
- Header-only design with only external dependency being MPI
- Creates a peer-to-peer MPI based interface between two or more codes
- MPI multi-program multi-data (MPMD) design allows large numbers of apps to be coupled together simultaneously
- No external dependency on other library (except MPI)

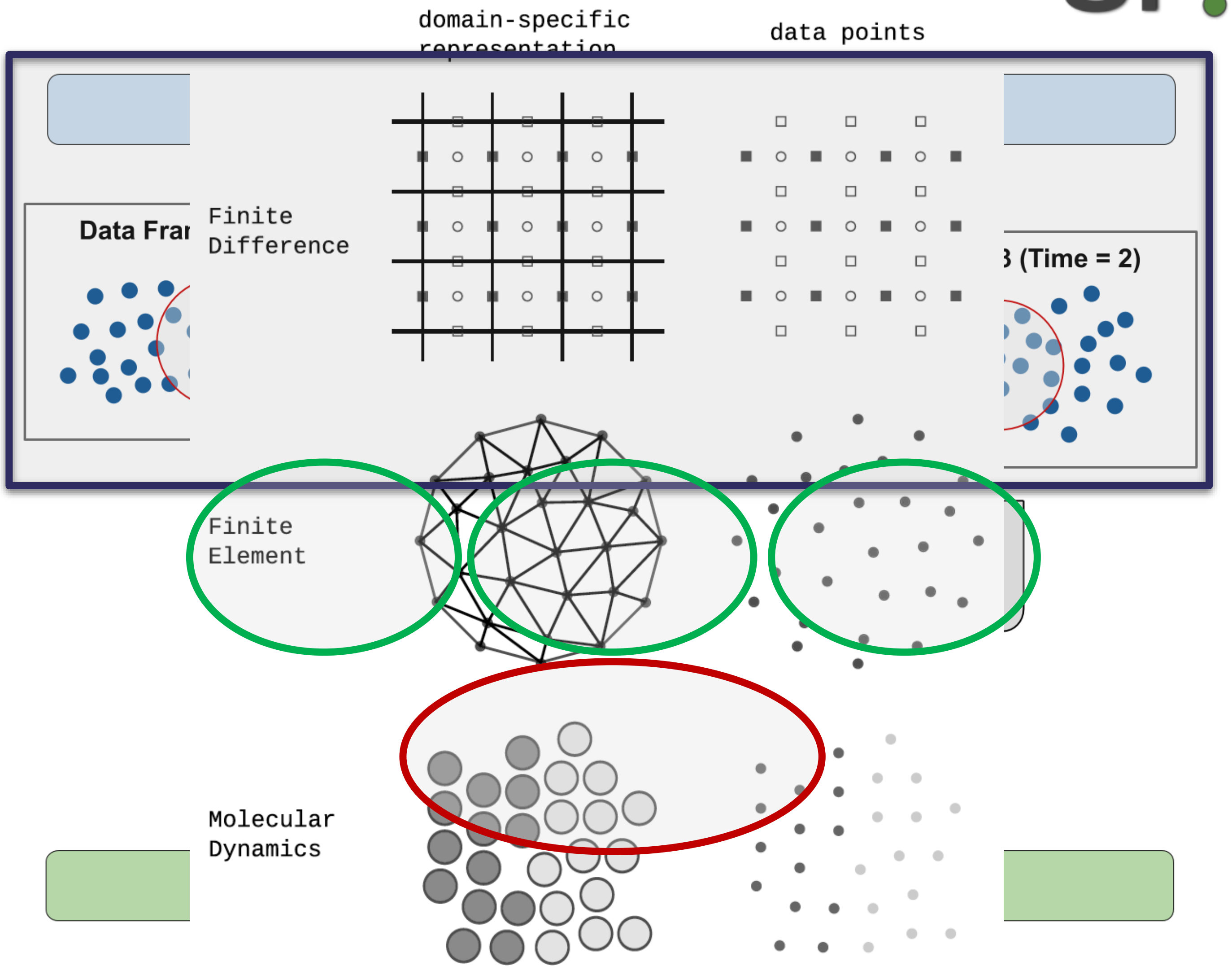
- https://github.com/MxUI/MUI



Science and Technology Facilities Council
Scientific Computing

MUI Wrokflow

- Couples using a set of discrete data samples and an **interface**:
  1. Convert domain-specific representations to a general form (a **cloud of points** with associated data)
  2. Solver **imparts** data (at a point) to interface with an **associated time-stamp** using **non-blocking** operations
  3. Other solver requests data at specific location and time from MUI interface using **spatial** and **temporal** sampler using **blocking** operations



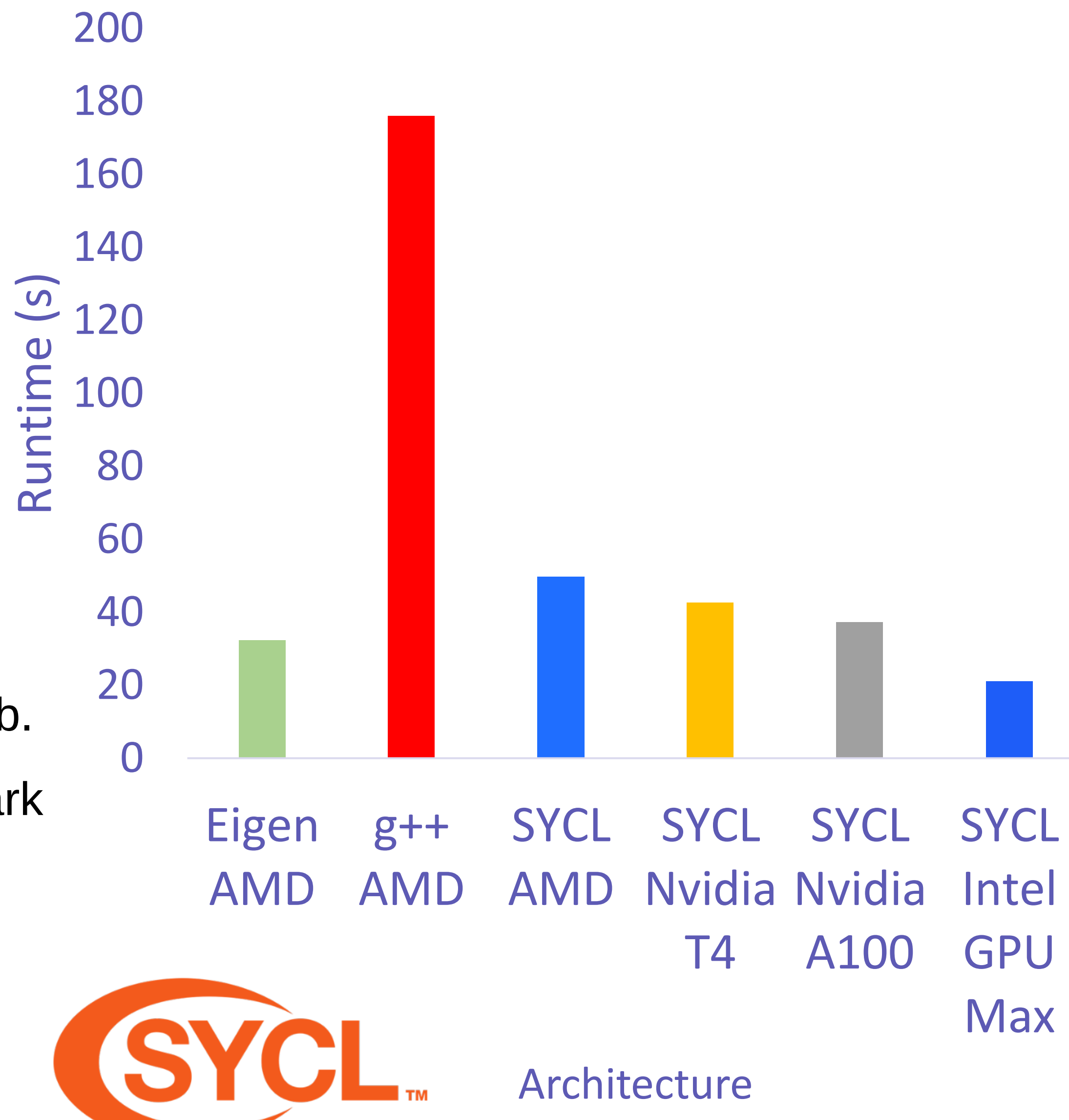
Science and Technology Facilities Council
Scientific Computing

What is in the MUI toolkit?

- API to create an MPI based interface between 2 or more apps
- Extensible frameworks of spatial and temporal samplers as well as coupling helpers:
  - **10 spatial samplers:** simple Gaussian, quintic SPH approach, Radial Basis Function (RBF) approach with both conservative and consistent modes and many others.
  - **Temporal samplers:** allowing simple concepts like summation or averaging in time but with scope for more complex operations.
  - **Coupling helpers**: enable approaches like the Aitken's and FR iterative methods
- A self-contained linear algebra library is part of MUI 2.0, currently used by the RBF spatial filter but able to be called from any filter or coupling helper

MUI Linear Algebra library using SYCL

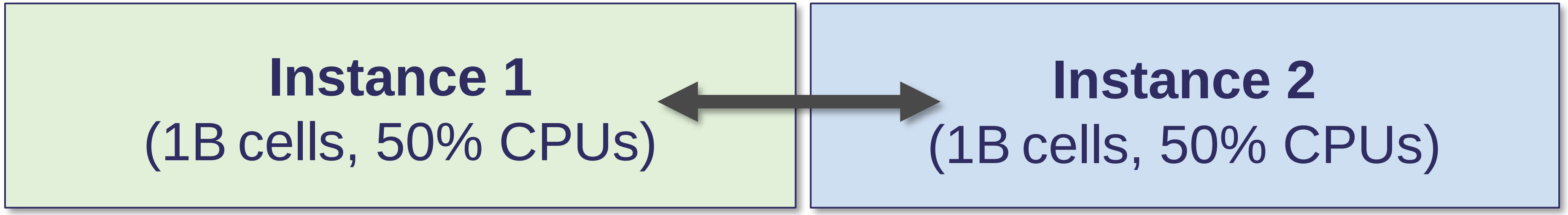
- **Solvers:** Conjugate Gradient, BiCGStab, Gauss Elimination
- **Preconditioner:** Diagonal, Incomplete Cholesky, Incomplete LU, SSOR
- The solver library is parallelized using SYCL.
  - SYCL is a royalty-free, cross-platform abstraction layer.
  - Enables code for heterogeneous and offload processors to be written using modern ISO C++
  - Work ongoing through an **EPSRC ExCALIBUR** project and **Intel Centre of Excellence** hosted at Daresbury Lab.
- Initial results confirm good speed up comparable to benchmark linear solver library Eigen



Science and Technology Facilities Council
Scientific Computing

MUI Performance Benchmark

- AMD EPYC HPE Cray EX (~750K cores)
- Representative of a typical 3D CFD problem coupled to itself:
  - Simulated local computation load
  - Simulated local MPI transfer using standard MPI 3D Cartesian decomposition
  - **Assumes linear scaling of CFD solver**
- 1 billion points transferred per instance (2B total) – full volumetric coupling
- Total of 48GB of data transferred via MUI
- Both with and without Gaussian spatial interpolation

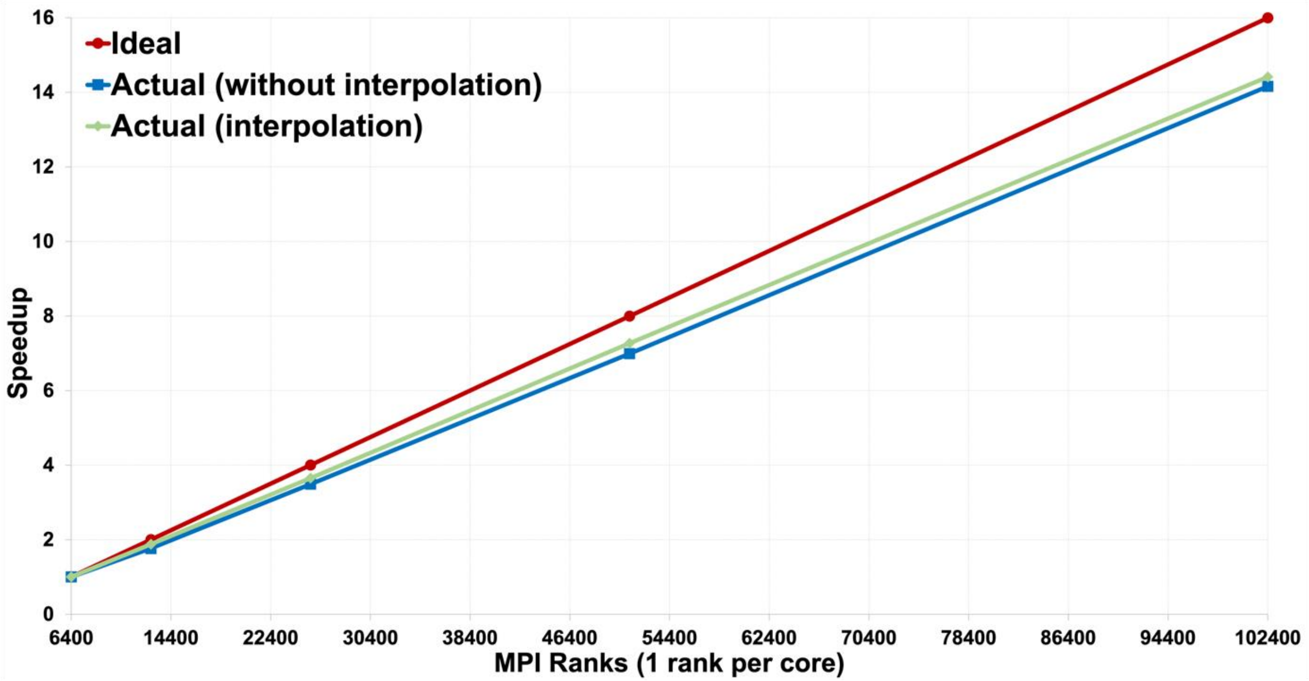


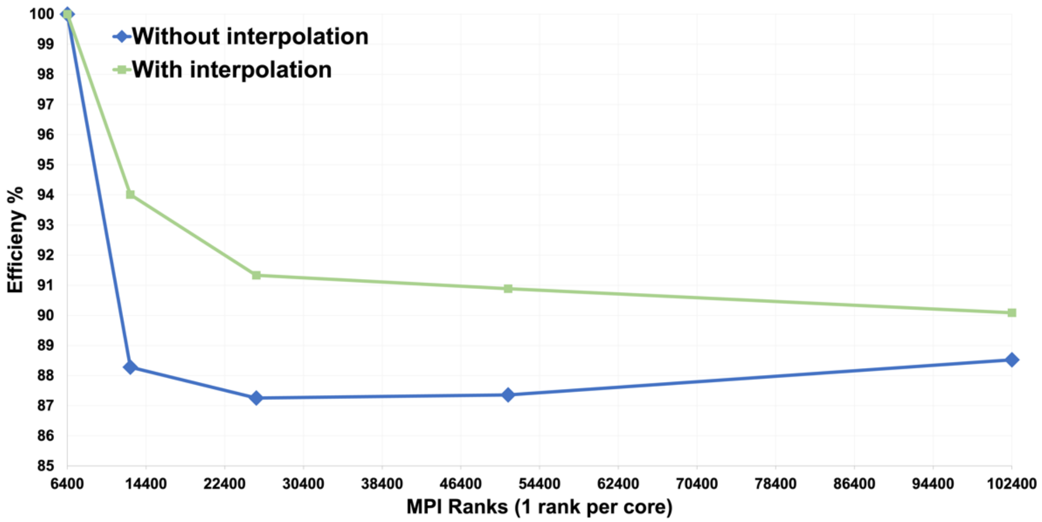
Science and Technology Facilities Council
Scientific Computing

https://github.com/MxUI/MUI-Testing

archer2

MUI Performance





Science and Technology Facilities Council

Scientific Computing

archer2

MUI Coupled Applications

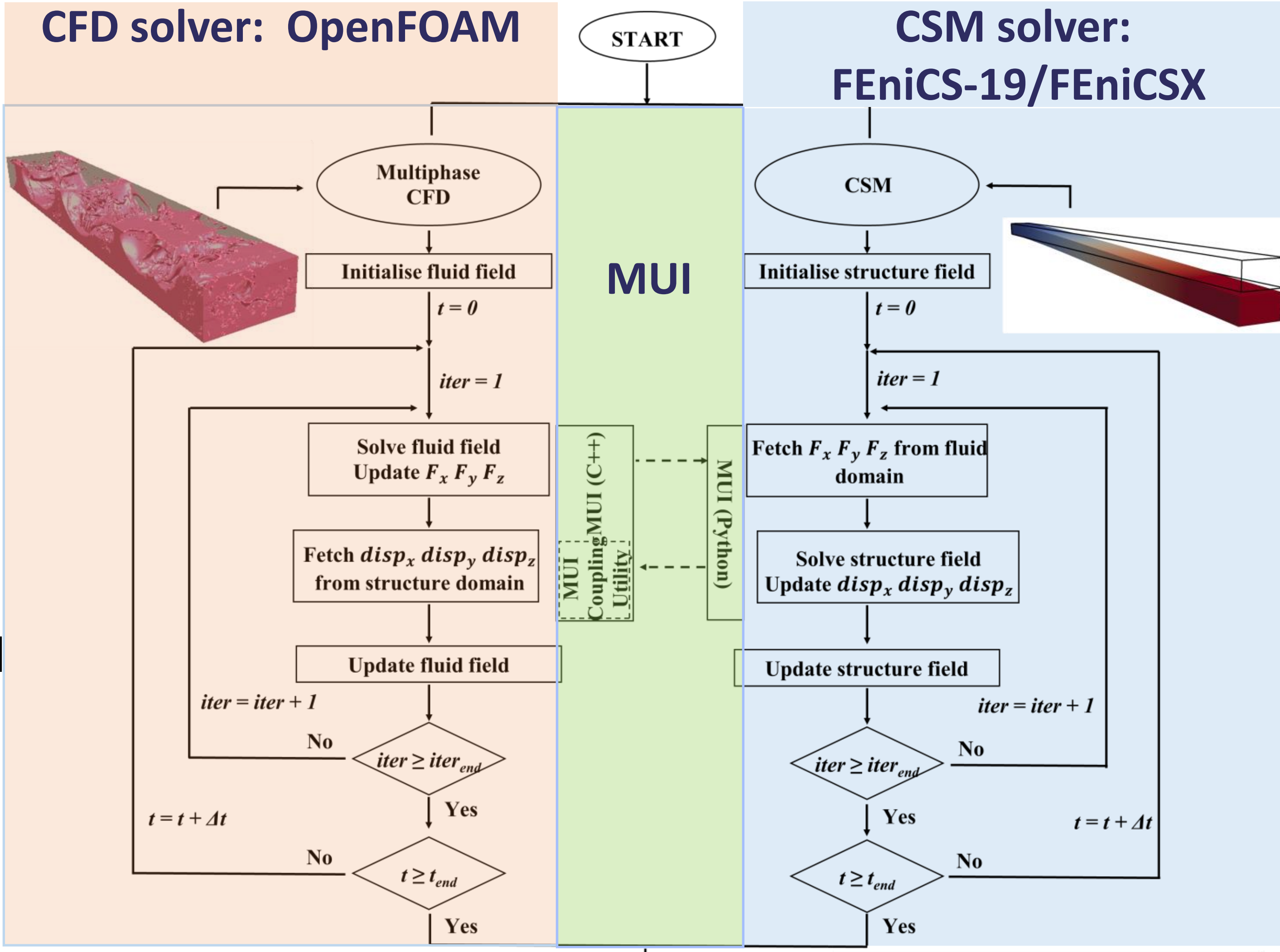
Science and Technology Facilities Council
Scientific Computing

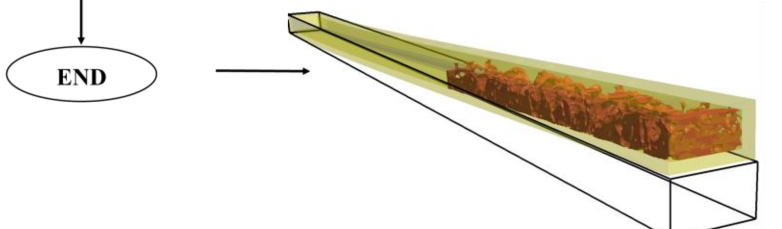
ParaSiF - Fluid Structure Interaction Framework

- Mesh-based solvers
- **OpenFOAM (ESI 2206)**
  - PimpleFSIFoam (Incompressible)
  - InterFSIFoam (Two-phase flow)
  - Single Point Coupling (most solvers)

- **CSM solver FEniCS-19 and FEniCSx**
  - generalized harmonic oscillator model
  - hyper-elastic model

https://github.com/ParaSiF

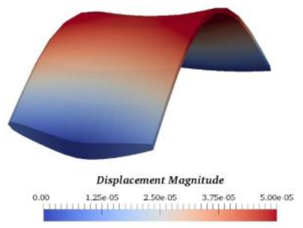




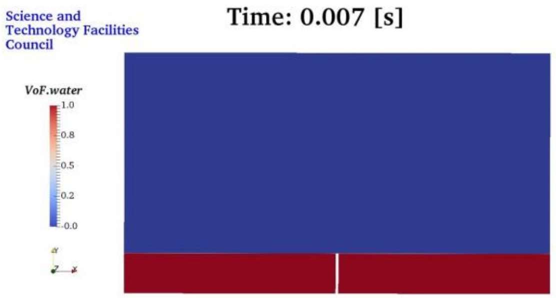
Science and Technology Facilities Council
Scientific Computing

W Liu et al. 'A high-performance open-source solution for multiphase fluid-structure interaction.' International Journal of Offshore and Polar Engineering 32(1), 24-30, 2022

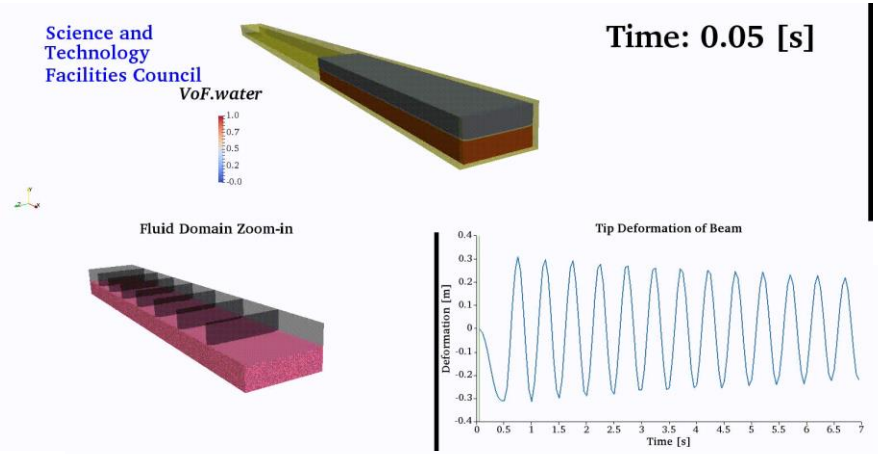
ParaSiF Cases

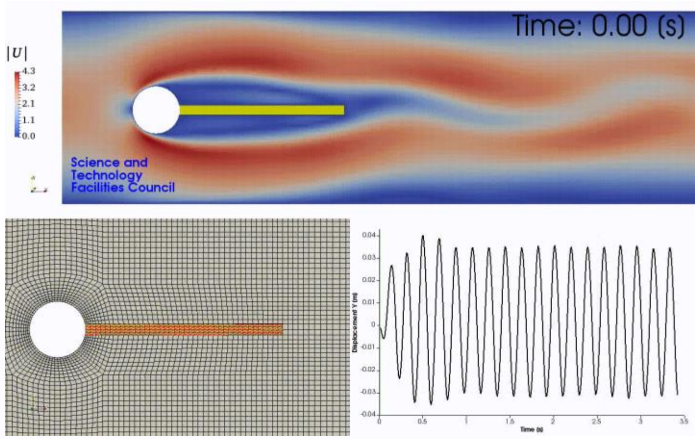


Vortex-induced vibration of the trailing edge of a hydrofoil

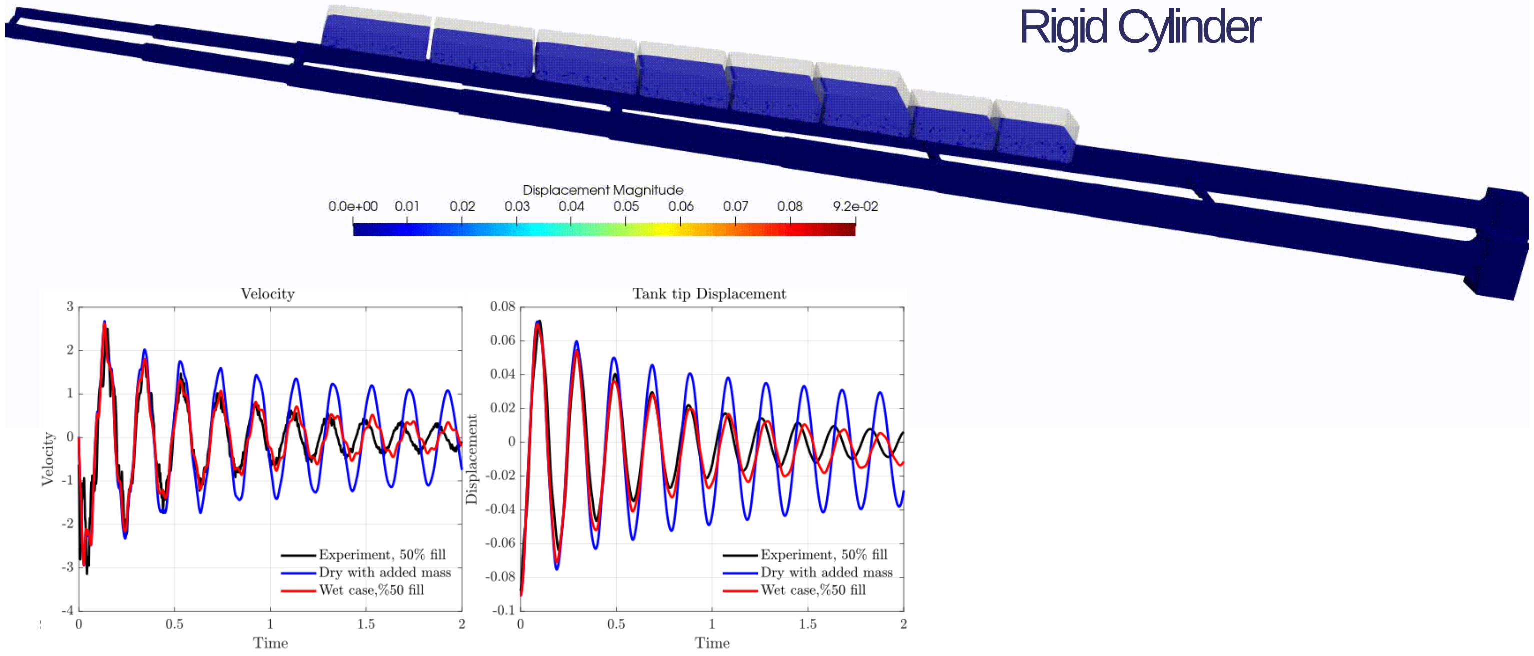


2-D roll tank with flexible beam





2-D Flow Pass Elastic Plate Behind a Rigid Cylinder

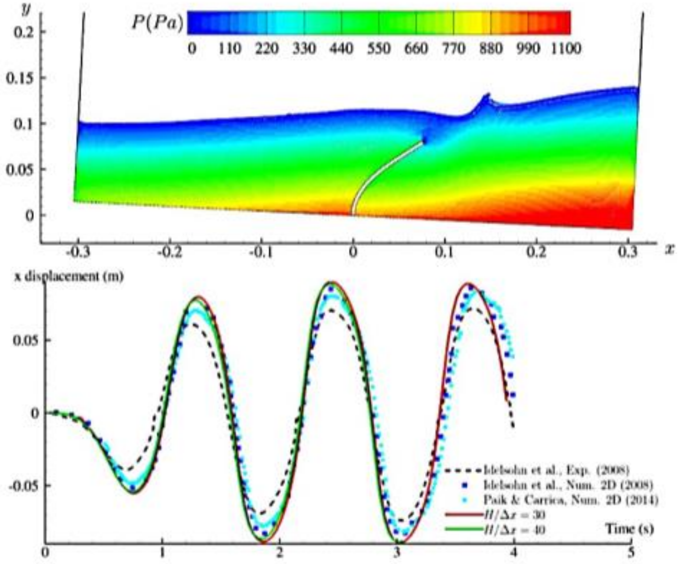


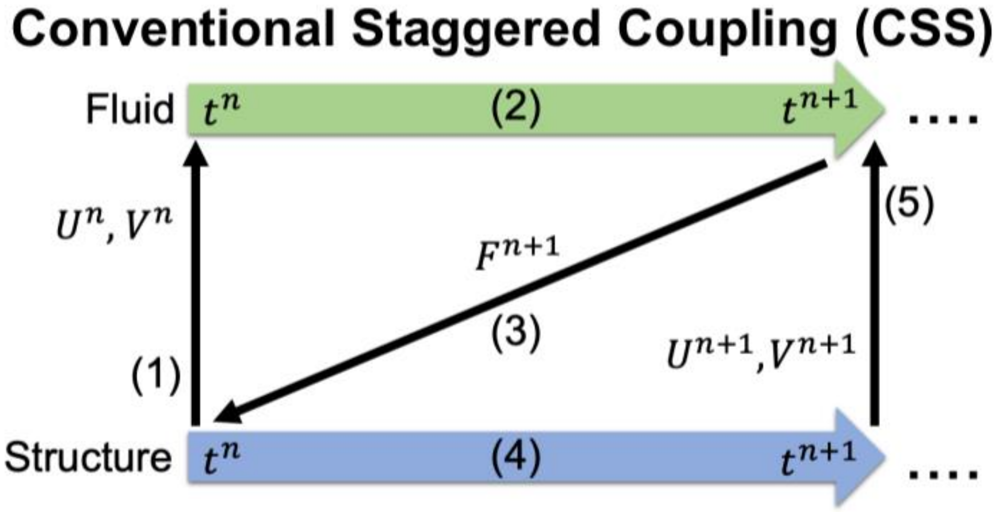
The Damping of fuel Sloshing in Wing-Like structures

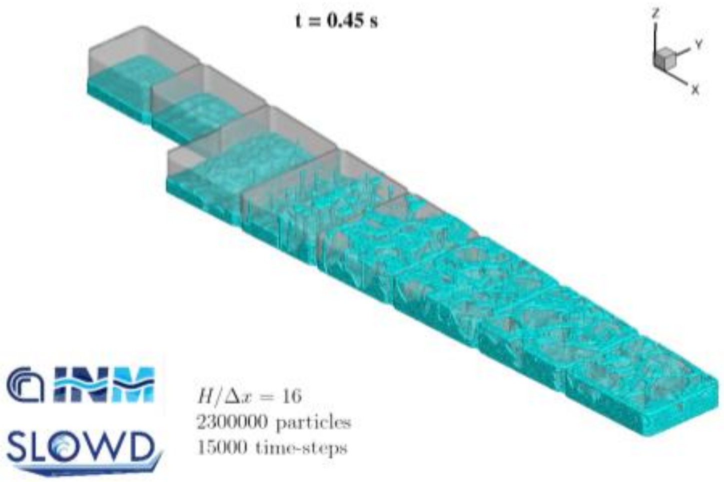
Science and Technology Facilities Council
Scientific Computing

Fluid Structure Interaction (FSI)

- Coupling CFD (SPH Flow) with FEA (MSC Nastran) for sloshing problems:
- Commercial SPH solver – explicit time-stepping; mesh-based boundary condition
- Commercial Finite Element (FE) solver – implicit time-stepping







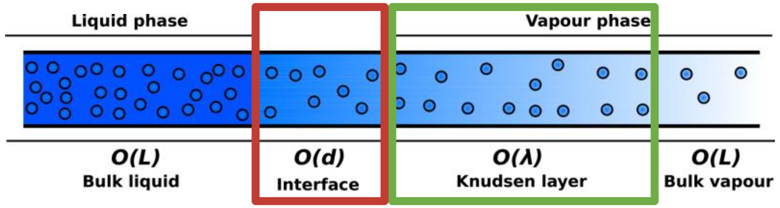
Science and Technology Facilities Council
Scientific Computing

S. M. Longshaw et al. A Coupled FSI Framework Using the Multiscale Universal Interface. International Forum on Aeroelasticity and Structural Dynamics, Madrid, Spain. 2022.

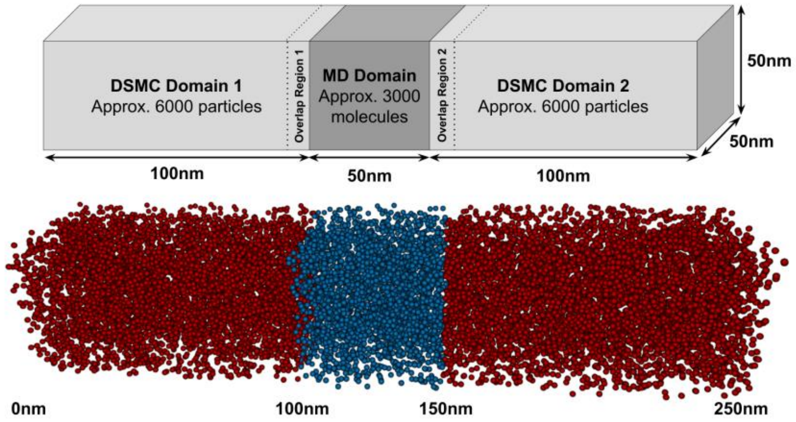
Molecular Modelling of Gas Dynamics

Coupling OpenFOAM based Molecular Dynamics (MD) with Direct Simulation Monte Carlo (DSMC) to simulate the process of evaporation

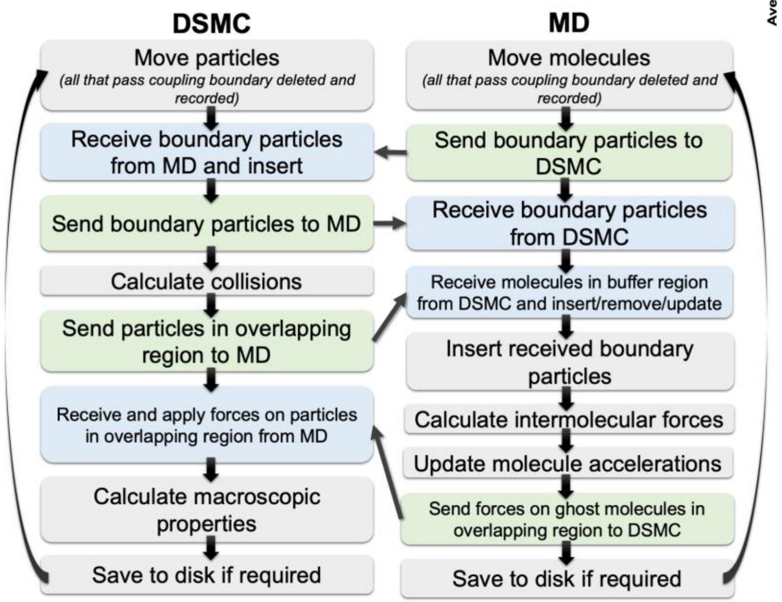
- Canonical NVT (contant number of atoms, volume and temp)

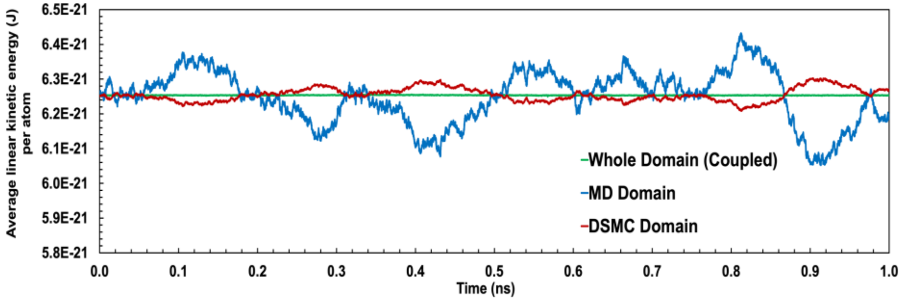


MD DSMC



https://github.com/MicroNanoFlows

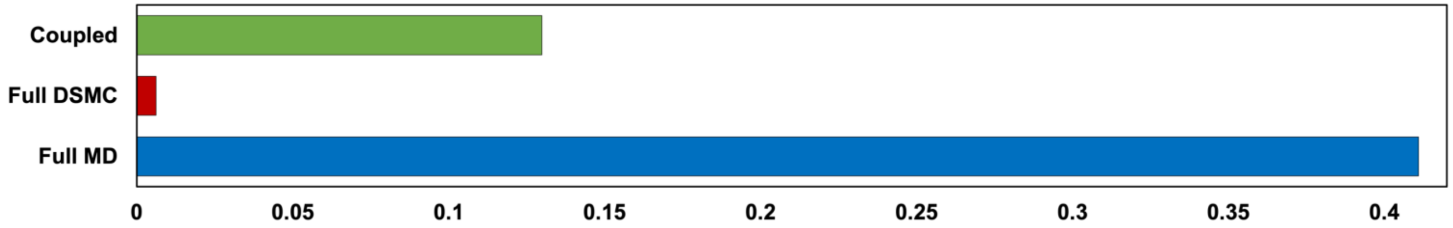




Coupled Linear Kinetic Energy (J)

S. M. Longshaw et al., Coupling Molecular Dynamics and Direct Simulation Monte Carlo using a general and high-performance code coupling library, Computers & Fluids, 213, 104726, 2020.

Computational time per step (s)



Science and Technology Facilities Council

Scientific Computing

Conclusions

Science and Technology Facilities Council
Scientific Computing

Conclusions

The **M**ultiscale **U**niversal **I**nterface (MUI) is a general purpose particle-based scientific code coupling library:

- Open-source and available on GitHub
- Header only with no external dependency allow for simple implementation
- Utilises MPI and provides a simple C++ library for many-to-many couplings (it has C, Fortran and python wrappers)
- Provides a variety of spatial and temporal interpolation methods
- Supports coupling of solvers designed for heterogenous systems
- Parallel scalability normally not limited by library but by **design of the couplings** and **the codes being coupled**
- Applicable to many coupled problem types

Science and Technology Facilities Council

Scientific Computing

https://github.com/MxUI/MUI