Mango-Selm Software:
Fluid-Structure Interaction subject to Thermal Fluctuations
Soft Materials, Biophysics, Fluidics

August 2021, LAMMPS Workshop

Paul J. Atzberger

Department of Mathematics

Department of Mechanical Engineering

University of California Santa Barbara

DOE ASCR CM4
DE-SC0009254

DOE ASCR PhILMS
DE-SC0019246

NSF Grant
DMS - 1616353

NSF CAREER Grant
DMS-0956210

Tips for SELM Package: Installation and Usage

Instructions

Pre-compiled binaries: for (Debian 9+/Ubuntu and Centos 7+, Python 3.6+).

Quick start:
>> pip install selm-lammps
test: >> python -c "from selm_lammps.tests import t1; t1.test()“

Alternatively, can use *docker run -it ubuntu:20.04 /bin/bash; (use python3).*

Examples, docs, source: see http://mango-selm.org/

Directories:

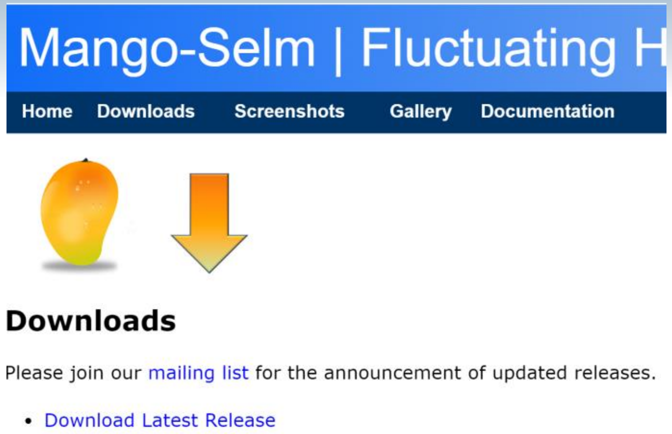
- src/USER-SELM: main C/C++ interface codes for SELM package
- examples/USER/selm: example models and simulation scripts
- lib/selm: library of SELM integrators and related algorithms

Installing Python components (uses lammps.py, lammps.so):

- Recommended to use *conda* or *virtualenv* with python >=3.6
- Examples for Jupyter notebooks, python scripts, for running simulations.
- See README files for more details.

Jupyter Notebooks and Python-based Simulations:

- Uses python-interfaces to LAMMPs.
- L.command(cmd_str); runs the command in cmd_str.
- Model building using python wrapper
  - sets up particles
  - sets up interactions, many possible types and potential available
  - sets up the simulation parameters.



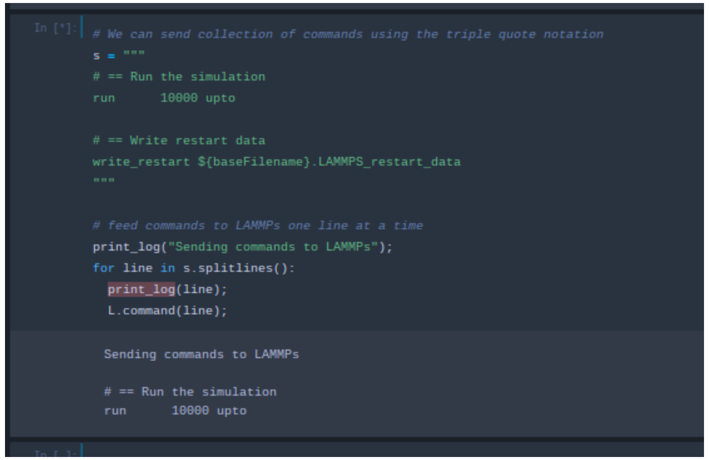
Downloads

Please join our mailing list for the announcement of updated releases.

- Download Latest Release

Download: http://mango-selm.org/

Python and Jupyter notebook interfaces

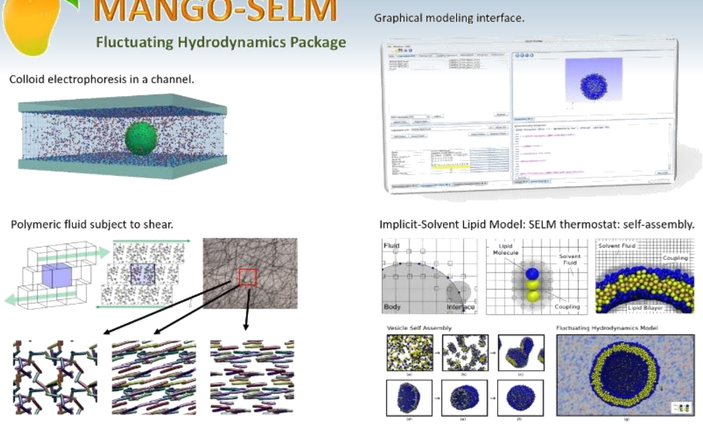


```
In [*]: # We can send collection of commands using the triple quote notation
        s = """
        # == Run the simulation
        run      10000 upto

        # == Write restart data
        write_restart ${baseFilename}.LAMMPS_restart_data
        """

        # feed commands to LAMMPs one line at a time
        print_log("Sending commands to LAMMPs");
        for line in s.splitlines():
          print_log(line);
          L.command(line);

         Sending commands to LAMMPs

         # == Run the simulation
         run      10000 upto

In [ ]:
```

Mango-Selm Software: Overview

MANGO-SELM

Fluctuating Hydrodynamics Package



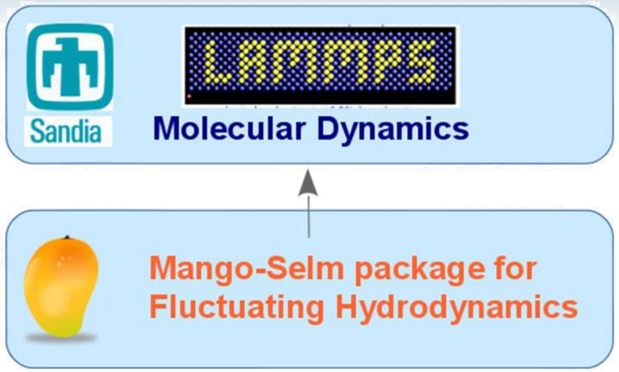
Colloid electrophoresis in a channel.

Graphical modeling interface.

Polymeric fluid subject to shear.

Implicit-Solvent Lipid Model: SELM thermostat: self-assembly.

Molecular dynamics Integration



Sandia

LAMMPS

Molecular Dynamics

Mango-Selm package for Fluctuating Hydrodynamics

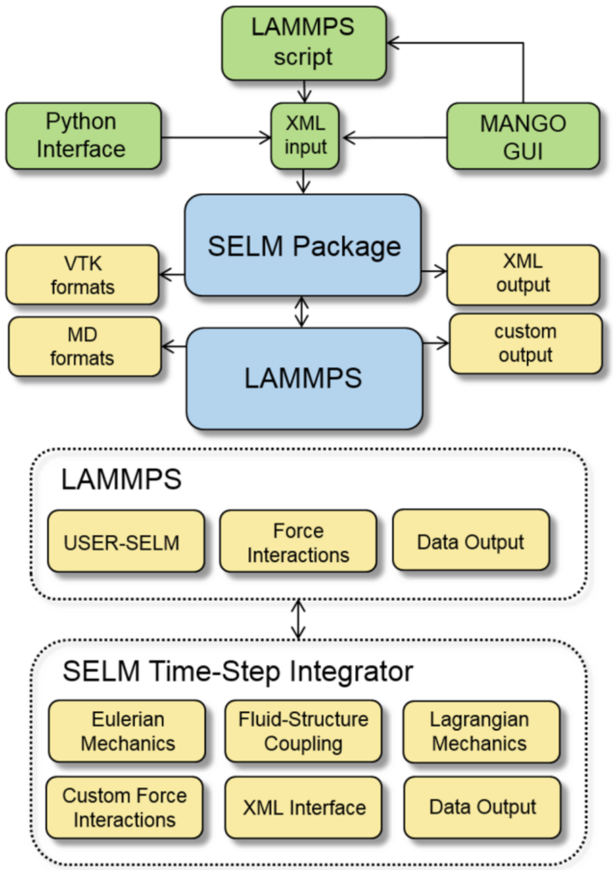
Download: http://mango-selm.org/

Python and Jupyter notebook interfaces

jupyter

python

Modular Design and Extendible



Atzberger 2016

Selm - Simulation Package:

SELM fluctuating hydrodynamics for fluid-structure interactions subject to thermal fluctuations.

- **Python, Jupyter notebooks**, and other scripting for model building and simulation.
- **Stochastic numerical time-step integrators** for inertial and quasi-steady physical regimes in (C/C++).

Molecular dynamics integration with LAMMPS

- **Modeling, interactions, many potentials**, **statistical analysis**.
- **Thermostats and many ensembles possible** such as Lees-Edwards for shear simulations.

Standardized formats

- **XML** for parametrization and data output.
- **VTK** output for continuum fields and microstructures (visualization / analysis).

Mango – GUI for Model Building:

Graphical User Interface (GUI) for setting up model geometry and simulation parameters.

- **Generates scripts and data files** for SELM fluctuating hydrodynamics simulations.

Extendible object-oriented architectures for inclusion of new numerical methods.

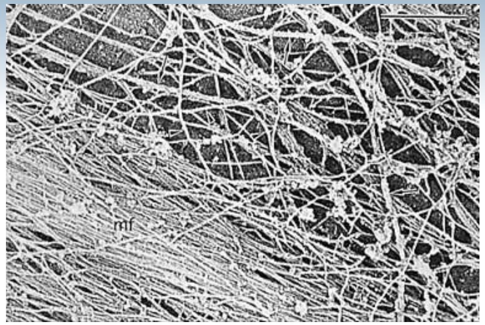
Motivations

Stochastic Immersed Boundary Methods (SIBMs)

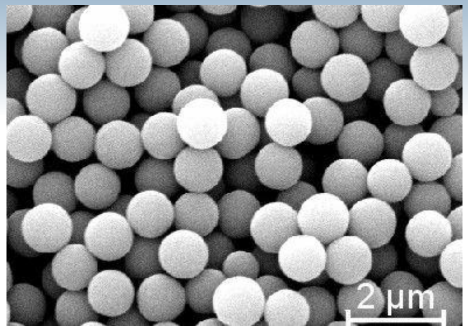
Eulerian-Lagrangian Methods (ELMs)

Implicit-Solvent Coarse-Grained (IS-CG) Simulations

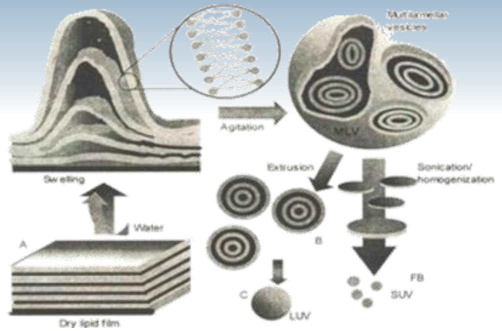
Motivations: Soft Materials, Complex Fluids, and Other Applications



Gels (Actin)



Colloids



Membranes (lipids)

Soft Materials / Complex Fluids

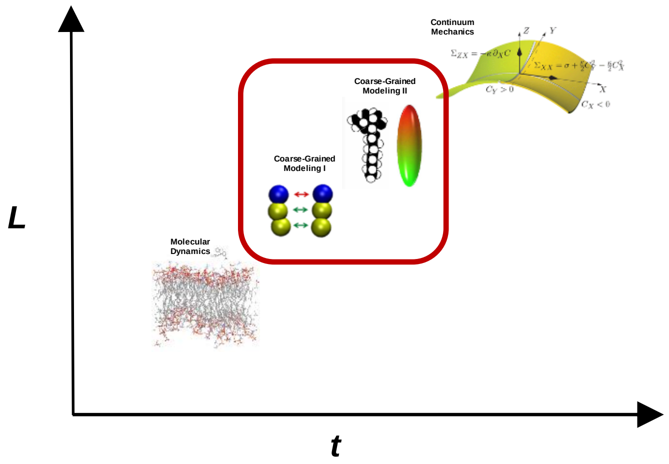
- Microstructure interactions on the order of K_BT.
- Properties arise from balance of entropy-enthalpy.
- Solvent plays important role (interactions / dynamic responses).

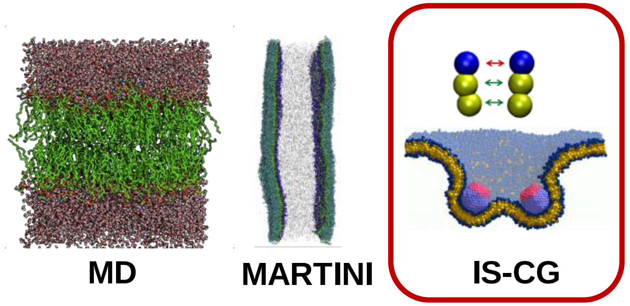
Approaches

- Atomistic Molecular Dynamics.
- Continuum Mechanics.
- Coarse-Grained Particle Models (solvated or implicitly treated).
- Challenges from phenomena spanning wide temporal-spatial scales.

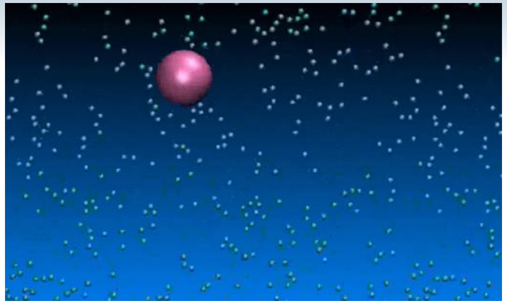
Simulation Aims

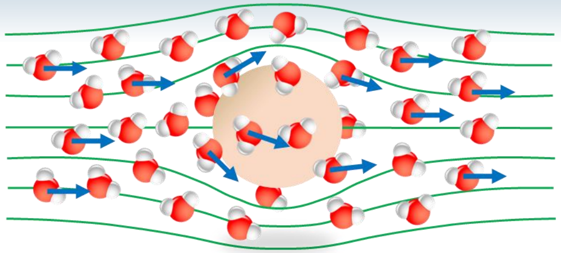
- Investigate how larger-scale mechanics arise from microstructure interactions / kinetics.
- Capture roles of solvent mediated interactions efficiently (i.e. continuum level).
- Resolve microstructure mechanics and dynamics.
- Computational efficiencies allow for accessing larger length and time-scales for investigating wider class of phenomena.

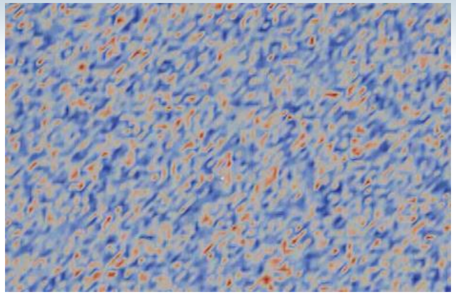

L
t
Molecular Dynamics
Coarse-Grained Modeling I
Coarse-Grained Modeling II
Continuum Mechanics
$\Sigma_{ZX} = -\kappa\partial_X C$
$\Sigma_{XX} = \sigma + \frac{\kappa}{2}C_Y^2 - \frac{\kappa}{2}C_X^2$
Z
Y
X
$C_Y > 0$
$C_X < 0$



Fluctuating Hydrodynamics


Brownian Motion: Molecular Collisions


Hydrodynamics + Fluctuations


Continuum Gaussian Random Field

Landau-Lifschitz fluctuating hydrodynamics

$$\rho\left(\frac{\partial \mathbf{u}(\mathbf{x},t)}{\partial t} + \mathbf{u}(\mathbf{x},t)\cdot\nabla\mathbf{u}(\mathbf{x},t)\right) = \mu\Delta\mathbf{u}(\mathbf{x},t) - \nabla p(\mathbf{x},t) + \nabla\cdot\mathbf{\Sigma}(\mathbf{x},t).$$

$$\nabla\cdot\mathbf{u}(\mathbf{x},t) = 0.$$

$$\langle \Sigma_{ij}(\mathbf{x},t)\Sigma_{kl}(\mathbf{y},s)\rangle = 2\mu k_B T\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right)\delta(\mathbf{x}-\mathbf{y})\delta(t-s).$$

Fluctuations arise from spontaneous momentum transfer from molecular-level collisions.

Stochastic model of thermal fluctuations captured through random stress Σ ~ Gaussian.

Challenges for analysis and numerical methods presented from the δ-correlation in space-time.

Fluid-structure interactions: How to incorporate tractably?

Stochastic Eulerian Lagrangian Methods (SELMs) for Fluid-Structure Interactions

Fluid Equations

$$
\begin{aligned}
\rho \frac{\partial \mathbf{u}}{\partial t} &= \mathcal{L}\mathbf{u} + \Lambda[\Upsilon(\mathbf{v} - \Gamma\mathbf{u})] + \lambda + \mathbf{f}_{\mathrm{thm}} \\
\nabla \cdot \mathbf{u} &= 0
\end{aligned}
$$

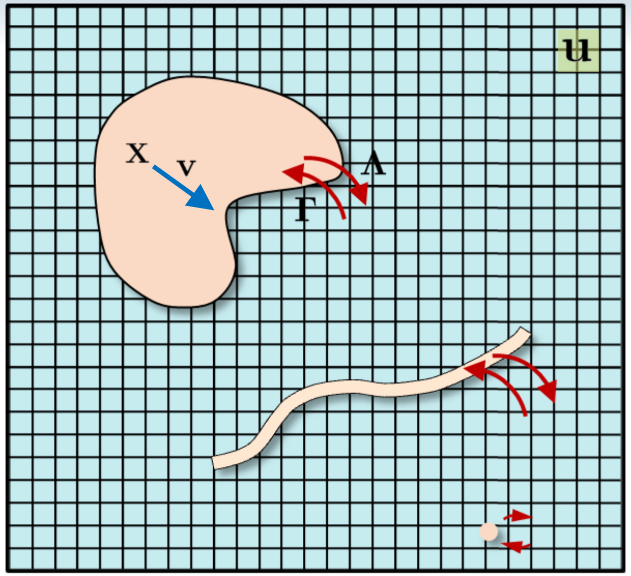
Microstructure Equations

$$
\begin{aligned}
\frac{d\mathbf{X}}{dt} &= \mathbf{v} \\
m\frac{d\mathbf{v}}{dt} &= -\Upsilon\left(\mathbf{v} - \Gamma\mathbf{u}\right) - \nabla_{\mathbf{X}}\Phi[\mathbf{X}] + \zeta + \mathbf{F}_{\mathrm{thm}}
\end{aligned}
$$

Thermal Fluctuations

$$
\begin{aligned}
\langle \mathbf{f}_{\mathrm{thm}}(s)\mathbf{f}_{\mathrm{thm}}^{T}(t)\rangle &= -\left(2k_BT\right)\left(\mathcal{L} - \Lambda\Upsilon\Gamma\right)\delta(t-s) \\
\langle \mathbf{F}_{\mathrm{thm}}(s)\mathbf{F}_{\mathrm{thm}}^{T}(t)\rangle &= \left(2k_BT\right)\Upsilon\,\delta(t-s) \\
\langle \mathbf{f}_{\mathrm{thm}}(s)\mathbf{F}_{\mathrm{thm}}^{T}(t)\rangle &= -\left(2k_BT\right)\Lambda\Upsilon\,\delta(t-s).
\end{aligned}
$$

Eulerian-Lagrangian Approach



Operators:

$\mathcal{L} \longrightarrow$ Fluid dissipation (viscosity).

$\Upsilon \longrightarrow$ Structure “slip” relative to local flow field.

$\Gamma \longrightarrow$ Kinematic particle velocity for given flow.

$\Lambda \longrightarrow$ Induced fluid force density from particle.

Notation:

$\mathbf{u} = \mathbf{u}(\mathbf{x}, t) \longrightarrow$ Fluid velocity.

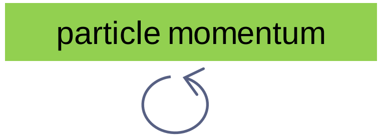
$\mathbf{X} = \mathbf{X}(\mathbf{q}, t) \longrightarrow$ Structure configuration

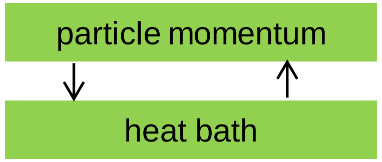
$\mathbf{v} = \mathbf{v}(\mathbf{q}, t) \longrightarrow$ Structure velocity.

Peskin, Kramer, Atzberger 2007, Atzberger 2011

Thermostats

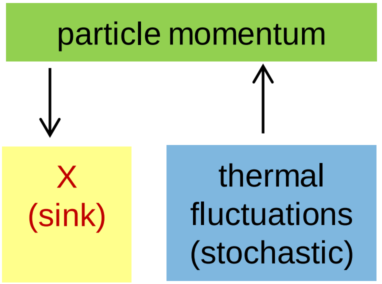
Berendson, Nose-Hoover


particle momentum


particle momentum

heat bath

Langevin


particle momentum

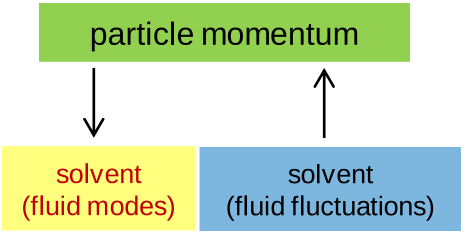
X (sink)

thermal fluctuations (stochastic)

$$
\begin{aligned}
m\frac{d\mathbf{V}}{dt} &= -\gamma\mathbf{V} - \nabla\Phi(\mathbf{X}) + \sqrt{2k_BT\gamma}\,\frac{d\mathbf{B}_t}{dt} \\
\frac{d\mathbf{X}}{dt} &= \mathbf{V}.
\end{aligned}
$$

missing correlations through solvent!

Fluctuating Hydrodynamics


particle momentum

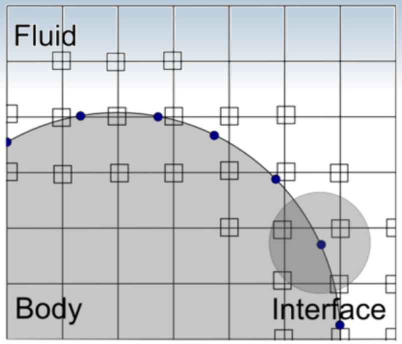
solvent (fluid modes)

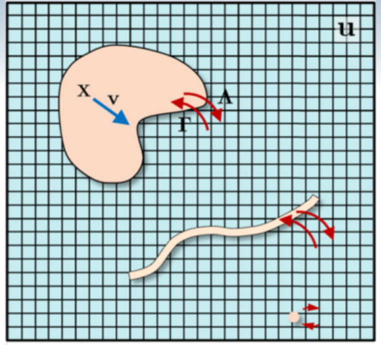
solvent (fluid fluctuations)

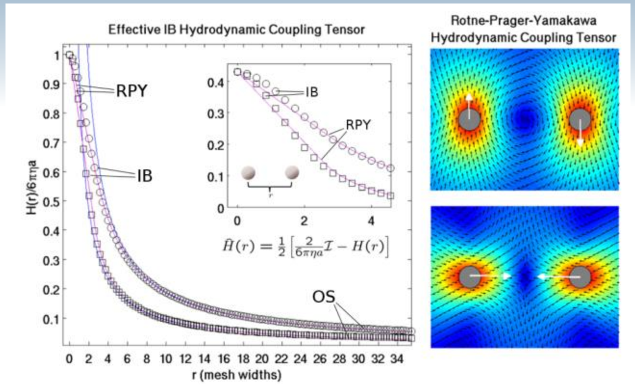
$$
\begin{aligned}
\frac{d\mathbf{X}}{dt} &= \mathbf{v} \\
m\frac{d\mathbf{v}}{dt} &= -\Upsilon\left(\mathbf{v} - \Gamma\mathbf{u}\right) - \nabla_{\mathbf{X}}\Phi[\mathbf{X}] + \zeta + \mathbf{F}_{\text{thm}}
\end{aligned}
$$

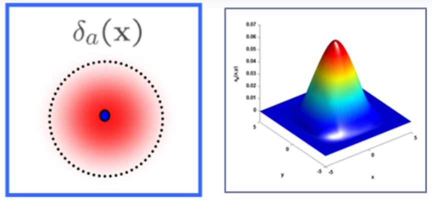
lateral momentum transfer : correlations

Coupling Operators, Immersed Boundary Methods









Peskin delta-function

SELM Coupling:

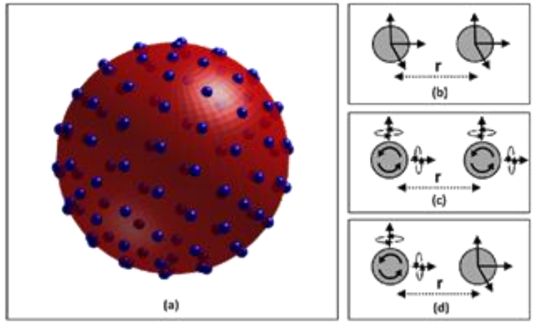
adjoint condition

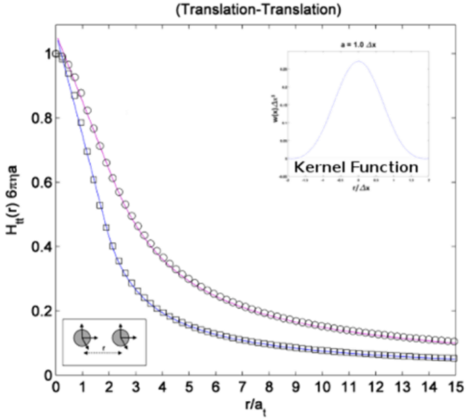
$$\langle \Gamma \mathbf{v}, \mathbf{F} \rangle = \sum_i [\Gamma \mathbf{v}]_i \cdot [\mathbf{F}]_i = \int_\Omega \mathbf{v}(\mathbf{x}) \cdot (\Lambda \mathbf{F})(\mathbf{x}) d\mathbf{x} = \langle \mathbf{v}, \Lambda \mathbf{F} \rangle$$

IB-Kernel coupling:

$$\Gamma \mathbf{u} = \int_\Omega \eta(\mathbf{y} - \mathbf{X}(t))\, \mathbf{u}(\mathbf{y}, t) d\mathbf{y}$$

$$\Lambda \mathbf{F} = \eta(\mathbf{x} - \mathbf{X}(t))\, \mathbf{F}.$$





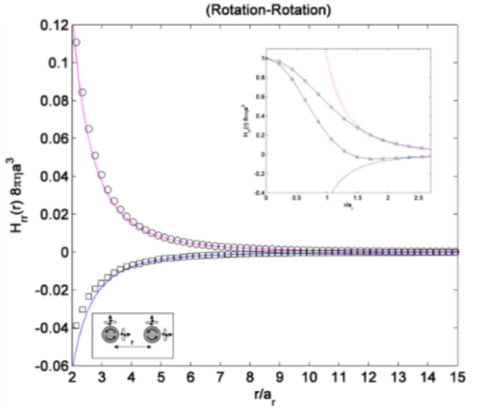
Generalized Coupling (Faxen)

$$\Gamma_0 \mathbf{u} = \sum_{\mathbf{m}} \left\langle\, \eta_0(\mathbf{y}_{\mathbf{m}} - (\mathbf{X}_{\mathrm{cm}} + \mathbf{z}))\, \mathbf{u}_{\mathbf{m}} \,\right\rangle_{\tilde{\mathcal{S}},|\mathbf{z}|=R} \Delta x_{\mathbf{m}}^3$$

$$\Gamma_1 \mathbf{u} = \frac{3}{2R^2} \sum_{\mathbf{m}} \left\langle\, \eta_1(\mathbf{y}_{\mathbf{m}} - (\mathbf{X}_{\mathrm{cm}} + \mathbf{z}))\, (\mathbf{z} \times \mathbf{u}_{\mathbf{m}}) \,\right\rangle_{\tilde{\mathcal{S}},|\mathbf{z}|=R} \Delta x_{\mathbf{m}}^3.$$

$$\Lambda_0(\mathbf{x}_{\mathbf{m}}) = \left(\left\langle\, \eta_0(\mathbf{x}_{\mathbf{m}} - (\mathbf{X}_{\mathrm{cm}} + \mathbf{z})) \,\right\rangle_{\tilde{\mathcal{S}},|\mathbf{z}|=R} \right) \mathbf{F}$$

$$\Lambda_1(\mathbf{x}_{\mathbf{m}}) = -\frac{3}{2R^2} \left(\left\langle\, \mathbf{z}\eta_1(\mathbf{x}_{\mathbf{m}} - (\mathbf{X}_{\mathrm{cm}} + \mathbf{z})) \,\right\rangle_{\tilde{\mathcal{S}},|\mathbf{z}|=R} \right) \times \mathbf{T}.$$



Summary of Regimes for SELMs

Stochastic Eulerian Lagrangian Methods (SELMs)

Fluid dynamics:

$$\begin{aligned} \rho\frac{\partial \mathbf{u}}{\partial t} &= \mu\Delta\mathbf{u} - \nabla p + \Lambda[\Upsilon(\mathbf{v} - \Gamma\mathbf{u})] + \mathbf{f}_{\mathrm{thm}} \\ \nabla\cdot\mathbf{u} &= 0 \end{aligned}$$

Structure dynamics:

$$\begin{aligned} \frac{d\mathbf{X}}{dt} &= \mathbf{v} \\ m\frac{d\mathbf{v}}{dt} &= -\Upsilon(\mathbf{v} - \Gamma\mathbf{u}) - \nabla_{\mathbf{X}}\Phi[\mathbf{X}] + \zeta + \mathbf{F}_{\mathrm{thm}} \end{aligned}$$

Thermal Fluctuations

$$\begin{aligned} \langle \mathbf{f}_{\mathrm{thm}}(s)\mathbf{f}_{\mathrm{thm}}^T(t)\rangle &= -(2k_BT)\left(\mu\Delta - \Lambda\Upsilon\Gamma\right)\delta(t-s) \\ \langle \mathbf{F}_{\mathrm{thm}}(s)\mathbf{F}_{\mathrm{thm}}^T(t)\rangle &= (2k_BT)\,\Upsilon\,\delta(t-s) \\ \langle \mathbf{f}_{\mathrm{thm}}(s)\mathbf{F}_{\mathrm{thm}}^T(t)\rangle &= -(2k_BT)\,\Lambda\Upsilon\,\delta(t-s). \end{aligned}$$

Microstructure density matched with fluid

$m \ll \rho\ell^3$

Fluid-structure dynamics:

$$\begin{aligned} \frac{d\mathbf{p}}{dt} &= \rho^{-1}\mathcal{L}\mathbf{p} + \Lambda[-\nabla_{\mathbf{X}}\Phi(\mathbf{X})] + \boxed{(\nabla_{\mathbf{X}}\cdot\Lambda)\,k_BT} + \lambda + \mathbf{g}_{\mathrm{thm}} \\ \frac{d\mathbf{X}}{dt} &= \rho^{-1}\Gamma\mathbf{p} + \Upsilon^{-1}[-\nabla_{\mathbf{X}}\Phi(\mathbf{X})] + \zeta + \mathbf{G}_{\mathrm{thm}}. \\ \nabla_{\mathbf{X}}\cdot\Lambda &= \mathrm{Tr}[\nabla_{\mathbf{X}}\Lambda] \end{aligned}$$

Phase space compressibility (p,X).

Thermal Fluctuations:

$$\begin{aligned} \langle \mathbf{g}_{\mathrm{thm}}(s)\mathbf{g}_{\mathrm{thm}}^T(t)\rangle &= -(2k_BT)\,\mathcal{L}\,\delta(t-s) \\ \langle \mathbf{G}_{\mathrm{thm}}(s)\mathbf{G}_{\mathrm{thm}}^T(t)\rangle &= (2k_BT)\,\Upsilon^{-1}\,\delta(t-s) \\ \langle \mathbf{g}_{\mathrm{thm}}(s)\mathbf{G}_{\mathrm{thm}}^T(t)\rangle &= 0. \end{aligned}$$

- Structure momentum no longer tracked.
- Removes a source of stiffness.
- Non-conjugate Hamiltonian formulation yields metric-factor in phase-space.

Microstructure-fluid no-slip coupling (S-Immersed-Boundary)

$\Upsilon \to \infty$

Fluid-Structure Equations:

$$\begin{aligned} \frac{d\mathbf{p}}{dt} &= \rho^{-1}\mathcal{L}\mathbf{p} + \Lambda[-\nabla_{\mathbf{X}}\Phi(\mathbf{X})] + \boxed{(\nabla_{\mathbf{X}}\cdot\Lambda)\,k_BT} + \lambda + \mathbf{g}_{\mathrm{thm}} \\ \frac{d\mathbf{X}}{dt} &= \rho^{-1}\Gamma\mathbf{p} \end{aligned}$$

Thermal Fluctuations:

$$\langle \mathbf{g}_{\mathrm{thm}}(s)\mathbf{g}_{\mathrm{thm}}^T(t)\rangle = -(2k_BT)\,\mathcal{L}\,\delta(t-s).$$

- Structure dynamics no-longer inertial.
- Removes additional sources of stiffness.
- Regime of the Stochastic Immersed Boundary Method.
- Phase-space metric reflected in the drift term.

Microstructure-fluid stress balance

$\mu \to \infty$

Fluid-Structure Equations:

$$\begin{aligned} \frac{d\mathbf{X}}{dt} &= H_{\mathrm{SELM}}[-\nabla_{\mathbf{X}}\Phi(\mathbf{X})] + \boxed{(\nabla_{\mathbf{X}}\cdot H_{\mathrm{SELM}})k_BT} + \mathbf{h}_{\mathrm{thm}} \\ H_{\mathrm{SELM}} &= \Gamma(-\wp\mathcal{L})^{-1}\Lambda \end{aligned}$$

Thermal Fluctuations:

$$\langle \mathbf{h}_{\mathrm{thm}}(s)\mathbf{h}_{\mathrm{thm}}^T(t)\rangle = (2k_BT)\,H_{\mathrm{SELM}}\,\delta(t-s).$$

- Fluid momentum no longer tracked.
- Balance of hydrodynamic stresses with elastic stresses.
- Removes additional sources of stiffness.
- Regime of the Stokesian-Brownian Dynamics (Brady 1980, McCammond 1980's).
- Phase-space metric reflected in the drift term.

Atzberger & Tabak 2015

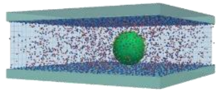
SELM-LAMMPS Integration

Molecular Dynamics and Coarse-Grained Modeling Approaches

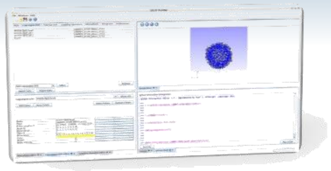
Mango-Selm Software: Overview

MANGO-SELM
Fluctuating Hydrodynamics Package

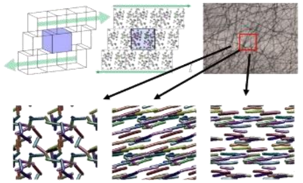
Colloid electrophoresis in a channel.



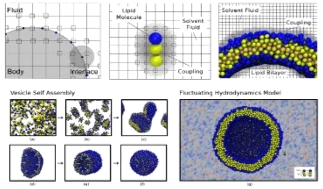
Graphical modeling interface.



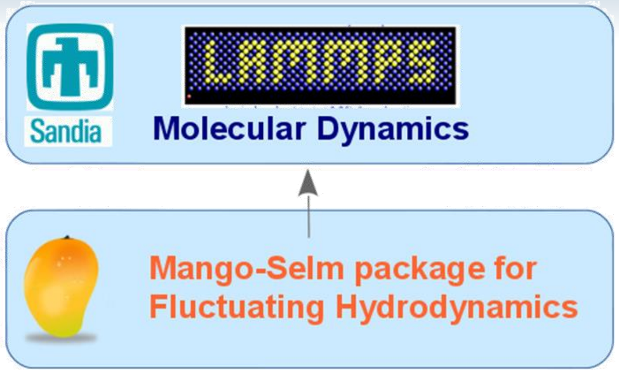
Polymeric fluid subject to shear.



Implicit-Solvent Lipid Model: SELM thermostat: self-assembly.



Molecular dynamics Integration

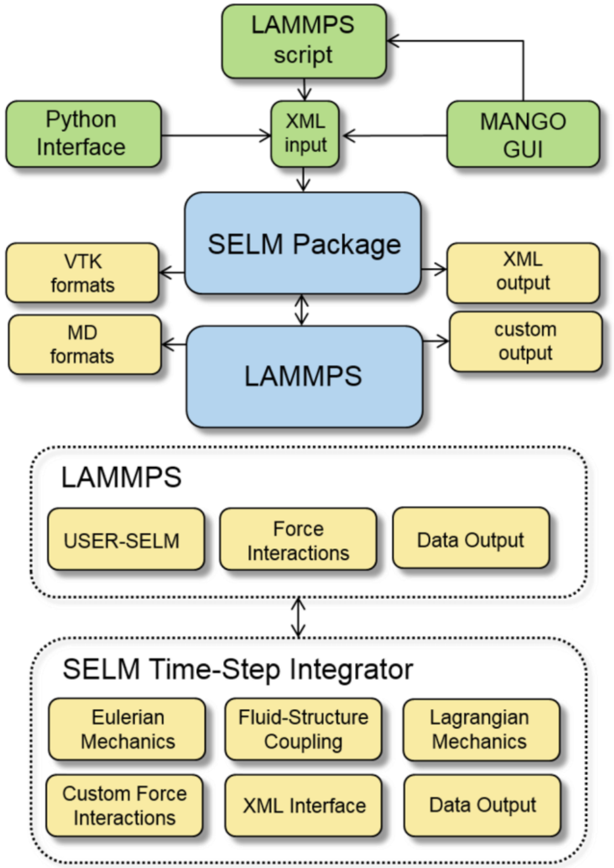


Download: http://mango-selm.org/

Python and Jupyter notebook interfaces

jupyter python

Modular Design and Extendible



Atzberger 2016

Selm - Simulation Package:

SELM fluctuating hydrodynamics for fluid-structure interactions subject to thermal fluctuations.

- **Python, Jupyter notebooks**, and other scripting for model building and simulation.
- **Stochastic numerical time-step integrators** for inertial and quasi-steady physical regimes in (C/C++).

Molecular dynamics integration with LAMMPS

- **Modeling, interactions, many potentials**, **statistical analysis**.
- **Thermostats and many ensembles possible** such as Lees-Edwards for shear simulations.

Standardized formats

- **XML** for parametrization and data output.
- **VTK** output for continuum fields and microstructures (visualization / analysis).

Mango – GUI for Model Building:

Graphical User Interface (GUI) for setting up model geometry and simulation parameters.

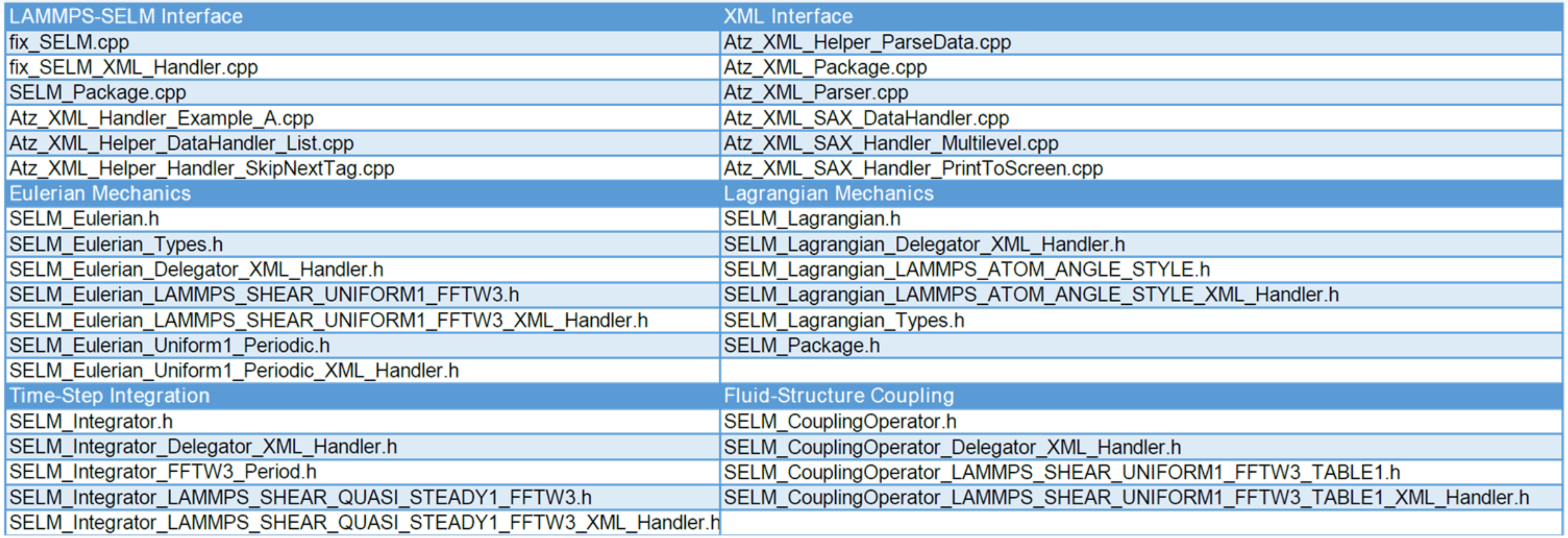
- **Generates scripts and data files** for SELM fluctuating hydrodynamics simulations.

Extendible object-oriented architectures for inclusion of new numerical methods.

Mango-Selm Implementation

Selm Codes: Design and Implementation

SELM – Source Codes:



LAMMPS-SELM Interface	XML Interface
fix_SELM.cpp	Atz_XML_Helper_ParseData.cpp
fix_SELM_XML_Handler.cpp	Atz_XML_Package.cpp
SELM_Package.cpp	Atz_XML_Parser.cpp
Atz_XML_Handler_Example_A.cpp	Atz_XML_SAX_DataHandler.cpp
Atz_XML_Helper_DataHandler_List.cpp	Atz_XML_SAX_Handler_Multilevel.cpp
Atz_XML_Helper_Handler_SkipNextTag.cpp	Atz_XML_SAX_Handler_PrintToScreen.cpp
Eulerian Mechanics	Lagrangian Mechanics
SELM_Eulerian.h	SELM_Lagrangian.h
SELM_Eulerian_Types.h	SELM_Lagrangian_Delegator_XML_Handler.h
SELM_Eulerian_Delegator_XML_Handler.h	SELM_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE.h
SELM_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3.h	SELM_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE_XML_Handler.h
SELM_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3_XML_Handler.h	SELM_Lagrangian_Types.h
SELM_Eulerian_Uniform1_Periodic.h	SELM_Package.h
SELM_Eulerian_Uniform1_Periodic_XML_Handler.h	
Time-Step Integration	Fluid-Structure Coupling
SELM_Integrator.h	SELM_CouplingOperator.h
SELM_Integrator_Delegator_XML_Handler.h	SELM_CouplingOperator_Delegator_XML_Handler.h
SELM_Integrator_FFTW3_Period.h	SELM_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1.h
SELM_Integrator_LAMMPS_SHEAR_QUASI_STEADY1_FFTW3.h	SELM_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1_XML_Handler.h
SELM_Integrator_LAMMPS_SHEAR_QUASI_STEADY1_FFTW3_XML_Handler.h	

Features:

Object-oriented C++ classes mirroring parts of SELM with **XML parameter files.**

Delegator design pattern is used to control the work-flow.

Four main SELM classes correspond to:

- Eulerian Mechanics
- Lagrangian Mechanics
- Fluid-Structure Coupling (Eulerian-Lagrangian communication)
- Time-Step Integration

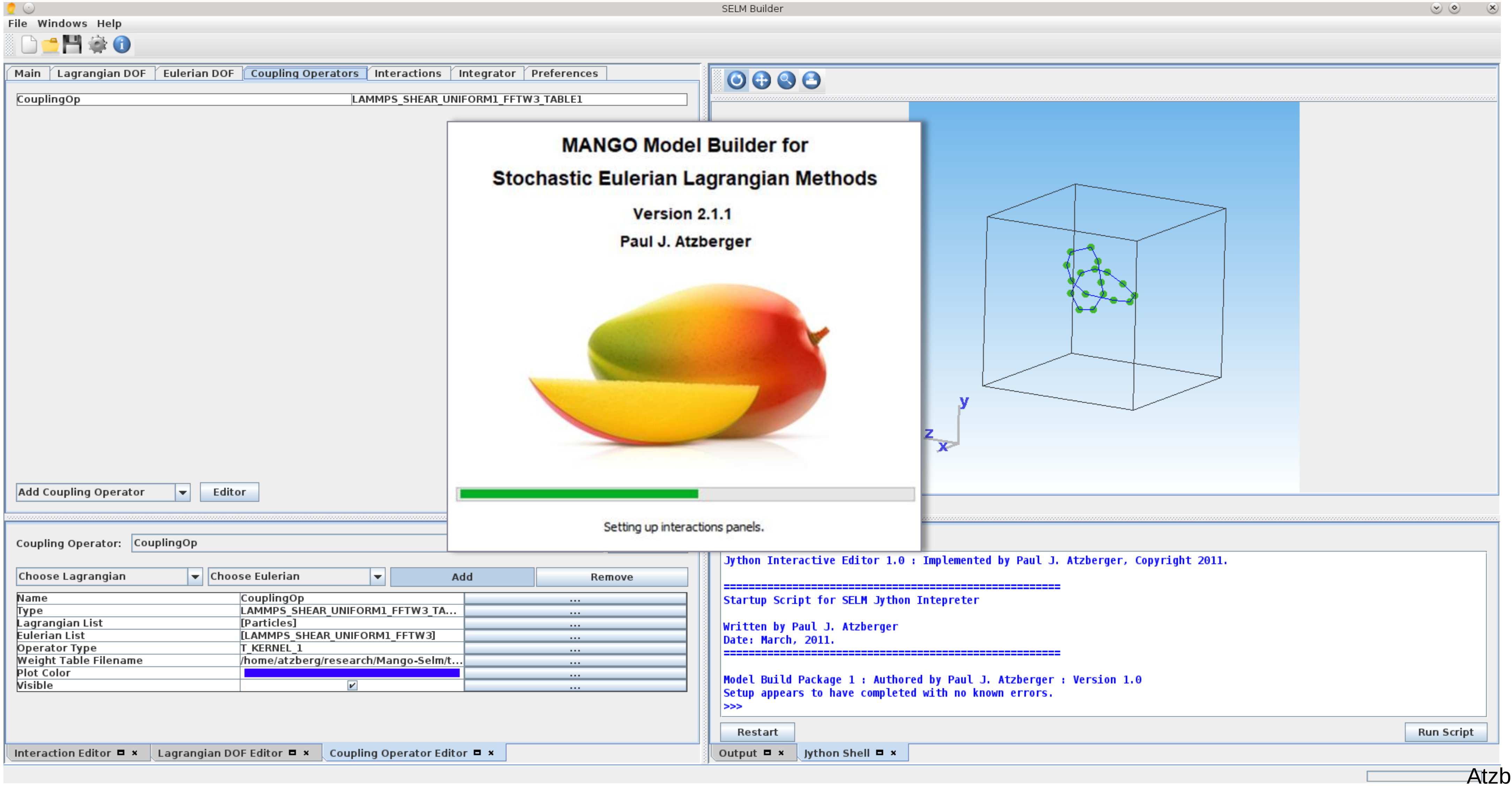
Additional classes for XML parsing, data generation.

Designed to be easily extended for new types of SELM formulations and integrators.

Atzberger 2016

Mango GUI Model Builder

Mango - Modeling Software:



CouplingOp LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1

MANGO Model Builder for

Stochastic Eulerian Lagrangian Methods

Version 2.1.1

Paul J. Atzberger

Setting up interactions panels.

Coupling Operator: CouplingOp

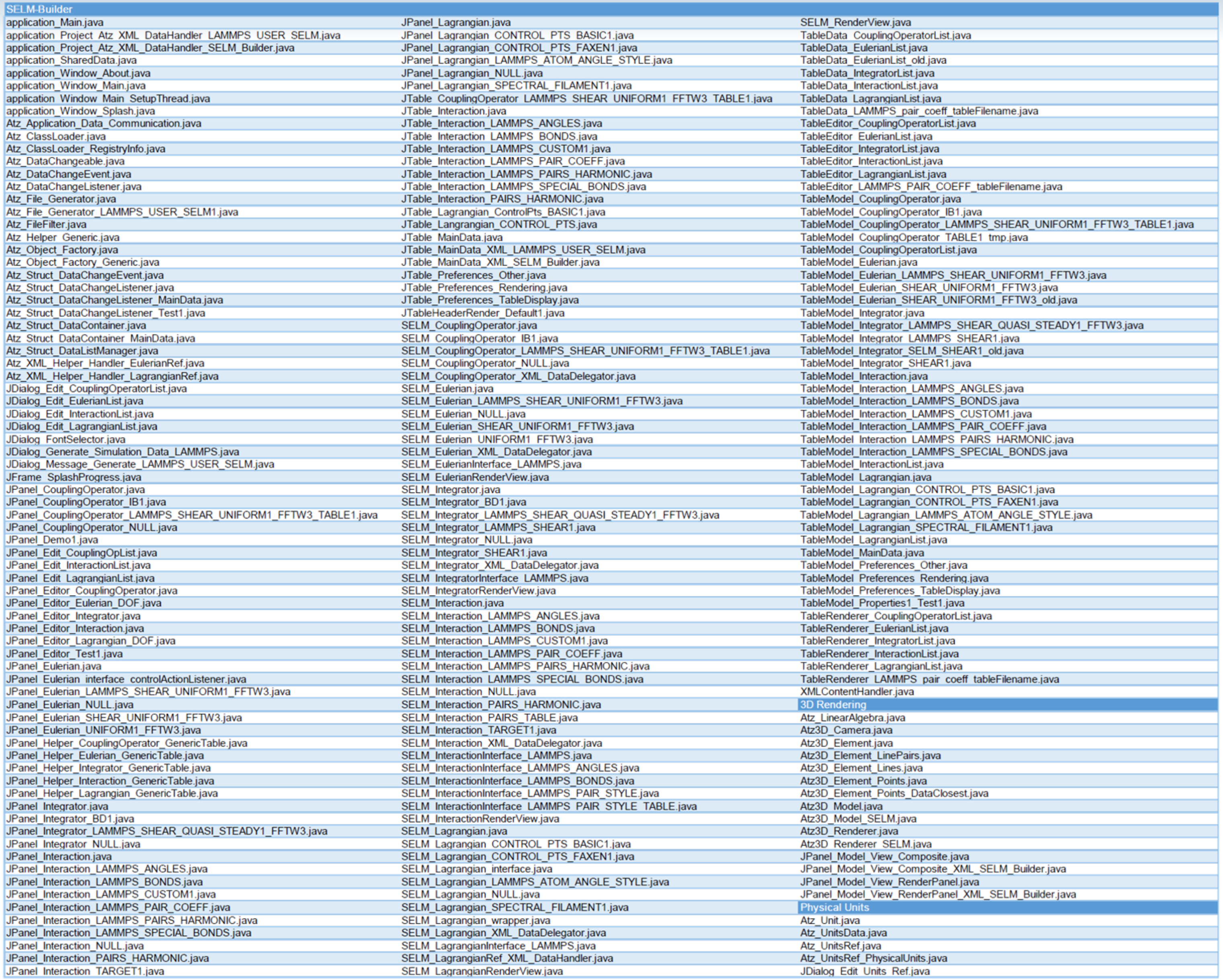
Name	CouplingOp	...
Type	LAMMPS_SHEAR_UNIFORM1_FFTW3_TA...	...
Lagrangian List	[Particles]	...
Eulerian List	[LAMMPS_SHEAR_UNIFORM1_FFTW3]	...
Operator Type	T_KERNEL_1	...
Weight Table Filename	/home/atzberg/research/Mango-Selm/t...	...
Plot Color	🟦	...
Visible	☑	...

```
Jython Interactive Editor 1.0 : Implemented by Paul J. Atzberger, Copyright 2011.

======================================================
Startup Script for SELM Jython Intepreter

Written by Paul J. Atzberger
Date: March, 2011.
======================================================

Model Build Package 1 : Authored by Paul J. Atzberger : Version 1.0
Setup appears to have completed with no known errors.
>>>
```

Atzberger 2016

Mango Modeling Software

Mango - Codes



SELM-Builder

- application_Main.java
- application_Project_Atz_XML_DataHandler_LAMMPS_USER_SELM.java
- application_Project_Atz_XML_DataHandler_SELM_Builder.java
- application_SharedData.java
- application_Window_About.java
- application_Window_Main.java
- application_Window_Main_SetupThread.java
- application_Window_Splash.java
- Atz_Application_Data_Communication.java
- Atz_ClassLoader.java
- Atz_ClassLoader_RegistryInfo.java
- Atz_DataChangeable.java
- Atz_DataChangeEvent.java
- Atz_DataChangeListener.java
- Atz_File_Generator.java
- Atz_File_Generator_LAMMPS_USER_SELM1.java
- Atz_FileFilter.java
- Atz_Helper_Generic.java
- Atz_Object_Factory.java
- Atz_Object_Factory_Generic.java
- Atz_Struct_DataChangeEvent.java
- Atz_Struct_DataChangeListener.java
- Atz_Struct_DataChangeListener_MainData.java
- Atz_Struct_DataChangeListener_Test1.java
- Atz_Struct_DataContainer.java
- Atz_Struct_DataContainer_MainData.java
- Atz_Struct_DataListManager.java
- Atz_XML_Helper_Handler_EulerianRef.java
- Atz_XML_Helper_Handler_LagrangianRef.java
- JDialog_Edit_CouplingOperatorList.java
- JDialog_Edit_EulerianList.java
- JDialog_Edit_InteractionList.java
- JDialog_Edit_LagrangianList.java
- JDialog_FontSelector.java
- JDialog_Generate_Simulation_Data_LAMMPS.java
- JDialog_Message_Generate_LAMMPS_USER_SELM.java
- JFrame_SplashProgress.java
- JPanel_CouplingOperator.java
- JPanel_CouplingOperator_IB1.java
- JPanel_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1.java
- JPanel_CouplingOperator_NULL.java
- JPanel_Demo1.java
- JPanel_Edit_CouplingOpList.java
- JPanel_Edit_InteractionList.java
- JPanel_Edit_LagrangianList.java
- JPanel_Editor_CouplingOperator.java
- JPanel_Editor_Eulerian_DOF.java
- JPanel_Editor_Integrator.java
- JPanel_Editor_Interaction.java
- JPanel_Editor_Lagrangian_DOF.java
- JPanel_Editor_Test1.java
- JPanel_Eulerian.java
- JPanel_Eulerian_interface_controlActionListener.java
- JPanel_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3.java
- JPanel_Eulerian_NULL.java
- JPanel_Eulerian_SHEAR_UNIFORM1_FFTW3.java
- JPanel_Eulerian_UNIFORM1_FFTW3.java
- JPanel_Helper_CouplingOperator_GenericTable.java
- JPanel_Helper_Eulerian_GenericTable.java
- JPanel_Helper_Integrator_GenericTable.java
- JPanel_Helper_Interaction_GenericTable.java
- JPanel_Helper_Lagrangian_GenericTable.java
- JPanel_Integrator.java
- JPanel_Integrator_BD1.java
- JPanel_Integrator_LAMMPS_SHEAR_QUASI_STEADY1_FFTW3.java
- JPanel_Integrator_NULL.java
- JPanel_Interaction.java
- JPanel_Interaction_LAMMPS_ANGLES.java
- JPanel_Interaction_LAMMPS_BONDS.java
- JPanel_Interaction_LAMMPS_CUSTOM1.java
- JPanel_Interaction_LAMMPS_PAIR_COEFF.java
- JPanel_Interaction_LAMMPS_PAIRS_HARMONIC.java
- JPanel_Interaction_LAMMPS_SPECIAL_BONDS.java
- JPanel_Interaction_NULL.java
- JPanel_Interaction_PAIRS_HARMONIC.java
- JPanel_Interaction_TARGET1.java
- JPanel_Lagrangian.java
- JPanel_Lagrangian_CONTROL_PTS_BASIC1.java
- JPanel_Lagrangian_CONTROL_PTS_FAXEN1.java
- JPanel_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE.java
- JPanel_Lagrangian_NULL.java
- JPanel_Lagrangian_SPECTRAL_FILAMENT1.java
- JTable_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1.java
- JTable_Interaction.java
- JTable_Interaction_LAMMPS_ANGLES.java
- JTable_Interaction_LAMMPS_BONDS.java
- JTable_Interaction_LAMMPS_CUSTOM1.java
- JTable_Interaction_LAMMPS_PAIR_COEFF.java
- JTable_Interaction_LAMMPS_PAIRS_HARMONIC.java
- JTable_Interaction_LAMMPS_SPECIAL_BONDS.java
- JTable_Interaction_PAIRS_HARMONIC.java
- JTable_Lagrangian_ControlPts_BASIC1.java
- JTable_Langrangian_CONTROL_PTS.java
- JTable_MainData.java
- JTable_MainData_XML_LAMMPS_USER_SELM.java
- JTable_MainData_XML_SELM_Builder.java
- JTable_Preferences_Other.java
- JTable_Preferences_Rendering.java
- JTable_Preferences_TableDisplay.java
- JTableHeaderRender_Default1.java
- SELM_CouplingOperator.java
- SELM_CouplingOperator_IB1.java
- SELM_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1.java
- SELM_CouplingOperator_NULL.java
- SELM_CouplingOperator_XML_DataDelegator.java
- SELM_Eulerian.java
- SELM_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3.java
- SELM_Eulerian_NULL.java
- SELM_Eulerian_SHEAR_UNIFORM1_FFTW3.java
- SELM_Eulerian_UNIFORM1_FFTW3.java
- SELM_Eulerian_XML_DataDelegator.java
- SELM_EulerianInterface_LAMMPS.java
- SELM_EulerianRenderView.java
- SELM_Integrator.java
- SELM_Integrator_BD1.java
- SELM_Integrator_LAMMPS_SHEAR_QUASI_STEADY1_FFTW3.java
- SELM_Integrator_LAMMPS_SHEAR1.java
- SELM_Integrator_NULL.java
- SELM_Integrator_SHEAR1.java
- SELM_Integrator_XML_DataDelegator.java
- SELM_IntegratorInterface_LAMMPS.java
- SELM_IntegratorRenderView.java
- SELM_Interaction.java
- SELM_Interaction_LAMMPS_ANGLES.java
- SELM_Interaction_LAMMPS_BONDS.java
- SELM_Interaction_LAMMPS_CUSTOM1.java
- SELM_Interaction_LAMMPS_PAIR_COEFF.java
- SELM_Interaction_LAMMPS_PAIRS_HARMONIC.java
- SELM_Interaction_LAMMPS_SPECIAL_BONDS.java
- SELM_Interaction_NULL.java
- SELM_Interaction_PAIRS_HARMONIC.java
- SELM_Interaction_PAIRS_TABLE.java
- SELM_Interaction_TARGET1.java
- SELM_Interaction_XML_DataDelegator.java
- SELM_InteractionInterface_LAMMPS.java
- SELM_InteractionInterface_LAMMPS_ANGLES.java
- SELM_InteractionInterface_LAMMPS_BONDS.java
- SELM_InteractionInterface_LAMMPS_PAIR_STYLE.java
- SELM_InteractionInterface_LAMMPS_PAIR_STYLE_TABLE.java
- SELM_InteractionRenderView.java
- SELM_Lagrangian.java
- SELM_Lagrangian_CONTROL_PTS_BASIC1.java
- SELM_Lagrangian_CONTROL_PTS_FAXEN1.java
- SELM_Lagrangian_interface.java
- SELM_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE.java
- SELM_Lagrangian_NULL.java
- SELM_Lagrangian_SPECTRAL_FILAMENT1.java
- SELM_Lagrangian_wrapper.java
- SELM_Lagrangian_XML_DataDelegator.java
- SELM_LagrangianInterface_LAMMPS.java
- SELM_LagrangianRef_XML_DataHandler.java
- SELM_LagrangianRenderView.java
- SELM_RenderView.java
- TableData_CouplingOperatorList.java
- TableData_EulerianList.java
- TableData_EulerianList_old.java
- TableData_IntegratorList.java
- TableData_InteractionList.java
- TableData_LagrangianList.java
- TableData_LAMMPS_pair_coeff_tableFilename.java
- TableEditor_CouplingOperatorList.java
- TableEditor_EulerianList.java
- TableEditor_IntegratorList.java
- TableEditor_InteractionList.java
- TableEditor_LagrangianList.java
- TableEditor_LAMMPS_PAIR_COEFF_tableFilename.java
- TableModel_CouplingOperator.java
- TableModel_CouplingOperator_IB1.java
- TableModel_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1.java
- TableModel_CouplingOperator_TABLE1_tmp.java
- TableModel_CouplingOperatorList.java
- TableModel_Eulerian.java
- TableModel_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3.java
- TableModel_Eulerian_SHEAR_UNIFORM1_FFTW3.java
- TableModel_Eulerian_SHEAR_UNIFORM1_FFTW3_old.java
- TableModel_Integrator.java
- TableModel_Integrator_LAMMPS_SHEAR_QUASI_STEADY1_FFTW3.java
- TableModel_Integrator_LAMMPS_SHEAR1.java
- TableModel_Integrator_SELM_SHEAR1_old.java
- TableModel_Integrator_SHEAR1.java
- TableModel_Interaction.java
- TableModel_Interaction_LAMMPS_ANGLES.java
- TableModel_Interaction_LAMMPS_BONDS.java
- TableModel_Interaction_LAMMPS_CUSTOM1.java
- TableModel_Interaction_LAMMPS_PAIR_COEFF.java
- TableModel_Interaction_LAMMPS_PAIRS_HARMONIC.java
- TableModel_Interaction_LAMMPS_SPECIAL_BONDS.java
- TableModel_InteractionList.java
- TableModel_Lagrangian.java
- TableModel_Lagrangian_CONTROL_PTS_BASIC1.java
- TableModel_Lagrangian_CONTROL_PTS_FAXEN1.java
- TableModel_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE.java
- TableModel_Lagrangian_SPECTRAL_FILAMENT1.java
- TableModel_LagrangianList.java
- TableModel_MainData.java
- TableModel_Preferences_Other.java
- TableModel_Preferences_Rendering.java
- TableModel_Preferences_TableDisplay.java
- TableModel_Properties1_Test1.java
- TableRenderer_CouplingOperatorList.java
- TableRenderer_EulerianList.java
- TableRenderer_IntegratorList.java
- TableRenderer_InteractionList.java
- TableRenderer_LagrangianList.java
- TableRenderer_LAMMPS_pair_coeff_tableFilename.java
- XMLContentHandler.java

3D Rendering

- Atz_LinearAlgebra.java
- Atz3D_Camera.java
- Atz3D_Element.java
- Atz3D_Element_LinePairs.java
- Atz3D_Element_Lines.java
- Atz3D_Element_Points.java
- Atz3D_Element_Points_DataClosest.java
- Atz3D_Model.java
- Atz3D_Model_SELM.java
- Atz3D_Renderer.java
- Atz3D_Renderer_SELM.java
- JPanel_Model_View_Composite.java
- JPanel_Model_View_Composite_XML_SELM_Builder.java
- JPanel_Model_View_RenderPanel.java
- JPanel_Model_View_RenderPanel_XML_SELM_Builder.java

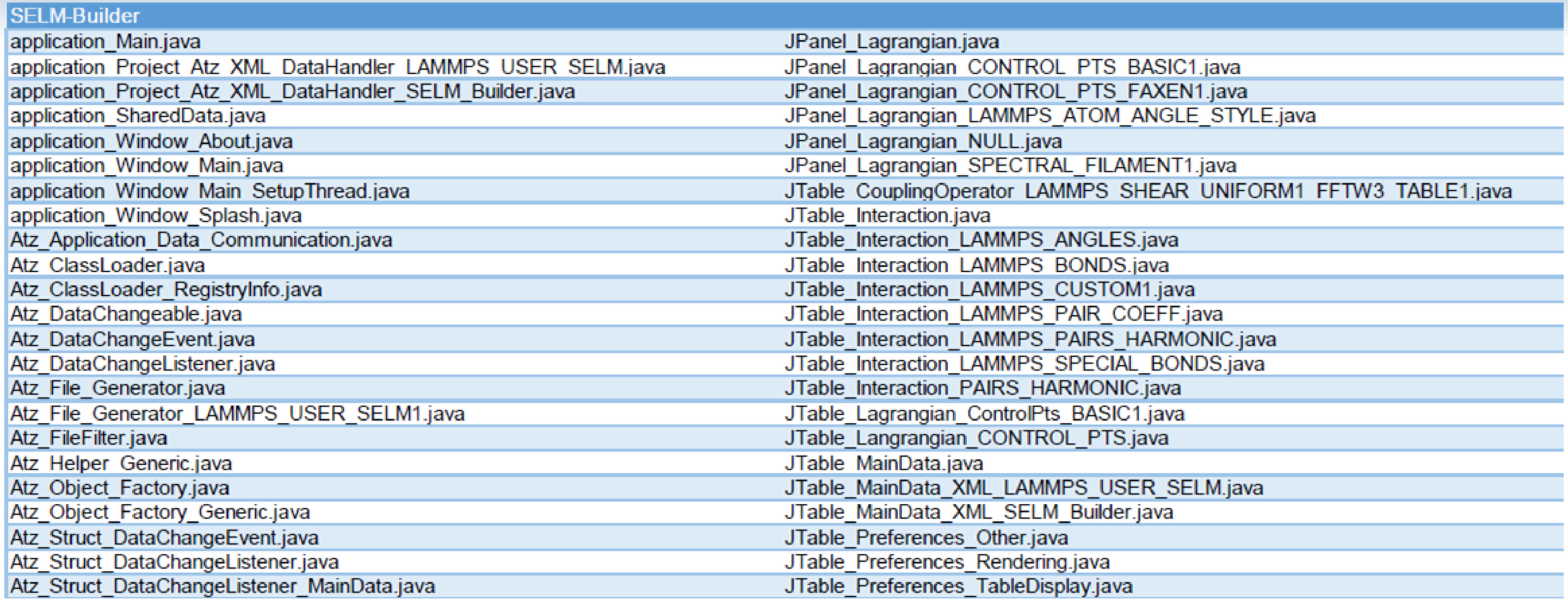
Physical Units

- Atz_Unit.java
- Atz_UnitsData.java
- Atz_UnitsRef.java
- Atz_UnitsRef_PhysicalUnits.java
- JDialog_Edit_Units_Ref.java

Atzberger 2016

Mango Modeling Software

MANGO - Codes



SELM-Builder	
application_Main.java	JPanel_Lagrangian.java
application_Project_Atz_XML_DataHandler_LAMMPS_USER_SELM.java	JPanel_Lagrangian_CONTROL_PTS_BASIC1.java
application_Project_Atz_XML_DataHandler_SELM_Builder.java	JPanel_Lagrangian_CONTROL_PTS_FAXEN1.java
application_SharedData.java	JPanel_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE.java
application_Window_About.java	JPanel_Lagrangian_NULL.java
application_Window_Main.java	JPanel_Lagrangian_SPECTRAL_FILAMENT1.java
application_Window_Main_SetupThread.java	JTable_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1.java
application_Window_Splash.java	JTable_Interaction.java
Atz_Application_Data_Communication.java	JTable_Interaction_LAMMPS_ANGLES.java
Atz_ClassLoader.java	JTable_Interaction_LAMMPS_BONDS.java
Atz_ClassLoader_RegistryInfo.java	JTable_Interaction_LAMMPS_CUSTOM1.java
Atz_DataChangeable.java	JTable_Interaction_LAMMPS_PAIR_COEFF.java
Atz_DataChangeEvent.java	JTable_Interaction_LAMMPS_PAIRS_HARMONIC.java
Atz_DataChangeListener.java	JTable_Interaction_LAMMPS_SPECIAL_BONDS.java
Atz_File_Generator.java	JTable_Interaction_PAIRS_HARMONIC.java
Atz_File_Generator_LAMMPS_USER_SELM1.java	JTable_Lagrangian_ControlPts_BASIC1.java
Atz_FileFilter.java	JTable_Langrangian_CONTROL_PTS.java
Atz_Helper_Generic.java	JTable_MainData.java
Atz_Object_Factory.java	JTable_MainData_XML_LAMMPS_USER_SELM.java
Atz_Object_Factory_Generic.java	JTable_MainData_XML_SELM_Builder.java
Atz_Struct_DataChangeEvent.java	JTable_Preferences_Other.java
Atz_Struct_DataChangeListener.java	JTable_Preferences_Rendering.java
Atz_Struct_DataChangeListener_MainData.java	JTable_Preferences_TableDisplay.java

Features:

Object-oriented classes in Java mirroring parts of SELM.

Dynamic object loaders for delegator design pattern for control flow (extension after compiled byte-codes).

Four main SELM classes correspond to:

- Eulerian Mechanics
- Lagrangian Mechanics
- Fluid-Structure Coupling (Eulerian-Lagrangian communication)
- Time-Step Integration

Designed to be easily extended for new types of SELM formulations and integrators.

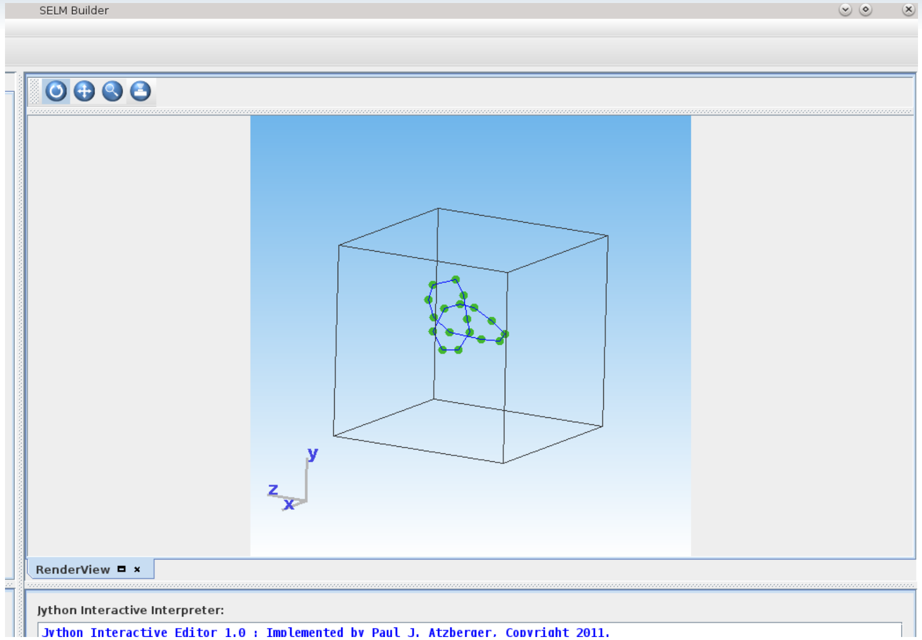
Custom classes and interfaces for rendering models in 3D and interactively editing models.

Atzberger 2016

Mango Modeling Software

Mango – Codes:

3D Rendering
Atz_LinearAlgebra.java
Atz3D_Camera.java
Atz3D_Element.java
Atz3D_Element_LinePairs.java
Atz3D_Element_Lines.java
Atz3D_Element_Points.java
Atz3D_Element_Points_DataClosest.java
Atz3D_Model.java
Atz3D_Model_SELM.java
Atz3D_Renderer.java
Atz3D_Renderer_SELM.java
JPanel_Model_View_Composite.java
JPanel_Model_View_Composite_XML_SELM_Builder.java
JPanel_Model_View_RenderPanel.java
JPanel_Model_View_RenderPanel_XML_SELM_Builder.java
Physical Units
Atz_Unit.java
Atz_UnitsData.java
Atz_UnitsRef.java
Atz_UnitsRef_PhysicalUnits.java
JDialog_Edit_Units_Ref.java



Jython Interactive Interpreter:

Jython Interactive Editor 1.0 : Implemented by Paul J. Atzberger. Copyright 2011.

Features:

Rendering in 3D for interactively editing for model geometry.

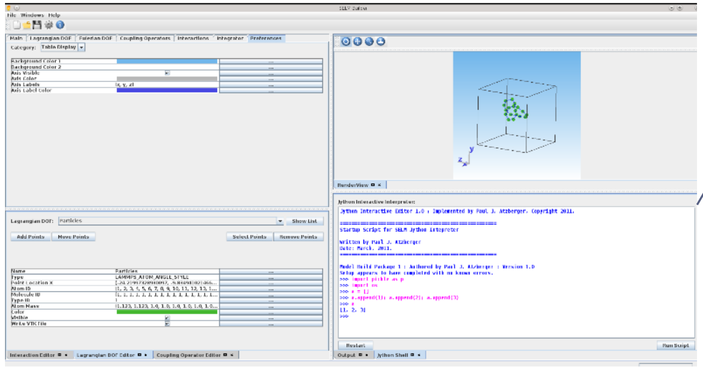
Interactive editor features allow for

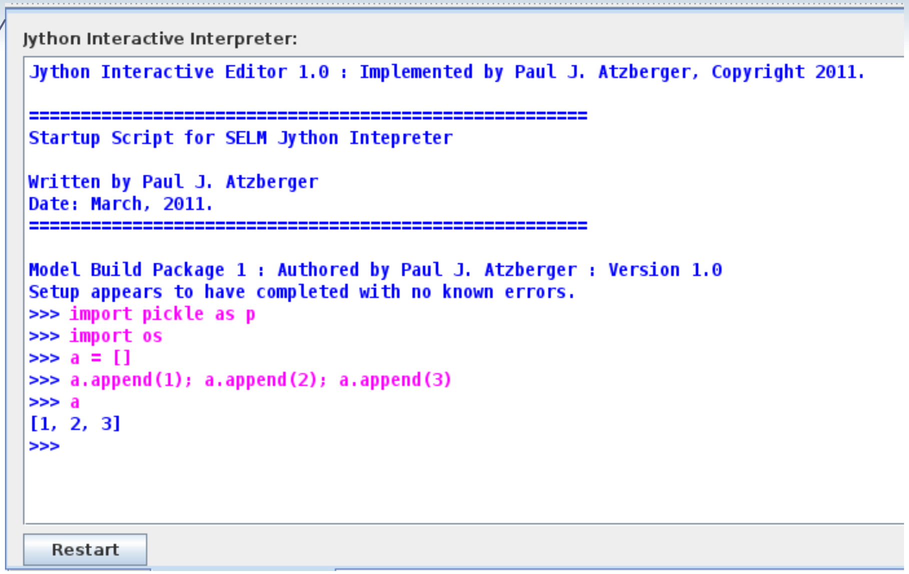
- interactive views of model
- adding / removing control points
- adding / removing bonds between points
- adding custom force interactions

Custom classes implemented for tracking physical units in tables.

Mango GUI Jython Interface

MANGO - Modeling Software:





```
Jython Interactive Editor 1.0 : Implemented by Paul J. Atzberger, Copyright 2011.

=====================================================
Startup Script for SELM Jython Intepreter

Written by Paul J. Atzberger
Date: March, 2011.
=====================================================

Model Build Package 1 : Authored by Paul J. Atzberger : Version 1.0
Setup appears to have completed with no known errors.
>>> import pickle as p
>>> import os
>>> a = []
>>> a.append(1); a.append(2); a.append(3)
>>> a
[1, 2, 3]
>>>
```

Jython Terminal:

Custom classes implement interactive terminal based on Jython.
Wrapper jython classes implemented for MANGO interface and SELM data structures.
Editor features allow for

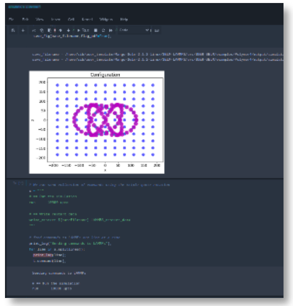
- jython/python scripting to construct models
- custom GUI windows : interactive components in MANGO
- post-processing scripts
- generation of SELM XML files from the constructed MANGO data structures.

Jupyter notebooks and Python interface now also available (directly with Selm-Lammps library).

Python and Jupyter notebook interfaces

jupyter

python



Atzberger 2016

Selm-Lammps Integration

Python and Jupyter Notebook Interfaces

Tips for SELM Package: Installation and Usage

Instructions

Pre-compiled binaries: for (Debian 9+/Ubuntu and Centos 7+, Python 3.6+).

Quick start:
>> pip install selm-lammps
test: >> python -c "from selm_lammps.tests import t1; t1.test()“

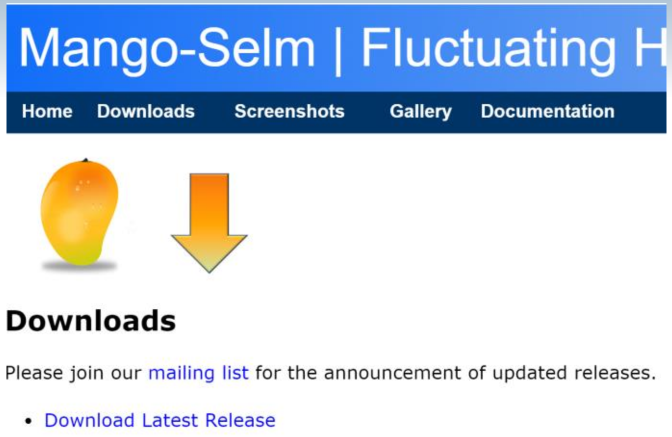
Alternatively, can use *docker run -it ubuntu:20.04 /bin/bash; (use python3).*

Examples, docs, source: see http://mango-selm.org/

Directories:
- src/USER-SELM: main C/C++ interface codes for SELM package
- examples/USER/selm: example models and simulation scripts
- lib/selm: library of SELM integrators and related algorithms

Installing Python components (uses lammps.py, lammps.so):
- Recommended to use *conda* or *virtualenv* with python >=3.6
- Examples for Jupyter notebooks, python scripts, for running simulations.
- See README files for more details.

Jupyter Notebooks and Python-based Simulations:
- Uses python-interfaces to LAMMPs.
- L.command(cmd_str); runs the command in cmd_str.
- Model building using python wrapper
    - sets up particles
    - sets up interactions, many possible types and potential available
    - sets up the simulation parameters.



Mango-Selm | Fluctuating H

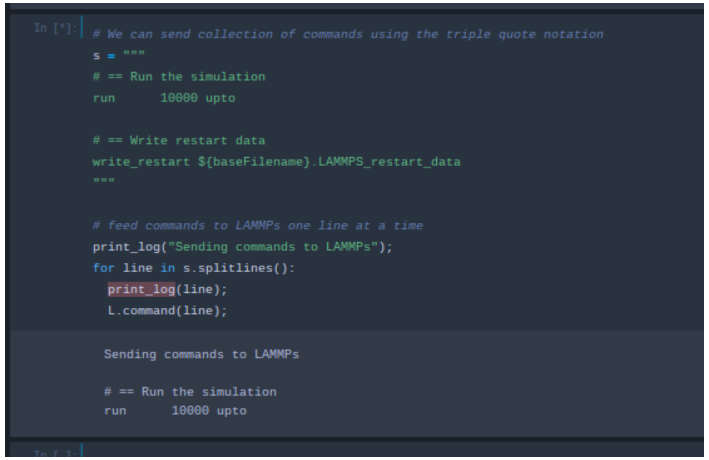
Downloads

Please join our mailing list for the announcement of updated releases.

- Download Latest Release

Download: http://mango-selm.org/

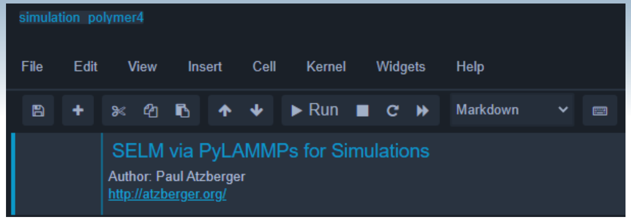
Python and Jupyter notebook interfaces



```
In [*]: # We can send collection of commands using the triple quote notation
        s = """
        # == Run the simulation
        run      10000 upto

        # == Write restart data
        write_restart ${baseFilename}.LAMMPS_restart_data
        """

        # feed commands to LAMMPs one line at a time
        print_log("Sending commands to LAMMPs");
        for line in s.splitlines():
          print_log(line);
          L.command(line);

         Sending commands to LAMMPs

         # == Run the simulation
         run      10000 upto

In [ ]:
```

Jupyter Notebook and Python Interface

Python Jupyter Notebook

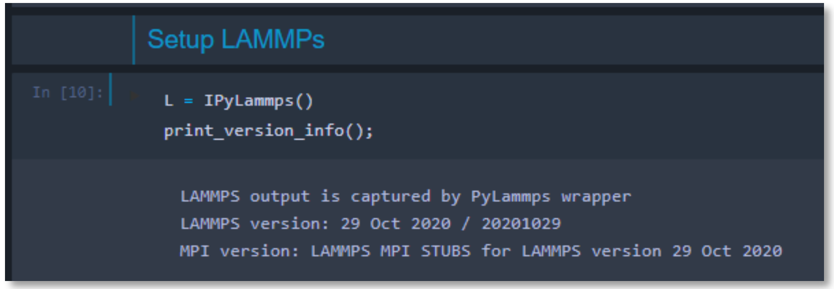


SELM via PyLAMMPs for Simulations

Author: Paul Atzberger

http://atzberger.org/

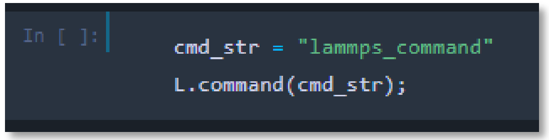
Create Selm-Lammps instance



Setup LAMMPs

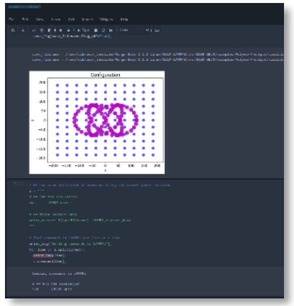
```
In [10]:  L = IPyLammps()
          print_version_info();

   LAMMPS output is captured by PyLammps wrapper
   LAMMPS version: 29 Oct 2020 / 20201029
   MPI version: LAMMPS MPI STUBS for LAMMPS version 29 Oct 2020
```

Issuing commands

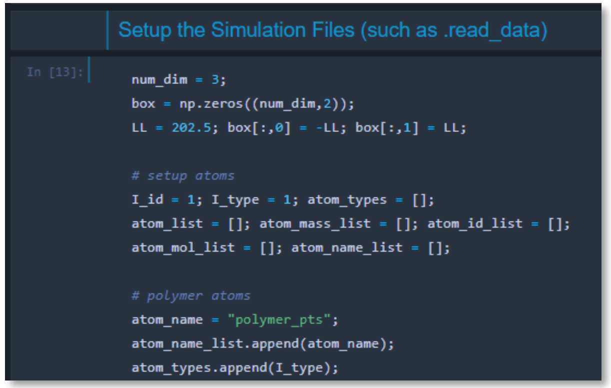


```
In [ ]:    cmd_str = "lammps_command"
           L.command(cmd_str);
```

Python and Jupyter notebook interfaces

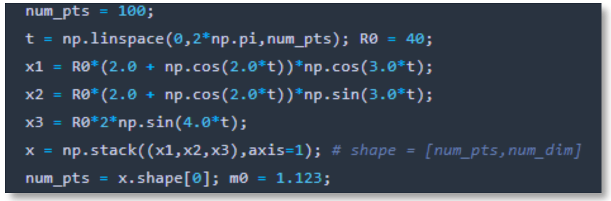


Setup model geometry and interactions



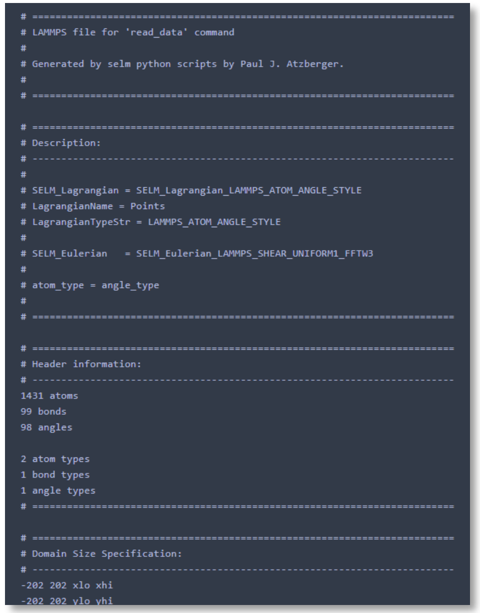
Setup the Simulation Files (such as .read_data)

```
In [13]:  num_dim = 3;
          box = np.zeros((num_dim,2));
          LL = 202.5; box[:,0] = -LL; box[:,1] = LL;

          # setup atoms
          I_id = 1; I_type = 1; atom_types = [];
          atom_list = []; atom_mass_list = []; atom_id_list = [];
          atom_mol_list = []; atom_name_list = [];

          # polymer atoms
          atom_name = "polymer_pts";
          atom_name_list.append(atom_name);
          atom_types.append(I_type);
```



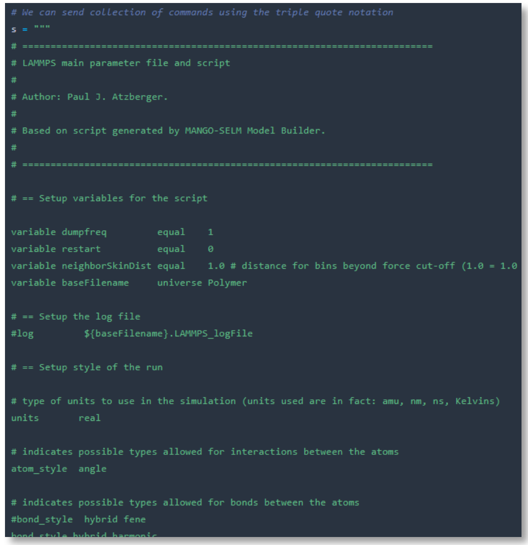
```
num_pts = 100;
t = np.linspace(0,2*np.pi,num_pts); R0 = 40;
x1 = R0*(2.0 + np.cos(2.0*t))*np.cos(3.0*t);
x2 = R0*(2.0 + np.cos(2.0*t))*np.sin(3.0*t);
x3 = R0*2*np.sin(4.0*t);
x = np.stack((x1,x2,x3),axis=1); # shape = [num_pts,num_dim]
num_pts = x.shape[0]; m0 = 1.123;
```

Model configuration (generated file)



```
# ==========================================================================
# LAMMPS file for 'read_data' command
#
# Generated by selm python scripts by Paul J. Atzberger.
#
# ==========================================================================

# ==========================================================================
# Description:
# --------------------------------------------------------------------------
#
# SELM_Lagrangian = SELM_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE
# LagrangianName = Points
# LagrangianTypeStr = LAMMPS_ATOM_ANGLE_STYLE
#
# SELM_Eulerian   = SELM_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3
#
# atom_type = angle_type
#
# ==========================================================================

# ==========================================================================
# Header information:
# --------------------------------------------------------------------------
1431 atoms
99 bonds
98 angles

2 atom types
1 bond types
1 angle types
# ==========================================================================

# ==========================================================================
# Domain Size Specification:
# --------------------------------------------------------------------------
-202 202 xlo xhi
-202 202 ylo yhi
```

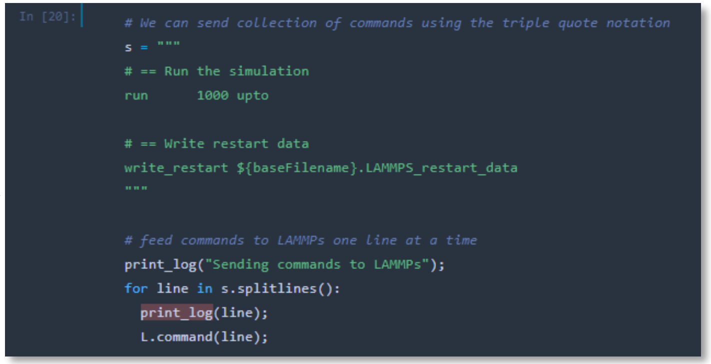
Setup parameters



```
# We can send collection of commands using the triple quote notation
s = """
# ======================================================================
# LAMMPS main parameter file and script
#
# Author: Paul J. Atzberger.
#
# Based on script generated by MANGO-SELM Model Builder.
#
# ======================================================================

# == Setup variables for the script

variable dumpfreq         equal    1
variable restart          equal    0
variable neighborSkinDist equal    1.0 # distance for bins beyond force cut-off (1.0 = 1.0
variable baseFilename     universe Polymer

# == Setup the log file
#log         ${baseFilename}.LAMMPS_logFile

# == Setup style of the run

# type of units to use in the simulation (units used are in fact: amu, nm, ns, Kelvins)
units       real

# indicates possible types allowed for interactions between the atoms
atom_style  angle

# indicates possible types allowed for bonds between the atoms
#bond_style  hybrid fene
bond_style hybrid harmonic
```

SELM XML files

Run simulation and analysis



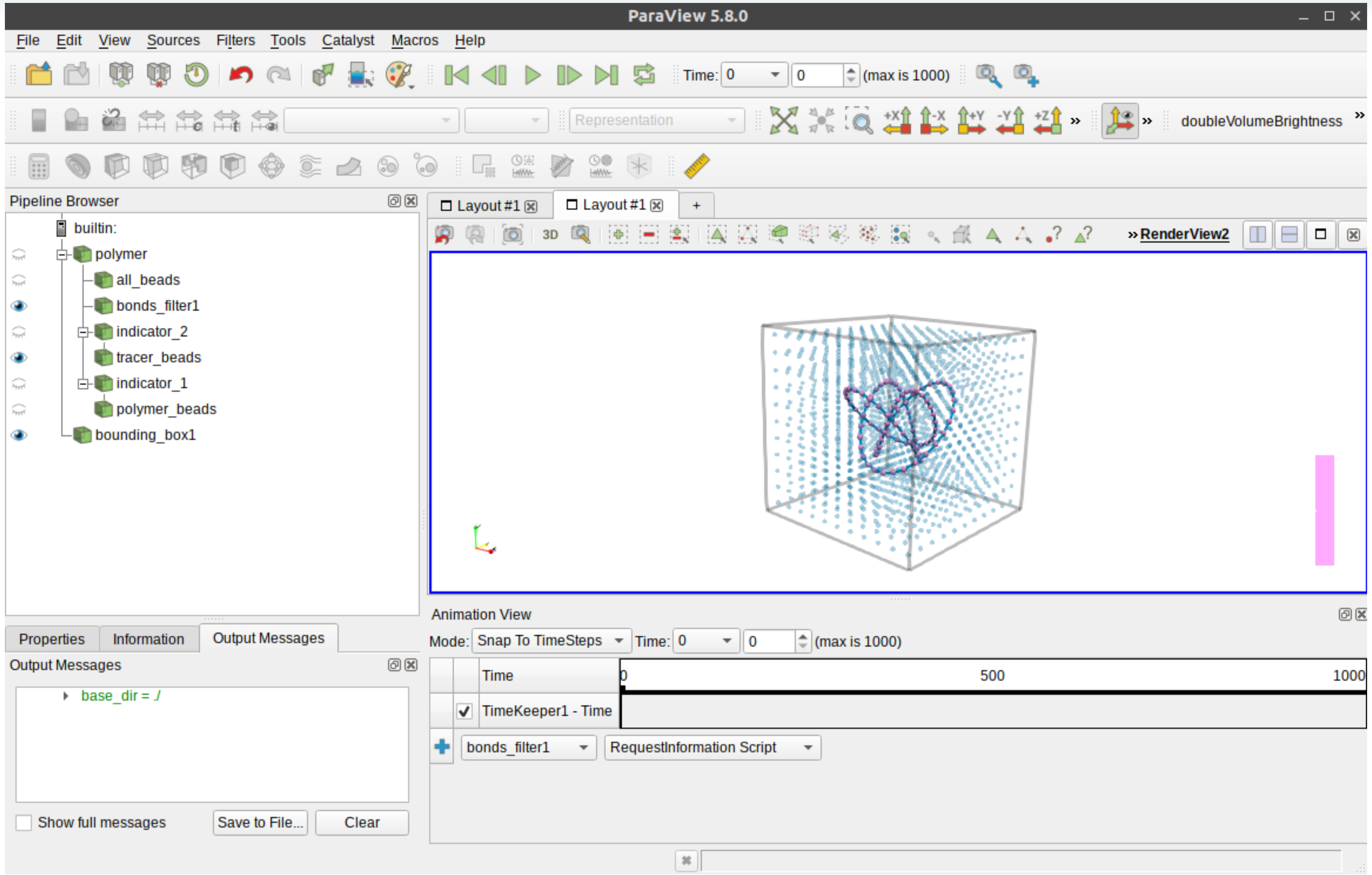
```
In [20]:  # We can send collection of commands using the triple quote notation
          s = """
          # == Run the simulation
          run      1000 upto

          # == Write restart data
          write_restart ${baseFilename}.LAMMPS_restart_data
          """

          # feed commands to LAMMPS one line at a time
          print_log("Sending commands to LAMMPS");
          for line in s.splitlines():
            print_log(line);
            L.command(line);
```

Interface allows for:

- checking intermediate results
- resuming simulations
- perform analysis and visualization.

Paraview for Visualization

ParaView
Kitware
Sandia National Laboratories
Los Alamos NATIONAL LABORATORY EST. 1943
ASC
ARL



```
base_dir = ./
```

Features:

- Processing VTK data output files (.vtp, .vtu, .vtr).
- Selm python scripts for loading and visualizing models.
- Rendering capabilities and movie generation.
- Analysis capabilities for statistics and other model properties.

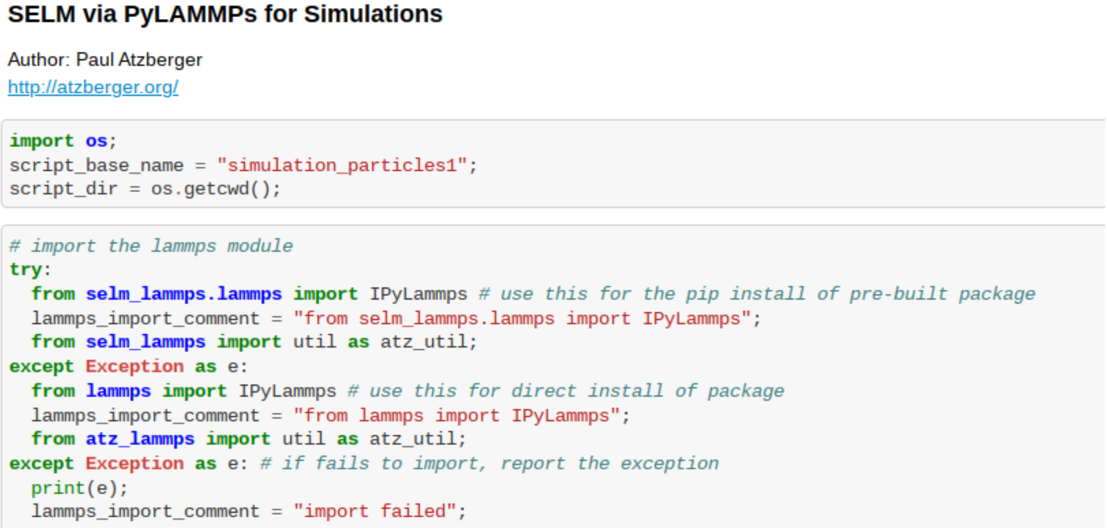
How to Setup Model in Practice

Examples

Particles: Thermal Fluctuations with Hydrodynamic Correlations

Directory:
examples/hydro1

Jupyter notebook:
simulation_hydro1.ipynb

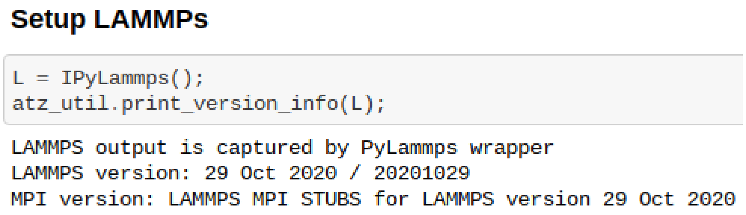


SELM via PyLAMMPs for Simulations

Author: Paul Atzberger
http://atzberger.org/

```
import os;
script_base_name = "simulation_particles1";
script_dir = os.getcwd();
```

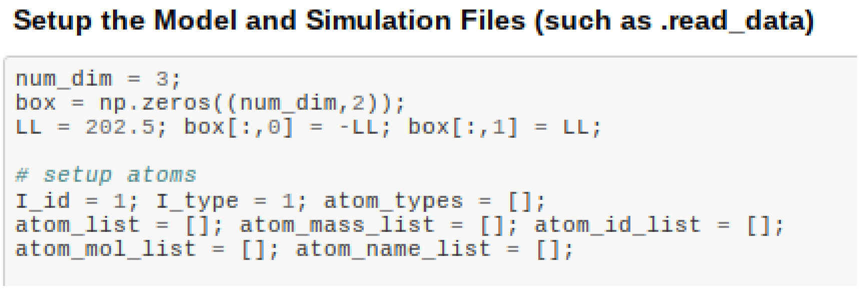
```
# import the lammps module
try:
  from selm_lammps.lammps import IPyLammps # use this for the pip install of pre-built package
  lammps_import_comment = "from selm_lammps.lammps import IPyLammps";
  from selm_lammps import util as atz_util;
except Exception as e:
  from lammps import IPyLammps # use this for direct install of package
  lammps_import_comment = "from lammps import IPyLammps";
  from atz_lammps import util as atz_util;
except Exception as e: # if fails to import, report the exception
  print(e);
  lammps_import_comment = "import failed";
```



Setup LAMMPs

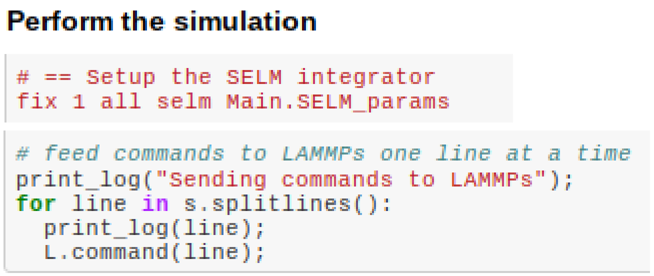
```
L = IPyLammps();
atz_util.print_version_info(L);
```

```
LAMMPS output is captured by PyLammps wrapper
LAMMPS version: 29 Oct 2020 / 20201029
MPI version: LAMMPS MPI STUBS for LAMMPS version 29 Oct 2020
```



Setup the Model and Simulation Files (such as .read_data)

```
num_dim = 3;
box = np.zeros((num_dim,2));
LL = 202.5; box[:,0] = -LL; box[:,1] = LL;

# setup atoms
I_id = 1; I_type = 1; atom_types = [];
atom_list = []; atom_mass_list = []; atom_id_list = [];
atom_mol_list = []; atom_name_list = [];
```

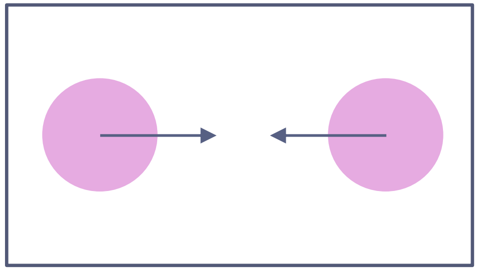


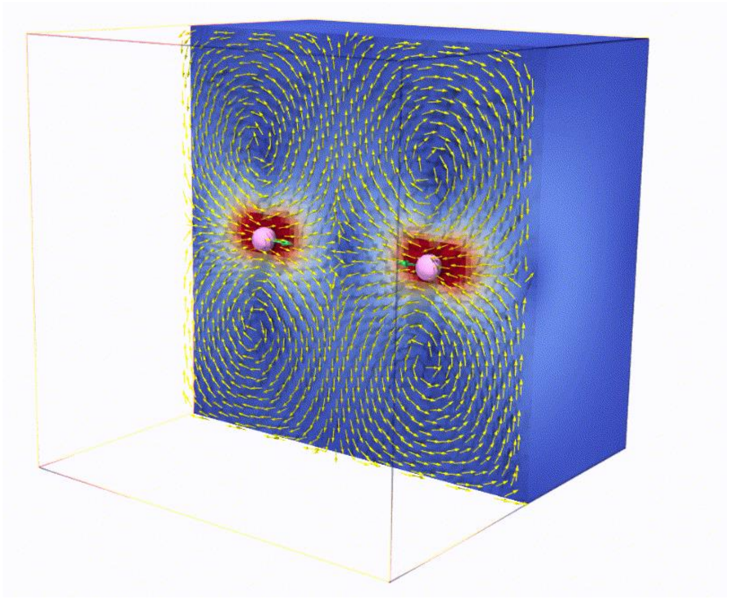
Perform the simulation

```
# == Setup the SELM integrator
fix 1 all selm Main.SELM_params
```

```
# feed commands to LAMMPs one line at a time
print_log("Sending commands to LAMMPs");
for line in s.splitlines():
  print_log(line);
  L.command(line);
```

particles w/ hydrodynamics

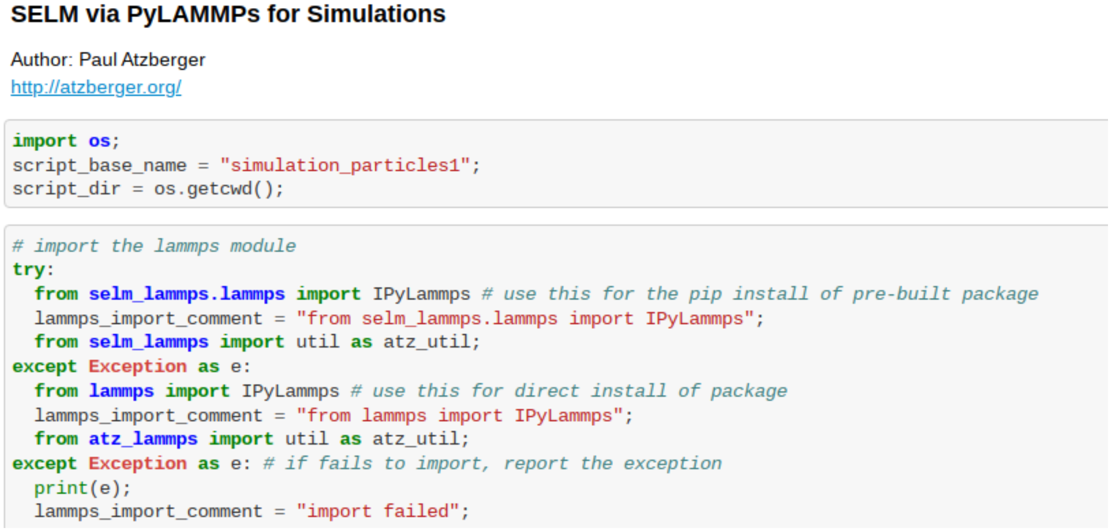




Particles: Thermal Fluctuations with Hydrodynamic Correlations

Directory:
examples/particles1

Jupyter notebook:
simulation_particles1.ipynb

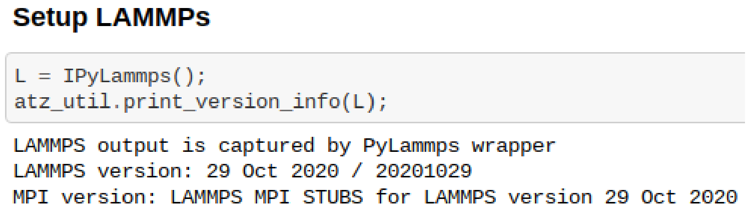


SELM via PyLAMMPs for Simulations

Author: Paul Atzberger
http://atzberger.org/

```
import os;
script_base_name = "simulation_particles1";
script_dir = os.getcwd();
```

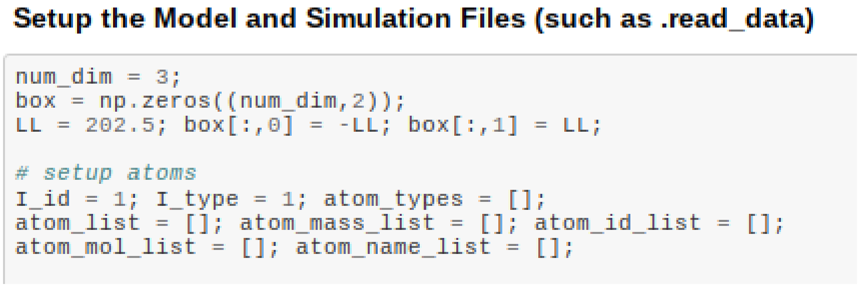
```
# import the lammps module
try:
  from selm_lammps.lammps import IPyLammps # use this for the pip install of pre-built package
  lammps_import_comment = "from selm_lammps.lammps import IPyLammps";
  from selm_lammps import util as atz_util;
except Exception as e:
  from lammps import IPyLammps # use this for direct install of package
  lammps_import_comment = "from lammps import IPyLammps";
  from atz_lammps import util as atz_util;
except Exception as e: # if fails to import, report the exception
  print(e);
  lammps_import_comment = "import failed";
```



Setup LAMMPs

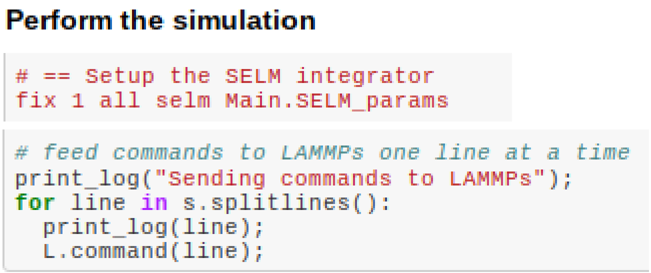
```
L = IPyLammps();
atz_util.print_version_info(L);
```

```
LAMMPS output is captured by PyLammps wrapper
LAMMPS version: 29 Oct 2020 / 20201029
MPI version: LAMMPS MPI STUBS for LAMMPS version 29 Oct 2020
```



Setup the Model and Simulation Files (such as .read_data)

```
num_dim = 3;
box = np.zeros((num_dim,2));
LL = 202.5; box[:,0] = -LL; box[:,1] = LL;

# setup atoms
I_id = 1; I_type = 1; atom_types = [];
atom_list = []; atom_mass_list = []; atom_id_list = [];
atom_mol_list = []; atom_name_list = [];
```

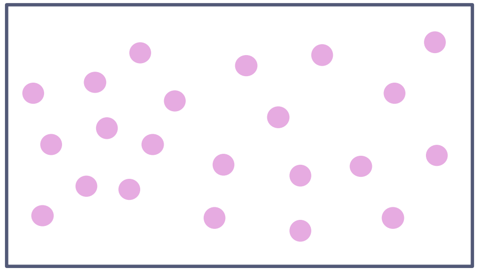


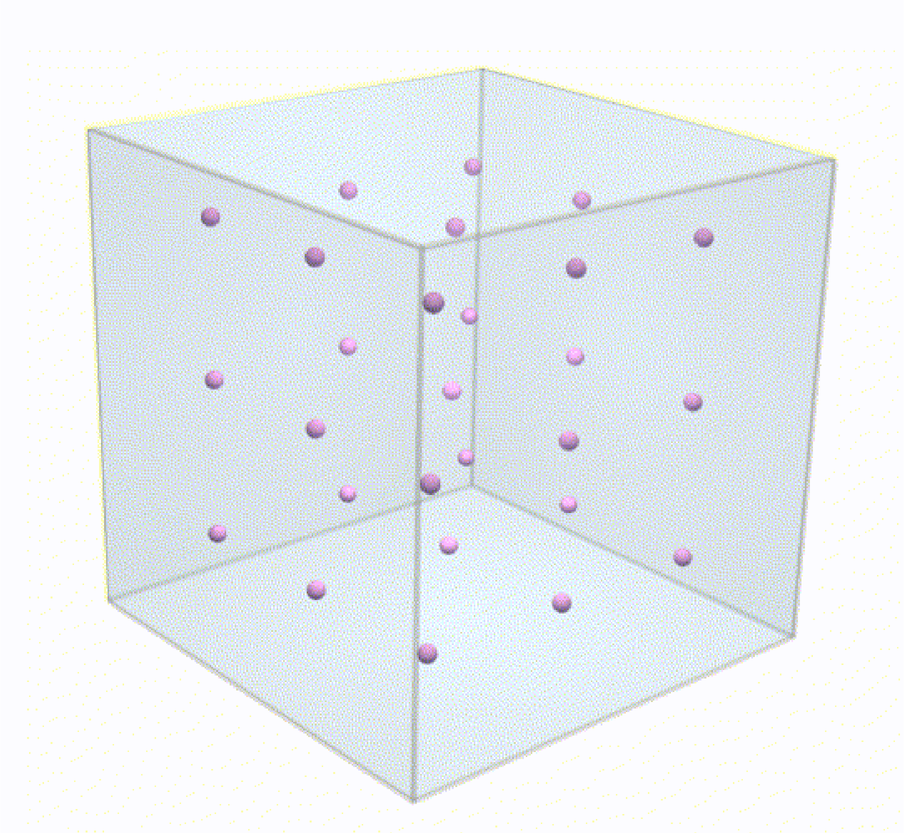
Perform the simulation

```
# == Setup the SELM integrator
fix 1 all selm Main.SELM_params
```

```
# feed commands to LAMMPS one line at a time
print_log("Sending commands to LAMMPs");
for line in s.splitlines():
  print_log(line);
  L.command(line);
```

particle diffusion w/
hydrodynamic correlations



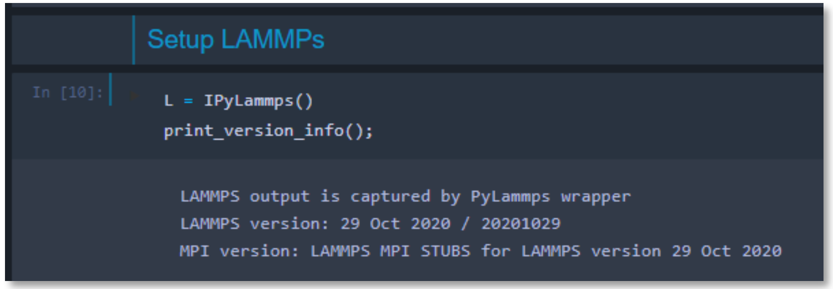


Polymer: Thermal Fluctuations with Hydrodynamic Correlations

Directory:
examples/polymer4

Jupyter notebook:
simulation_polymer4.ipynb

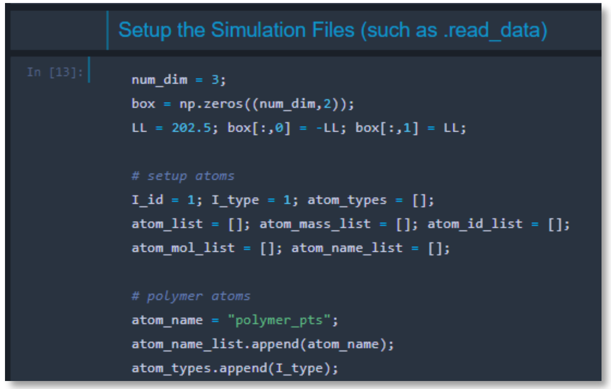
Create Selm-Lammps instance



Setup LAMMPs

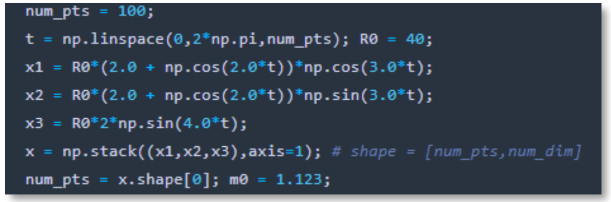
```
In [10]:
    L = IPyLammps()
    print_version_info();

      LAMMPS output is captured by PyLammps wrapper
      LAMMPS version: 29 Oct 2020 / 20201029
      MPI version: LAMMPS MPI STUBS for LAMMPS version 29 Oct 2020
```

Setup model geometry and interactions



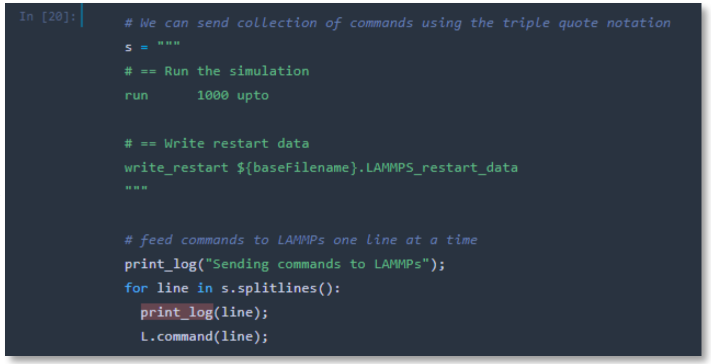
Setup the Simulation Files (such as .read_data)

```
In [13]:
    num_dim = 3;
    box = np.zeros((num_dim,2));
    LL = 202.5; box[:,0] = -LL; box[:,1] = LL;

    # setup atoms
    I_id = 1; I_type = 1; atom_types = [];
    atom_list = []; atom_mass_list = []; atom_id_list = [];
    atom_mol_list = []; atom_name_list = [];

    # polymer atoms
    atom_name = "polymer_pts";
    atom_name_list.append(atom_name);
    atom_types.append(I_type);
```



```
num_pts = 100;
t = np.linspace(0,2*np.pi,num_pts); R0 = 40;
x1 = R0*(2.0 + np.cos(2.0*t))*np.cos(3.0*t);
x2 = R0*(2.0 + np.cos(2.0*t))*np.sin(3.0*t);
x3 = R0*2*np.sin(4.0*t);
x = np.stack((x1,x2,x3),axis=1); # shape = [num_pts,num_dim]
num_pts = x.shape[0]; m0 = 1.123;
```

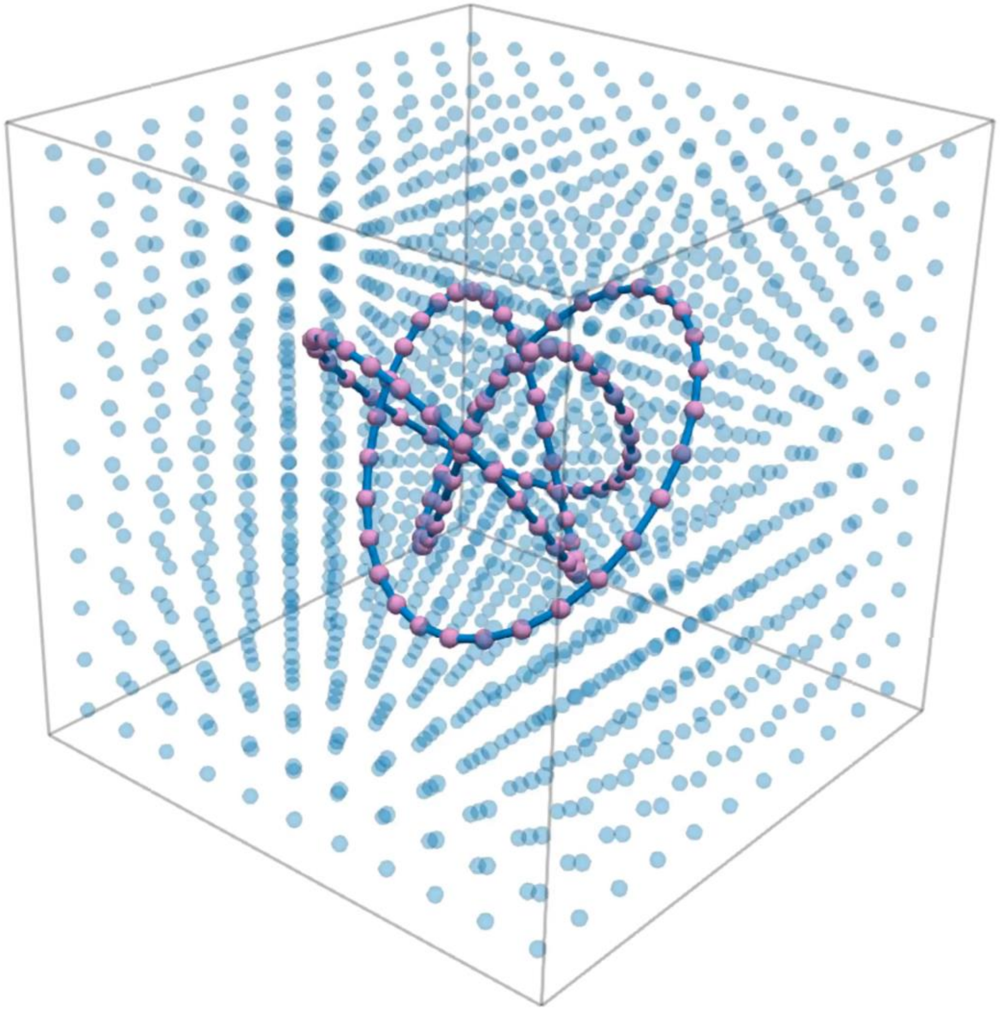
Run simulation and analysis



```
In [20]:
    # We can send collection of commands using the triple quote notation
    s = """
    # == Run the simulation
    run      1000 upto

    # == Write restart data
    write_restart ${baseFilename}.LAMMPS_restart_data
    """


    # feed commands to LAMMPs one line at a time
    print_log("Sending commands to LAMMPs");
    for line in s.splitlines():
      print_log(line);
      L.command(line);
```

SELM XML files

Polymer knot fluctuations w/ hydrodynamic correlations

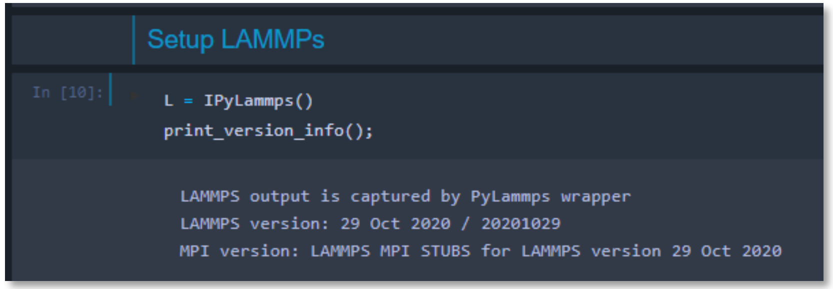


Polymer Knot: Subject to Shearing and Thermal Fluctuations

Directory:
examples/polymer4

Jupyter notebook:
simulation_polymer4.ipynb

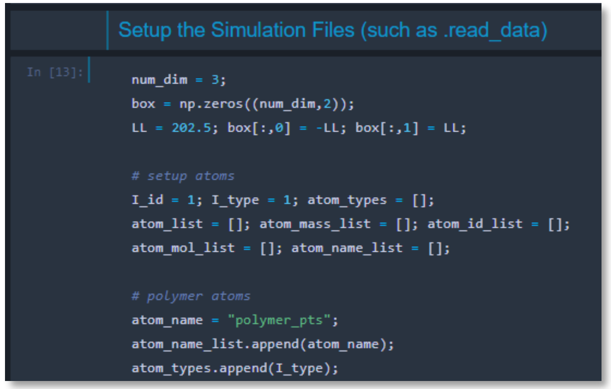
Create Selm-Lammps instance



Setup LAMMPs

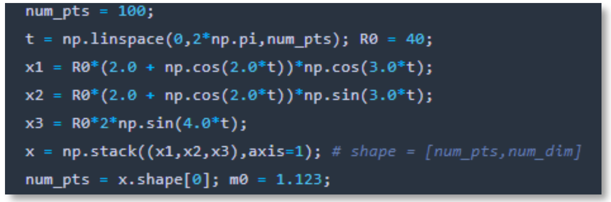
```
In [10]:  L = IPyLammps()
          print_version_info();

            LAMMPS output is captured by PyLammps wrapper
            LAMMPS version: 29 Oct 2020 / 20201029
            MPI version: LAMMPS MPI STUBS for LAMMPS version 29 Oct 2020
```

Setup model geometry and interactions



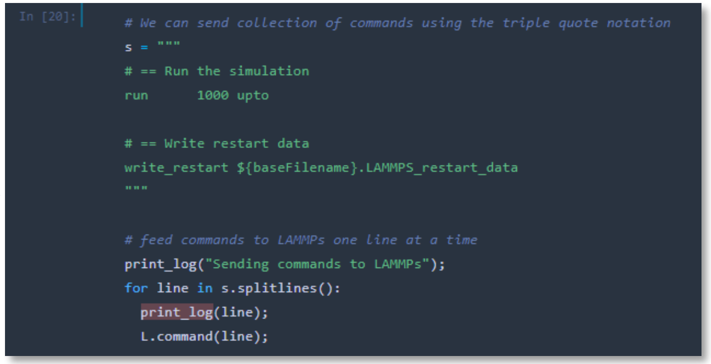
Setup the Simulation Files (such as .read_data)

```
In [13]:  num_dim = 3;
          box = np.zeros((num_dim,2));
          LL = 202.5; box[:,0] = -LL; box[:,1] = LL;

          # setup atoms
          I_id = 1; I_type = 1; atom_types = [];
          atom_list = []; atom_mass_list = []; atom_id_list = [];
          atom_mol_list = []; atom_name_list = [];

          # polymer atoms
          atom_name = "polymer_pts";
          atom_name_list.append(atom_name);
          atom_types.append(I_type);
```



```
num_pts = 100;
t = np.linspace(0,2*np.pi,num_pts); R0 = 40;
x1 = R0*(2.0 + np.cos(2.0*t))*np.cos(3.0*t);
x2 = R0*(2.0 + np.cos(2.0*t))*np.sin(3.0*t);
x3 = R0*2*np.sin(4.0*t);
x = np.stack((x1,x2,x3),axis=1); # shape = [num_pts,num_dim]
num_pts = x.shape[0]; m0 = 1.123;
```

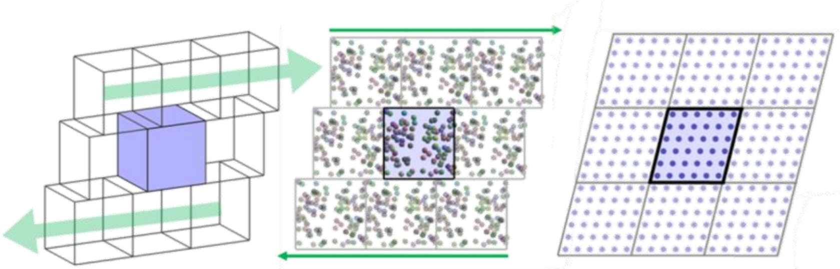
Run simulation and analysis



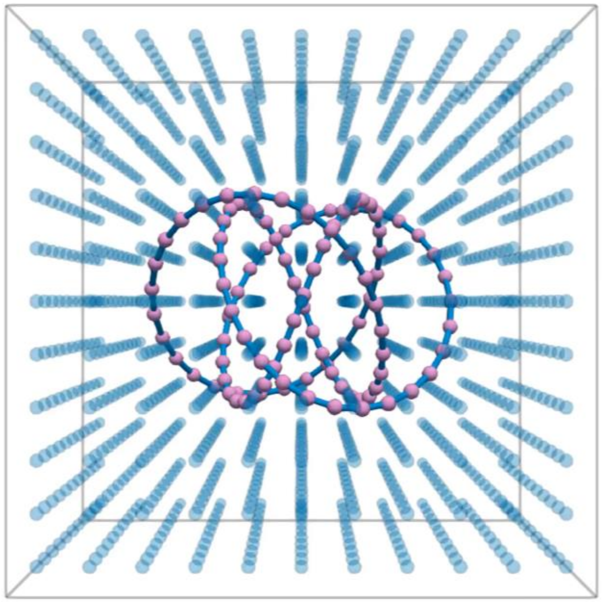
```
In [20]:  # We can send collection of commands using the triple quote notation
          s = """
          # == Run the simulation
          run       1000 upto

          # == Write restart data
          write_restart ${baseFilename}.LAMMPS_restart_data
          """


          # feed commands to LAMMPs one line at a time
          print_log("Sending commands to LAMMPs");
          for line in s.splitlines():
            print_log(line);
            L.command(line);
```

SELM XML files

Shearing boundaries: Less-Edwards conditions

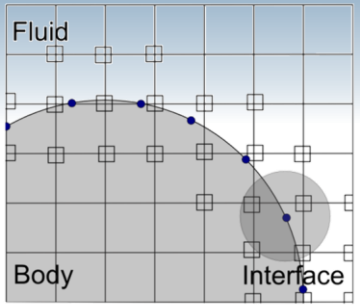


Polymer knot fluctuations in shear flow

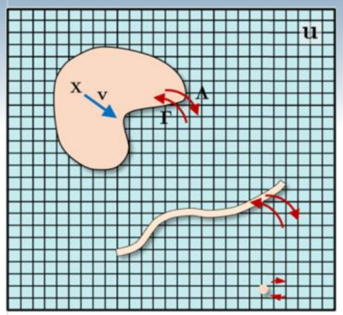


Further Simulations with SELMs

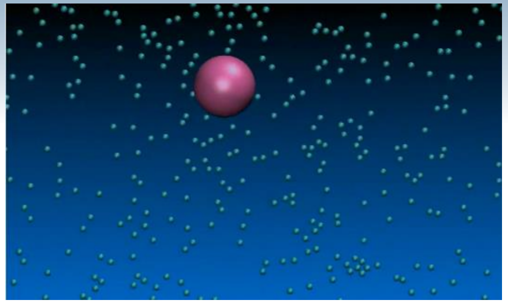
SELM Fluctuating Hydrodynamics



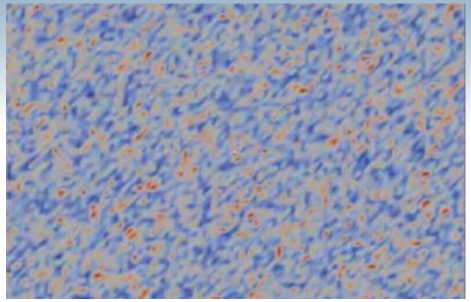
Eulerian-Lagrangian Mechanics



Eulerian-Lagrangian Coupling



Brownian Motion: Molecular Collisions



Continuum Gaussian Random Field

SELM Inertial Regime I:

hydrodynamics

$$\begin{aligned} \rho \frac{\partial \mathbf{u}}{\partial t} &= \mu \Delta \mathbf{u} - \nabla p + \Lambda[\Upsilon(\mathbf{v} - \Gamma \mathbf{u})] + \mathbf{f}_{thm} \\ \nabla \cdot \mathbf{u} &= 0. \end{aligned}$$

microstructure

$$\begin{aligned} \frac{d\mathbf{X}}{dt} &= \mathbf{v} \\ m\frac{d\mathbf{v}}{dt} &= -\Upsilon(\mathbf{v} - \Gamma \mathbf{u}) - \nabla_X \Phi[X] + \mathbf{F}_{thm}. \end{aligned}$$

thermal fluctuations

$$\begin{aligned} \langle \mathbf{f}_{\mathrm{thm}}(s)\mathbf{f}_{\mathrm{thm}}^T(t)\rangle &= -(2k_BT)\,(\mathcal{L} - \Lambda\Upsilon\Gamma)\,\delta(t-s) \\ \langle \mathbf{F}_{\mathrm{thm}}(s)\mathbf{F}_{\mathrm{thm}}^T(t)\rangle &= (2k_BT)\,\Upsilon\,\delta(t-s) \\ \langle \mathbf{f}_{\mathrm{thm}}(s)\mathbf{F}_{\mathrm{thm}}^T(t)\rangle &= -(2k_BT)\,\Lambda\Upsilon\,\delta(t-s). \end{aligned}$$

SELM Overdamped Regime IV:

microstructure + hydrodynamics (quasi-steady)

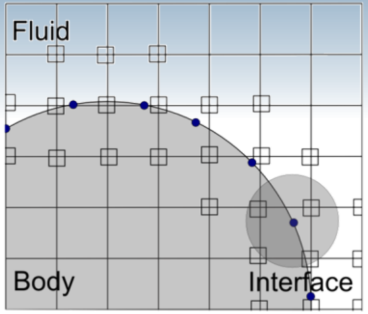
$$\begin{aligned} \frac{d\mathbf{X}}{dt} &= H_{\mathrm{SELM}}[-\nabla_{\mathbf{X}}\Phi(\mathbf{X})] \\ &+ (\nabla_{\mathbf{X}} \cdot H_{\mathrm{SELM}})k_BT + \mathbf{h}_{\mathrm{thm}} \\ H_{\mathrm{SELM}} &= \Gamma(-\wp\mathcal{L})^{-1}\Lambda \end{aligned}$$

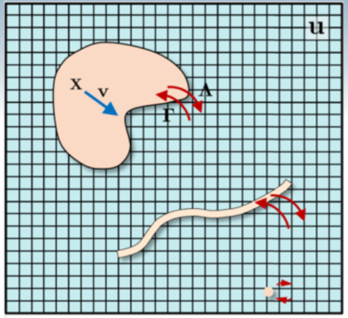
thermal fluctuations

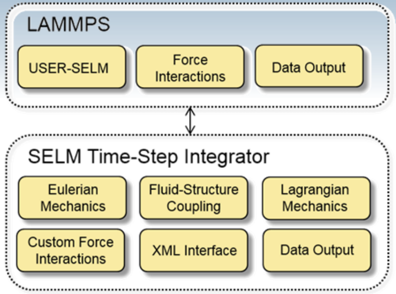
$$\langle \mathbf{h}_{\mathrm{thm}}(s)\mathbf{h}_{\mathrm{thm}}^T(t)\rangle = (2k_BT)\,H_{\mathrm{SELM}}\,\delta(t-s).$$

Atzberger 2015

MANGO-SELM Simulation Software







SELM Inertial Regime I (Verlet-stye temporal integration):

microstructure

$$\begin{aligned}
\mathbf{v}^{n+\frac{1}{2}} &= \mathbf{v}^{n} + \frac{\Delta t}{2} m^{-1}\mathbf{F}^{n} \\
&+ \frac{\Delta t}{2}\left(-m^{-1}\Upsilon\left(\mathbf{v}^{n-\frac{1}{2}} - \Gamma^{n}\mathbf{u}^{n-\frac{1}{2}}\right)\right. \\
&\quad \left. + m^{-1}\mathbf{g}^{n-\frac{1}{2}}\right) \\
\mathbf{X}^{n+1} &= \mathbf{X}^{n} + \mathbf{v}^{n+\frac{1}{2}}\Delta t
\end{aligned}$$

hydrodynamics

$$\begin{aligned}
\mathbf{u}^{n+\frac{1}{2}} &= \mathbf{u}^{n} + \frac{\Delta t}{2}\rho^{-1}\mu L\mathbf{u}^{n-\frac{1}{2}} \\
&- \frac{\Delta t}{2}\left(\rho^{-1}\Lambda^{n}\left[-\Upsilon\left(\mathbf{v}^{n-\frac{1}{2}} - \Gamma^{n}\mathbf{u}^{n-\frac{1}{2}}\right)\right.\right. \\
&\quad \left.\left. + \mathbf{g}^{n-\frac{1}{2}}\right]\right) \\
&+ \mathbf{h}^{n-\frac{1}{2}}
\end{aligned}$$

thermal fluctuations

$$\begin{aligned}
\langle \mathbf{g}^{n-\frac{1}{2}}\mathbf{g}^{n-\frac{1}{2}T}\rangle &= 4k_B T\Upsilon/\Delta t \\
\langle \mathbf{h}^{n}\mathbf{h}^{nT}\rangle &= 4k_B T\rho^{-2}\mu L/\Delta t.
\end{aligned}$$

microstructure

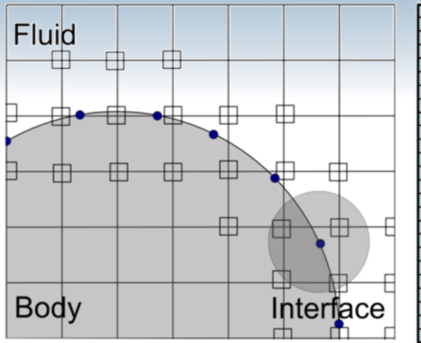
$$\begin{aligned}
\mathbf{v}^{n+1} &= \mathbf{v}^{n+\frac{1}{2}} + \frac{\Delta t}{2} m^{-1}\mathbf{F}^{n+1} \\
&+ \frac{\Delta t}{2}\left(-m^{-1}\Upsilon\left(\mathbf{v}^{n+\frac{1}{2}} - \Gamma^{n+1}\mathbf{u}^{n+\frac{1}{2}}\right)\right. \\
&\quad \left. + m^{-1}\mathbf{g}^{n+\frac{1}{2}}\right)
\end{aligned}$$

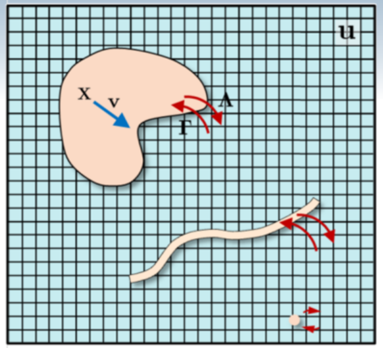
hydrodynamics

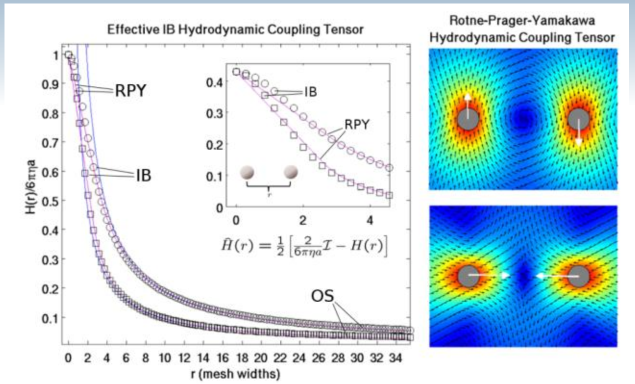
$$\begin{aligned}
\mathbf{u}^{n+1} &= \mathbf{u}^{n+\frac{1}{2}} + \frac{\Delta t}{2}\rho^{-1}\mu L\mathbf{u}^{n+\frac{1}{2}} \\
&- \frac{\Delta t}{2}\left(\rho^{-1}\Lambda^{n+1}\left[-\Upsilon\left(\mathbf{v}^{n+\frac{1}{2}} - \Gamma^{n+1}\mathbf{u}^{n+\frac{1}{2}}\right)\right.\right. \\
&\quad \left.\left. + \mathbf{g}^{n+\frac{1}{2}}\right]\right) \\
&+ \mathbf{h}^{n+\frac{1}{2}}.
\end{aligned}$$

Atzberger 2015

MANGO-SELM Simulation Software



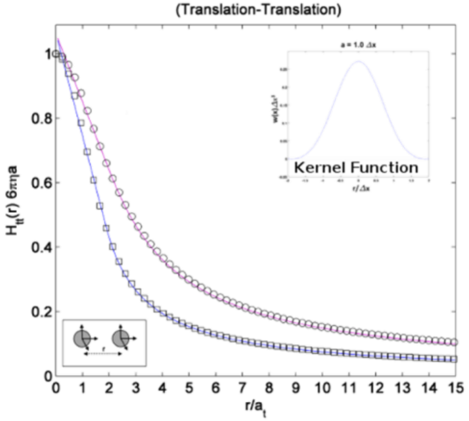




SELM Coupling:

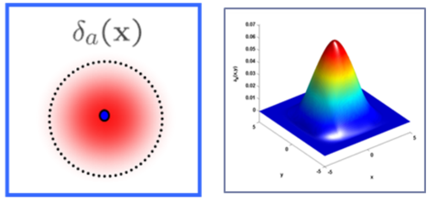
adjoint condition

$$\langle \Gamma \mathbf{v}, \mathbf{F}\rangle = \sum_i [\Gamma \mathbf{v}]_i \cdot [\mathbf{F}]_i = \int_\Omega \mathbf{v}(\mathbf{x}) \cdot (\Lambda \mathbf{F})(\mathbf{x}) d\mathbf{x} = \langle \mathbf{v}, \Lambda \mathbf{F}\rangle$$



IB-Kernel coupling:

$$\begin{aligned} \Gamma \mathbf{u} &= \int_\Omega \eta\left(\mathbf{y} - \mathbf{X}(t)\right) \mathbf{u}(\mathbf{y}, t) d\mathbf{y} \\ \Lambda \mathbf{F} &= \eta\left(\mathbf{x} - \mathbf{X}(t)\right) \mathbf{F}. \end{aligned}$$



Peskin delta-function

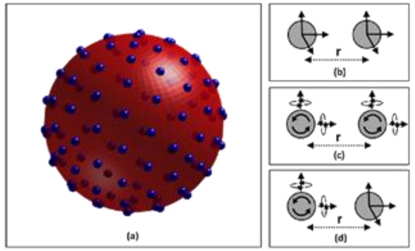
Generalized Coupling (Faxen)

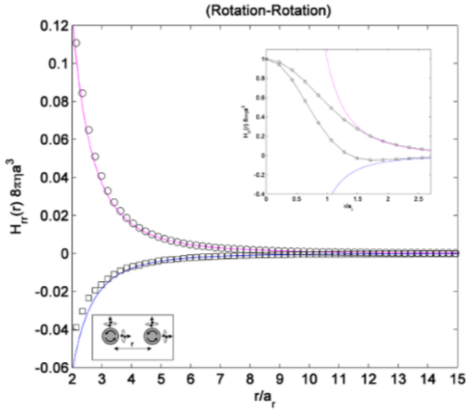
$$\Gamma_0 \mathbf{u} = \sum_{\mathbf{m}} \left\langle\, \eta_0(\mathbf{y}_{\mathbf{m}} - (\mathbf{X}_{\rm cm} + \mathbf{z}))\; \mathbf{u}_{\mathbf{m}} \,\right\rangle_{\tilde{\mathcal{S}}, |\mathbf{z}|=R} \Delta x_{\mathbf{m}}^3$$

$$\Gamma_1 \mathbf{u} = \frac{3}{2R^2} \sum_{\mathbf{m}} \left\langle\, \eta_1(\mathbf{y}_{\mathbf{m}} - (\mathbf{X}_{\rm cm} + \mathbf{z}))\, (\mathbf{z} \times \mathbf{u}_{\mathbf{m}}) \,\right\rangle_{\tilde{\mathcal{S}}, |\mathbf{z}|=R} \Delta x_{\mathbf{m}}^3.$$

$$\Lambda_0(\mathbf{x}_{\mathbf{m}}) = \left(\left\langle\, \eta_0(\mathbf{x}_{\mathbf{m}} - (\mathbf{X}_{\rm cm} + \mathbf{z})) \,\right\rangle_{\tilde{\mathcal{S}}, |\mathbf{z}|=R} \right) \mathbf{F}$$

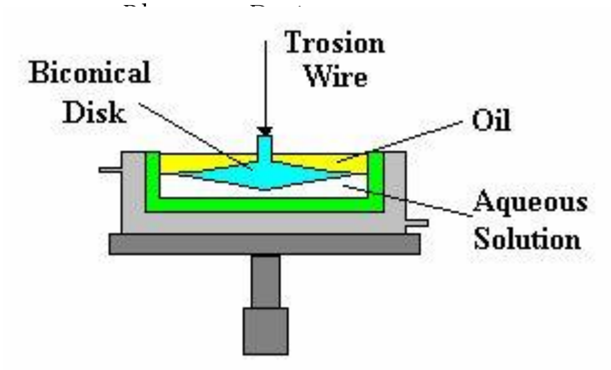
$$\Lambda_1(\mathbf{x}_{\mathbf{m}}) = -\frac{3}{2R^2} \left(\left\langle\, \mathbf{z}\eta_1(\mathbf{x}_{\mathbf{m}} - (\mathbf{X}_{\rm cm} + \mathbf{z})) \,\right\rangle_{\tilde{\mathcal{S}}, |\mathbf{z}|=R} \right) \times \mathbf{T}.$$

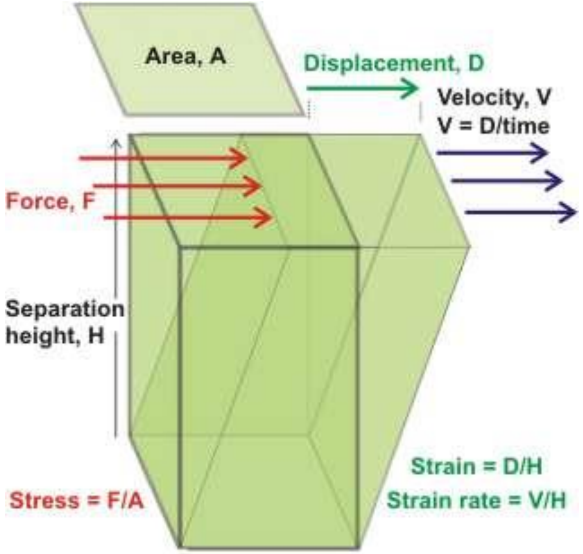




Atzberger 2011, Atzberger 2015

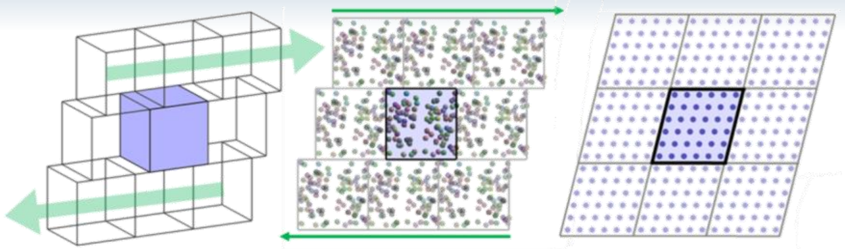
Rheological Properties and Microstructure Dynamics



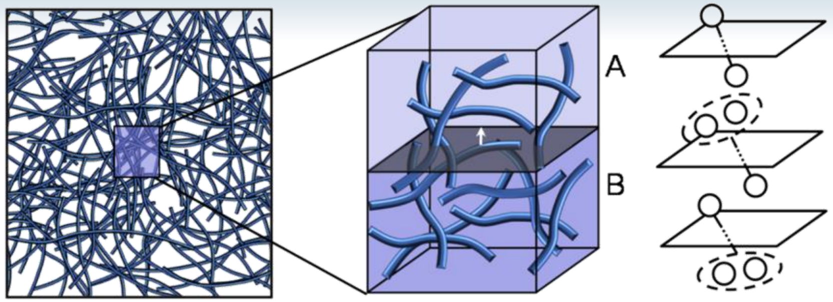


MANGO-SELM Simulation Software

Lees-Edwards Boundary Conditions:



Material Stress ←Microscopic Forces



Stress Tensor Estimator:

$$\sigma^{(n)}_{\ell,z} = \frac{1}{AL}\sum_{\mathbf{q}\in\mathcal{Q}_n}\sum_{j=1}^{n-1}\left\langle \mathbf{f}^{(\ell)}_{\mathbf{q},j}\cdot\left(\mathbf{x}^{*,(z)}_{q_n} - \mathbf{x}^{*,(z)}_{q_j}\right)\right\rangle$$

Fluctuating hydrodynamics (moving frame):

$$\begin{aligned}
\rho\frac{d\mathbf{w}}{dt} &= L(t)\mathbf{w} + \lambda + \Lambda[-\nabla_{\mathbf{X}}\Phi] + (\nabla_{\mathbf{X}}\cdot\Lambda)\,k_BT + \mathbf{J} + \mathbf{h}_{\mathrm{thm}} \\
S(t)\cdot\mathbf{w} &= \mathbf{K} \\
\frac{d\mathbf{X}}{dt} &= \Gamma\mathbf{w}.
\end{aligned}$$

$$\begin{aligned}
S(t)\cdot\mathbf{w} &= D\cdot\mathbf{w} + \mathbf{e}_z^T\; G\mathbf{w}\;\mathbf{e}_x\dot{\gamma}t \\
L(t)\mathbf{w} &= \mu\left[\mathbf{e}_d - \delta_{d,3}\dot{\gamma}t\mathbf{e}_x\right]^T A\mathbf{w}\left[\mathbf{e}_d - \delta_{d,3}\dot{\gamma}t\mathbf{e}_x\right]
\end{aligned}$$

$$\begin{aligned}
G(s,t) &= \langle\mathbf{h}_{\mathrm{thm}}(s)\mathbf{h}_{\mathrm{thm}}(t)^T\rangle \\
G(s,t) &= -2\wp(t)L(t)C\delta(t-s)
\end{aligned}$$

$$D\cdot\mathbf{w} = \sum_{d=1}^{3}\frac{\mathbf{w}^{(d)}(\mathbf{q}+\mathbf{e}_d) - \mathbf{w}^{(d)}(\mathbf{q}-\mathbf{e}_d)}{2\Delta x}$$

$$[G\mathbf{w}]_{ij} = \frac{\mathbf{w}^{(i)}(\mathbf{q}+\mathbf{e}_j) - \mathbf{w}^{(i)}(\mathbf{q}-\mathbf{e}_j)}{2\Delta x}$$

$$\begin{aligned}
[A\mathbf{w}]_{ii} &= \frac{\mathbf{w}^{(i)}(\mathbf{q}+\mathbf{e}_i) - 2\mathbf{w}^{(i)}(\mathbf{q}) + \mathbf{w}^{(i)}(\mathbf{q}-\mathbf{e}_i)}{\Delta x^2} \\
[A\mathbf{w}]_{ij} &= \frac{\mathbf{w}^{(d)}(\mathbf{q}+\mathbf{e}_i+\mathbf{e}_j) - \mathbf{w}^{(d)}(\mathbf{q}-\mathbf{e}_i+\mathbf{e}_j)}{4\Delta x^2} \\
&\quad - \frac{\mathbf{w}^{(d)}(\mathbf{q}+\mathbf{e}_i-\mathbf{e}_j) - \mathbf{w}^{(d)}(\mathbf{q}-\mathbf{e}_i-\mathbf{e}_j)}{4\Delta x^2},\;\; i\neq j.
\end{aligned}$$

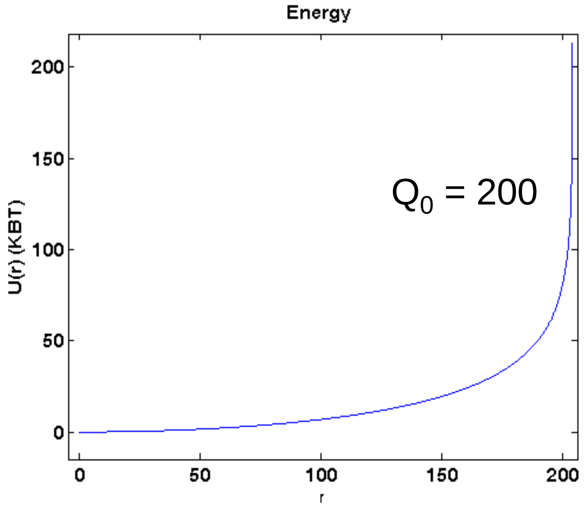
Atzberger 2013

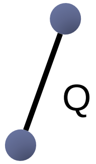
MANGO-SELM Simulation Software

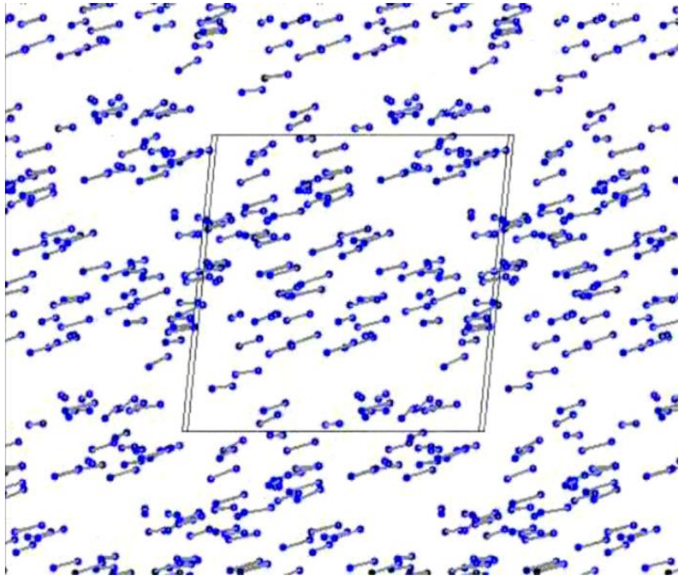
Example System : Finitely Extensible Nonlinear Elastic (FENE) Dimers:

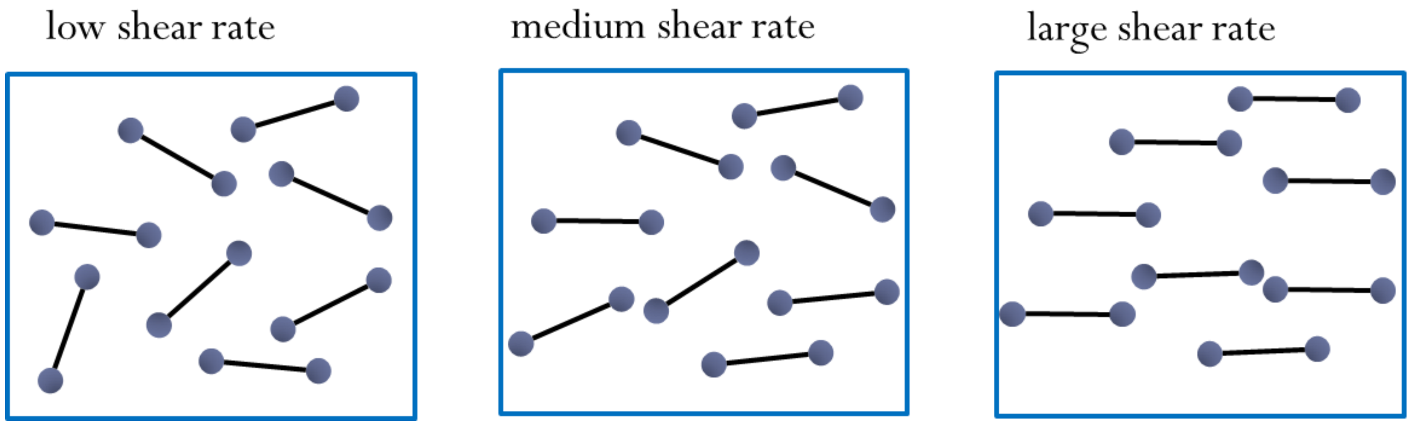
Potential Energy:

$$U(r) = -\frac{K}{2} Q_0^2 \log(1 - (Q/Q_0)^2)$$



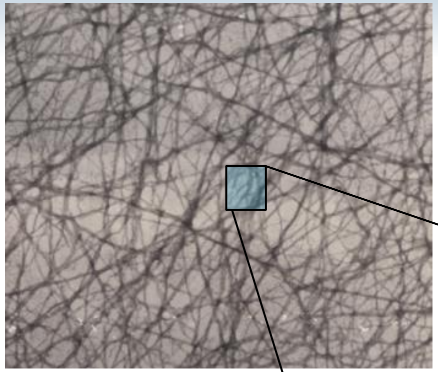


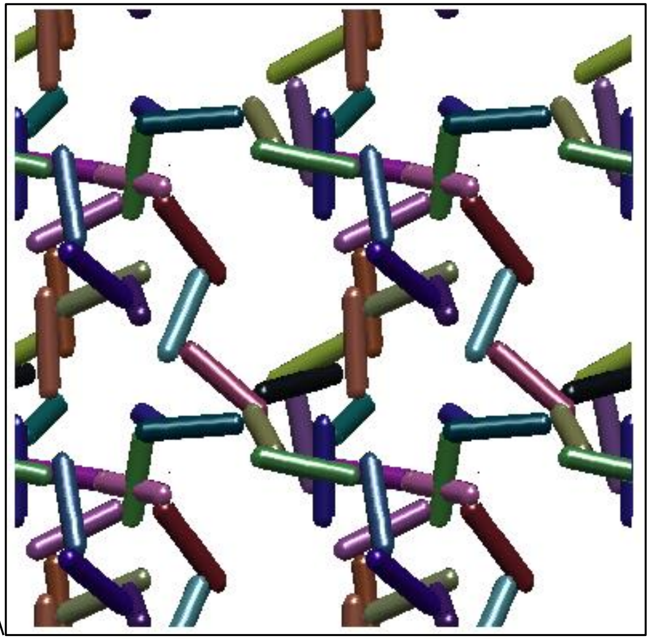




Atzberger 2013

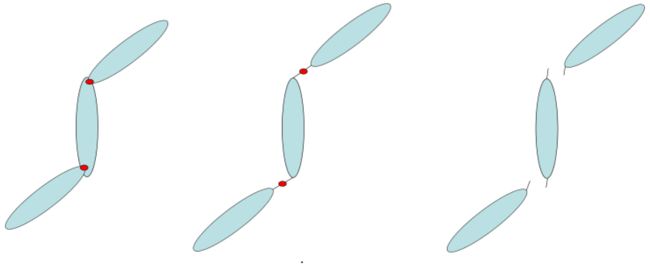
Gel-like Materials



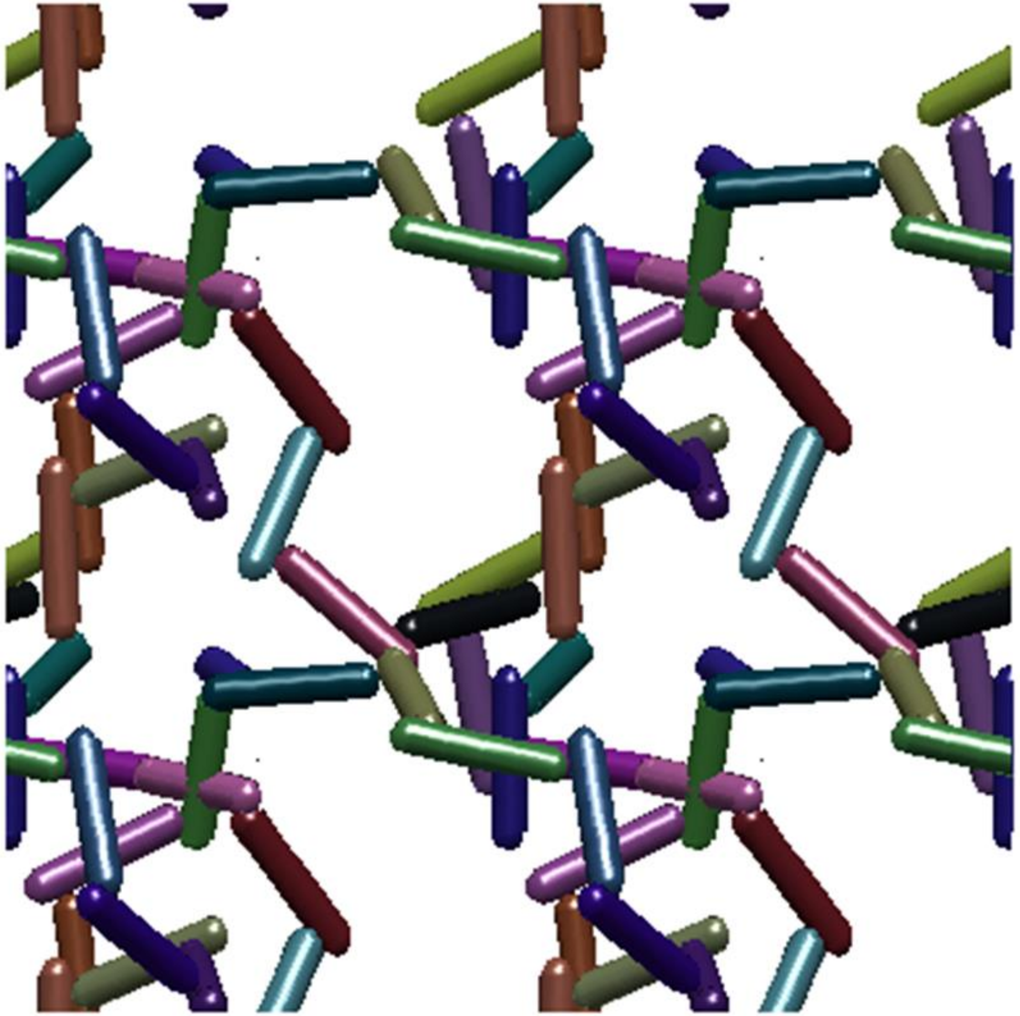


Polymeric Gel Model:

- Worm-like chain elasticity for individual polymers.
- Near-end segments “sticky”.
- Branches bond angles are 70 degrees.
- Yield of “sticky” ends after 50% rest-length stretchi

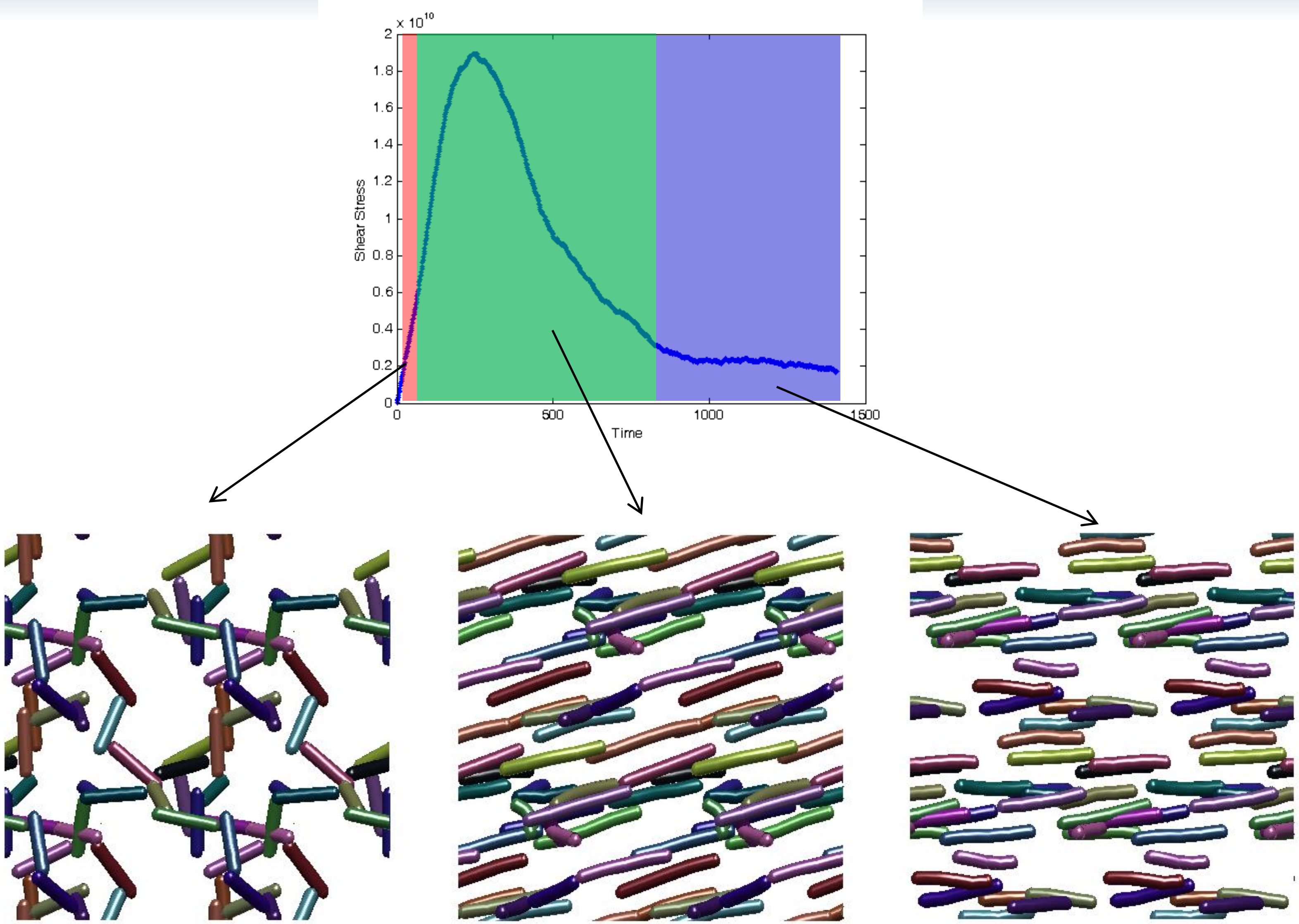


Gel-like Materials



Gel-like Materials

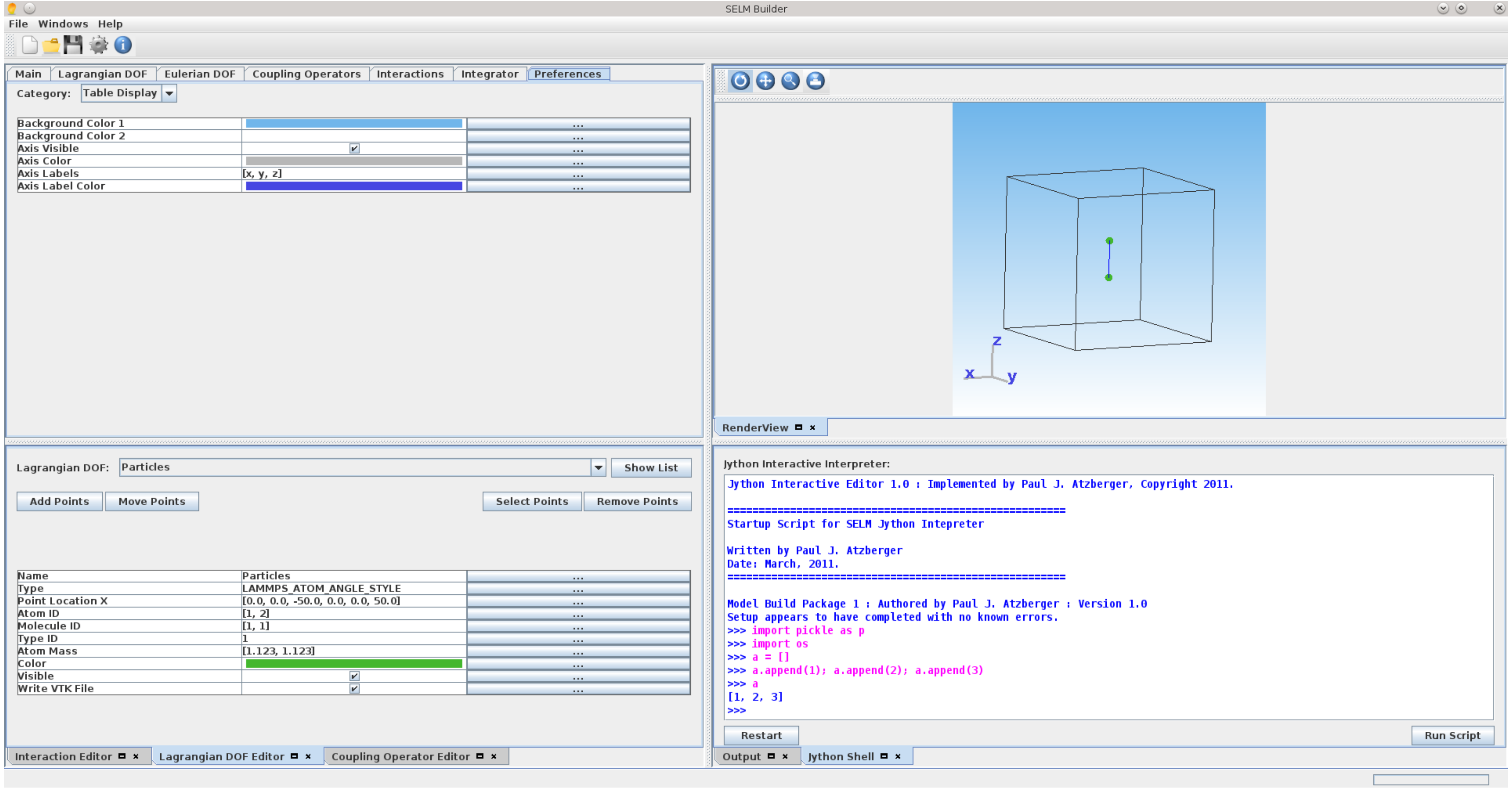
Time Dependence of Shear Stress



Mango-Selm GUI
(previous versions)

MANGO-SELM Simulation Software

MANGO - Modeling Software:



Category: Table Display

Background Color 1	🟦	...
Background Color 2		...
Axis Visible	☑	...
Axis Color	▨	...
Axis Labels	[x, y, z]	...
Axis Label Color	🟦	...

Lagrangian DOF: Particles

Name	Particles	...
Type	LAMMPS_ATOM_ANGLE_STYLE	...
Point Location X	[0.0, 0.0, -50.0, 0.0, 0.0, 50.0]	...
Atom ID	[1, 2]	...
Molecule ID	[1, 1]	...
Type ID	1	...
Atom Mass	[1.123, 1.123]	...
Color	🟩	...
Visible	☑	...
Write VTK File	☑	...

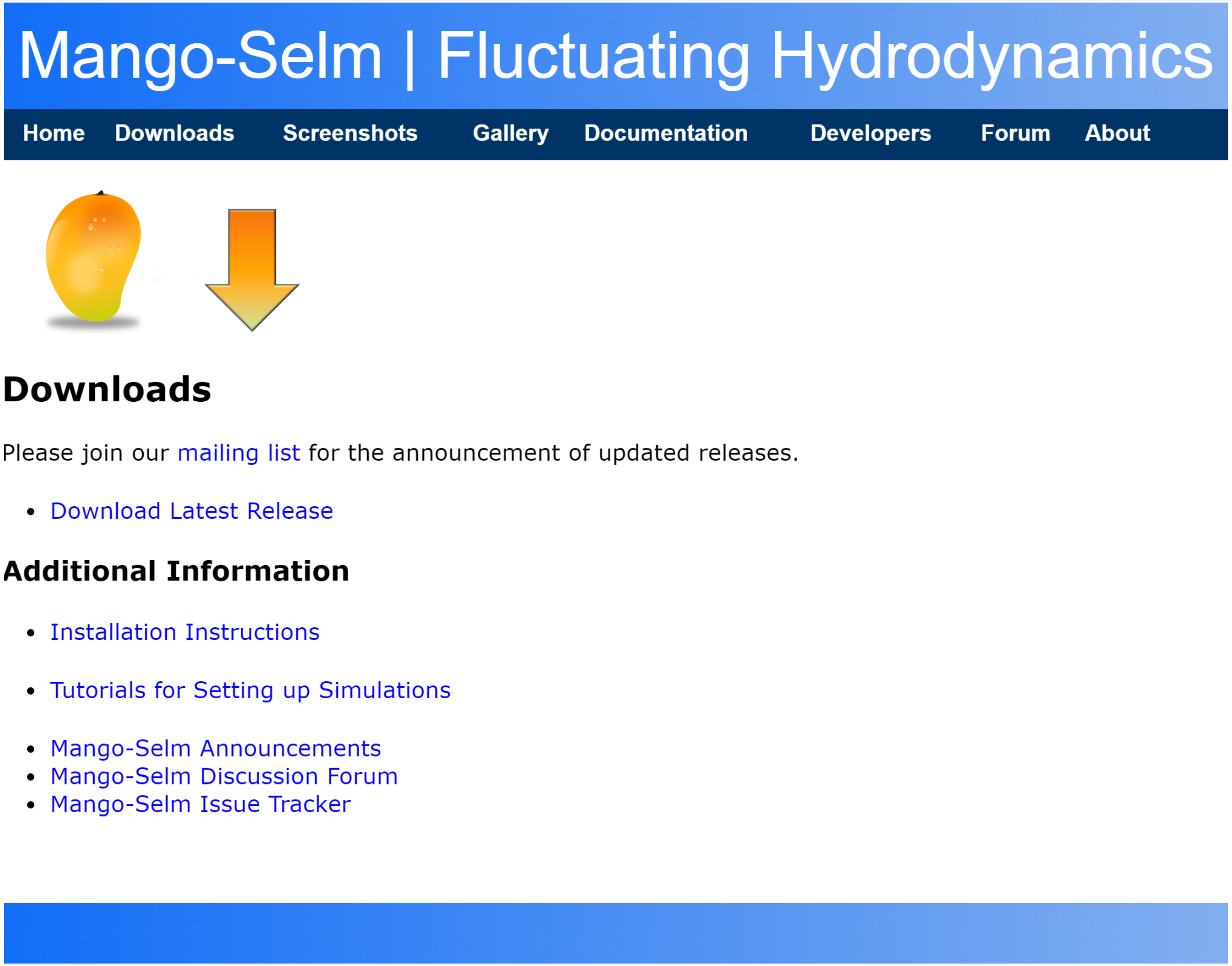
Jython Interactive Interpreter:

```
Jython Interactive Editor 1.0 : Implemented by Paul J. Atzberger, Copyright 2011.

======================================================
Startup Script for SELM Jython Intepreter

Written by Paul J. Atzberger
Date: March, 2011.
======================================================

Model Build Package 1 : Authored by Paul J. Atzberger : Version 1.0
Setup appears to have completed with no known errors.
>>> import pickle as p
>>> import os
>>> a = []
>>> a.append(1); a.append(2); a.append(3)
>>> a
[1, 2, 3]
>>>
```

Atzberger 2016

MANGO-SELM Simulation Software

MANGO-SELM – Download: http://mango-selm.org/



Downloads

Please join our mailing list for the announcement of updated releases.

- Download Latest Release

Additional Information

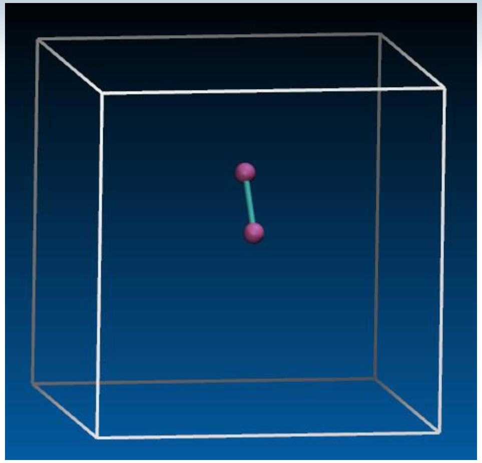
- Installation Instructions

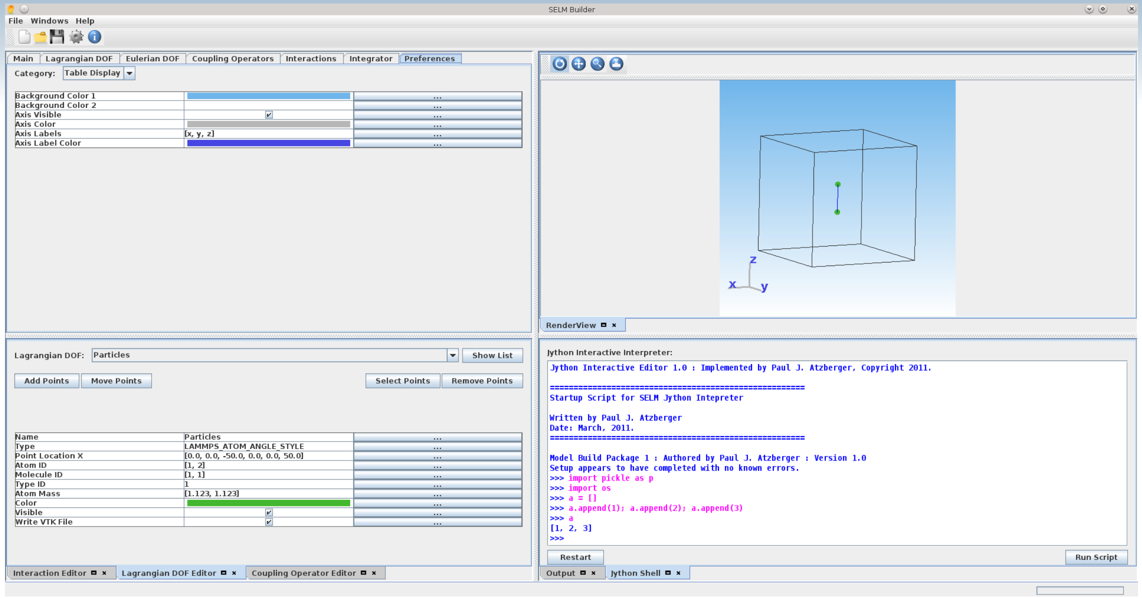
- Tutorials for Setting up Simulations

- Mango-Selm Announcements
- Mango-Selm Discussion Forum
- Mango-Selm Issue Tracker

MANGO-SELM Simulation Software

Demo Live: FENE_Dimer





Category: Table Display

Background Color 1	🟦	...
Background Color 2		...
Axis Visible	☑	...
Axis Color	▨	...
Axis Labels	[x, y, z]	...
Axis Label Color	🟦	...

Lagrangian DOF: Particles

Name	Particles	...
Type	LAMMPS_ATOM_ANGLE_STYLE	...
Point Location X	[0.0, 0.0, -50.0, 0.0, 0.0, 50.0]	...
Atom ID	[1, 2]	...
Molecule ID	[1, 1]	...
Type ID	1	...
Atom Mass	[1.123, 1.123]	...
Color	🟩	...
Visible	☑	...
Write VTK File	☑	...

Jython Interactive Interpreter:

```
Jython Interactive Editor 1.0 : Implemented by Paul J. Atzberger, Copyright 2011.

=======================================================
Startup Script for SELM Jython Intepreter

Written by Paul J. Atzberger
Date: March, 2011.
=======================================================

Model Build Package 1 : Authored by Paul J. Atzberger : Version 1.0
Setup appears to have completed with no known errors.
>>> import pickle as p
>>> import os
>>> a = []
>>> a.append(1); a.append(2); a.append(3)
>>> a
[1, 2, 3]
>>>
```

Steps:

1. Use File→Open project.
2. Load: FeneTest1/FeneModel.SELM_Builder_Project
3. Adjust coupling operator table to /common/CouplingOp_T_KERNEL_1.SELM_CouplingOperator_weightTable
4. Gear Icon → generate SELM simulation files.
5. Link executable in ln –s ../common/ SELM_LAMMPS_2.0.0a_serial_x86_Ubuntu run
6. run –in Fene_Dimer.LAMMPS_script
7. Generates output data → .dcd file.
8. Run ./vis_FENE.vmd to visualize the model.

Important files:

FENE_Dimer.LAMMPS_script
FENE_Dimer.LAMMPS_read_data
FENE_Dimer.SELM_Info
FENE_Dimer.SELM_InfoExtra
FENE_Dimer.SELM_params
FENE-bonds.SELM_Interaction

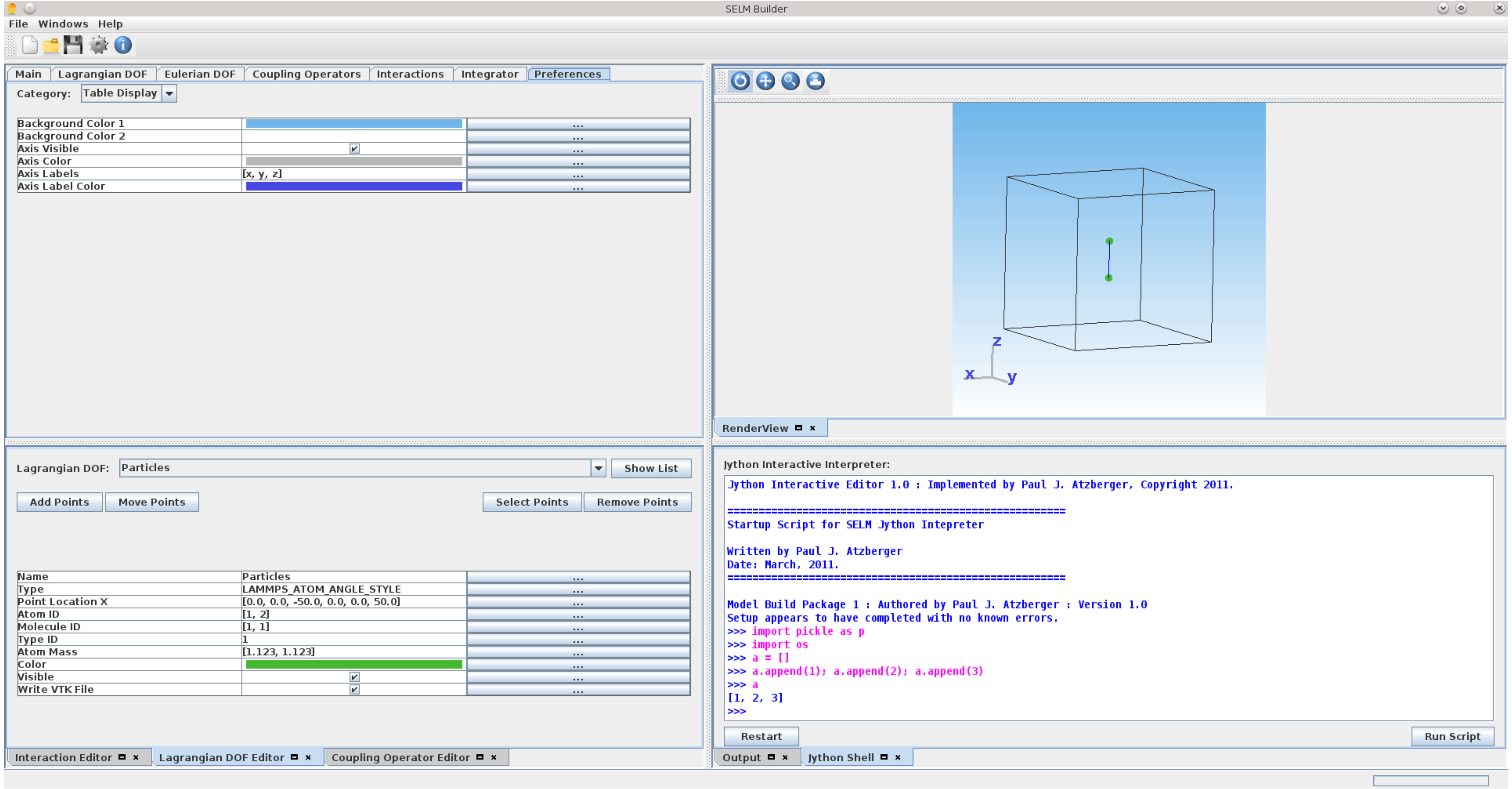
CouplingOp.SELM_CouplingOperator
CouplingOp_T_KERNEL_1.SELM_CouplingOperator_weightTable

LAMMPS_SHEAR_QUASI_STEADY1_FFTW3.SELM_Integrator
LAMMPS_SHEAR_UNIFORM1_FFTW3.SELM_Eulerian
Particles.SELM_Lagrangian

SELM_LAMMPS_serial_x86_Ubuntu
*.dcd
vis1.vmd

MANGO-SELM Simulation Software

Demo Live: FENE



Category: Table Display

Background Color 1	🟦	...
Background Color 2		...
Axis Visible	☑	...
Axis Color	▨	...
Axis Labels	[x, y, z]	...
Axis Label Color	🟦	...

Lagrangian DOF: Particles

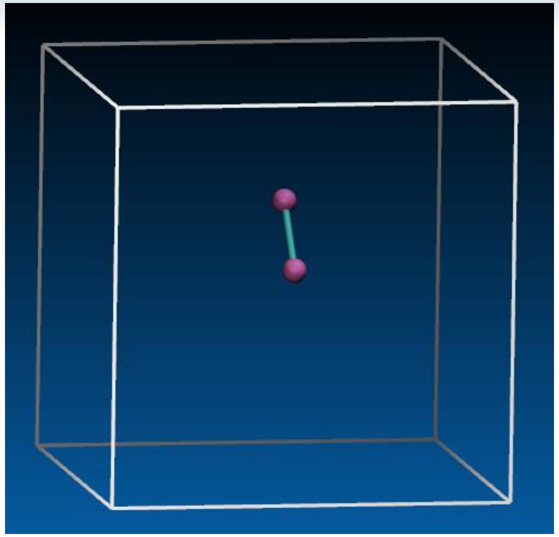
Name	Particles	...
Type	LAMMPS_ATOM_ANGLE_STYLE	...
Point Location X	[0.0, 0.0, -50.0, 0.0, 0.0, 50.0]	...
Atom ID	[1, 2]	...
Molecule ID	[1, 1]	...
Type ID	1	...
Atom Mass	[1.123, 1.123]	...
Color	🟩	...
Visible	☑	...
Write VTK File	☑	...

Jython Interactive Interpreter:

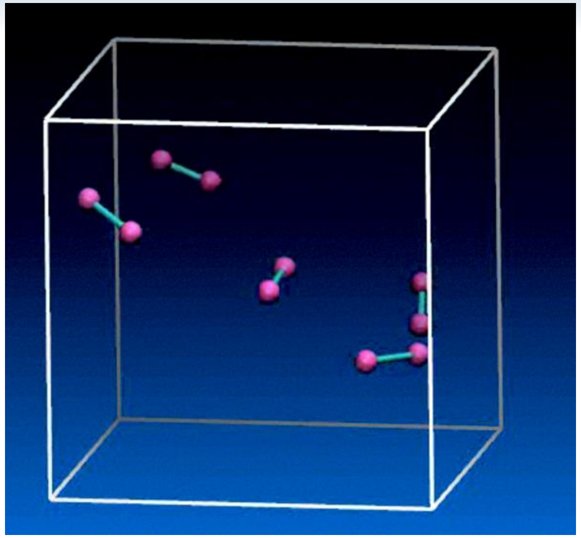
```
Jython Interactive Editor 1.0 : Implemented by Paul J. Atzberger, Copyright 2011.

=======================================================
Startup Script for SELM Jython Intepreter

Written by Paul J. Atzberger
Date: March, 2011.
=======================================================

Model Build Package 1 : Authored by Paul J. Atzberger : Version 1.0
Setup appears to have completed with no known errors.
>>> import pickle as p
>>> import os
>>> a = []
>>> a.append(1); a.append(2); a.append(3)
>>> a
[1, 2, 3]
>>>
```

MANGO-SELM Simulation Software

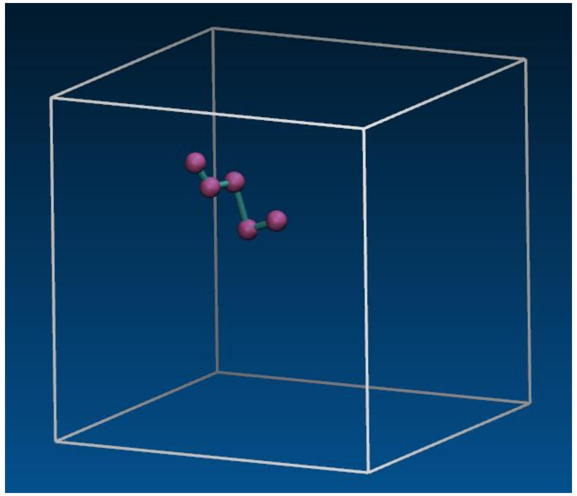
Other Demos:



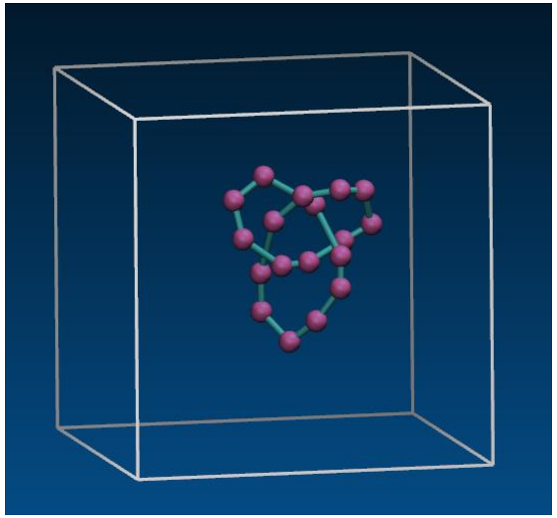
Sheared FENE Dimer



FENE Fluid



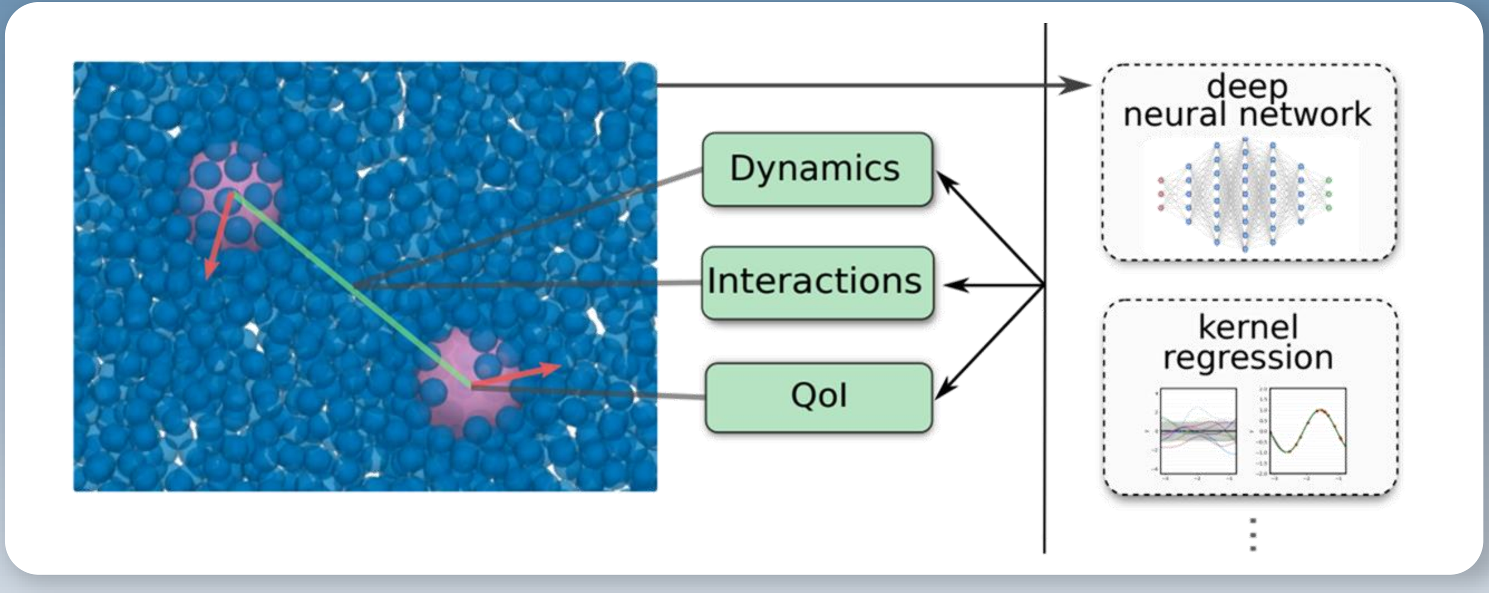
Polymer Chain

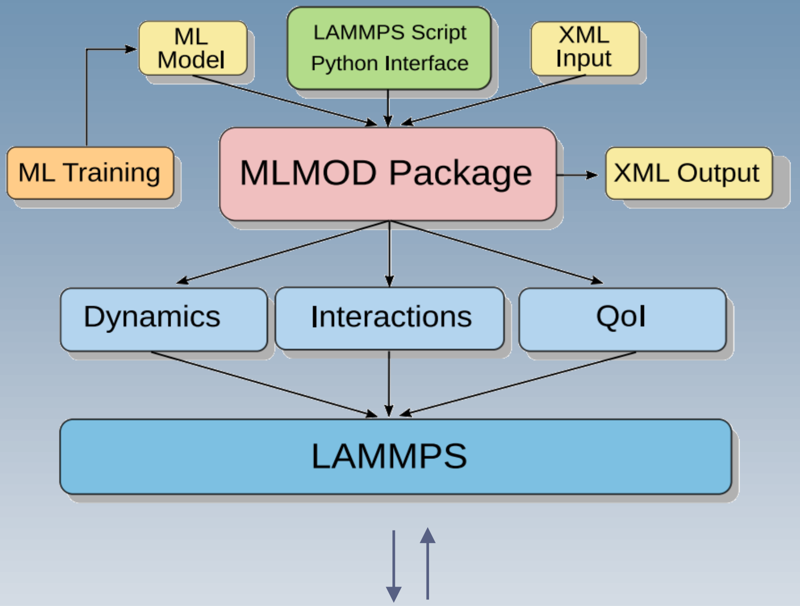


Polymer Knot

MLMOD Package

MLMOD Package: Machine Learning Methods for Data-Driven Modeling





LAMMPS

SELM
Fluctuating Hydrodynamics

LAMMPS Workshop
August 2021

Paul J. Atzberger
University of California Santa Barbara

DOE ASCR PhILMS
DE-SC0019246

NSF Grant
DMS - 1616353

http://software.atzberger.org

paper: https://arxiv.org/abs/2107.14362

MLMOD Package: Data-Driven Modeling

Software

MLMOD

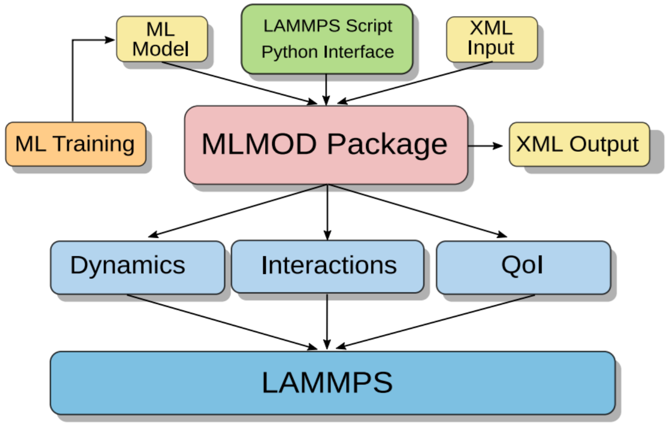
Machine Learning Modeling Package

C/C++ python PyTorch jupyter notebooks

http://software.atzberger.org

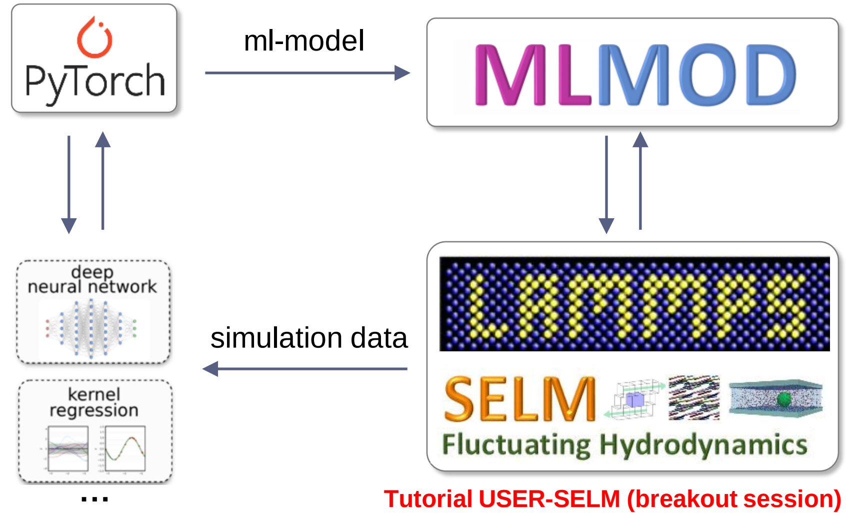
paper: https://arxiv.org/abs/2107.14362

Package Structure



Machine Learning (training)

Scientific Computation (simulation)

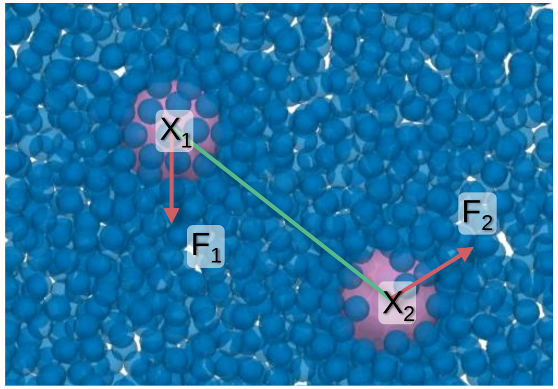


Tutorial USER-SELM (breakout session)

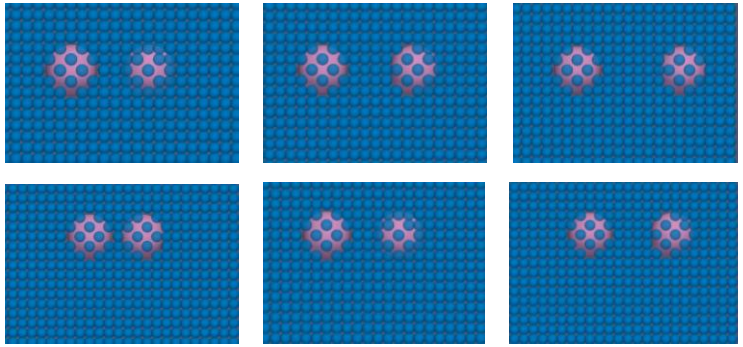
Friday, 8/13, 12:50pm EDT

Example use case

Colloid + Solvent System



Simulation Data



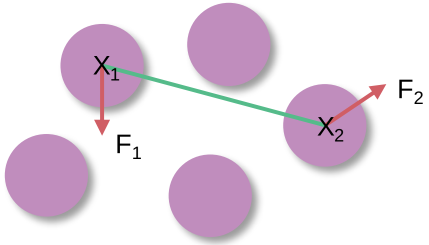
...

Reduced Model

$$\frac{d\mathbf{X}}{dt} = \mathbf{M}F + k_B T \nabla_X \cdot \mathbf{M} + \mathbf{F}_{thm}$$

$$\langle \mathbf{F}_{thm}(s)\mathbf{F}_{thm}(t)^T \rangle = 2k_B T \mathbf{M}\delta(t-s).$$

$\mathrm{M}=\mathrm{M}(\mathrm{X})$



DOE ASCR PhILMS DE-SC0019246

NSF Grant DMS - 1616353

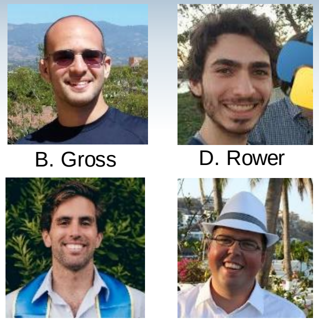
More Information:

http://software.atzberger.org

paper: https://arxiv.org/abs/2107.14362

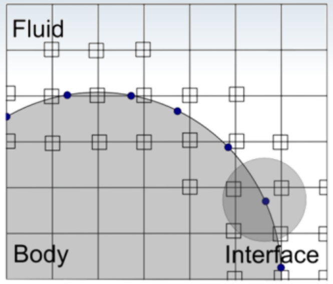
Summary / Conclusions

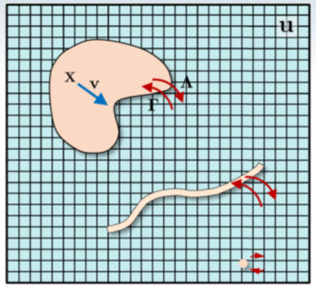
Conclusions

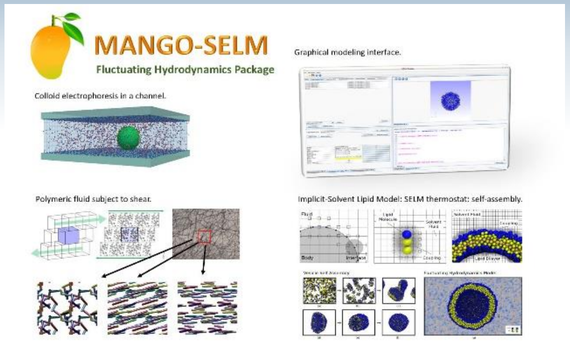

B. Gross D. Rower

M. Padidar J. Sigurdsson

UCSB Recent Student Collaborators






MANGO-SELM
Fluctuating Hydrodynamics Package
Colloid electrophoresis in a channel.
Graphical modeling interface.
Polymeric fluid subject to shear.
Implicit-Solvent Lipid Model: SELM thermostat: self-assembly.

Paper
2016 Atzberger & Sigurdsson

Summary

***Stochastic Immersed Boundary Methods** with numerical solvers preserving statistical mechanics properties.*
***Stochastic Eulerian Lagrangian Methods** for inertial and overdamped regimes, various boundary conditions.*
***Python interface** for setting up simulations, **LAMMPS molecular dynamics integration** (modeling, analysis).*
***Applications** in soft materials, complex fluids, rheology, microfluidics, biophysics, lipid bilayer membranes.*

Mango-Selm package for Fluctuating Hydrodynamics
http://atzberger.org

Sandia LAMMPS Molecular Dynamics

Papers

A Stochastic Immersed Boundary Method for Fluid-Structure Dynamics at Microscopic Length Scales,
P.J. Atzberger, P.R. Kramer, and C.S. Peskin, J. Comp. Phys., Vol. 224, Iss. 2, (2007).

Stochastic Eulerian Lagrangian Methods for Fluid Structure Interactions with Thermal Fluctuations,
P.J. Atzberger, J. of Comp. Phys., 230, pp. 2821--2837, (2011).

Surface Fluctuating Hydrodynamics Methods for the Drift-Diffusion Dynamics of Particles and Microstructures within Curved Fluid Interfaces,
D. Rower, M. Padidar, and P. J. Atzberger, arXiv:1906.01146, (2019).

Meshfree Methods on Manifolds for Hydrodynamic Flows on Curved Surfaces: A Generalized Moving Least-Squares (GMLS) Approach,
Gross B. J., Kuberry P. A., Trask N., Atzberger P. J., J. Comp. Phys., 409, 15 May (2020).

UCSB Recent Student Collaborators

B. Gross, M. Padidar, D. Rower, J. K. Sigurdsson.

Sandia Collaborators

N. Trask, P. Kuberry, J. Hu, C. Siefert, and others.

Funding

DOE ASCR CM4
DE-SC0009254

DOE ASCR PhILMS
DE-SC0019246

NSF Grant
DMS - 1616353

NSF CAREER Grant
DMS-0956210

More information: http://atzberger.org/

Publications

A Stochastic Immersed Boundary Method for Fluid-Structure Dynamics at Microscopic Length Scales,
P.J. Atzberger, P.R. Kramer, and C.S. Peskin, J. Comp. Phys., Vol. 224, Iss. 2, (2007). http://dx.doi.org/10.1016/j.jcp.2006.11.015

Stochastic Eulerian Lagrangian Methods for Fluid Structure Interactions with Thermal Fluctuations,
P.J. Atzberger, J. of Comp. Phys., 230, pp. 2821--2837, (2011). http://dx.doi.org/10.1016/j.jcp.2010.12.028

Fluctuating Hydrodynamics Methods for Dynamic Coarse-Grained Implicit-Solvent Simulations in LAMMPS, Y. Wang, J. K. Sigurdsson, and P.J. Atzberger, SIAM J. Sci. Comp. (accepted), (2016). https://doi.org/10.1137/15M1026390

MLMOD Package: Machine Learning Methods for Data-Driven Modeling in LAMMPS, P.J. Atzberger, arXiv (2021), https://arxiv.org/abs/2107.14362.

Surface Fluctuating Hydrodynamics Methods for the Drift-Diffusion Dynamics of Particles and Microstructures within Curved Fluid Interfaces,
D. Rower, M. Padidar, and P. J. Atzberger, arXiv:1906.01146, (2019). https://arxiv.org/abs/1906.01146

Meshfree Methods on Manifolds for Hydrodynamic Flows on Curved Surfaces: A Generalized Moving Least-Squares (GMLS) Approach,
Gross B. J., Kuberry P. A., Trask N., Atzberger P. J., J. Comp. Phys., 409, 15 May (2020). https://doi.org/10.1016/j.jcp.2020.109340

Systematic Stochastic Reduction of Inertial Fluid-Structure Interactions subject to Thermal Fluctuations, G. Tabak and P.J. Atzberger, SIAM J. Appl. Math., 75(4), 1884–1914, (2015). http://dx.doi.org/10.1137/15M1019088

Dynamic Implicit-Solvent Coarse-Grained Models of Lipid Bilayer Membranes : Fluctuating Hydrodynamics Thermostat, Y. Wang, J. K. Sigurdsson, E. Brandt, and P.J. Atzberger, Phys. Rev. E 88, 023301, (2013). http://dx.doi.org/10.1103/PhysRevE.88.023301

Spatially Adaptive Stochastic Methods for Fluid-Structure Interactions Subject to Thermal Fluctuations in Domains with Complex Geometries,
P. Plunkett, J. Hu, C. Siefert, P.J. Atzberger, Journal of Computational Physics, Vol. 277, 15 Nov. 2014, pg. 121--137, (2014).
http://dx.doi.org/10.1016/j.jcp.2014.07.051

Incorporating Shear into Stochastic Eulerian Lagrangian Methods for Rheological Studies of Complex Fluids and Soft Materials,
P.J. Atzberger, Physica D, Vol. 265, pg. 57–70, (2013). http://dx.doi.org/10.1016/j.physd.2013.09.002

For additional information: http://atzberger.org/