Implementation of a meshless MLS scheme for simulations of suspension flows

Amanda Howard[1], Martin Maxey[1], and Nathaniel Trask[2]

[1]Brown University, [2]Sandia National Laboratories

Introduction

Suspension flows represent a complex dynamical system due to the nonlinear coupling of the flow and the particle positions, which in turn determine the flow. Even in the zero Reynolds' number Stokes flow limit, where the flow is linear and reversible, the presence of particles induces chaos and irreversibility [1, 2, 3]. Here, we present work on a novel meshless particle method using Generalized Moving Least Squares (GMLS) for high-order simulations of suspension Stokes flows with arbitrary particle shapes and complex, deforming domain boundaries [4, 5, 6]. Applications of such flows occur both in nature and in industry, and include microfluidic systems for cell sorting and inertial focusing [7], the microstructure that appears due to particle stresses with spherical particles at high volume fraction [8], and particle migration and segregation by size or density [9, 10, 11].

Generalized Moving Least Squares (GMLS)

We want to approximate a function u near a point x_i. Define

$$u_h(x; x_i) = q^*(x_i)$$

where q^* is the solution to a weighted l_2 optimization problem:

$$q^* = \arg\min_{q \in \pi_m} \sum_{i=1}^{N} [u(x_j) - q(x_j)]^2 W_{ij}$$

Operator D^α is found by applying D^α to the reconstruction:

$$D^\alpha \mathbf{u}_i \approx D_h^\alpha \mathbf{u}_i := D^\alpha q^*(\mathbf{x}_i)$$

BROWN

GMLS for Stokes Flow

- Recently developed Staggered MLS scheme for numerical solutions of Stokes flow
- Polynomial interpolants are used to represent the flow using least squares minimization
- High order accurate (4th or 6th order easily obtained by changing the order of the polynomial interpolants.)
- Force-free and torque-free colloids with position $\mathbf{X}_i$, orientation Θ_i, and boundary $\partial\Omega_i$.

$$\begin{cases} -\nu\nabla^2\mathbf{u} + \nabla p = \mathbf{f} & \mathbf{x} \in \Omega \\ \nabla \cdot \mathbf{u} = 0 & \mathbf{x} \in \Omega \\ \mathbf{u} = \mathbf{w} & \mathbf{x} \in \partial\Omega \\ \mathbf{u} = \dot{\mathbf{X}}_i + \dot{\Theta}_i \times (\mathbf{x} - \mathbf{X}_i) & \mathbf{x} \in \partial\Omega_i \end{cases}$$

- Choose $\mathbf{u}$ in the space of divergence free vector fields.
- Each operator found as the minimization over the space of divergence-free polynomials in $\mathbb{R}^3$ of degree n.

Implementation

- Because of the form of minimizing a polynomial, we can write each quantity as a linear combination, forming a global matrix:

$$\nabla_h^2 p_i = \sum_{j \in supp(W_{ij})} \alpha_{ij}^1 p_j$$

$$\nabla_h p_i = \sum_{j \in supp(W_{ij})} \alpha_{ij}^2 p_j$$

$$\nabla \times \nabla \times_h \mathbf{u}_i = \sum_{j \in supp(W_{ij})} \alpha_{ij}^3 \mathbf{u}_j$$

- Dirichlet boundary conditions are inforced on the global matrix.
- Lack of symmetry makes it difficult to provide divergence free. The flow is only divergence free in the local polynomial reconstruction.
- Currently implementing a massively parallel code in collaboration with the Compadre project at Sandia National Laboratories
- Uses the Trilinos linear algebra package and Nanoflann near-neighbor search
- To do: implement more efficient preconditioners

Results

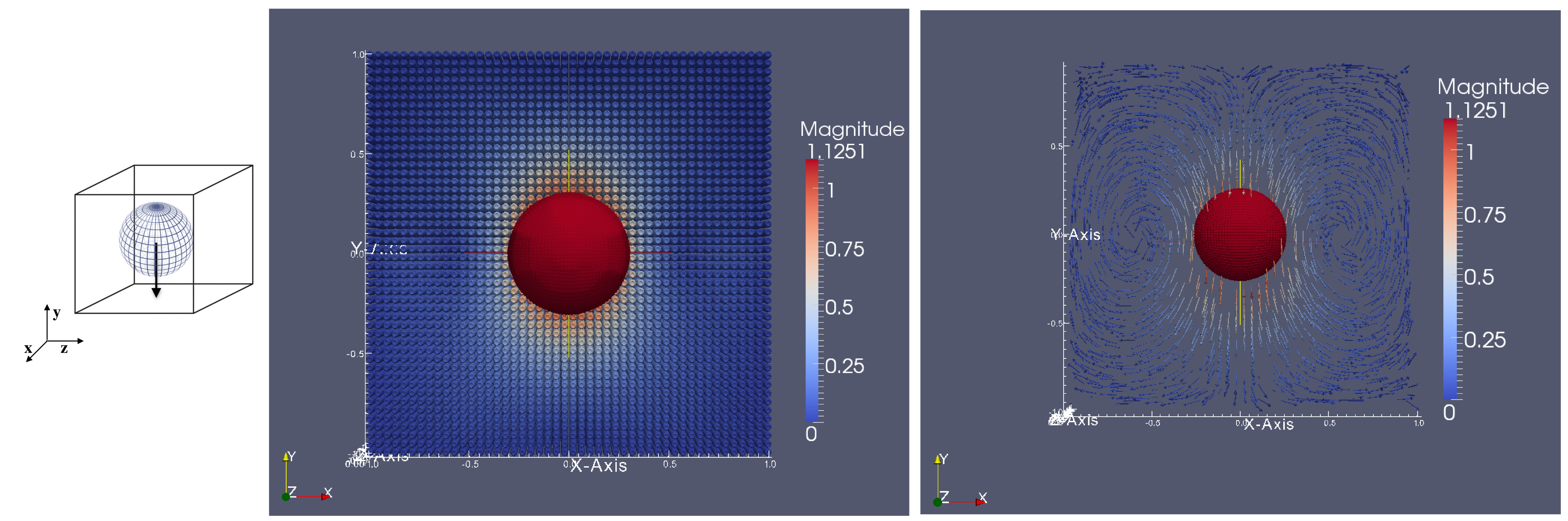

Figure 1: GMLS results for a solid sphere settling in a unit box with Dirichlet boundary conditions. Left: diagram of setup. Middle: locations of GMLS collocation points, colored by the velocity magnitude. Right: velocity vector field.

Conclusions

- Developed a high-order, scalable meshfree discretization for solving systems of partial differential equations using Generalized Moving Least Squares (GMLS) polynomial reconstructions to provide a computationally efficient method for use with general boundary conditions while maintaining stability.
- Methods are fast and stable and use only the graph of neighbor connectivity
- Use of the scientific computing packages available in the Trilinos Project allows for solving large-scale, complex problems in parallel.

References

[1] Pine, D. J., Gollub, J. P., Brady, J. F., and Leshansky, A. M. Dec 2005 *Nature* **438(7070)**, 997–1000.
[2] Metzger, B., Pham, P., and Butler, J. E. May 2013 *Phys. Rev. E* **87(5)**, 052304.
[3] Corté, L., Chaikin, P. M., Gollub, J. P., and Pine, D. J. Mar 2008 *Nat. Phys.* **4(5)**, 420–424.
[4] Trask, N., Maxey, M., and Hu, X. (2017) *J. Comput. Phys.*
[5] Trask, N., Perego, M., and Bochev, P. Jan 2017 *SIAM J. Sci. Comput.* **39(2)**, A479–A502.
[6] Mirzaei, D., Schaback, R., and Dehghan, M. Jul 2012 *IMA J. Numer. Anal.* **32(3)**, 983–1000.
[7] Di Carlo, D. (2009) *Lab Chip* **9**, 3038–3046.
[8] Morris, J. F. mar 2009 *Rheol. Acta* **48(8)**, 909–923.
[9] Kanehl, P. and Stark, H. Jun 2015 *J. Chem. Phys.* **142(21)**, 214901.
[10] Lyon, M. K. and Leal, L. G. May 1998 *J. Fluid Mech.* **363**, 57–77.
[11] Batchelor, G. K. and Van Rensburg, J. (1986) *J. Fluid Mech.* **166**, 379–407.

Acknowledgements

This work is supported by the NSF Graduate Research Fellowship under Grant No. DGE 1058262 and the U.S. Department of Energy Office of Science under Award Number DE-SC0009247.