The Journal of Open Source Software

PyStokes: phoresis and Stokesian hydrodynamics in Python

Rajesh Singh[1] **and Ronojoy Adhikari**[1]

1 DAMTP, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge CB3 0WA, UK

DOI: 10.21105/joss.02318

Software

- Review
- Repository
- Archive

Editor: Alice Harpole

Reviewers:

- @fcooper8472
- @khinsen

Submitted: 06 June 2020
Published: 30 June 2020

Summary

PyStokes is a Python library for studying phoretic and hydrodynamic interactions between spherical particles when these interactions can be described by the solutions of, respectively, the Laplace and Stokes equations. The library has been specifically designed for studying these interactions in suspensions of active particles, which are distinguished by their ability to produce flow, and thus motion, in the absence of external forces or torques. Such particles are endowed with a mechanism to produce hydrodynamic flow in a thin interfacial layer, which may be due to the motion of cilia, as in microorganisms (Brennen & Winet, 1977) or osmotic flows of various kinds in response to spontaneously generated gradients of phoretic fields (Ebbens & Howse, 2010). The latter, often called autophoresis, is a generalisation of well-known phoretic phenomena including, *inter alia*, electrophoresis (electric field), diffusiophoresis (chemical field) and thermophoresis (temperature field) that occur in response to externally imposed gradients of phoretic fields (Anderson, 1989).

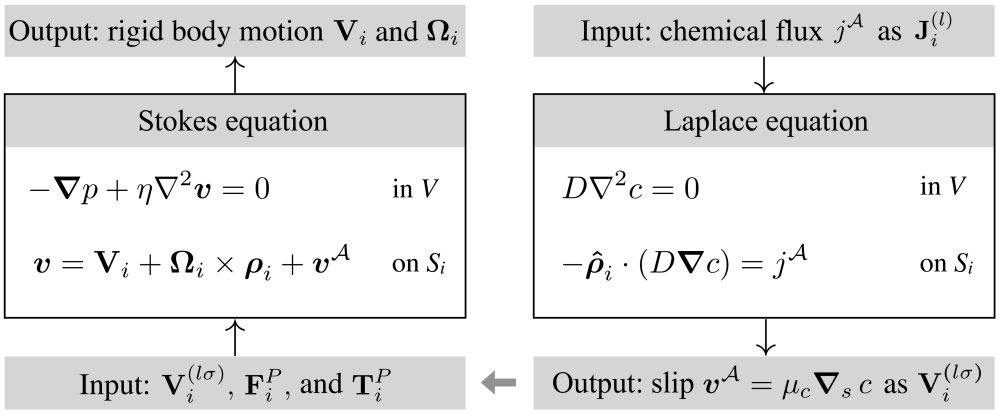

Figure 1: Input and output structure of PyStokes to determine the hydrodynamic and phoretic interactions between active particles in a three-dimensional domain V. The equations are coupled by active boundary conditions on the surface S_i of the particles. Particle indices are $i = 1, \ldots, N$ and harmonic indices are $l = 1, 2, \ldots$ and $\sigma = s, a, t$ (see text).

Hydrodynamic and phoretic interactions between active particles in a viscous fluid are central to the understanding of their collective dynamics. Under experimentally relevant conditions, the motion of the fluid is governed by the Stokes equation and that of the phoretic field, if one is present, by the Laplace equation. The "activity" appears in these equations as boundary conditions on the particle surfaces that prescribe the slip velocity in the Stokes equation and flux of the phoretic field in the Laplace equation (see Figure 1). The slip velocity and the

phoretic flux are related by a linear constitutive law that can be derived from a detailed analysis of the boundary layer physics (Anderson, 1989). The Stokes and Laplace equations are coupled by this linear constitutive law only at the particle boundaries. The linearity of the governing equations and the coupling boundary conditions allows for a formally exact solution of the problem of determining the force per unit area on the particle surfaces. This formally exact solution can be approximated to any desired degree of accuracy by a truncated series expansion in a complete basis of functions on the particle boundaries. This, in turn, leads to an efficient and accurate numerical method for computing hydrodynamic and phoretic interactions between active particles.

In addition to the joint computation of phoretic and hydrodynamic interactions, the PyStokes library can be used to compute the hydrodynamically interacting motion of squirming particles where the slip is specified independently of a phoretic field, or the dynamics of passive suspensions where the slip vanishes and forces and torques are prescribed. The PyStokes library can also compute hydrodynamically correlated Brownian motion, and thus, allows the study of the interplay between passive, active, and Brownian contributions to motion.

The PyStokes library has been used to model suspensions of microorganisms (Bolitho, Singh, & Adhikari, 2020; Singh & Adhikari, 2016), synthetic autophoretic particles (Singh & Adhikari, 2016; R. Singh et al., 2019) and self-propelling droplets (Thutupalli, Geyer, Singh, Adhikari, & Stone, 2018). Our software implementation uses a polyglot programming approach that combines the readability of Python with the speed of Cython and retains the advantages of a high-level, dynamically typed, interpreted language without sacrificing performance.

Methods

Our method relies on the reduction of a linear elliptic partial differential equation (PDE) to systems of linear algebraic equations using the following steps:

$$\text{elliptic PDE} \xrightarrow{1} \text{boundary integral} \xrightarrow{2} \text{spectral expansion} \xrightarrow{3} \text{Ritz-Galerkin discretization} \xrightarrow{4} \text{truncation}$$

Figure 2: Key mathematical steps underpinning the PyStokes codebase.

The *first* step is the representation of the solution of an elliptic PDE in a three-dimensional volume V as an integral over the boundary of the surface S (Cheng & Cheng, 2005; Ladyzhenskaia, 1969; Muldowney & Higdon, 1995; Odqvist, 1930; Pozrikidis, 1992; Singh, Ghose, & Adhikari, 2015; Youngren & Acrivos, 1975; Zick & Homsy, 1982). For the Laplace equation, this is the classical theorem of Green (Jackson, 1962); for the Stokes equation, it is the generalization obtained by Lorentz (Ladyzhenskaia, 1969; Lorentz, 1896; Odqvist, 1930). The integral representation leads to a linear integral equation that provides a functional relation between the field and its flux on S. Thus, if the surface flux in the Laplace equation is specified, the surface concentration is determined by the solution of the Laplace boundary integral equation. Similarly, if the surface velocity in the Stokes equation is specified, the surface traction is determined by the solution of the Stokes boundary integral equation. This transformation of the governing PDE is the most direct way of relating boundary conditions (surface flux, slip velocities) to boundary values (surface concentration, surface traction). It reduces the dimensionality of the problem from a three-dimensional one in V to a two-dimensional one on S. The *second* step is the spectral expansion of the field and its flux in terms of global basis functions on S. We use the geometry-adapted tensorial spherical harmonics, which provide a unified way of expanding both scalar and vector quantities on the surface of a sphere. These functions are both complete and orthogonal and provide representations of the three-dimensional rotation group (Hess, 2015). Thus, symmetries of the active boundary conditions can be represented straightforwardly and transparently. The *third* step is the discretization

of the integral equation using the procedure of Ritz and Galerkin (Boyd, 2000; Finlayson & Scriven, 1966), which reduces it to an infinite-dimensional self-adjoint linear system in the expansion coefficients. This exploits the orthogonality of the basis functions on the sphere. The matrix elements of the linear system can be evaluated analytically in terms of the Green's functions of the respective elliptic equations. The *fourth* step is the truncation of the infinite-dimensional linear system to a finite-dimensional one that can be solved by standard methods of linear algebra adapted for self-adjoint systems (Saad, 2003). The analytical solution can be obtained by Jacobi iteration, which is equivalent to Smoluchowski's method of reflection. Numerical solutions can be obtained by the conjugate gradient method, at a cost quadratic in the number of unknowns. From this solution, we can reconstruct the field and the flux on the boundary, use these to determine the fields in the bulk, and from there, compute derived quantities.

The above steps have been elaborated in several papers (Singh & Adhikari, 2016, 2017, 2018; Singh et al., 2019, 2015) and we do not repeat them in detail here. Briefly, the expansion coefficients of the slip can be either specified or obtained as a solution of Laplace equation. Once the coefficients are determined, the following equation are solved numerically to obtain velocity and angular velocity of the i-th particle

$$\mathbf{V}_i = \boldsymbol{\mu}_{ij}^{TT} \cdot \mathbf{F}_j^P + \boldsymbol{\mu}_{ij}^{TR} \cdot \mathbf{T}_j^P + \sum_{l\sigma=1s}^{\infty} \boldsymbol{\pi}_{ij}^{(T,l\sigma)} \cdot \mathbf{V}_j^{(l\sigma)}, \qquad \boldsymbol{\mu}_{ij}^{\alpha\beta} \;:\; \text{mobility matrices},$$

$$\boldsymbol{\Omega}_i = \underbrace{\boldsymbol{\mu}_{ij}^{RT} \cdot \mathbf{F}_j^P + \boldsymbol{\mu}_{ij}^{RR} \cdot \mathbf{T}_j^P}_{\text{Passive}} + \sum_{l\sigma=1s}^{\infty} \underbrace{\boldsymbol{\pi}_{ij}^{(R,l\sigma)} \cdot \mathbf{V}_j^{(l\sigma)}}_{\text{Active}}, \qquad \boldsymbol{\pi}_{ij}^{(\alpha,l\sigma)} \;:\; \text{propulsion tensors}.$$

In the above, $\alpha, \beta = (T, R)$, repeated particle index j is summed, and

$$\mathbf{F}_i^P : \text{body force}, \quad \mathbf{T}_i^P : \text{body torque}, \quad \mathbf{V}_i^{(l\sigma)} : l\sigma\text{-th expansion coefficients of active slip}.$$

Thus, hydrodynamic interactions between particles with no-slip boundary conditions can be computed entirely in terms of mobility matrices, as implemented in existing numerical libraries (Hinsen, 1995; Ichiki, 2002), to study suspensions of passive particles. The active contributions due to the slip boundary condition is given in terms of propulsion tensors (Singh et al., 2015). To the best of our knowledge, PyStokes is the only numerical implementation of propulsion tensors to model suspensions of active particles.

To summarize, the principal features that set our method apart are (a) the restriction of independent fluid and phoretic degrees of freedom to the particle boundaries (b) the freedom from grids, both in the bulk of the fluid and on the particle boundaries and (c) the ability to handle, within the same numerical framework, a wide variety of geometries and boundary conditions, including unbounded volumes, volumes bounded by plane walls or interfaces, periodic volumes and, indeed, any geometry-boundary condition combination for which the Green's functions of the governing equations are simply evaluated.

The PyStokes library can be instantiated in the following way to

- obtain phoretic field created by active particles at a given set of points

```
phoreticField = pystokes.phoreticUnbounded.Field(radius=1, particles=1,
                phoreticConstant=1, gridpoints=4096)
```

- evaluate fluid flow created by active particles at a given set of points

```
Flow = pystokes.unbounded.Flow(radius=1, particles=1, viscosity=1,
       gridpoints=4096)
```

- determine phoretic field at surface of active particles

```
phoresis = pystokes.phoreticUnbounded.Phoresis(radius=1, particles=1024,
            phoreticConstant=1)
```

- compute rigid body motion of hydrodynamically interacting particles

```
Rbm = pystokes.unbounded.Rbm(radius=1, particles=1024, viscosity=1)
```

The above instantiation can then be used to compute `Flow` and `Rbm` due to body forces, body torques, and each irreducible mode of the surface slip in various geometries of Stokes flow, by replacing `unbounded` with `wallBounded`, `periodic`, etc. `pystokes.forceFields` contains an implementation of force fields commonly used in colloidal systems for completeness. The arXiv preprint (Singh & Adhikari, 2019) of this article contains more detailed documentation and examples.

Acknowledgements

We thank A Banerjee, ME Cates, S Date, A Donev, E Eiser, D Frenkel, S Ghose, R Goldstein, J Hinch, A Laskar, AJC Ladd, RK Manna, I Pagonabarraga, DJ Pine, T Pradeep, R Simon, HA Stone, G Subramanian, PB Sunil Kumar, and S Thutupalli for useful discussions; Rajeev Singh and Abhrajit Laskar for code contributions in the initial stages of development. This work was funded in parts by the European Research Council under the EU's Horizon 2020 Program, Grant No. 740269; a Royal Society-SERB Newton International Fellowship to RS; and an Early Career Grant to RA from the Isaac Newton Trust.

References

Anderson, J. L. (1989). Colloid transport by interfacial forces. *Annu. Rev. Fluid Mech.*, *21*(1), 61–99. doi:10.1146/annurev.fl.21.010189.000425

Bolitho, A., Singh, R., & Adhikari, R. (2020). Periodic orbits of active particles induced by hydrodynamic monopoles. *Phys. Rev. Lett.*, *124*(8), 088003. doi:10.1103/PhysRevLett.124.088003

Boyd, J. P. (2000). *Chebyshev and fourier spectral methods*. Dover. Retrieved from https://store.doverpublications.com/0486411834.html

Brennen, C., & Winet, H. (1977). Fluid mechanics of propulsion by cilia and flagella. *Annu. Rev. Fluid Mech.*, *9*(1), 339–398. doi:10.1146/annurev.fl.09.010177.002011

Cheng, A. H.-D., & Cheng, D. T. (2005). Heritage and early history of the boundary element method. *Eng. Anal. Bound. Elem.*, *29*(3), 268–302. doi:10.1016/j.enganabound.2004.12.001

Ebbens, S. J., & Howse, J. R. (2010). In pursuit of propulsion at the nanoscale. *Soft Matter*, *6*(4), 726–738. doi:10.1039/B918598D

Finlayson, B. A., & Scriven, L. E. (1966). The method of weighted residuals - a review. *Appl. Mech. Rev*, *19*(9), 735–748. Retrieved from http://faculty.washington.edu/finlayso/MWR-AReview.pdf

Hess, S. (2015). *Tensors for physics*. Springer. doi:10.1007/978-3-319-12787-3

Hinsen, K. (1995). HYDROLIB: A library for the evaluation of hydrodynamic interactions in colloidal suspensions. *Comp. Phys. Commun.*, *88*(2), 327–340. doi:10.1016/0010-4655(95)00029-F

Ichiki, K. (2002). libstokes. *GitHub repository*. https://github.com/kichiki/libstokes; GitHub.

Jackson, J. D. (1962). *Classical electrodynamics*. Wiley. Retrieved from http://as.wiley.com/WileyCDA/WileyTitle/productCd-047130932X.html

Ladyzhenskaia, O. A. (1969). *The mathematical theory of viscous incompressible flow*. Mathematics and its applications. Gordon; Breach. Retrieved from https://books.google.co.in/books?id=qVXvAAAAMAAJ

Lorentz, H. A. (1896). A general theorem concerning the motion of a viscous fluid and a few consequences derived from it. *Versl. Konigl. Akad. Wetensch. Amst*, *5*, 168–175.

Muldowney, G. P., & Higdon, J. J. L. (1995). A spectral boundary element approach to three-dimensional Stokes flow. *J. Fluid Mech.*, *298*, 167–192. doi:10.1017/S0022112095003260

Odqvist, F. K. G. (1930). Über die Bandwertaufgaben der Hydrodynamik zäher Flüssig-keiten. *Mathematische Zeitschrift*, *32*, 329–375. doi:10.1007/BF01194638

Pozrikidis, C. (1992). *Boundary integral and singularity methods for linearized viscous flow*. Cambridge University Press. doi:10.1017/CBO9780511624124

Saad, Y. (2003). *Iterative methods for sparse linear systems*. SIAM. doi:10.1137/1.9780898718003.bm

Singh, R., & Adhikari, R. (2016). Universal hydrodynamic mechanisms for crystallization in active colloidal suspensions. *Phys. Rev. Lett.*, *117*(22), 228002. doi:10.1103/PhysRevLett.117.228002

Singh, R., & Adhikari, R. (2017). Fluctuating hydrodynamics and the Brownian motion of an active colloid near a wall. *Eur. J. Comp. Mech*, *26*(1-2), 78–97. doi:10.1080/17797179.2017.1294829

Singh, R., & Adhikari, R. (2018). Generalized Stokes laws for active colloids and their applications. *J. Phys. Commun.*, *2*(2), 025025. doi:10.1088/2399-6528/aaab0d

Singh, R., & Adhikari, R. (2019). Hydrodynamic and phoretic interactions of active particles in Python. *arXiv:1910.00909*. Retrieved from https://arxiv.org/abs/1910.00909

Singh, R., Adhikari, R., & Cates, M. E. (2019). Competing chemical and hydrodynamic interactions in autophoretic colloidal suspensions. *J. Chem. Phys.*, *151*, 044901. doi:10.1063/1.5090179

Singh, R., Ghose, S., & Adhikari, R. (2015). Many-body microhydrodynamics of colloidal particles with active boundary layers. *J. Stat. Mech*, *2015*(6), P06017. doi:10.1088/1742-5468/2015/06/p06017

Thutupalli, S., Geyer, D., Singh, R., Adhikari, R., & Stone, H. A. (2018). Flow-induced phase separation of active particles is controlled by boundary conditions. *Proc. Natl. Acad. Sci.*, *115*(21), 5403–5408. doi:10.1073/pnas.1718807115

Youngren, G., & Acrivos, A. (1975). Stokes flow past a particle of arbitrary shape: A numerical method of solution. *J. Fluid Mech.*, *69*(02), 377–403. doi:10.1017/S0022112075001486

Zick, A. A., & Homsy, G. M. (1982). Stokes flow through periodic arrays of spheres. *J. Fluid Mech.*, *115*, 13–26. doi:10.1017/S0022112082000627