Statement of Research Interest

Saibal De

Introduction

Over the last decades, first-principle physics-based models have gained recognition in many natural sciences and engineering fields. Simulations utilizing these models make fewer simplifying assumptions about the underlying system, leading to highly accurate predictions. However, despite this advantage of high-fidelity outputs, we can rarely employ these models in practice due to their high computational cost.

A key aspect of many computational research projects is reducing this cost of full-fidelity simulations. We have developed numerous approaches to meet this challenge:

- **Computing Hardware:** Our computing nodes possess increasingly powerful processors and highly parallel accelerators (e.g., GPUs). High-speed communication channels also enable us to group these nodes into a single computing network, pushing the limit on the size of a problem we can solve even further.

- **Scalable Programming:** Software development has kept up pace with the hardware improvements. MPI, the de-facto standard for programming distributed machines, was introduced almost three decades ago. Since then, we have developed many domain-specific state-of-the-art open-source distributed libraries (e.g., PETSc and Trilinos for sparse linear algebra). These libraries allow us to design scalable programs that run on thousands of nodes, reducing the computational time by dividing a large problem into small manageable chunks.

- **Fast Algorithms:** We can often take advantage of the mathematical structure in a problem and fundamentally reduce computational complexity. For instance, the fast multipole method (FMM) can reduce the cost of computing gravitational potentials in an n-body problem by a full order of magnitude [1]. The development of such fast algorithms is essential for reducing the computational cost of full-scale simulations.

- **Novel Computing Paradigms:** Very recently, quantum computing has gained recognition for its (theoretical) ability to outperform even a traditional supercomputer at specific tasks. In the context of scientific computing, the potential for this technology is largely unexplored — but it will likely enable us to accelerate full-scale simulations beyond classical limits.

- **Reduced-Order Modeling:** Many learning applications, such as optimization, control, and uncertainty quantification, require thousands of data points. In these scenarios, and especially for time-critical online learning, it is impossible to run full-fidelity simulations. Reduced models provide a viable alternative in many of these tasks.

In my research, I develop physics-based simulation frameworks for physical phenomena with potential applications in mathematical microbiology and terramechanics/vehicle mobility. In the rest of this statement, I describe my current projects, emphasizing some of the approaches described above. I also outline how I plan to extend and combine these projects.

Fast Algorithms for Solving Boundary Integral Equations

A wide range of problems in computational physics and biology involves the numerical solution of constant-coefficient partial differential equations (PDEs), e.g., Stokes equation for bacterial locomotion and Maxwell's equations for electromagnetic scattering. Solving these PDEs using standard discretization methods (i.e., finite element methods), especially in complex geometries, is challenging. I employ an alternate approach in my research by recasting these partial differential equations as boundary integral equations (BIEs). This approach has several advantages:

- The integral equations are defined only on the boundary of the domain, which reduces the dimensionality of the problem by one, and significantly reduces the number of unknowns.
- Since we only need to discretize the boundary and not the entire domain, integral equations are ideal for handling complex, multi-connected geometries. Also, in moving geometry, we rarely, if ever, need to re-mesh.
- Linear systems obtained from discretizing an integral equation are as well-conditioned as the physics allows, leading to highly accurate computations.

We can often reduce the computational cost even further by taking advantage of symmetries in the problem. For instance, [2] developed a robust numerical framework for Laplace and Helmholtz BIEs on axisymmetric geometries.

In collaboration with Shravan Veerapaneni (my Ph.D. advisor, University of Michigan) and Bogdan Vioreanu, I extended the aforementioned scheme to Stokes BIEs. I utilize the Fourier series to decompose a single BIE on a two-dimensional axisymmetric surface embedded in $\mathbb{R}^3$ to a series of BIEs defined on a one-dimensional curve embedded in $\mathbb{R}^2$. This reduction allows us to take advantage of well-developed one-dimensional quadrature schemes to obtain more accurate results. We also get significant savings in terms of computational complexity:

- Using n_f Fourier modes and n_p quadrature panels to discretize the surface, solving the 2D BIE using direct methods involves inverting an $\mathcal{O}(n_p n_f)$ matrix, with computational cost $\mathcal{O}(n_p^3 n_f^3)$.
- By comparison, with the Fourier decomposition scheme, we need to solve an $\mathcal{O}(n_f)$ number of $\mathcal{O}(n_p)$ sized linear systems. This reduced computation costs $\mathcal{O}(n_p^3 n_f + n_p^2 n_f \log n_f)$ using direct solvers and taking the Fourier decoupling into account.

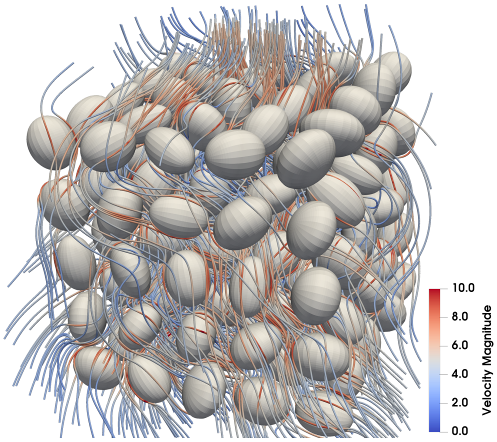

FIGURE 1: Streamlines from a Stokes flow past 125 prolate and oblate ellipsoids arranged in a cubic lattice, with a slip velocity profile assigned to each of the particles. The flow was computed using our fast BIE solver scheme, combined with FMM and iterative algorithms.

Finally, combining our solution scheme with the fast multipole method (FMM) allows us to simulate Stokes flow past multiple axisymmetric particles. In Figure 1, we demonstrate the streamlines from such a simulation, generated using our fast BIE solver scheme combined with FMM and bi-conjugate gradient descent iterative algorithm. We are currently writing up the theory and numerical experiments involving this solver framework for publication.

In the future, I plan to extend this project. Our scheme can simulate Stokes flow past axisymmetric bodies with very high length-to-width aspect ratios. I want to expand our solvers' capabilities to handle thin bodies with a nonlinear axis of symmetries. This model would then provide an efficient and accurate simulator for flow past slender bodies, e.g., bacterium's flagella.

Scalable Framework for Rigid Multi-Body Systems with Contact

One of the potential issues in simulating Stokes flow past moving geometries is when the particles collide with each other. In fact, this phenomenon of rigid body contact is also vital in many engineering applications, such as in soil-vehicle simulations in terramechanics. The discrete element method (DEM), which tracks each of the constituent particles individually and evolve them in time via either fluid flow or Newton's equations, has proven to be an excellent modeling setup in terms of accuracy and capacity to capture complex interactions. Many state-of-the-art software packages (e.g., LAMMPS and LIGGGHTS) have already implemented the traditional penalty-based contact formulation (DEM-P). But due to the nature of the imposed contact forces, the resulting system of ordinary differential equations (ODEs) is often very stiff, and the maximum timestep size tends to be very small for stability. An alternate complementarity based approach (DEM-C) resolves this issue. But at the same time is computationally expensive since we need to solve an optimization problem at every timestep. While libraries implementing the DEM-C framework, such as Chrono, are available, these are quite limited as they primarily target one-node simulations. The main challenge in implementing a distributed memory DEM-C framework is keeping the inter-note communication.

In collaboration with Shravan Veerapaneni, Eduardo Corona (New York Institute of Technology), Paramsothy Jayakumar, David Gorsich (US Army CCDC GVSC), Wen Yan (Flatiron Institute), and Dhairya Malhotra (NYU Courant), I developed a hybrid MPI-OpenMP scalable framework for simulating granular media using the DEM-C modeling approach. The full details are available in the publication [3]; I only highlight the main results here. Figure 2 demonstrates the near-optimal scalability of our implementation:

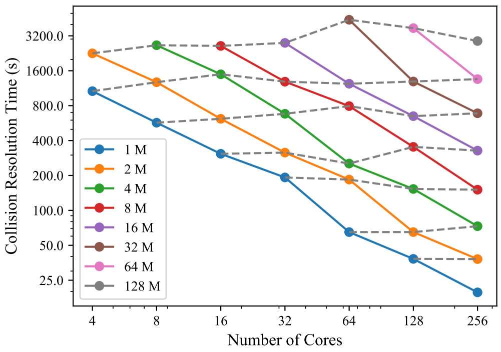

FIGURE 2: Near-optimal strong and weak scalability (solid and dashed lines respectively) of the collision resolution phase in the distributed memory rigid body collision framework.

- **Strong Scalability:** As we double the number of computing cores for a problem of the same size, the computing time reduces approximately by a factor of two.
- **Weak Scalability:** As we double the problem size and the number of cores, keeping the problem-size-per-core ratio fixed, the computational time increases only slightly.

We were also able to simulate granular material consisting of up to 256 million particles and approximately 324 million pairwise contacts using 512 computing cores. Recently, this framework was utilized as a key component to simulate a suspension of spherical rotors in Stokes fluid [4].

In the future, we plan to combine this collision detection and resolution framework with the axisymmetric BIE solver implementation. This combined framework would give us a prototype model for cellular mechanics in microbiology.

Quantum Computing for Solving Optimization Problems

The main bottleneck in DEM-C based rigid body contact simulations is the cost of solving an optimization problem at every timestep. Even utilizing the distributed memory computing framework, the computing time is still significant for large-scale problems. But the emerging quantum computing technologies have the potential to provide efficient methods to solve these optimization problems.

The quantum computing community has already designed algorithms for solving NP-hard combinatorial optimization problems [5]. Let us briefly review the MAXCUT problem: given an undirected graph (V, E) specified by its vertex and edge sets

$$V = \{1, \ldots, n\}, \quad E \subseteq V \times V = \{(i, j) : i, j \in V\} \tag{1}$$

we aim to minimize the functional

$$C(\sigma) = \sum_{(i,j) \in E} w_{ij} \sigma_i \sigma_j \quad \text{subject to} \quad \sigma : V \to \{\pm 1\}^n \tag{2}$$

where w_{ij} are weights assigned to the edges. In the context of quantum computing, this optimization target is recast as an eigenvalue minimization problem for the Hamiltonian operator

$$H_C = \sum_{(i,j) \in E} w_{ij} Z_i Z_j \quad \text{on Hilbert space} \quad \mathbb{C}^{2 \otimes n} = \mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2 \tag{3}$$

where Z_i's are Pauli-Z quantum gates

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{with eigenvalues } \pm 1 \tag{4}$$

Then the quantum optimization technique essentially boils down to this: we start from the easy-to-prepare ground state of a simpler Hamiltonian H_B, then slowly change the Hamiltonian over time: $H(t) = (1-t)H_B + tH_C$. The adiabatic principle states that if this change is gradual enough, then at $t = 1$ the quantum state will transform to the lowest-energy eigenstate of H_C. The quantum annealers produced by D-Wave Systems has already implemented this approach. Optimization algorithms such as variational Monte Carlo (VMC) and quantum approximate optimization algorithm (QAOA) designed for more general-purpose quantum computers are also available.

In collaboration with Shravan Veerapaneni, James Stokes, and Giuseppe Carleo (Flatiron Institute), I am trying to extend these algorithms to solve continuous optimization problems. We propose to achieve this by utilizing general quantum rotor models with the following Hamiltonian

$$H_C = -\frac{U}{2} \sum_{j=1}^{d} \frac{\partial}{\partial \theta_j^2} - \sum_{(i,j) \in E} J_{ij} \cos(\theta_i - \theta_j) \quad \text{on Hilbert space} \quad \mathcal{H}^1_{\text{periodic}}([0, 2\pi)^n) \tag{5}$$

Since qubits, the key-components of current quantum computers, can be considered as the zero-dimensional analog of quantum rotors, we expect much of the theory for MAXCUT-type combinatorial optimization problems to also hold true in this context of continuous optimization. We are currently developing VMC solvers for quantum rotor networks in high-dimensions, and comparing them against benchmark solutions obtained with PDE eigensolvers.

Reduced Order Model Generation with Tensor Factorizations

Even as quantum computers continue to evolve rapidly, they might not be available for general-purpose use for a few years yet. On the other hand, many engineering tasks, such as design optimization and active control, require a large number of simulations to produce an optimal answer. For such time-critical applications, reduced-order models (ROMs) that faithfully reproduce the results from the full-scale simulations at a fraction of the cost are instrumental. Deep neural networks have proven to be very useful in generating ROMs in many different fields. But one drawback of these models is the lack of interpretation — it is not straightforward to connect the weights of a neural network layer to one particular attribute of the model.

In collaboration with Shravan Veerapaneni, Eduardo Corona, and Xun Huan (my Ph.D. co-advisor, University of Michigan), I have recently started using tensor-decomposition based methods instead. These factorization models generalize the singular value decomposition (SVD), and can sometimes be interpreted as high dimensional analogs of principal component analysis (PCA). Our preliminary results show that, for Laplace boundary integral operator evaluation, we can approximate a 3D function on a $1001 \times 1001 \times 1001$ grid to machine precision by evaluating the function only at approximately 0.1% of the grid points. We intend to develop a machine-learning framework, utilizing tensor decompositions instead of neural networks, and attempt to model more complex functions arising from physical processes.

The second advantage of tensor decompositions lies in the field of data compression. In time-series data generated from a simulation, we typically expect time and space correlated simulation outputs. Tensor factorizations can exploit this redundancy and reduce the dataset to the bare essentials. For example, we have utilized the tensor-train (TT) decomposition to compress a 12.8 GB stress-strain response dataset from a multi-scale soil-wheel simulation (provided to us by Hiroyuki Sugiyama and his group at University of Iowa) to 215.0 MB with 10^{-1} relative reconstruction accuracy. Incorporating tensor factorizations into existing physics-based simulations frameworks would reduce the data output significantly. This data compression framework can potentially speed up any machine learning algorithms that would utilize the simulation data.

Bayesian Approach to Tensor Factorizations

Bayesian inference has the potential to be an ideal framework for data-driven learning tasks. With the advancement in both hardware and software, computers are finally powerful enough to run fully Bayesian data analysis, which has lead to robust predictions in many fields such as climatology and finance. A Bayesian treatment of low-rank matrix and tensor factorizations is an active area of research due to its varied potential applications. For instance, using tensor completion, one can infer the results of a simulation at new parameter values from an existing library of simulations.

In collaboration with Alex Gorodetsky and Haid Salehi (University of Michigan), I have recently demonstrated that it is possible to provably improve the existing methodology for matrix factorizations. The main result and supporting numerical experiments are available in [6], and it is a first step towards the Bayesian tensor factorization framework. We are currently developing a similar technique for low-rank tensor factorizations, and in particular, the tensor-train decomposition.

Conclusion

In summary, the projects I plan to work on will use state-of-the-art computing facilities and novel fast algorithms to lighten the computational burden of physics-based simulations. This development would increase the potential for widely utilizing these highly accurate models in application areas such as computational microbiology and vehicle manufacturing. Also, for time-critical applications, where running thousands of high-fidelity simulations would be impractical, I propose developing a robust data-driven reduced-order model generation framework that will treat the full simulation as a black box and automatically construct reduced models.

References

[1] John Ambrosiano, Leslie Greengard, and Vladimir Rokhlin. The fast multipole method for gridless particle simulation. *Computer Physics Communications*, 48(1):117–125, 1988.

[2] P Young, Sijia Hao, and Per-Gunnar Martinsson. A high-order Nyström discretization scheme for boundary integral equations defined on rotationally symmetric surfaces. *Journal of Computational Physics*, 231(11):4142–4159, 2012.

[3] **Saibal De**, Eduardo Corona, Paramsothy Jayakumar, and Shravan Veerapaneni. Scalable solvers for cone complementarity problems in frictional multibody dynamics. In *2019 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–7. IEEE, 2019.

[4] Wen Yan, Eduardo Corona, Dhairya Malhotra, Shravan Veerapaneni, and Michael Shelley. A scalable computational platform for particulate Stokes suspensions. *Journal of Computational Physics*, page 109524, 2020.

[5] Arnab Das and Bikas K Chakrabarti. *Quantum annealing and related optimization methods*, volume 679. Springer Science & Business Media, 2005.

[6] **Saibal De**, Hadi Salehi, and Alex Gorodetsky. Efficient mcmc sampling for bayesian matrix factorization by breaking posterior symmetries. *arXiv preprint arXiv:2006.04295*, 2020.