Research Statement

Yu Jun Loo

I am a researcher in scientific computing and fluid dynamics. I develop numerical methods for flows with moving and deforming bodies, particularly in difficult physical regimes where standard formulations are unstable or expensive. My work begins with simulation: I ask what model and algorithm will make a difficult physical regime computable, then use analysis to exploit mathematical structure to design a suitable algorithm and explain the simulation results. My early research with Alexander Tumanov in the analysis of PDEs (optimal Hölder regularity of the $\bar{\partial}$-equation on polydiscs [1], improving on [6, 7, 8]) has had a strong influence on my style of work. I have since used similar techniques in three directions: to enable simulations of falling plates and zero-rigidity membrane flags; to explain the wrinkling and tip-snapping that emerge in those systems; and to develop fast, differentiable vortex and Poisson solvers for modern hardware, whose gradients I am now extending toward learned control. In each case, simulation is the object of the work, while analysis is the enabling and explanatory tool.

Vortex-sheet methods for fluid–structure interaction

Computing the full, viscous flow around a macroscopic deforming body at moderate Reynolds numbers can be prohibitively expensive, since fine-scale vortex structures form on a surface of complex, time-varying shape. In the inviscid limit, the boundary layer collapses to a vortex sheet of zero thickness that can separate from the body and continue to shape the pressure forces on it. This simplified model is orders of magnitude faster than direct simulation, enabling the rapid exploration of highly unsteady flows and its embedding in an optimization or control loop. However, the mathematical and computational foundations of such methods remain far less developed than those of grid-based methods. My work addresses part of that gap by developing a framework for continuous, stable vortex shedding from both edges of a moving or deforming body [2, 3]. Prior vortex-sheet methods suppressed leading-edge shedding for numerical stability [9, 10, 11], restricting simulations to trailing-edge separation or short time horizons. I developed a specialized quadrature rule and regularization of the Birkhoff–Rott equations to stabilize leading-edge shedding (figure 1(a)), and used a projection procedure to control the regularization error over long times. This produced the first long-time simulations of a falling plate in an inviscid fluid with continuous shedding from both edges (video link). The paper was selected as an Editors' Suggestion in *Phys. Rev. Fluids*. Across roughly seventy rigid-body motions, the solver reproduced direct Navier–Stokes forces to within 10–20% while operating at orders-of-magnitude lower computational cost [3].

Computing singular limits in elastic membrane flow

My most extensive body of work couples the fluid solver above to an extensible elastic membrane and asks how the computed dynamics change as the bending rigidity R_2 is taken to zero [4]. This is a singular perturbation limit in which the governing PDE itself degenerates. The fluid side remains a singular integral equation, and its numerical pressure solve has a conservation-law interpretation: projecting the Euler momentum equation onto the membrane and taking its jump across the surface shows that the pressure jump is the flux transporting bound vortex-sheet strength along the body.

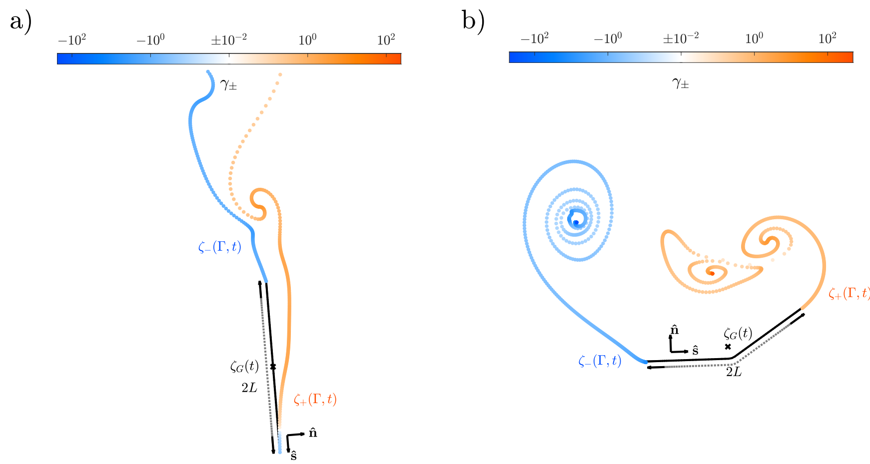

Figure 1: A flat (a) and V-shaped (b) plate falling under gravity. As they fall, they shed vortex sheets colored by their strengths in blue-orange.

This explains why bound-vorticity and pressure extrema coincide throughout the simulations and why shed vortices alternate in sign.

Three results follow from analyzing the equation's singular limits directly rather than smoothing over the associated numerical difficulties.

- **Compression wrinkling.** At zero bending rigidity the membrane equation is ill-posed under compression: arbitrarily short perturbations grow arbitrarily fast as the equation degenerates from hyperbolic to elliptic in time. For any positive rigidity, however, the same equation is provably well-posed independently of the sign of the tension. Thus, after a vortex detaches and unloads the membrane, wrinkles form on the bending-regularized, well-posed branch of the otherwise ill-posed equation.
- **Trailing-edge tip-snapping.** A membrane pinned at one end and free at the other is formally underdetermined at zero bending rigidity because the tension vanishes at the free edge. I show that a finite stretching energy constraint supplements the missing condition and enforce it through a weak spectral Galerkin method. This permits, to my knowledge, the first simulations of a fixed-free membrane at exactly zero bending rigidity. The resulting solutions reveal rapid tip-snapping: pressure disturbances cannot propagate past the zero-wave-speed free edge and instead accumulate there.
- **Inverse-logarithmic scaling.** As the bending rigidity tends to zero, the tip-snapping frequency converges to its singular limit at the unusually slow rate $O(1/|\log R_2|)$, confirmed numerically to $R^2 = 0.989$ (figure 2). Linear analysis traces this slow convergence to the logarithmically-singular branch of the zero-rigidity problem, whose coefficient must vanish inverse-logarithmically as the bending layer contracts toward the free edge.

Accelerating singular integral equations on modern hardware

The third thread concerns high-performance scientific computing: I compress singular, translation-invariant kernels into fast, differentiable solvers designed to run as static tensor-network computations on modern accelerator hardware.

Existing quantum-inspired CFD solvers compress the velocity field itself as a matrix product

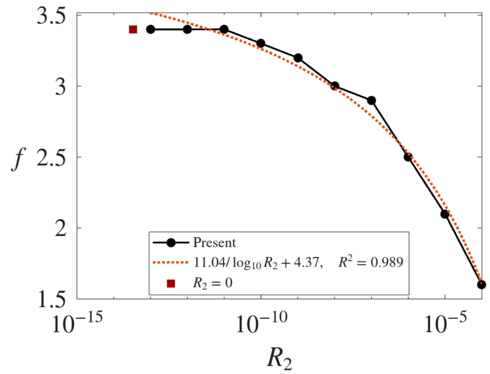

Figure 2: Tip-snapping frequency f vs. bending rigidity R_2, with an inverse-logarithmic fit (dashed) and the $R_2 = 0$ limit (red square).

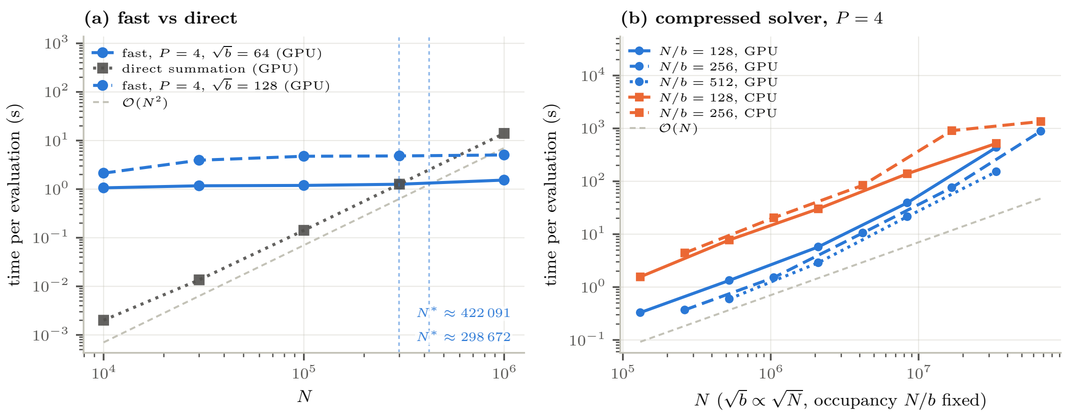

Figure 3: Wall-clock time per evaluation for DiffVortex vs. particle count N. (a) At fixed grid size, the compressed solver overtakes direct $O(N^2)$ summation beyond $N^* \approx 3 \times 10^5$. (b) At fixed occupancy, cost scales linearly in N past 3×10^7 particles.

state [12, 13]. This works well for low-Reynolds-number flows in simple geometries, but breaks down for deforming bodies, where the field must be re-masked at every step and thin shed vortex sheets resist low-rank compression. DiffVortex [5] is a differentiable fluid solver (hence the name) that instead compresses the Biot–Savart convolution operator. The operator is fixed for the entire simulation and compressed once, offline, while Lagrangian vortex particles carry the flow without re-masking as the body moves or deforms. My contribution to DiffVortex is the tensor-network formulation, the code implementation, and the lifting theorem that makes this static operator compression possible: a d-dimensional kernel of matrix-product-state rank r yields a convolution operator of matrix-product-operator rank at most $2^d r$, independent of grid resolution and with a sharp constant. The theorem turns the moment-to-local stage of a single-level fast multipole method into a precomputed tensor-network contraction, giving an $O(N)$ solver for uniformly distributed particles (figure 3) that runs natively on GPU accelerators and is differentiable end-to-end. Its gradients are therefore available to a downstream optimization or control law with no separate adjoint solve.

In ongoing work with Jason Kaye, Marc Ritter, and Shravan Veerapaneni, I designed and implemented a high-order QTT solver for the free-space Poisson equation. The main analytical obstacle is high-order quadrature for kernels singular at a lattice point. I generalize the classi-

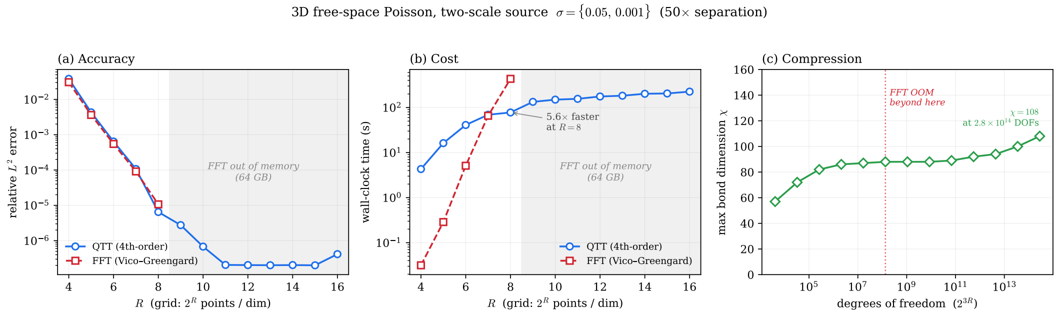

Figure 4: QTT Poisson solver vs. FFT: (a) accuracy and (b) runtime vs. resolution R (2^R points/dimension); (c) solution bond dimension stays nearly flat as resolution grows.

cal Euler–Maclaurin formula to this setting and show that the resulting global corrections remain compressible in the QTT format. Combined with the lifting theorem above, this produces a sixth-order accurate solver whose cost grows logarithmically with resolution. It outperforms FFT-based free-space solvers by an order of magnitude under large scale separation and continues past the resolutions at which the FFT exhausts memory (figure 4). The Euler–Maclaurin generalization extends beyond the Poisson kernel, and I am developing it into a general theory of tensor-compressible corrected quadratures.

Across these projects, the recurring problem is the same: the physically interesting simulation regimes are often the ones in which the obvious numerical formulation fails. My approach is to use analysis to identify the structure responsible for that failure and then build computational solutions around it.

Future directions

My long-term goal is to develop real-time, three-dimensional unsteady-flow solvers for control. Reaching that goal requires advances at several levels: reduced-order models that accurately represent forces, algorithms that accelerate these models at large scale, and differentiable implementations that expose the resulting dynamics to optimization. My current work provides pieces of each, and I plan to develop them into an integrated computational framework.

One direction is differentiable flow control. DiffVortex and the QTT Poisson solver already propagate gradients end-to-end, so they can be inserted directly into optimization and learning pipelines. I plan to use this capability to design control laws that exploit, rather than suppress, vortex formation and shedding in highly unsteady regimes, particularly for flexible UAVs and energy-harvesting devices.

The second direction is three-dimensional vortex-sheet simulation. In three dimensions, shedding occurs from moving edge curves and generates vortex sheets that stretch, fold, and change topology. These dynamics make stable simulation substantially more difficult while greatly increasing the computational cost. I plan to develop stable shedding and regularization procedures for these evolving sheets and combine them with fast tensor-compressed interaction algorithms. The goal is to make repeated three-dimensional simulation inexpensive enough for optimization and eventually real-time control. As throughout my work, I will use analysis to identify the mechanisms that limit the computation, guide the design of stable algorithms, and explain the dynamics that emerge from the simulations.

References

[1] Y. J. Loo and A. Tumanov, *Hölder regularity of the $\bar{\partial}$-equation on the polydisc*, Complex Var. Elliptic Equ., pp. 1–9 (2025).

[2] Y. J. Loo and S. Alben, *Falling plates with leading-edge vortex shedding*, Phys. Rev. Fluids **10**, 104701 (2025).

[3] Y. J. Loo and S. Alben, *Comparison of inviscid and viscous vortex shedding from translating and rotating plates*, arXiv:2602.06299 (2026).

[4] Y. J. Loo and S. Alben, *Extensible membranes in inviscid flow: aerodynamics and singular limits*, preprint (2026).

[5] Y. J. Loo, S. Alben, and S. Veerapaneni, *DiffVortex: A quantum-inspired fast solver for incompressible flows*, preprint (2026).

[6] Y. Pan and Y. Zhang, *Hölder estimates for the $\bar{\partial}$ problem for (p, q) forms on product domains*, Internat. J. Math. **32**, no. 3, 2150014 (2021).

[7] Y. Pan and Y. Zhang, *Cauchy singular integral operator with parameters in log-Hölder spaces*, J. Anal. Math. **145**, no. 1, 357–379 (2021).

[8] Y. Zhang, *Optimal Hölder regularity for the $\bar{\partial}$ problem on product domains in $\mathbb{C}^2$*, Proc. Amer. Math. Soc. **150**, 2937–2943 (2022).

[9] S. Alben, *Simulating the dynamics of flexible bodies and vortex sheets*, J. Comput. Phys. **228**, no. 7, 2587–2603 (2009).

[10] C. Mavroyiakoumou and S. Alben, *Large-amplitude membrane flutter in inviscid flow*, J. Fluid Mech. **891**, A23 (2020).

[11] C. Mavroyiakoumou and S. Alben, *Sail dynamics during tacking motions*, Phys. Rev. Fluids **10**, 073901 (2025).

[12] N. Gourianov, M. Lubasch, S. Dolgov, Q. Y. van den Berg, H. Babaee, P. Givi, M. Kiffner, and D. Jaksch, *A quantum-inspired approach to exploit turbulence structures*, Nat. Comput. Sci. **2**, 30–37 (2022).

[13] R. D. Peddinti, S. Pisoni, A. Marini, P. Lott, H. Argentieri, E. Tiunov, and L. Aolita, *Quantum-inspired framework for computational fluid dynamics*, Commun. Phys. **7**, 135 (2024).

[14] V. Kazeev, B. N. Khoromskij, and E. E. Tyrtyshnikov, *Multilevel Toeplitz matrices generated by tensor-structured vectors and convolution with logarithmic complexity*, SIAM J. Sci. Comput. **35**, no. 3, A1511–A1536 (2013).