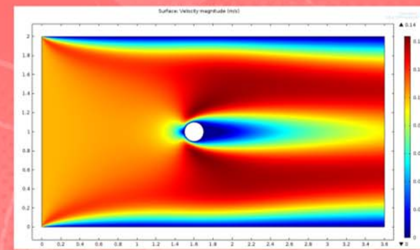
The Effect of Mesh Size and Mach Number on Computational Fluid Dynamics

Adams, Benson, Brewer, Lisk

Dr. Pingen, EGR 250 faculty advisor

Computational Fluid Dynamics (CFD) is a process used to model different fluid flow scenarios. Most engineers solve the Navier-Stokes Equation to do this, while we are using the Boltzmann Transport Equation. These create accurate models of overall drag forces and flow speeds at all points in a specified flow region. Even with modern computing, some flow situations may be impossibly slow to process accurately. Modeling flow through a pipe and around a round obstacle is often used as a benchmark for fluid dynamics solvers. We accordingly compared our results to standards prepared by Schaefer and Turek.

David Makhija, along with Georg Pingen and Kurt Maute, published research using the immersed boundary method of solving the Boltzmann's Transport Equation. They ran into a peculiarity related to the flow velocity of the fluid, as seen in the results in the graph *Results from Makhija, Pingen, Maute Paper* below. Most fluid flow solvers, including the immersed boundary method, ignore the compressibility of the fluid. This is generally accurate, as most fluids appear incompressible. However, at speeds approaching the speed of sound, most fluids begin to be compressible.



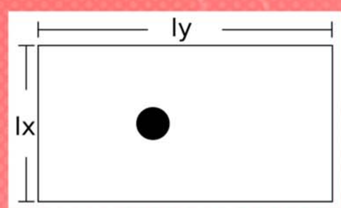
CFD Problem in Comsol Multiphysics

Flow Analysis Results

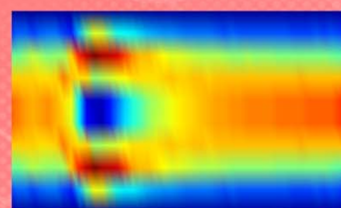
Ma	lx	Re	DOF	Cd	Time Steps	t_{TS}
0.001	10	20	4050	-76.95059	25233	0.00167
0.001	20	20	17820	3.61306	52451	0.00237
0.001	30	20	41040	5.92844	81480	0.00416
0.001	40	20	74160	5.51493	114170	0.00728
0.001	50	20	117000	5.30169	144288	0.01632
0.001	100	20	475200	5.39895	333576	0.06211
0.01	10	20	4050	-3.48934	7176	0.00129
0.01	20	20	17820	4.72271	22855	0.0029
0.01	30	20	41040	5.9119	32690	0.00407
0.01	40	20	74160	5.55452	42658	0.00787
0.01	50	20	117000	5.29053	52200	0.01238
0.01	100	20	475200	5.38772	116352	0.05679
0.023	10	20	4050	1.69281	5481	0.00101
0.023	20	20	17820	4.83253	15428	0.00266
0.023	30	20	41040	5.89741	21820	0.00452
0.023	40	20	74160	5.52172	29022	0.00731
0.023	50	20	117000	5.2791	36252	0.0133
0.023	100	20	475200	5.38042	74088	0.06921
0.023	200	20	1915200	5.33634	155304	0.26121
0.05	50	20	117000	5.27266	22640	0.02766
0.05	100	20	475200	5.38148	39380	0.05425
0.05	200	20	1915200	5.34185	100296	0.39063
0.058	10	20	4050	4.2212	3705	0.00117
0.058	20	20	17820	4.9145	9016	0.00289
0.058	30	20	41040	5.88083	12980	0.00456
0.058	40	20	74160	5.49454	17822	0.0094
0.058	50	20	117000	5.27481	22500	0.01587
0.058	100	20	475200	5.38728	45540	0.05268
0.058	200	20	1915200	5.34778	92808	0.30518
0.1	10	20	4050	4.89325	2748	0.13695
0.1	20	20	17820	5.00293	6209	0.00385
0.1	30	20	41040	5.90854	9200	0.02624
0.1	40	20	74160	5.53123	12740	0.01754
0.1	50	20	117000	5.31961	16092	0.02986
0.1	100	20	475200	5.44347	32580	0.18567
0.1	200	20	1915200	5.40952	65592	0.31618
0.1	500	20	12033000	5.38621	167622	3.16177
0.15	50	20	117000	5.43527	10520	0.01048
0.15	100	20	475200	5.57038	18460	0.05195
0.15	200	20	1915200	5.54273	49608	0.2421
0.2	50	20	117000	5.62045	8460	0.01086
0.2	100	20	475200	5.77294	14260	0.05865
0.2	200	20	1915200	5.74536	40032	0.26457
0.25	50	20	117000	5.88497	6780	0.0107
0.25	100	20	475200	6.05133	11400	0.06126
0.25	200	20	1915200	6.02846	35208	0.24254
0.3	50	20	117000	6.24755	6500	0.01006
0.3	100	20	475200	6.4393	7800	0.05354
0.35	50	20	117000	6.74196	5840	0.01192
0.35	100	20	475200	6.94125	9380	0.0519
0.4	50	20	117000	7.41504	5220	0.01054
0.4	100	20	475200	7.66534	7780	0.0579
0.45	50	20	117000	8.37042	4660	0.00933
0.45	100	20	475200	8.64094	8900	0.05966
0.5	50	20	117000	9.7661	4300	0.01002

Tabulated Experimental Results

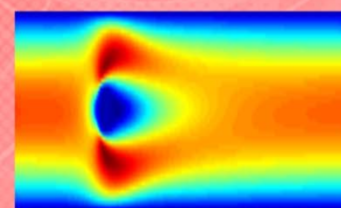
We designed a similar experiment to demonstrate a comparable system for the Lattice-Boltzmann method, a compressible solver for the Boltzmann Transport Equation. Our goal was to determine the mesh size and flow velocity required to determine accurate drag force in a reasonable amount of computation time. The Lattice-Boltzmann method requires the region of interest to be divided into a mesh of arbitrary dimensions. It then analyzes the behavior of each individual unit in the mesh. Smaller meshes are pretty quick to solve, but they result in relatively imprecise results (examples of both small and larger meshes can be seen below). We need large meshes to produce results with a reasonable degree of accuracy. Unfortunately, analyzing large meshes increases the computational time required exponentially. To put it in perspective, our custom-built research workstation with an eight-core processor, 32GiB of ram, and a high-end graphics card--over eight times more powerful than most standard computers-- took over a day to run a single simulation at our large mesh size.



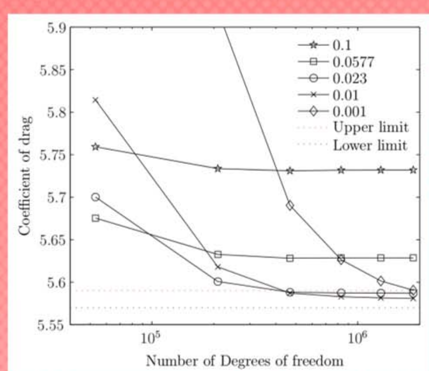
Problem Setup



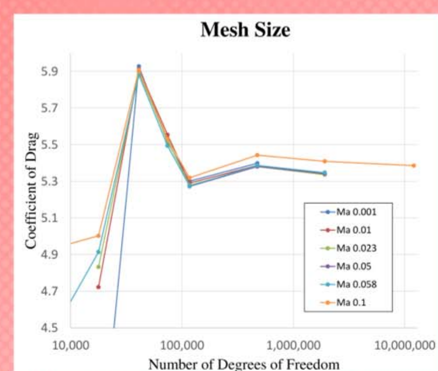
Small Mesh



Larger Mesh

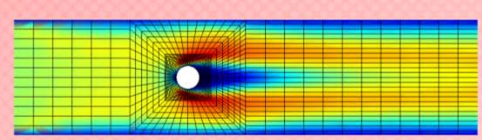


Results from Makhija, Pingen, Maute Paper

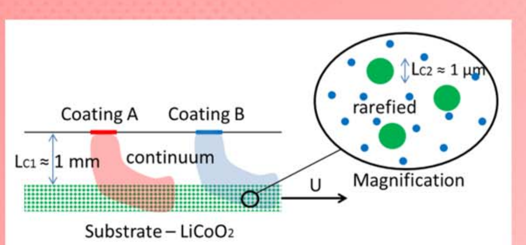


Our Experimental Results

While the experimental results shown above were obtained with a finite difference based Lattice Boltzmann (LBM) algorithm, **our research partners at the University of Colorado utilize a highly optimized parallel solver based on a stabilized Galerkin finite-element (FEA) formulation.** In order to better understand the fundamentals of finite element formulations, Andrew Tan has been working on the implementation of a Navier-Stokes finite element solver in Matlab to simulate fluid flow over the 2D cylinder previously discussed. The developed solver permits limited localized mesh refinement and utilizes a Newton solver for the solution of large nonlinear systems of equations. Current analysis is limited to low Reynolds number flows due to the lack of a stabilization procedure, which is our current research focus. The results shown in the figure to the right show the x-velocities of flow across the cylinder at RE=20, as well as the mesh used.



Navier-Stokes Cylinder Model



Real-world Application

This project constitutes the first steps towards the development of a multi-scale kinetic theory model for macro and micro-scale flow applications which is required to simulate, for example, the chemical coating processes for lithium-ion batteries as illustrated in the figure to the left. Here we present the first steps towards the development and testing of the necessary flow solvers, an effort that we are undertaking in collaboration with researchers at the University of Colorado at Boulder. We would like to thank Union University for supporting this project with an undergraduate research grant.