I propose investigating airfoil shape optimization using a novel combination of recently developed techniques, including pseudospectral (PS), volume penalization, and adjoint looping optimization methods. I will perform simulations using the `Dedalus` PS python framework which I have used to study Rayleigh-Bénard convection and magnetohydrodynamics throughout the last two years. Volume penalization for external flow simulations has already been implemented with `Dedalus` [Hester 2020] and there exists a substantial body of airfoil shape optimization research for comparison. I am currently developing the necessary adjoint looping optimization infrastructure which uses `Dedalus` to solve the adjoint system, thereby allowing us to approximate gradients for arbitrary metrics. The proposed technique offers key advantages when compared to the conventional approach which relies on finite volume simulations and finite difference gradient-based optimization.

1. High-fidelity physics simulations: PS methods are well-suited for simulating turbulent fluid flow. Finite volume methods are subject to small grid-scale numerical errors which can excite nominally stable modes in turbulent shear flows, thereby triggering artificial instabilities [Lecoanet 2015]. In the context of shape optimization, these instabilities could propagate tremendous differences in the resulting optimized geometries.
2. Modular flow regimes: the `Dedalus` pseudospectral python framework interprets equations symbolically, allowing for rapid development and future shape optimization research in compressible, high-Mach number, multiphase, and combustive flow regimes using the same codes.
3. Efficient gradient-based optimization: adjoint looping allows us to approximate gradients rapidly and accurately in high dimensional parameter spaces [Lecoanet 2015]. The conventional finite difference method is costly (and sometimes inaccurate) because it involves a large number of simulations.
4. Flexible domain representation: the volume penalty method allows us to model external flows around an arbitrary spatial mask function, thereby eliminating the need to continuously remesh as the design is updated and ultimately reconstruct a palatable design from the optimized mesh.

Airfoils offer an ideal context for investigation because they operate in a wide range of flow regimes. Airfoil geometries are generally smooth, allowing us to represent them with truncated Fourier series. Smooth design representation has tangible benefits from an engineering perspective and it will allow us to analyze optimizated geometries from a scientific perspective. For instance, we will vary the system's Reynolds number and compare the optimized Fourier coefficient profiles.

Methods: The complicated domains necessary for shape optimization are difficult to represent via spectral basis functions. Volume penalization is the robust solution, where a time-invariant mask function Φ is included in the relevant PDE to represent the immersed solid. We combine this technique with the adjoint looping method, which has been used to optimize initial conditions in similar 2D incompressible flows [Kerswell 2014].

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \boldsymbol{\nabla} \boldsymbol{u} + \frac{1}{\rho} \boldsymbol{\nabla} p = \nu \nabla^2 \boldsymbol{u} - \frac{1}{\tau} \Phi(\boldsymbol{x})(\boldsymbol{u} - \boldsymbol{U}) \qquad \text{and} \qquad \boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 \tag{1}$$

where $\boldsymbol{U} = \hat{\boldsymbol{x}} U$ is the airfoil's velocity frame; $\tau \ll 1$ is the damping timescale; and $\boldsymbol{u} = u\hat{\boldsymbol{x}} + v\hat{\boldsymbol{y}}$, p, ρ, and ν are the fluid's velocity, pressure, density, and viscosity respectively. The aerodynamic force acting on the immersed interface is given by

$$\boldsymbol{F} = F_D \hat{\boldsymbol{x}} + F_L \hat{\boldsymbol{y}} = \frac{\rho}{\tau} \langle \Phi(\boldsymbol{x})(\boldsymbol{u} - \boldsymbol{U}) \rangle \tag{2}$$

where $\langle \cdot \rangle$ denotes the spatial integral over the domain $-L_x/2 < x < L_x/2$, $-L_y/2 < y < L_y/2$.

The mask function is a smooth indicator

$$\Phi(\boldsymbol{x}) = \frac{1}{2}\big[1 - \tanh\big(d_r^{-1}\mathrm{SDF}(\boldsymbol{x})\big)\big] \tag{3}$$

where d_r is proportional to the minimum resolvable length scale. The signed distance function $\mathrm{SDF}(\boldsymbol{x})$ is bijective with a closed airfoil contour $\{\boldsymbol{x} : \mathrm{SDF}(\boldsymbol{x}) = 0\}$

Our optimization problem maximizes the time-integrated lift $\int_0^T F_L(\boldsymbol{u}; \Phi(\boldsymbol{x}))dt$ while penalizing the drag's time-integrated deviation $\frac{\alpha}{2}\big[\int_0^T F_D(\boldsymbol{u}; \Phi(\boldsymbol{x})) - \overline{F_D}dt\big]^2$ from some prescribed drag target $\overline{F_D}$ where $T \gg U/L_x$.

$$\begin{aligned}\mathcal{L}(\boldsymbol{u}, p, \Phi, \boldsymbol{\mu}, \pi) &= \frac{1}{\tau}\int_0^T \langle\Phi(\boldsymbol{x})(\boldsymbol{u}-\boldsymbol{U})\cdot\hat{\boldsymbol{y}}\rangle - \frac{\alpha}{2}\big\langle\Phi(\boldsymbol{x})(\boldsymbol{u}-\boldsymbol{U})\cdot\hat{\boldsymbol{x}} - \overline{F_D}\big\rangle^2 dt + \ldots \\ &\int_0^T \Big\langle\boldsymbol{\mu}(\boldsymbol{x},t)\cdot\big(\partial_t\boldsymbol{u} + \boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{u} + \frac{1}{\rho}\boldsymbol{\nabla}p - \nu\nabla^2\boldsymbol{u} + \frac{1}{\tau}\Phi(\boldsymbol{x})(\boldsymbol{u}-\boldsymbol{U})\big)\Big\rangle dt + \int_0^T \big\langle\pi(\boldsymbol{x},t)\boldsymbol{\nabla}\cdot\boldsymbol{u}\big\rangle dt\end{aligned}$$

where

$$\begin{aligned}\mathcal{J} &\equiv \frac{1}{\tau}\int_0^T \langle\Phi(\boldsymbol{x})(\boldsymbol{u}-\boldsymbol{U})\cdot\hat{\boldsymbol{y}}\rangle - \frac{\alpha}{2}\big\langle\Phi(\boldsymbol{x})(\boldsymbol{u}-\boldsymbol{U})\cdot\hat{\boldsymbol{x}} - \overline{F_D}\big\rangle^2 dt \\ &= \frac{1}{\rho}\int_0^T F_L(\boldsymbol{u};\Phi(\boldsymbol{x})) - \frac{\alpha}{2}\Big[F_D(\boldsymbol{u};\Phi(\boldsymbol{x})) - \overline{F_D}\Big]^2 dt\end{aligned}$$

is the scalar quantity to be maximized and the remaining terms impose the Navier-Stokes constraint. We aim to compute the function $\mathrm{SDF}(\boldsymbol{x})$ which maximizes $\mathcal{J}$. The coefficient α is left arbitrary.

The adjoint system

$$\begin{aligned}\frac{\delta\mathcal{L}}{\delta p(\boldsymbol{x},t)} &= 0 = -\boldsymbol{\nabla}\cdot\boldsymbol{\mu} \\ \frac{\delta\mathcal{L}}{\delta\boldsymbol{u}(\boldsymbol{x},t)} &= \boldsymbol{0} = -\partial_t\boldsymbol{\mu} - \boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{\mu} + \boldsymbol{\mu}\cdot\big(\boldsymbol{\nabla}\boldsymbol{u}\big)^T - \boldsymbol{\nabla}\pi - \nu\nabla^2\boldsymbol{\mu} + \frac{1}{\tau}\Phi\,\boldsymbol{\mu} + \frac{\delta\mathcal{J}}{\delta\boldsymbol{u}} \\ &\implies \boldsymbol{0} = \partial_t\boldsymbol{\mu} + \boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{\mu} - \boldsymbol{\mu}\cdot\big(\boldsymbol{\nabla}\boldsymbol{u}\big)^T + \boldsymbol{\nabla}\pi + \nu\nabla^2\boldsymbol{\mu} - \frac{1}{\tau}\Phi\,\boldsymbol{\mu} - \frac{1}{\tau}\Phi(\boldsymbol{x})\Big[\hat{\boldsymbol{y}} - \alpha(F_D - \overline{F_D})\hat{\boldsymbol{x}}\Big] \\ &\implies \boldsymbol{0} = \partial_t\boldsymbol{\mu} + \boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{\mu} - \boldsymbol{\mu}\cdot\big(\boldsymbol{\nabla}\boldsymbol{u}\big)^T + \boldsymbol{\nabla}\pi + \nu\nabla^2\boldsymbol{\mu} - \frac{1}{\tau}\Phi(\boldsymbol{x})\Big[\boldsymbol{\mu} + \hat{\boldsymbol{y}} - \alpha(F_D - \overline{F_D})\hat{\boldsymbol{x}}\Big] \\ \frac{\delta\mathcal{L}}{\delta\boldsymbol{u}(\boldsymbol{x},T)} &= \boldsymbol{0} = \boldsymbol{\mu}(\boldsymbol{x},T)\end{aligned}$$

is evolved backward in time $t : T \to 0$. This renders the viscous term well-posed. The mask term has a dampening effect $\boldsymbol{\mu} + \hat{\boldsymbol{y}} \to 0$.

While solving the forward and adjoint systems, we can integrate

$$\frac{\delta\mathcal{L}}{\delta\Phi(\boldsymbol{x})} = \frac{1}{\tau}\int_0^T \Big(\boldsymbol{\mu} + \hat{\boldsymbol{y}} - \alpha(F_D - \overline{F_D})\hat{\boldsymbol{x}}\Big)\cdot\Big(\boldsymbol{u} - \boldsymbol{U}\Big)dt.$$

Or, for a time-dependent mask function $\Phi(\boldsymbol{x}, t)$,

$$\frac{\delta\mathcal{L}}{\delta\Phi(\boldsymbol{x}, t)} = \frac{1}{\tau}\Big(\boldsymbol{\mu} + \hat{\boldsymbol{y}} - \alpha(F_D - \overline{F_D})\hat{\boldsymbol{x}}\Big)\cdot\Big(\boldsymbol{u} - \boldsymbol{U}\Big).$$

However, the prospect of gradient-based optimization with respect to $\Phi(\boldsymbol{x}, t)$ poses a hypothetical problem: $\Phi(\boldsymbol{x}, t)$ is a "smooth" indicator where, overwhelmingly, we have either $\Phi(\boldsymbol{x}, t) = 0$ or $\Phi(\boldsymbol{x}, t) = 1$.

Rather than treating $\Phi(\boldsymbol{x}, t)$ as the optimization parameter, we can instead optimize the signed-distance function $\mathrm{SDF}(\boldsymbol{x}, t)$ subject to the eikonal equation constraint

$$|\boldsymbol{\nabla}[f(\boldsymbol{x}, t)]| = 1 = (\partial_x f)^2 + (\partial_y f)^2 = (\partial_x f + i\partial_y f)(\partial_x f - i\partial_y f) \tag{4}$$

which we must then impose somewhere in our alorithm. **projection:** At iteration n, suppose we have performed a loop on distance function $d_n(\boldsymbol{x}, t) = \mathrm{SDF}(\boldsymbol{x}, t)$, yielding $\frac{\delta\mathcal{L}}{\delta d_n(\boldsymbol{x},t)}$. Assuming

$$|\boldsymbol{\nabla} d_n| = 1 = (\partial_x d_n)^2 + (\partial_y d_n)^2 = (\partial_x d_n + i\partial_y d_n)(\partial_x d_n - i\partial_y d_n), \tag{5}$$

we want

$$|\boldsymbol{\nabla} d_n + \varepsilon\boldsymbol{\nabla}\frac{\delta\mathcal{L}}{\delta d_n}| = 1 = (\partial_x d_n + \varepsilon\partial_x\frac{\delta\mathcal{L}}{\delta d_n})^2 + (\partial_y d_n + \varepsilon\partial_y\frac{\delta\mathcal{L}}{\delta d_n})^2. \tag{6}$$

$$0 = 2\varepsilon\Big[\partial_x d_n\partial_x\frac{\delta\mathcal{L}}{\delta d_n} + \partial_y d_n\partial_y\frac{\delta\mathcal{L}}{\delta d_n}\Big] + \varepsilon^2\Big[(\partial_x\frac{\delta\mathcal{L}}{\delta d_n})^2 + (\partial_y\frac{\delta\mathcal{L}}{\delta d_n})^2\Big]) \tag{7}$$

for $\varepsilon \neq 0$,

$$\boldsymbol{\nabla} d_n \cdot \boldsymbol{\nabla}\frac{\delta\mathcal{L}}{\delta d_n} = -\frac{\varepsilon}{2}\boldsymbol{\nabla}\frac{\delta\mathcal{L}}{\delta d_n}\cdot\boldsymbol{\nabla}\frac{\delta\mathcal{L}}{\delta d_n} \tag{8}$$

The simplest implementation would be to interpret the updated airfoil contour

$$\Gamma(t) \equiv \{\boldsymbol{\xi}(t) \text{ s.t. } \mathrm{SDF}(\boldsymbol{\xi}(t), t) = 0\} \tag{9}$$

as a boundary condition. Using these data, we can solve 4 and take its solution to be the updated SDF.

The mask function

$$\Phi(\boldsymbol{x}, t) = \frac{1}{2}\big[1 - \tanh\big(d_r^{-1}\mathrm{SDF}(\boldsymbol{x}, t)\big)\big] \tag{10}$$

$$\frac{\delta\Phi(\boldsymbol{x}, t)}{\delta\mathrm{SDF}(\boldsymbol{x}, t)} = -\frac{1}{2d_r}\mathrm{sech}^2\big(d_r^{-1}\mathrm{SDF}(\boldsymbol{x}, t)\big) \tag{11}$$

$$\frac{\delta\mathcal{L}}{\delta\mathrm{SDF}(\boldsymbol{x}, t)} = -\frac{\mathrm{sech}^2\big(d_r^{-1}\mathrm{SDF}(\boldsymbol{x}, t)\big)}{2\tau d_r}\Big(\boldsymbol{\mu} + \hat{\boldsymbol{y}} - \alpha(F_D - \overline{F_D})\hat{\boldsymbol{x}}\Big) \cdot \Big(\boldsymbol{u} - \boldsymbol{U}\Big) \tag{12}$$