Numerical shape optimization for compressible flows

Praveen. C
praveen@math.tifrbng.res.in

Tata Institute of Fundamental Research
Center for Applicable Mathematics
Bangalore 560065
http://math.tifrbng.res.in

ICMPDE
TIFR-CAM, Bangalore
13–17 August, 2010

Shape optimization framework

- Shape is parameterized in terms of $x \in D \subset \mathbb{R}^d$
- PDE-constrained minimization

$$\min_{x \in D} J(x, u) \quad \text{s.t.} \quad R(x, u) = 0$$

- Solving $R = 0$ is computationally expensive
- d can be large - *Curse of dimensionality*

Navier-Stokes Equations I

- In d-dimensions

$$\frac{\partial U}{\partial t} + \sum_{i=1}^{d} \frac{\partial F_i}{\partial x_i} = \sum_{i=1}^{d} \frac{\partial G_i}{\partial x_i}$$

- Conserved quantities and fluxes

$$U = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ E \end{bmatrix}, \quad F_i = \begin{bmatrix} \rho u_i \\ p\delta_{i1} + \rho u_1 u_i \\ p\delta_{i2} + \rho u_2 u_i \\ p\delta_{i3} + \rho u_3 u_i \\ (E+p)u_i \end{bmatrix}, \quad G_i = \begin{bmatrix} 0 \\ \tau_{i1} \\ \tau_{i2} \\ \tau_{i3} \\ \tau_{ij}u_j - q_i \end{bmatrix}$$

$$\begin{aligned} \rho &= \text{Density} \\ (u_1, u_2, u_3) &= \text{Velocity} \\ p &= \text{Pressure} \\ E &= \text{Energy per unit volume} \\ \tau_{ij} &= \text{Viscous stress tensor} \\ q_i &= \text{Heat flux} \end{aligned}$$

Navier-Stokes Equations II

- Ideal gas equation of state

$$p = (\gamma - 1)\left[E - \frac{1}{2}\rho|u|^2\right]$$

- Constitutive law

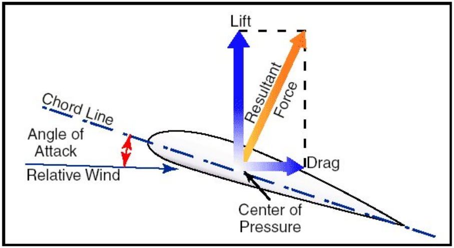
$$\begin{aligned}
\tau_{ij} &= (\mu + \mu_t)\left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}(\nabla \cdot u)\delta_{ij}\right] \\
q_i &= -\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x_i}
\end{aligned}$$

- Additional equations, *Turbulence models*, to determine μ_t

Quantities of interest

- Forces on a solid body

$$F_i = \int_S (-pn_i + \tau_{ij} n_j) \mathrm{d}S$$



- Lift and drag

$$L = F \cdot V_\infty^\perp \qquad D = F \cdot V_\infty$$

- Optimization problem

$$\min D \quad \text{s.t.} \quad L = W, \quad \text{etc.}$$

Finite volume method

$$\frac{\partial U}{\partial t} + \nabla \cdot H(U) = 0$$

- Divide Ω into non-overlapping, polygonal, *finite volumes* Ω_i

$$\Omega = \cup_i \Omega_i$$

- Conservation principle on each finite volume

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega_i} U \mathrm{d}x + \int_{\partial\Omega_i} H \cdot n \mathrm{d}s = 0$$

- Semi-discrete scheme

$$|\Omega_i| \frac{\mathrm{d}U_i}{\mathrm{d}t} + \sum_{j \in N(i)} H(U_i, U_j, n_{ij}) = 0$$

Classical approach

- PDE-constrained problem

$$\min_x J(u, x) \quad \text{s.t.} \quad R(u, x) = 0$$

or

$$L(x, u, v) = J(x, u) + (v, R(u, x))$$

- Need to develop complex adjoint solvers
- PDE models not well motivated, e.g., turbulence models

$$\min_x J_h(u_h, x) \quad \text{s.t.} \quad R_h(u_h, x) = 0$$

- Discrete approach: Automatic Differentiation
- Noisy objective functions
- Only local optimum

Discrete adjoint in presence of shocks

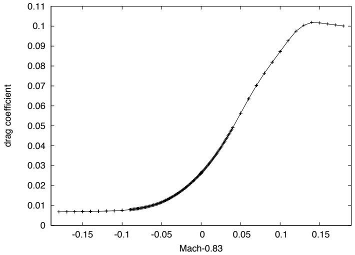


Figure 6: Drag with respect to Mach number in transonic regime.

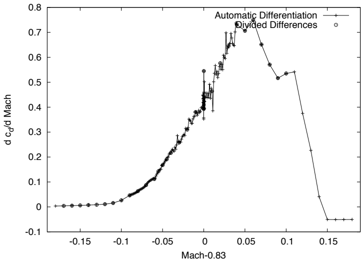


Figure 7: Drag derivatives with respect to Mach number in transonic regime.

(Martinelli et al.)

Discrete adjoint in presence of shocks

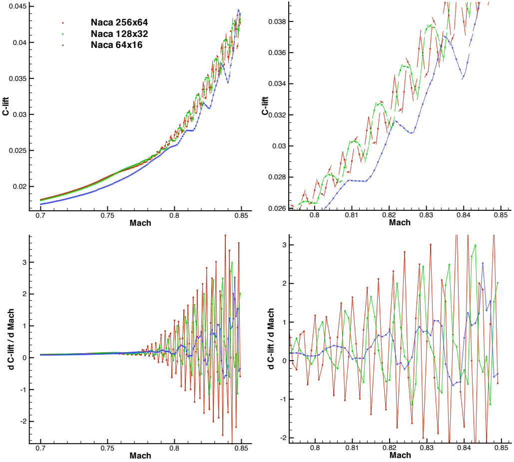


Figure 1. Lift coefficient and derivatives against Mach number for a transonic NACA0012 aerofoil on three grids. The bars in the top plots display the derivatives computed using the discrete adjoint.

(Dwight et al.)

Discrete adjoint: Frozen turbulence

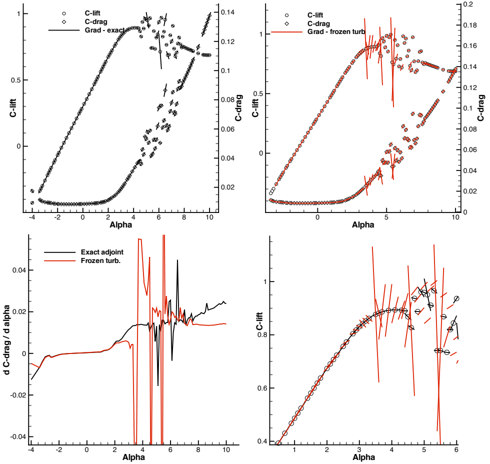


Figure 2. Lift and drag against angle-of-attack for an RAE 2822 aerofoil. Line segments represent gradients computed with a discrete adjoint code, with a full linearization of the turbulence model (black), and a frozen-turbulence approximation (red).

(Dwight et al.)

Derivative-free methods

- Using function values only
- Global, stochastic search
  - Genetic algorithm
  - Particle swarm method

$$\min_{x \in D} J(x), \quad D \subset \mathbb{R}^d$$

- Collection of N_p solutions at any iteration n

$$P^n = \{x_1^n, x_2^n, \ldots, x_{N_p}^n\} \subset D$$

- Solutions evolve according to some rules

$$P^{n+1} = E(P^n)$$

Particle swarm optimization

- Kennedy and Eberhart (1995)
- Modeled on behaviour of animal swarms: ants, bees, birds
- Cooperative behaviour of large number of individuals through simple rules
- Emergence of swarm intelligence

Optimization problem

$$\min_{x \in D} J(x), \quad D \subset \mathbb{R}^2$$

Particle swarm optimization

- Kennedy and Eberhart (1995)
- Modeled on behaviour of animal swarms: ants, bees, birds
- Cooperative behaviour of large number of individuals through simple rules
- Emergence of swarm intelligence

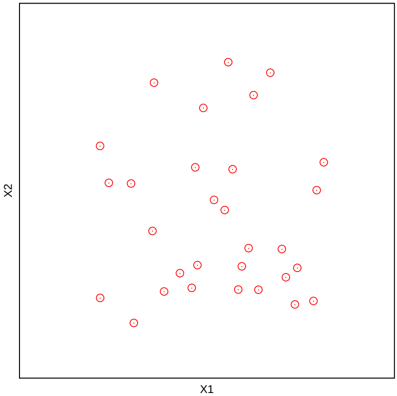
Optimization problem

$$\min_{x \in D} J(x), \quad D \subset \mathbb{R}^2$$

Particle swarm optimization

Particles distributed in design space

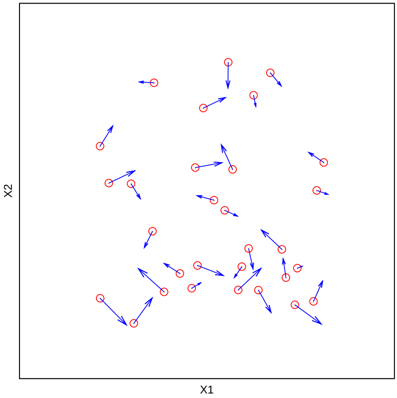
$$x_i \in D, \quad i = 1, ..., N_p$$



Particle swarm optimization

Each particle has a velocity

$$v_i \in \mathbb{R}^d, \quad i = 1, ..., N_p$$



Particle swarm optimization

- Particles have memory (t = iteration number)

$$\text{Local memory} \;\; : p_i^t = \underset{0 \le s \le t}{\operatorname{argmin}} J(x_i^s)$$

$$\text{Global memory} \;\; : p^t = \underset{i}{\operatorname{argmin}} J(p_i^t)$$

- Velocity update

$$v_i^{t+1} = \omega v_i^t + \underbrace{c_1 r_1^t \otimes (p_i^t - x_i^t)}_{Local} + \underbrace{c_2 r_2^t \otimes (p^t - x_i^t)}_{Global}$$

- Position update

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

Surrogate Models

Metamodels

- Replace

$$\min_{x \in D} J(x,u) \quad \text{s.t.} \quad R(x,u) = 0$$

  with

$$\min_{x \in D} \tilde{J}(x)$$

- Model error estimate/indicator $\sigma(x)$

$$\min_{x \in D} J_\rho(x) := \tilde{J}(x) - \rho\sigma(x), \quad \rho \geq 0$$

- Local and global search

$$x_0 = \operatorname*{argmin}_{x \in D} J_0(x) \qquad x_3 = \operatorname*{argmin}_{x \in D} J_3(x)$$

$$J(x_0), J(x_3) \Longrightarrow \text{Update } \tilde{J}$$

Kriging I

Unknown function $f : D \subset \mathbb{R}^d \to \mathbb{R}$

Given the data as $F_N = \{f_1, f_2, \ldots, f_N\} \subset \mathbb{R}$ sampled at $X_N = \{x_1, x_2, \ldots, x_N\} \subset D$, infer the function value at a new point $x_{N+1} \in D$.

Treat result of a computer simulation as a <u>fictional gaussian process</u>

F_N is assumed to be one sample of a multivariate Gaussian process with joint probability density

$$p(F_N) = \frac{\exp\left(-\frac{1}{2} F_N^\top C_N^{-1} F_N\right)}{\sqrt{(2\pi)^N \det(C_N)}} \tag{1}$$

where C_N is the $N \times N$ covariance matrix.

Kriging II

When adding a new point x_{N+1}, the resulting vector of function values F_{N+1} is assumed to be a realization of the $(N+1)$-variable Gaussian process with joint probability density

$$p(F_{N+1}) = \frac{\exp\left(-\frac{1}{2}F_{N+1}^{\top} C_{N+1}^{-1} F_{N+1}\right)}{\sqrt{(2\pi)^{N+1}\det(C_{N+1})}} \tag{2}$$

Using Baye's rule we can write the probability density for the unknown function value f_{N+1}, given the data (X_N, F_N) as

$$p(f_{N+1}|F_N) = \frac{p(F_{N+1})}{p(F_N)} = \frac{1}{Z}\exp\left[-\frac{(f_{N+1} - \hat{f}_{N+1})^2}{2\sigma_{f_{N+1}}^2}\right]$$

where

$$\underbrace{\hat{f}_{N+1} = k^{\top} C_N^{-1} F_N}_{\text{Inference}}, \qquad \underbrace{\sigma_{f_{N+1}}^2 = \kappa - k^{\top} C_N^{-1} k}_{\text{Error indicator}} \tag{3}$$

Kriging III

Covariance matrix: Given in terms of a correlation function, $C_N = [C_{mn}]$,

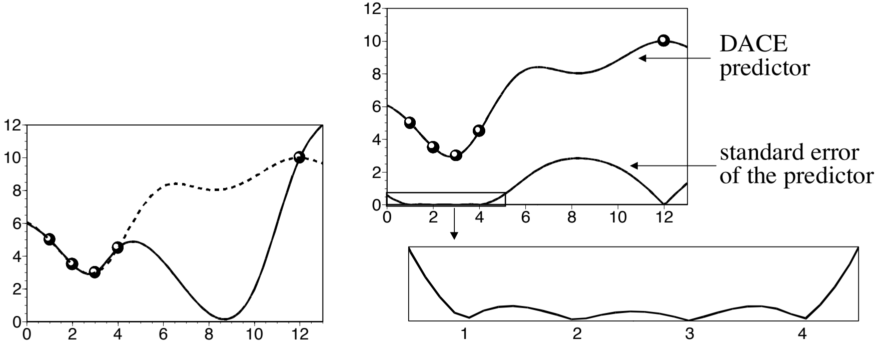
$$C_{mn} = \text{corr}(f_m, f_n) = c(x_m, x_n)$$

$$c(x, y) = \theta_1 \exp\left[-\frac{1}{2}\sum_{i=1}^{d}\frac{(x_i - y_i)^2}{{r_i}^2}\right] + \theta_2$$

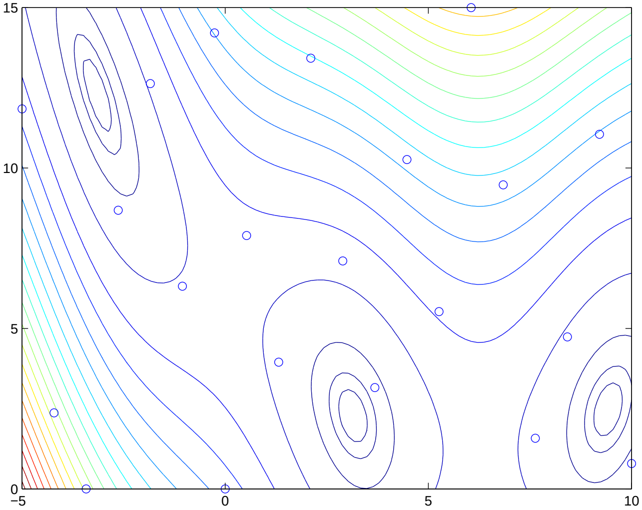
Parameters $\Theta = (\theta_1, \theta_2, r_1, r_2, \ldots, r_d)$ determined to maximize the likelihood of known data

$$\max_{\Theta} \log(p(F_N))$$

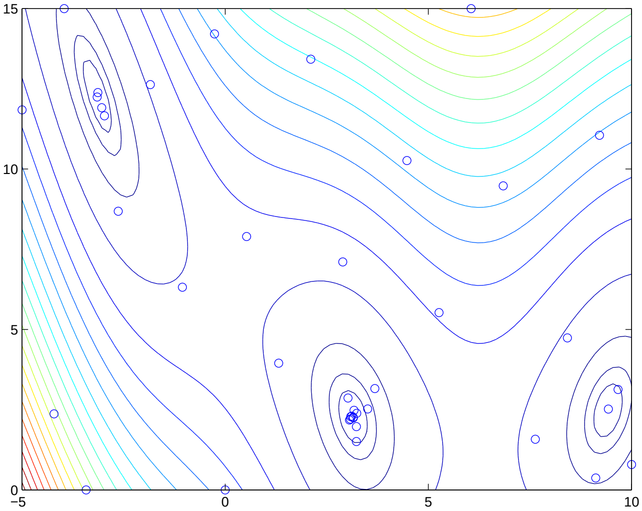
Kriging: Illustration



Minimization of 2-D Branin function: Initial database



Minimization of 2-D Branin function: after 20 iter

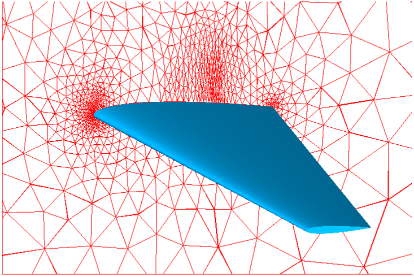


Transonic wing optimization

(with R. Duvigneau, INRIA, Sophia Antipolis)

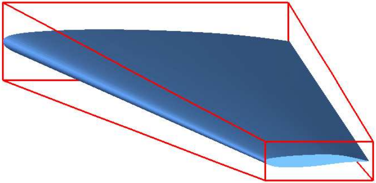
Transonic Wing shape optimization

$M_\infty = 0.83$, $\alpha = 2^o$



(Piaggio Aero. Ind.)
Grid: 31124 nodes

Free form deformation



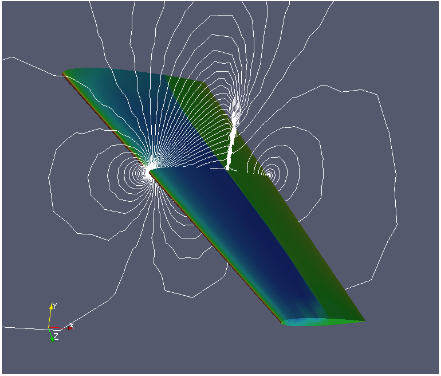
$$S_0 \xrightarrow{D(x)} S(x)$$

Minimize drag under lift and volume constraint

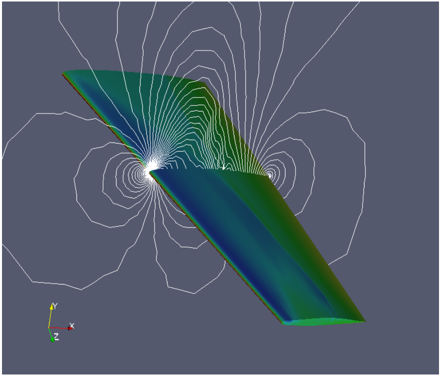
$$\min \frac{C_d}{C_{d_0}} \quad \text{s.t.} \quad \frac{C_l}{C_{l_0}} \geq 1, \quad \frac{V}{V_0} \geq 1$$

Wing optimization

Initial

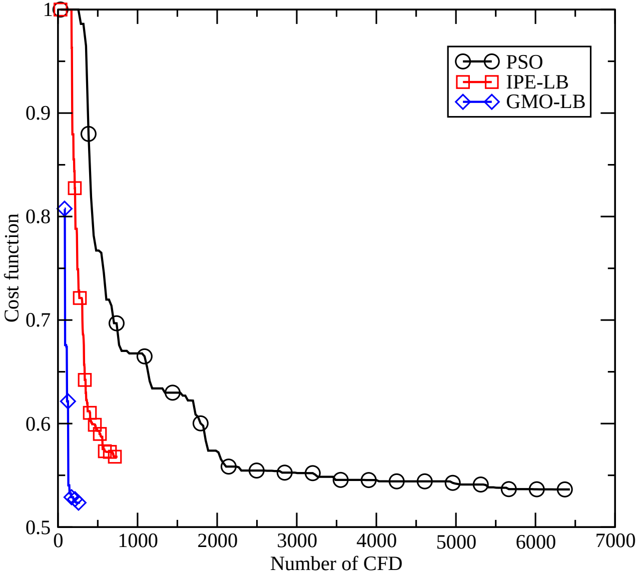


Optimized

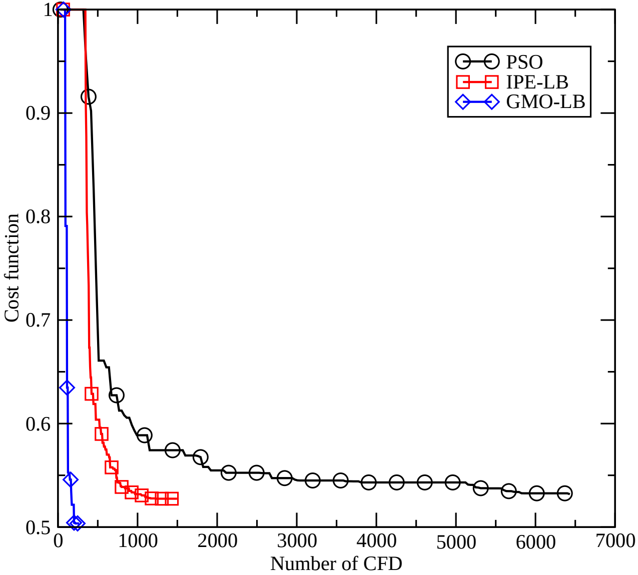


Pressure distribution

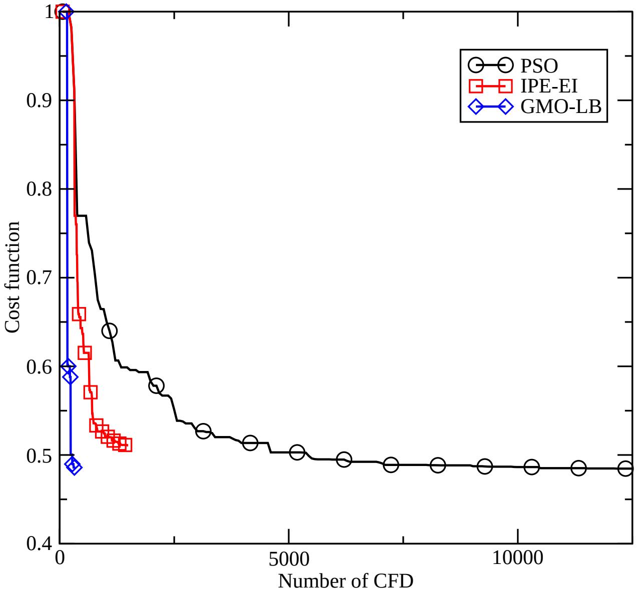
Transonic wing optimization: 8 design variables



Transonic wing optimization: 16 design variables



Transonic wing optimization: 32 design variables



Transonic, turbulent airfoil optimization

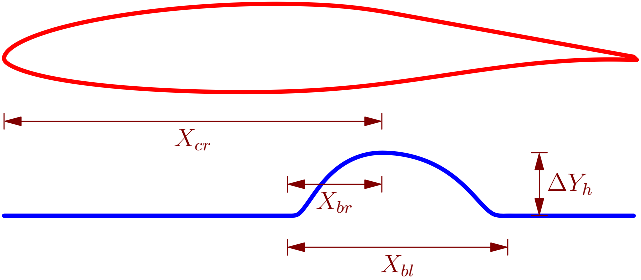
(with R. Duvigneau, INRIA, Sophia Antipolis)

TA5 test case

- Optimize RAE5243 airfoil to reduce drag under lift constraint

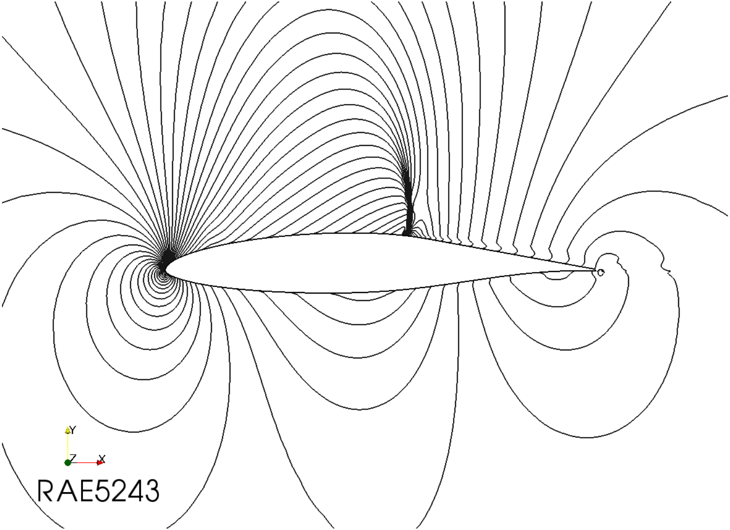
Mach	Re	C_l	Flow condition
0.68	19 million	0.82	Fully turbulent

- Modify shape of upper airfoil surface by adding a bump



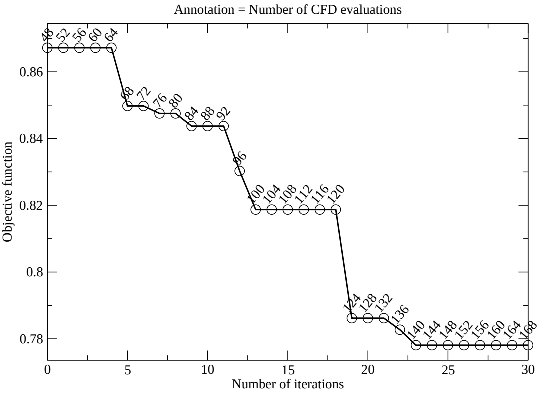
Reference solution: Pressure

$\alpha = 2.5$ deg.



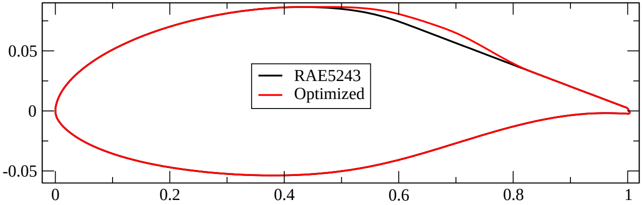
Optimization test

- 5 design variables
- Initial database of 48 using LHS
- 4 merit functions based on statistical lower bound with $\kappa = 0, 1, 2, 3$
- Gaussian process models
- Merit functions minimized using PSO



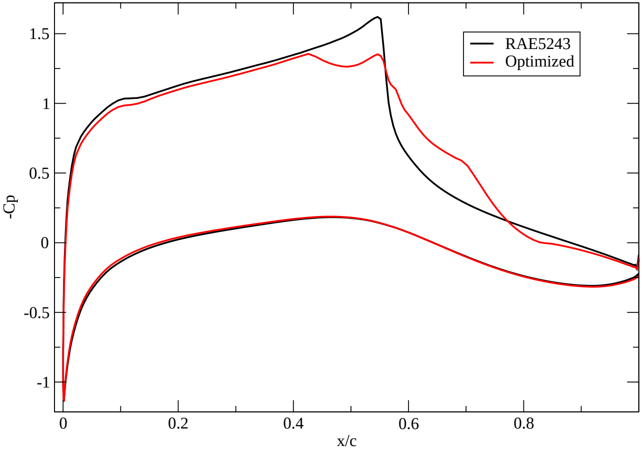
Shape parameters

Case	X_{cr}	X_{bl}	X_{br}	$\Delta Y_h \times 10^{-3}$
Present	0.688	0.399	0.257	8.578
Qin et al.	0.597	0.313	0.206	5.900

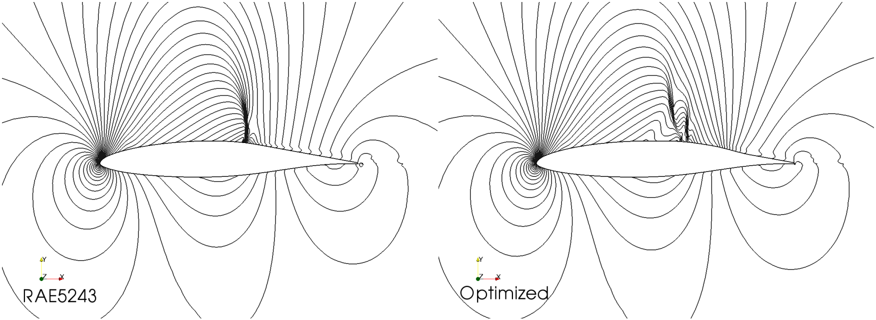


Force and Pressure coefficient

Case	C_d	ΔC_d	C_l	AOA
Present	0.01266	-22.2%	0.8204	2.19
Qin et al.	0.01326	-18.2%	0.82	-



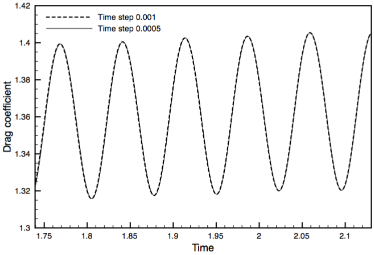
Pressure contours

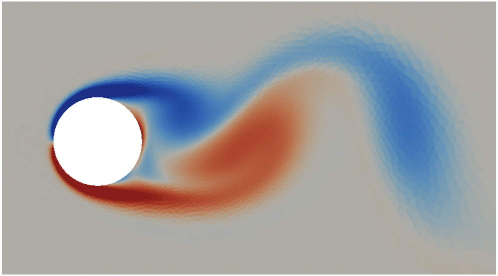


Unsteady cylinder flow

(with R. Duvigneau, INRIA, Sophia Antipolis)

- Flow past 2-D cylinder at $Re = 200$
- Periodic vortex shedding, oscillatory forces



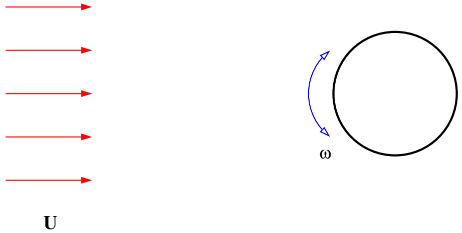


Ref.	St	Cd
Bergmann *et al.* (2005)	0.195	1.382
Braza *et al.* (1986)	0.200	1.400
Henderson (1997)	0.197	1.341
Homescu *et al.* (2002)	-	1.440
current study	0.198	1.370

Oscillating cylinder

Oscillating cylinder: Apply oscillating velocity boundary condition to cylinder wall

$$\omega(t) = A \sin(2\pi N t)$$



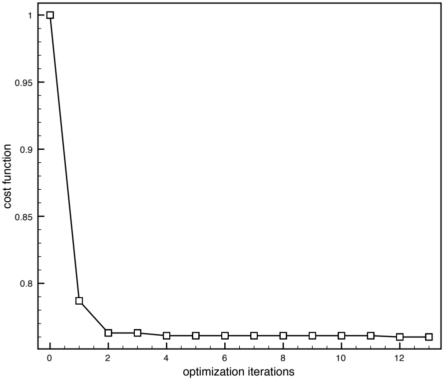
$$\text{Find } (A, N) \text{ to minimise } \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} C_D(t; A, N) dt$$

Non-dimensional variables and bounds:

$$A^* = \frac{AD}{U_\infty} \in [0, 5], \quad N^* = \frac{ND}{U_\infty} \in [0, 1]$$

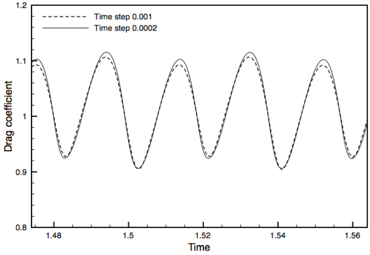
Optimization

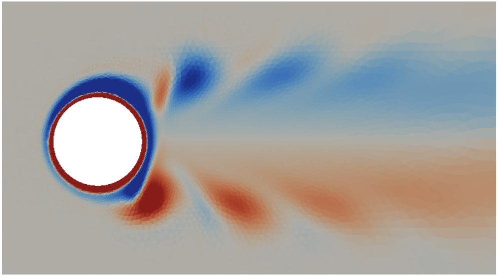
Initial sample of 16 using LHS



Good convergence in 3 iterations, 24 CFD solutions

Controlled case

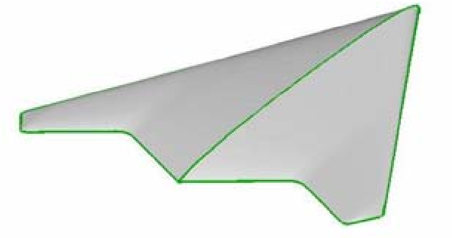




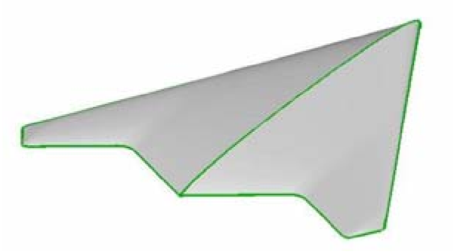
Ref.	Method	$A^\star$	$N^\star$	ΔCd
Bergmann *et al.*(2004)	POD	2.2	0.53	25%
Bergmann *et al.*(2004)	POD-ROM	4.25	0.74	30%
He *et al.*(2002)	NS 2D	3.00	0.75	30%
current study	NS 3D	3.20	0.80	25%

Optimization of flying wing

(with Biju Uthup, ADA, Bangalore)



Application to AURA



Optimization problem

Maximize Lift/Drag subject to volume constraint

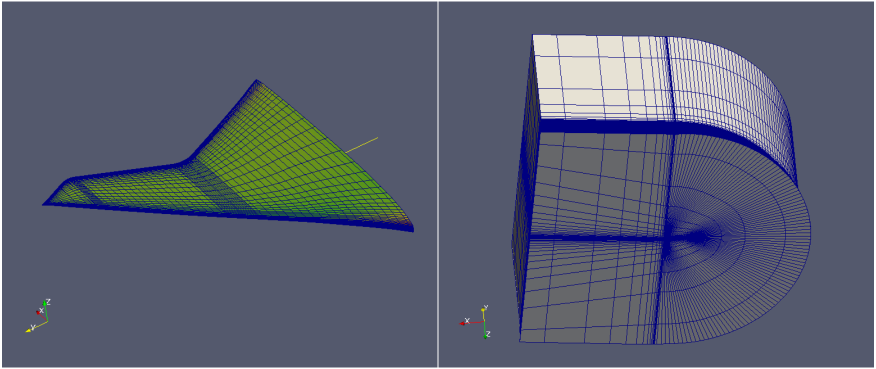
- Inviscid, compressible flow model (Euler equations)

Configuration C1A1

$$M_\infty = 0.75, \quad AOA = 2 \text{ deg.}$$

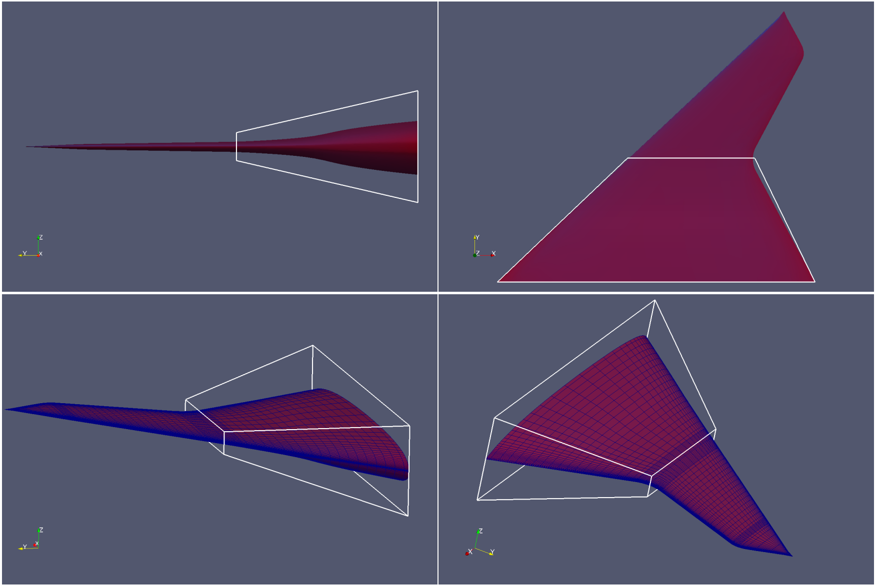
Grid for C1A1

$141 \times 20 \times 82$



On the wing: 100×72

FFD Box



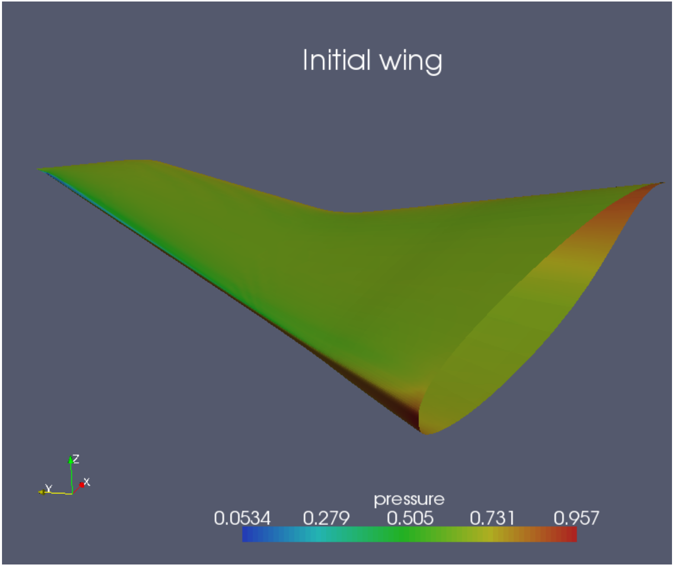
Optimization test

- 4 design variables
- Gaussian process metamodel
- Statistical lower bound merit function (4)
- Initial database of 100 designs using LHD

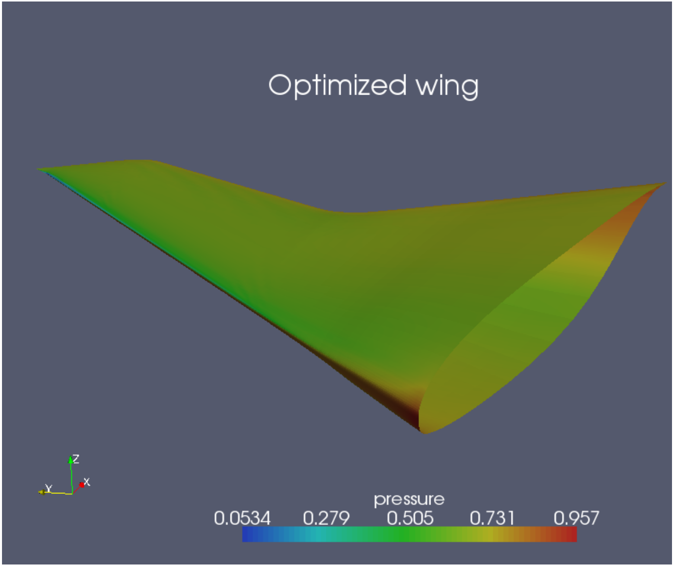
Config	L	$100D$	L/D	Improve
Initial	0.11529	0.47371	24.3	-
Optimized	0.08523	0.28928	29.4	21%

- 13 iterations, 150 CFD solutions in total
- Intel Xeon X5482 @ 3.2 GHz
- 6 process parallel job – about 7 hours

Initial wing

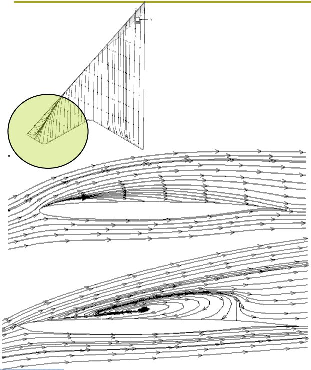


Optimized wing

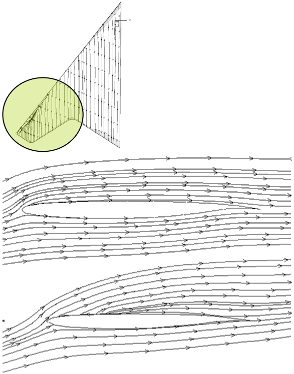


RANS computations

Studies to improve L/D for AURA : Aerodynamic Team



Un-optimized



optimized

Optimized wings for turboprops

(with R. Narasimha, S. M. Deshpande and B. R. Rakshith
JNCASR, Bangalore)

Subsonic aircraft

ATR (EADS)

RTA70 (India)

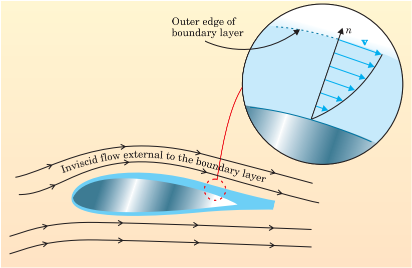
Potential flow model

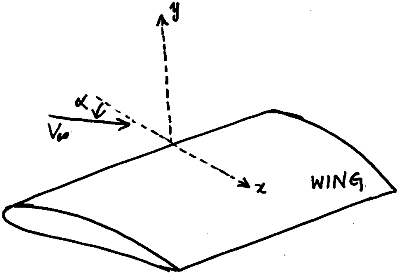
- Thin, attached boundary layers
- Velocity is irrotational

$$u = \nabla\phi$$

- Low speed flow

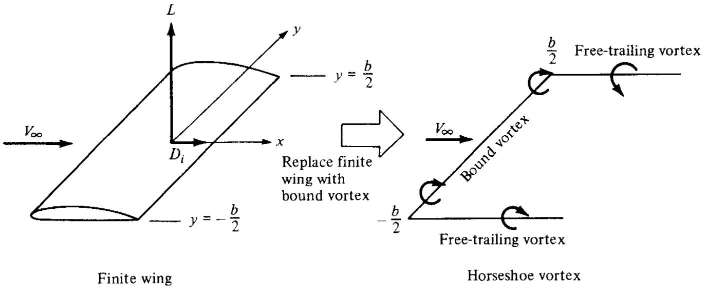
$$\begin{aligned} \Delta\phi &= 0 \\ \frac{\partial\phi}{\partial n} &= 0 \quad \text{on wing} \\ \phi &= V_\infty(x\cos\alpha + y\sin\alpha) \\ |\nabla\phi| & \quad \text{finite at TE} \end{aligned}$$





Prandtl's lifting-line model (1918)

Large aspect ratio $A = b/c \gg 1$



Formally obtained through asymptotic expansion with A^{-1} as small parameter (Van Dyke, 1975)

Concentrated line vorticity distribution

$$\omega(x, y, z) = \Gamma(y)\delta(x)\delta(z)$$

Prandtl's lifting-line model (1918)

- Integral equation for Γ

$$\Gamma(y) = \frac{1}{2}c(y)a(y)V_\infty \left[\alpha(y) - \frac{1}{4\pi V_\infty}\int_{-b/2}^{+b/2} \frac{1}{y-y'}\frac{\mathrm{d}\Gamma}{\mathrm{d}y'}\mathrm{d}y'\right]$$

with boundary conditions

$$\Gamma(-b/2) = \Gamma(+b/2) = 0$$

- Lift and drag

$$L = \int_{-b/2}^{+b/2} \rho V_\infty \Gamma(y)\mathrm{d}y$$

$$D_i = \frac{\rho}{4\pi}\int_{-b/2}^{+b/2}\int_{-b/2}^{+b/2} \frac{1}{y-y'}\Gamma(y)\frac{\mathrm{d}\Gamma}{\mathrm{d}y'}\mathrm{d}y'\mathrm{d}y$$

Optimal wings

$$\min_{c(y)} D_i, \quad L = \text{constant}$$

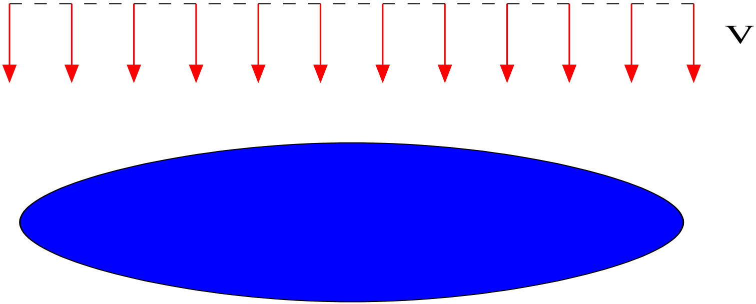
leads to elliptic circulation distribution

$$\frac{\Gamma}{\Gamma_0} + \frac{4y^2}{b^2} = 1$$

which can be achieved by elliptic wings

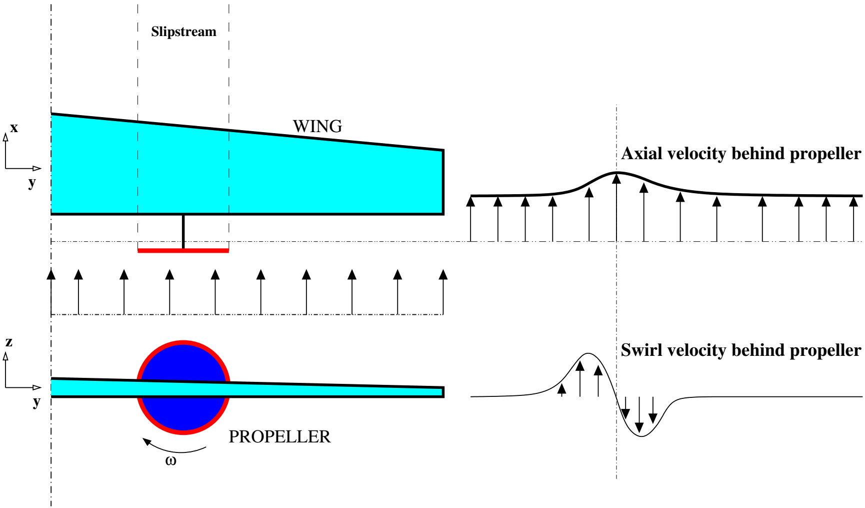
$$\frac{c^2}{c_0} + \frac{4y^2}{b^2} = 1$$

Elliptic wing



Propeller aircraft (ATR-72-600)

Propeller aircraft slipstream



Modified lifting-line model

- Velocity information by solving Euler/NS equations

$$V(y) = V_\infty + V_p(y), \qquad w_p(y)$$

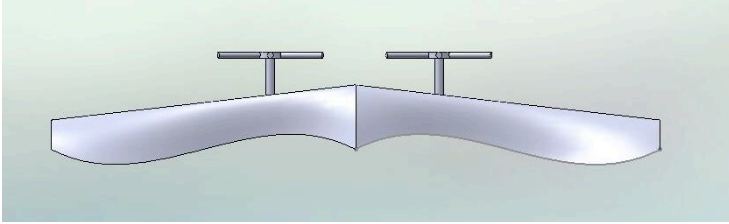
- Integral equation for Γ

$$\Gamma(y) = \frac{1}{2} c(y) a(y) V(y) \left[\alpha(y) - \frac{w_p(y)}{V(y)} - \frac{1}{4\pi V(y)} \int_{-b/2}^{+b/2} \frac{1}{y - y'} \frac{\mathrm{d}\Gamma}{\mathrm{d}y'} \mathrm{d}y' \right]$$

with boundary conditions

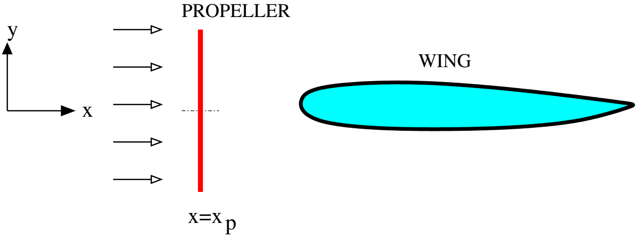
$$\Gamma(-b/2) = \Gamma(+b/2) = 0$$

Planform optimization for propeller aircraft



Induced drag reduced by 19%
Total drag reduced by 8%

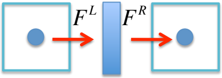
Euler/NS model for wing-propeller



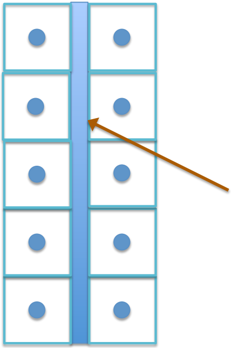
$$\frac{\partial U}{\partial t} + \nabla \cdot H = \begin{bmatrix} 0 \\ f_1 \\ f_2 \\ f_3 \\ u_1 f_1 + u_2 f_2 + u_3 f_3 \end{bmatrix} \delta(x - x_p)$$

Actuator disk: cell-based CFD

Cell centers of finite volume where flow solution is known



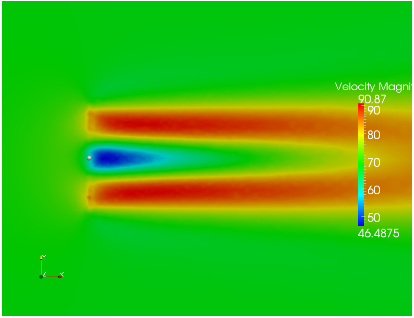
$$F^L \neq F^R$$



Actuator disk of infinitesimal thickness

Euler simulations

Axial velocity



Swirl velocity