Analytic Representations for Gradients in Shape Optimization

Thomas Slawig
TU Berlin, Institut für Mathematik
slawig@math.tu-berlin.de

MATHEON

Optimization in Aerodynamics HU Berlin May 05

Gradients in Shape Optimization for Fluid Mechanics/Aerodynamics

- What options do we have to compute gradients?
- Are there any analytic representations?
- Are they applicable?
- What has to be done to use them?

Shape optimization problem

state

control

$$\min_{\Omega} J(y,\Omega) \quad \text{s.t.} \quad e(y) = 0 \ \text{ in } \Omega$$

cost function:

drag, lift, vorticity,
prescribed state

state equation:
Potential flow
Navier-Stokes
Euler
LES
Turbulence Model

Re-write the problem...

$$\min_{\Omega} J(y,\Omega) \quad \text{s.t.} \quad e(y) = 0 \;\text{ in } \Omega$$

$$\Downarrow$$

$$\min_{\Omega} \underbrace{J(y(\Omega),\Omega)}_{=:F(\Omega)} \quad \text{s.t.} \quad e(y(\Omega),\Omega) = 0 \;\text{ in } Y$$

Computation of the gradient

via
adjoint
equation

direct:
differentiate

$$F(\Omega) = J(y(\Omega), \Omega)$$

algorithmically: AD
numerically: FD
analytically

Gradient via adjoint equation

$$F'(\Omega) = \frac{d}{d\Omega} J(y(\Omega), \Omega) = J_y(y, \Omega) y'(\Omega) + J_\Omega(y, \Omega)$$

Differentiate

$$e(y(\Omega), \Omega) = 0 \quad \text{in } Y:$$

$$e_y(y, \Omega) y'(\Omega) + e_\Omega(y, \Omega) = 0$$

Compute adjoint variable

$$\langle \lambda, e_y(y, \Omega) v \rangle = -J_y(y, \Omega) v \quad \forall v$$

Then

$$-\langle \lambda, e_\Omega(y, \Omega) \rangle = \langle \lambda, e_y(y, \Omega) y'(\Omega) \rangle = -J_y(y, \Omega) y'(\Omega)$$

Finally:

$$F'(\Omega) = e_\Omega(y, \Omega)^* \lambda + J_\Omega(y, \Omega)$$

... to compute

$$F'(\Omega) = e_\Omega(y,\Omega)^*\lambda + J_\Omega(y,\Omega)$$

Adjoint equation:

$$\langle \lambda, e_y(y,\Omega)v \rangle = -J_y(y,\Omega)v \quad \forall v$$

Direct ways to compute the gradient

- AD
- FD
- analytically ...

$$F'(\Omega)\omega := \lim_{t \to 0} \frac{F(\Omega + t\omega) - F(\Omega)}{t}$$

Questions:

How to define domain perturbation ω ?

How to compare solutions y on different domains?

Function space setting?

What has been done?

- Laplace equation in 2+3D *Potential Flow*
  Boundary integral representation
- *incompressible Navier-Stokes equations*
  low Reynolds number
  Fictitious domain formulation
- ...

Potential flow

Assumption: zero vorticity $\quad \text{rot}\,\vec{v} = 0$

Velocity Potential $\quad \Phi : \quad \vec{v} = \nabla \Phi$

solves Laplace equation

$$\begin{aligned} \Delta\Phi &= 0 && \text{in} \quad \Omega \\ \frac{\partial \Phi}{\partial n} &= \vec{v}\cdot\vec{n} && \text{on} \quad \partial\Omega \end{aligned}$$

integral representation

$$\Phi(x) = \int_{\partial\Omega} \left(U(x,y) \frac{\partial \Phi}{\partial n_y}(y) - \Phi(y) \frac{\partial U}{\partial n_y}(x,y) \right) ds_y$$

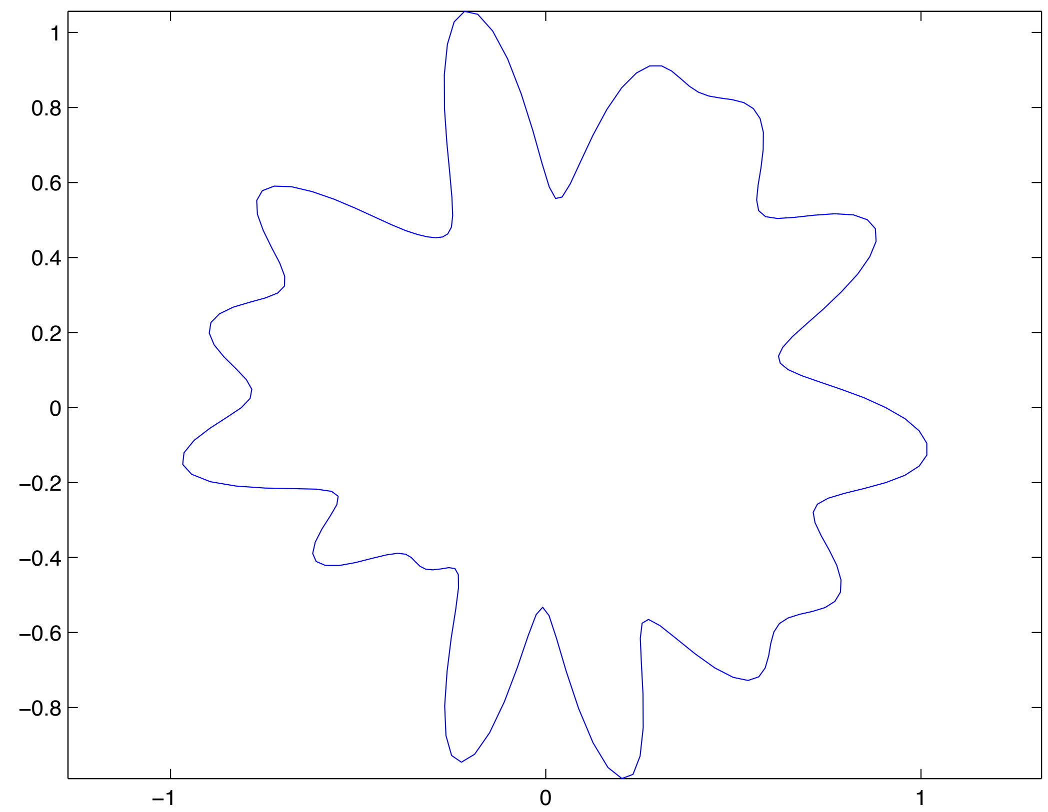
$$U(x,y) = (4\pi |x-y|)^{-1}$$

MATHEON

TU berlin

Star-shaped domains

$$\partial\Omega = \left\{ r(\phi) \begin{pmatrix} \cos(\phi) \\ \sin(\phi) \end{pmatrix} : \phi \in [0, 2\pi] \right\}, r \in C^2[0, 2\pi]$$



$$\partial\Omega_\varepsilon = \left\{ (r + \varepsilon s)(\phi) \begin{pmatrix} \cos(\phi) \\ \sin(\phi) \end{pmatrix} : \phi \in [0, 2\pi] \right\}, s \in C^2[0, 2\pi]$$

Gradient for star-shaped domains

(K. Eppler, Dirichlet b.c.)

$$\Phi(x) = (K\mu)(x) := \int_{\partial\Omega} \frac{\partial U(x,y)}{\partial n} \mu(y) ds$$

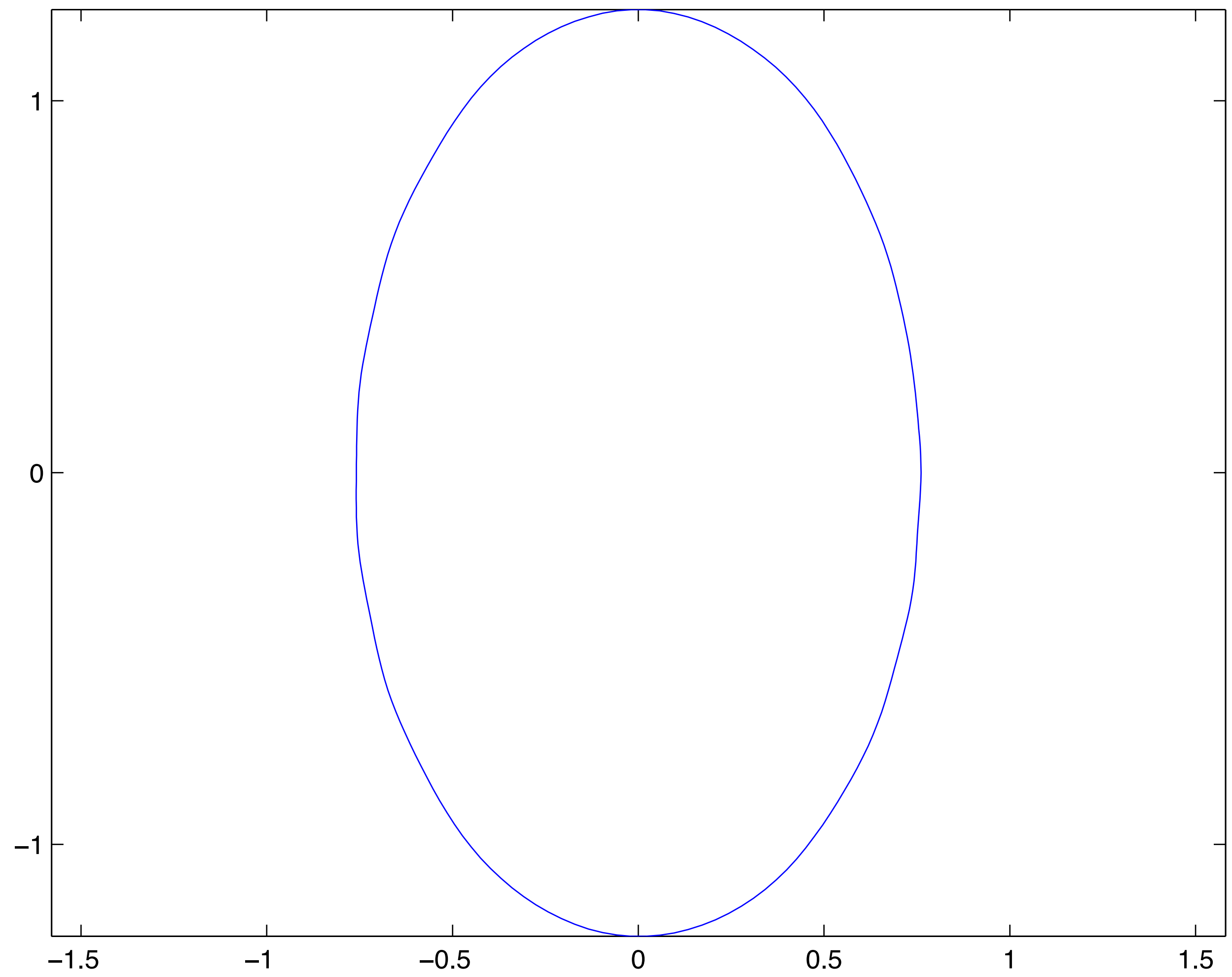
μ solves integral equation

$$\left(\frac{1}{2}I + K\right)\mu = g \quad \text{(Dirichlet data) on } \partial\Omega$$

its derivative μ' solves

$$\left(\frac{1}{2}I + K\right)\mu' = g' - K'\mu \quad \text{on } \partial\Omega$$

Example 2-D:



computed by K. Eppler & H. Harbrecht

Summary Potential Flow

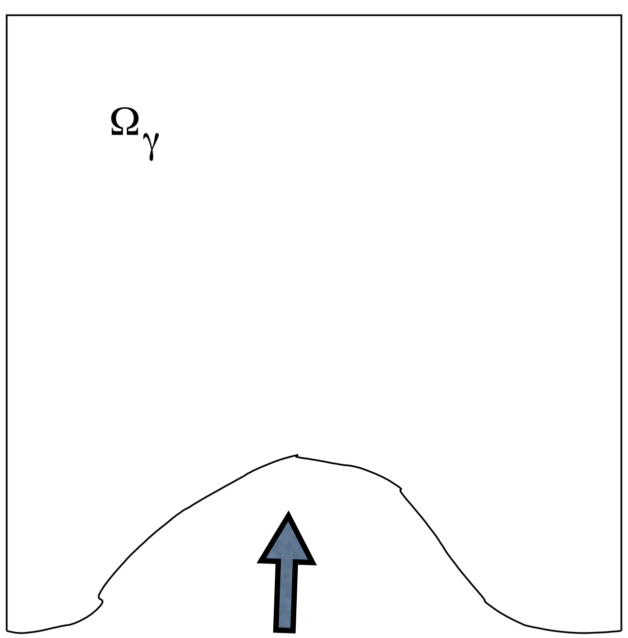
- Gradient for star-shaped domains available
- second derivatives
- state equation (BIE) solver may be used or modified
- BIE technique possible for Stokes

Questions

- Potential flow/Stokes flow useful for applications?
- Optimum from potential flow as first guess for other (more complex) equations?
- Star-shaped domains sufficient?

Incompressible Navier-Stokes equations: Embedding (Fictitious) Domain Technique

Idea: Instead of computing on varying domains Ω_γ during the optimization process ...

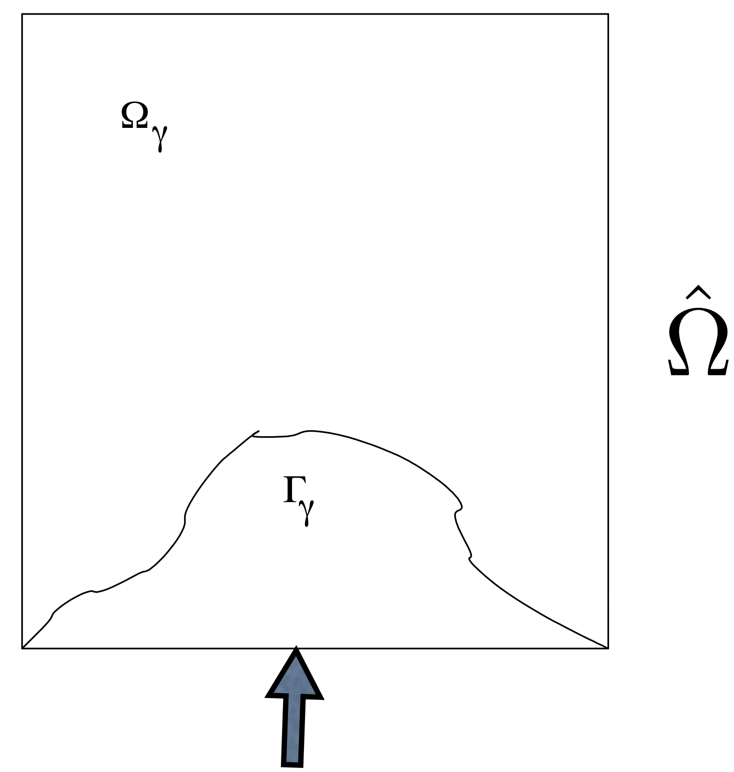


varying part of domain boundary

$$\Gamma_\gamma = \{(x, \gamma(x)) | x \in I\}$$

...compute on one fixed *fictitious* domain $\hat{\Omega}$ satisfying

$$\Omega_\gamma \subset \hat{\Omega} \quad \forall \gamma$$



fictitious boundary

Reformulation of the NSE

Navier-Stokes (weak formulation) on Ω_γ

$$
\begin{aligned}
(u \cdot \nabla u, v) + \nu(\nabla u, \nabla v) - (p, \operatorname{div} v) &= (f, v) \quad \forall v \in H_0^1(\Omega_\gamma)^d \\
(\operatorname{div} u, q) &= 0 \quad \forall q \in L_0^2(\Omega_\gamma).
\end{aligned}
$$

Fictitious Domain formulation on $\hat{\Omega}$

$$
\begin{aligned}
(\hat{u} \cdot \nabla \hat{u}, \hat{v}) + \nu(\nabla \hat{u}, \nabla \hat{v}) - (p, \operatorname{div} \hat{v}) - (g, v)_{\Gamma_\gamma} &= (\tilde{f}, \hat{v}) \quad \forall \hat{v} \in H_0^1(\hat{\Omega})^d \\
(\operatorname{div} \hat{u}, \hat{q}) &= 0 \quad \forall \hat{q} \in L_0^2(\hat{\Omega}) \\
u &= 0 \quad \text{on } \Gamma_\gamma
\end{aligned}
$$

Compute solution on $\hat{\Omega}$ and restrict it to Ω_γ :

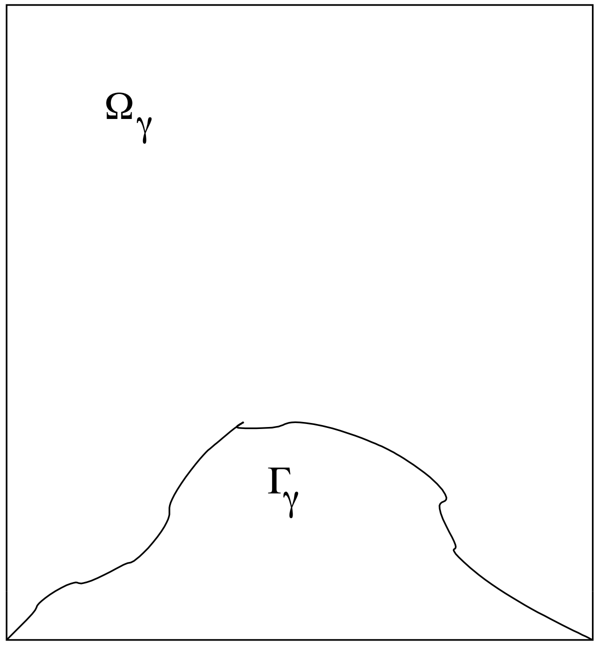
$$
(u, p) = (\hat{u}, \hat{p})|_{\Omega_\gamma}
$$

Taking advantage of the Lagrange multipier...

$$
\begin{aligned}
(\hat{u}\cdot\nabla\hat{u},\hat{v}) + \nu(\nabla\hat{u},\nabla\hat{v}) - (p,\operatorname{div}\hat{v}) - (g,v)_{\Gamma_\gamma} &= (\tilde{f},\hat{v}) && \forall\hat{v}\in H_0^1(\hat{\Omega})^d \\
(\operatorname{div}\hat{u},\hat{q}) &= 0 && \forall\hat{q}\in L_0^2(\hat{\Omega}) \\
u &= 0 && \text{on } \Gamma_\gamma
\end{aligned}
$$

Lagrange multiplier equals normal derivative of velocity:

$$g = \frac{\partial u}{\partial n} \quad \text{on } \Gamma_\gamma$$



... and can be used to compute the gradient

directional derivative of the cost function:

$$F'(\gamma)\bar{\gamma} = \frac{1}{\nu}\int_I \Big(g(x,\gamma(x))\cdot\chi(x,\gamma(x)) - p(x,\gamma(x))\mu(x,\gamma(x))\Big)\bar{\gamma}(x)dx$$

adjoint equation in *Fictitious Domain* formulation:

$$\begin{array}{rcll} \left.\begin{array}{r} (\hat{\lambda}, \hat{u}\cdot\nabla\hat{v} + \hat{v}\cdot\nabla\hat{u})_{\hat{\Omega}} \\ +\nu(\nabla\hat{\lambda},\nabla\hat{v})_{\hat{\Omega}} - (\hat{\mu}, \operatorname{div}\hat{v})_{\hat{\Omega}} - (\chi,\hat{v})_{\Gamma_\gamma} \end{array}\right\} & = & -J_u(u,\gamma)\hat{v} & \forall \hat{v}\in H_0^1(\hat{\Omega})^d \\ (\operatorname{div}\hat{\lambda},\hat{q})_{\hat{\Omega}} & = & 0 & \forall \hat{q}\in L_0^2(\hat{\Omega}) \\ \hat{\lambda} & = & 0 & \text{on } \Gamma_\gamma. \end{array}$$

Summary Fictitious Domain:

- Gradient can be computed with low effort
- Fictitious Domain formulation must be used
- Similar technique is used in FeatFlow for the solution on time-dependent domains

Questions for Fictitious domain method

- Sufficient to treat boundaries that are graphs?
- How big is the effort to introduce FD technique to really applicable solvers?
- Modifying FeatFlow?

Questions BIE method

- Potential flow/Stokes flow useful for applications?
- Optimum from potential flow as first guess for other (more complex) equations?
- Star-shaped domains sufficient?