Spline boxes for shape optimization

Shape parametrization and sensitivity analysis for optimal flow problems

Eduard Rohan Robert Cimrman

Katedra mechaniky, Fakulta aplikovaných věd,
Výzkumné centrum Nové technologie,
Západočeská univerzita v Plzni

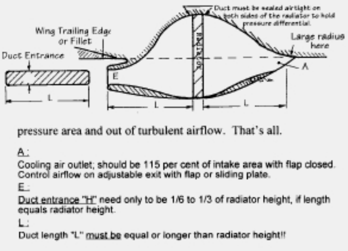
Seminář SIGA, ÚT AV ČR Praha, 9.2.2011

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Outline

Outline

Shape Optimization of Flow in Ducts



Applications

Automotive and Aerospace Industry

- efficient cooling
- exhaust piping
- jet engines, turbines ...
- wing and blade profiles
- vehicle aerodynamics ...

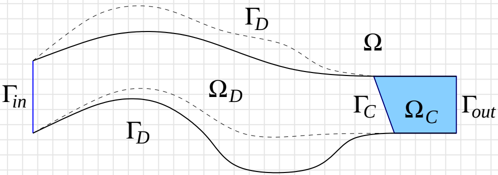
Merits of optimization

to obtain better velocity profiles

- desired distribution at the outlet ... "control" part
- minimizing pressure losses
- reducing wear of downstream parts
- flows in compliant vessels — passive control (material + design)

Outline

Optimization Problem



Objectives of the optimal flow problem

$$\Psi_\Gamma(\mathbf{u}) \equiv \frac{\nu}{2} \int_{\Omega_C} |\nabla \mathbf{u}|^2 \longrightarrow \min$$

- Merit: minimize gradients of solution (e.g. losses) in $\Omega_C \subset \Omega$
- by moving design boundary $\Gamma_D \subset \partial\Omega$
- perturbation of Γ_D by design velocity field $\vec{\mathcal{V}}$,

$$\Omega(t) = \Omega + \{t\vec{\mathcal{V}}(x)\}_{x\in\Omega} \quad \text{where } \vec{\mathcal{V}} = 0 \text{ in } \bar{\Omega}_C \cup \Gamma_{\text{in}}$$

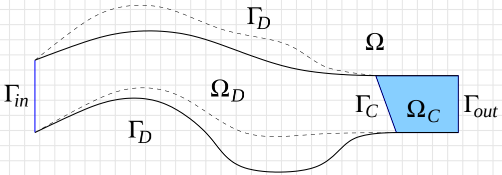
State problem

N.-S. for incompressible flow

$$-\nu\nabla^2\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p = 0 \quad \text{in } \Omega\ ,$$
$$\nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega\ ,$$

boundary conditions

$$\mathbf{u} = \mathbf{u}_0 \quad \text{on } \Gamma_{\text{in}}\ ,$$
$$-p\mathbf{n} + \nu\frac{\partial\mathbf{u}}{\partial n} = -\bar{p}\mathbf{n} \quad \text{on } \Gamma_{\text{out}}\ ,$$
$$\mathbf{u} = 0 \quad \text{on } \partial\Omega\setminus\Gamma_{\text{in-out}}\ .$$



State problem – Weak formulation

find $\mathbf{u} \in \mathbf{V}_{\bar{\mathbf{u}}}(\Omega)$ and $p \in L^2(\Omega)$ such that

$$a_\Omega(\mathbf{u},\, \mathbf{v}) + c_\Omega(\mathbf{u},\, \mathbf{u},\, \mathbf{v}) - b_\Omega(\mathbf{v},\, p) = -\bar{g}(\mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{V}_0\,,$$
$$b_\Omega(\mathbf{u},\, q) = 0 \quad \forall q \in L^2(\Omega)\,.$$

where

$$\mathbf{V}_{\bar{\mathbf{u}}} = \{\mathbf{v} \in \mathbf{H}^1(\Omega) \,|\, \mathbf{v} = \bar{\mathbf{u}} \text{ on } \partial\Omega \setminus \Gamma_{\mathrm{in-out}}\}\,,$$

functional forms:

$$a_\Omega(\mathbf{u},\, \mathbf{v}) = \nu \int_\Omega \nabla\mathbf{u} : \nabla\mathbf{v} = \nu \int_\Omega \frac{\partial u_i}{\partial x_k}\frac{\partial v_i}{\partial x_k}\,, \quad b_\Omega(\mathbf{u},\, p) = \int_\Omega p\,\mathrm{div}\mathbf{u}\,,$$
$$c_\Omega(\mathbf{w},\, \mathbf{u},\, \mathbf{v}) = \int_\Omega (\mathbf{w}\cdot\nabla\mathbf{u})\cdot\mathbf{v} = \int_\Omega w_k \frac{\partial u_i}{\partial x_k} v_i\,, \quad \bar{g}(\mathbf{v}) = \int_{\Gamma_{\mathrm{in-out}}} \bar{p}\,\mathbf{v}\cdot\mathbf{n}\,dS\,.$$

Optimal problem formulation

"Improvement" of the reference (initial) design Ω_0

$$\min_{\Gamma_D} \Psi(\mathbf{u}) \,,$$

$$\text{subject to: } (\mathbf{u}, p) \text{ satisfy the STATE PROBLEM,}$$

$$\Gamma_D \text{ in } \mathcal{U}_{ad}(\Omega_0) \,,$$

$$\text{where } \Psi(\mathbf{u}) = \frac{\nu}{2}\int_{\Omega_C} |\nabla \mathbf{u}|^2 = \frac{1}{2} a_{\Omega_C}(\mathbf{u}, \mathbf{u}) \,.$$

- $\Psi(\mathbf{u})$ does not depend on Γ_D explicitly,
- Shape design restriction related to Ω_0

Outline

Sensitivity analysis — Design velocity

Domain sensitivity approach

- $\Omega \subset \mathcal{B} \subset \mathbb{R}^3$
- design velocity field $\vec{\mathcal{V}} : \mathcal{B} \to \mathbb{R}^3$

$$\Omega(t) = \Omega + \{t\vec{\mathcal{V}}(x)\}_{x \in \Omega}$$

- total directional derivative δf
- partial design derivative $\delta_D f$
- total sensitivity: consider $f_\Omega : L^2(\Omega) \to \mathbb{R}$, then

$$\delta f_\Omega(u) = \delta_D f_\Omega(u) + \delta_u f_\Omega(u) \circ \delta u \ ,$$

Feasible design velocity fields

$$\mathrm{supp}\vec{\mathcal{V}} \cap \overline{\Omega} \subset \overline{\Omega_D} \text{ and } \vec{\mathcal{V}} = 0 \text{ on } \Gamma_{\mathrm{in}} \cup \Gamma_C \ ,$$

$$\vec{\mathcal{V}} \text{ is differentiable in } \Omega_D.$$

Partial design derivative I

Scalar parametrization of *design domain*

$$\overline{\Omega_D}(\tau) = \{y\} \quad \text{where} \quad y_i(x,\tau) = x_i + \tau \mathcal{V}_i(x)\ , \quad x \in \overline{\Omega_D},\ \tau \in \mathbb{R}\ .$$

Partial design derivative of $f_\Omega(u)$

$$\delta_D f_\Omega(u) = \frac{\mathrm{d}}{\mathrm{d}\,\tau} \left(f_{\Omega_D(\tau)}(u) \right)_{\tau=0}\ .$$

Auxiliary formulae: $J(y(x,\tau)) = \det[\partial y_i(x,\tau)/\partial x_j]$

$$\delta_D \left(\frac{\partial y_i}{\partial x_j} \right) = \frac{\mathrm{d}}{\mathrm{d}\,\tau} \left(\frac{\partial y_i(x,\tau)}{\partial x_j} \right)_{\tau=0} = \frac{\partial \mathcal{V}_i(x)}{\partial x_j}\ ,$$

$$\delta_D \left(\frac{\partial x_k}{\partial y_j} \right) = \frac{\mathrm{d}}{\mathrm{d}\,\tau} \left(\frac{\partial x_k}{\partial y_j(x,\tau)} \right)_{\tau=0} = - \frac{\partial \mathcal{V}_k(x)}{\partial x_j}\ ,$$

$$\delta_D \left(J(y) \right) = \frac{\mathrm{d}}{\mathrm{d}\,\tau} \left(J(y(x,\tau)) \right)_{\tau=0} = \frac{\partial \mathcal{V}_i(x)}{\partial x_i} = \mathrm{div} \vec{\mathcal{V}}\ .$$

Partial design derivative II (Application)

Rewrite the functional for the perturbed domain $\Omega_D(\tau)$:

$$a_{\Omega_{D(\tau)}}(\mathbf{u},\,\mathbf{w}) = \nu \int_{\Omega_D(\tau)} \frac{\partial u_i}{\partial y_k(\tau)} \frac{\partial w_i}{\partial y_k(\tau)}\, dy$$

$$= \nu \int_{\Omega_D} \frac{\partial u_i(x)}{\partial x_j} \frac{\partial x_j}{\partial y_k(x,\tau)} \frac{\partial w_i(x)}{\partial x_l} \frac{\partial x_l}{\partial y_k(x,\tau)} J(y(x,\tau))\, dx\,.$$

...and apply the differentiation $\delta_D(\cdot) = \frac{\mathrm{d}}{\mathrm{d}\,\tau}(\cdot)_{\tau=0}$

Shape sensitivities of the functionals involved: ...linear in $\mathcal{V}$

$$\delta_D a_\Omega(\mathbf{u},\,\mathbf{w}) = \nu \int_{\Omega_D} \left[\frac{\partial u_i}{\partial x_k}\frac{\partial w_i}{\partial x_k}\mathrm{div}\mathcal{V} - \frac{\partial \mathcal{V}_j}{\partial x_k}\frac{\partial u_i}{\partial x_j}\frac{\partial w_i}{\partial x_k} - \frac{\partial u_i}{\partial x_k}\frac{\partial \mathcal{V}_l}{\partial x_k}\frac{\partial w_i}{\partial x_l}\right]\,.$$

$$\delta_D c_\Omega(\mathbf{u},\,\mathbf{u},\,\mathbf{w}) = \int_{\Omega_D} \left[u_k \frac{\partial u_i}{\partial x_k}\, w_i\,\mathrm{div}\mathcal{V} - u_k \frac{\partial \mathcal{V}_j}{\partial x_k}\frac{\partial u_i}{\partial x_j}\, w_i \right]\,,$$

$$\delta_D b_\Omega(\mathbf{u},\,q) = \int_{\Omega_D} q\,\left[\mathrm{div}\mathbf{u}\,\mathrm{div}\mathcal{V} - \frac{\partial \mathcal{V}_k}{\partial x_i}\frac{\partial u_i}{\partial x_k}\right]\,.$$

Remark: *Domain vs. Boundary method*

Consider functional: $F_\Omega(\mathbf{u}) = \int_\Omega f(x, \mathbf{u})\, dx$

perturbed: $$F_{\Omega(\tau)}(\mathbf{u}(\tau)) = \int_{\Omega(\tau)} f(y(\tau), \mathbf{u}(\tau))\, dz$$

Boundary method — $\vec{\mathcal{V}}$ defined only on $\partial\Omega$:-)

$$\delta F_\Omega(\mathbf{u}) = \int_\Omega \left[\partial_\mathbf{u} f \left. \frac{\partial \mathbf{u}}{\partial \tau} \right|_{\tau=0} + \vec{\mathcal{V}} \cdot \nabla f + f \operatorname{div} \vec{\mathcal{V}} \right] dx$$
$$= \int_\Omega \partial_\mathbf{u} f \left. \frac{\partial \mathbf{u}}{\partial \tau} \right|_{\tau=0} dx + \int_{\partial\Omega} f(\mathbf{u}) \vec{\mathcal{V}} \cdot \vec{n}\, dS$$

Application: $f = \nabla \mathbf{u}$ . . . defined in Ω

$$\delta F_\Omega(\mathbf{u}) = \int_\Omega \nabla \left. \frac{\partial \mathbf{u}}{\partial \tau} \right|_{\tau=0} dx + \int_{\partial\Omega} \nabla \mathbf{u}\, \vec{\mathcal{V}} \cdot \vec{n}\, dS$$

. . . gradients $\nabla \mathbf{u}$ needed on boundary $\partial\Omega$! :-(

Remark: *Domain vs. Boundary method*

Domain method — $\vec{\mathcal{V}}$ defined in Ω_D :-/

It can be derived that:

$$\delta F_\Omega(\mathbf{u}) = \int_\Omega \left[\nabla \delta \mathbf{u} - (\nabla \vec{\mathcal{V}}) \cdot (\nabla \mathbf{u}) + \nabla \mathbf{u}\, \mathrm{div} \vec{\mathcal{V}}\right] \, dx$$

... gradients $\nabla \mathbf{u}$ needed only in Ω, no traces needed on $\partial\Omega$:-)

Advantages of the domain method:

- more accurate (well-defined gradients of the solution)
- can be efficient if $\vec{\mathcal{V}}$ in Ω can be "computed" in cheap way.
- $\Rightarrow$ use the FFD! (domain parametrization)

Sensitivity analysis

How to *avoid* computing the state sensitivity?

Lagrangian function

$$\mathcal{L}(\Gamma_D, \mathbf{u}, p, \mathbf{w}, q) = \Psi_\Gamma(\mathbf{u}) + a_\Omega(\mathbf{u}, \mathbf{w}) + c_\Omega(\mathbf{u}, \mathbf{u}, \mathbf{w}) - b_\Omega(\mathbf{w}, p) + b_\Omega(\mathbf{u}, q) + \bar{g}(\mathbf{u})$$

with $\mathbf{w} \in \mathbf{V}_0$, $q \in L^2(\Omega)$ ... the Lagrange multipliers

Consider paths of admissible states $(\mathbf{u}, p)$

Variation $\delta\mathcal{L}$ w.r.t. $(\Gamma_D, \mathbf{u}, p)$ in direction $(\mathcal{V}, \delta\mathbf{u}, \delta p) = ?$

$$\begin{aligned}\delta\mathcal{L} = {} & \delta_D a_\Omega(\mathbf{u}, \mathbf{w}) + \delta_D c_\Omega(\mathbf{u}, \mathbf{u}, \mathbf{w}) - \delta_D b_\Omega(\mathbf{w}, p) + \delta_D b_\Omega(\mathbf{u}, q) \\ & + a_\Omega(\delta\mathbf{u}, \mathbf{w}) + c_\Omega(\delta\mathbf{u}, \mathbf{u}, \mathbf{w}) + c_\Omega(\mathbf{u}, \delta\mathbf{u}, \mathbf{w}) - b_\Omega(\mathbf{w}, \delta p) \\ & + b_\Omega(\delta\mathbf{u}, q) + \delta_u\Psi(\mathbf{u}) \circ \delta\mathbf{u} + \text{ var. w.r.t. Lagr. mult.} \\ & \stackrel{!}{=} \delta\Psi(\mathbf{u}) ,\end{aligned}$$

Sensitivity analysis — Adjoint state

KKT conditions — state admissibility

Optimality conditions $\delta_{\mathbf{u},p}\mathcal{L} = 0$ yield Adjoint State Problem for $\mathbf{w}$, q

$$\delta_{\mathbf{u}}\mathcal{L} \circ \mathbf{v} = 0 = \delta_u \Psi(\mathbf{u}) \circ \mathbf{v} + a_\Omega(\mathbf{v}, \mathbf{w}) + c_\Omega(\mathbf{v}, \mathbf{u}, \mathbf{w}) + c_\Omega(\mathbf{u}, \mathbf{v}, \mathbf{w}) + b_\Omega(\mathbf{v}, q) \;,$$

$$\delta_p\mathcal{L} \circ \eta = 0 = b_\Omega(\mathbf{w}, \eta) \;,$$

for all $\mathbf{v} \in \mathbf{V}_0$ and for all $\eta \in L^2(\Omega)$.

Elimination of state variables sensitivity using $\mathbf{v} := \mathbf{w}$, $\eta := q$

$$\delta\Psi(\mathbf{u}) = \delta\mathcal{L} = \delta_D a_\Omega(\mathbf{u}, \mathbf{w}) + \delta_D c_\Omega(\mathbf{u}, \mathbf{u}, \mathbf{w}) - \delta_D b_\Omega(\mathbf{w}, p) + \delta_D b_\Omega(\mathbf{u}, q) + a_\Omega(\delta\mathbf{u}, \mathbf{w}) + c_\Omega(\delta\mathbf{u}, \mathbf{u}, \mathbf{w}) + c_\Omega(\mathbf{u}, \delta\mathbf{u}, \mathbf{w}) - b_\Omega(\mathbf{w}, \delta p) + b_\Omega(\delta\mathbf{u}, q) + \delta_u\Psi(\mathbf{u}) \circ \delta\mathbf{u}$$

$\Rightarrow$ only partial shape sensitivities must be evaluated (for a given $\vec{\mathcal{V}}$).

Outline

Domain vs. Surface parametrization

Usual types of surface parametrization

- nodal positions of the FE surface mesh (mesh dependence, smoothness constraints!)
- spline-parametrized surface – too much restriction ???

Difficulties :-(

- description of complex shapes
- need to compute $\vec{\mathcal{V}}$ $\Rightarrow$ domain parametrization

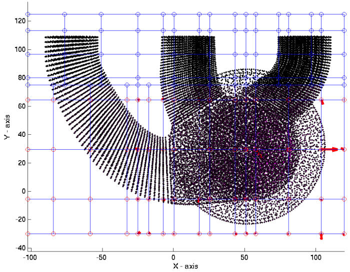
Free-Form Deformation – *Volume* parametrization

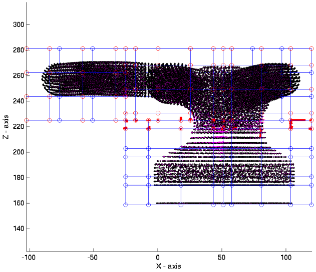
using B-spline or Bézier functions

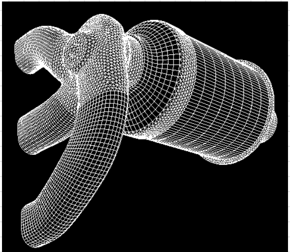
- direct domain parametrization — $\vec{\mathcal{V}}$ obtained in Ω_D :-)
- non-complicated handle to complex shapes :-)
- :-/ "bulk behaviour" (in general) $\Rightarrow$ rough shape modifications

Domain Parametrization — Spline Boxes

Complex structure partitioned using the Spline Boxes





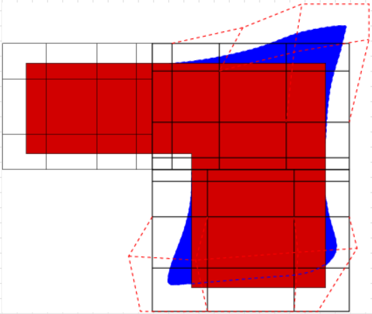


Domain Parametrization — Spline Boxes I

Spline Box S^l

$$\mathcal{B}^l \ni x = S^l(\{b\},\{N\},t) \equiv \left[\sum_{i=1}^{\bar{i}} \sum_{j=1}^{\bar{j}} \sum_{k=1}^{\bar{k}} b_{ijk} N_i^1(t_1) N_j^2(t_2) N_k^3(t_3) \right]_l ,$$

$t^\square = (t_1, t_2, t_3) \in \mathcal{B}_0^l$, knot vector $\tau^\square = (\tau_i^\square)$



- union of boxes: $\mathcal{B} = \sum_{l=1}^{NSB} \mathcal{B}^l$
- domain embedding: $\Omega \in \mathcal{B}$
- initial (ref.) body $\Omega_0 \subset \mathcal{B}_0$
- control polyhedron $\{b\}$, $b = (b_{ijk}^\square)$ for each l
- B-spline basis $N_{ijk}^\square(t^\square)$, $t \in]\underline{\tau}^\square, \overline{\tau}^\square[$

Domain decomposition using Spline Boxes

Ω designed by $\{b\}$

$$\overline{\Omega} = \bigcup_{I=1}^{NSB} S^I(\{b\}, \{N\}, \overline{\Omega_0^I})$$

Non-overlapping spline boxes with interface segments

$$\Omega_0 \subset \mathcal{B}_0 = \bigcup_{I=1}^{NSB} \mathcal{B}_0^I \ , \quad \mathcal{B}_0^I = \Pi_{k=1,2,3}]a_k^I, b_k^I[$$

such that $\mathcal{B}_0^I \cap \mathcal{B}_0^J = \emptyset$ for $I \neq J$,

$\overline{\mathcal{B}_0^I} \cap \overline{\mathcal{B}_0^J} \equiv \mathcal{S}_0^{IJ} \neq \emptyset$ iff $\exists k, l, m \in \{1, 2, 3\}$:

interface: $\mathcal{S}_0^{IJ} = x_m \times [a_k^I, b_k^I] \times [a_l^I, b_l^I]$

where $x_m = a_m^I = b_m^J$, or $x_m = b_m^I = a_m^J$.

Spline Boxes II — properties

Identity property: $\mathcal{B}_0^l = S^l(\{g\}, \{N\}, \mathcal{B}_0^l)$

Greville abscissae $\{g_r^{ijk}\}^l$ computed for defined parametrization $t \in \mathcal{B}_0^l$

$$\mathcal{B}_0^l \ni t = x = \left[\sum_{i=1}^{\bar{i}} \sum_{j=1}^{\bar{j}} \sum_{k=1}^{\bar{k}} g^{ijk} N^i(t_1) N^j(t_2) N^k(t_3) \right]_l .$$

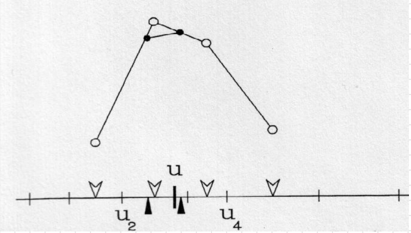
Let $\overline{\Omega_0^l} = \overline{\Omega_0 \cap \mathcal{B}_0^l} \Rightarrow \overline{\Omega_0} = \bigcup_{l=1}^{NSB} \overline{\Omega_0^l}$ and $\Omega_0^l = S^l(\{g\}, \{N\}, \Omega_0^l)$

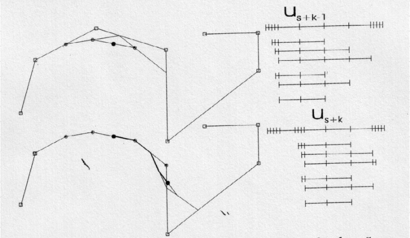
Details – cubic spline

- parametrization: $t \in [\underline{\tau}, \overline{\tau}]$,
- nodes: $\tau_k \in [\underline{\tau}, \overline{\tau}]$, $k = 1, \ldots, 4\times(1+num.seg.)$,
- Greville positions: $g^i = (\tau_{i-1} + \tau_i + \tau_{i+1})/3$

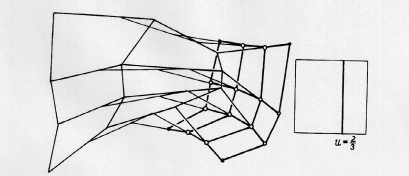
Elements of manipulation over splines . . .

- knot insertion
- segmentaion – Bézier representation
- degree elevation





Obr. 3.16: Vyčlenění sgmentu B–spline křivky $k = 5$.



Obr. 3.8: Rozdělení Bézierovy bikubické plochy podle parametrické křivky $u = \frac{2}{3}$.

Spline Boxes III — C^0 continuity constraints

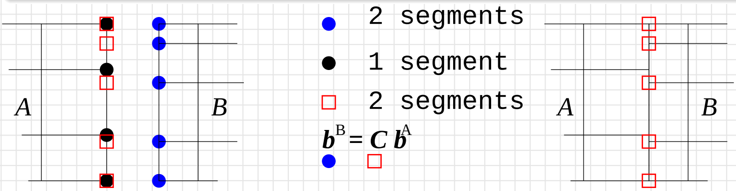
Control points $\{[b]_I\}_{I=1}^{NSB}$ subject to interface conditions

$$P^{IJ}[b]_J = P^{JI}[b]_I \qquad \text{for any } I, J : \quad \overline{\mathcal{B}_0^I} \cap \overline{\mathcal{B}_0^J} \equiv \mathcal{S}_0^{IJ} \text{ and } \mathcal{S}_0^{IJ} \cap \Omega_0 \neq \emptyset \,,$$

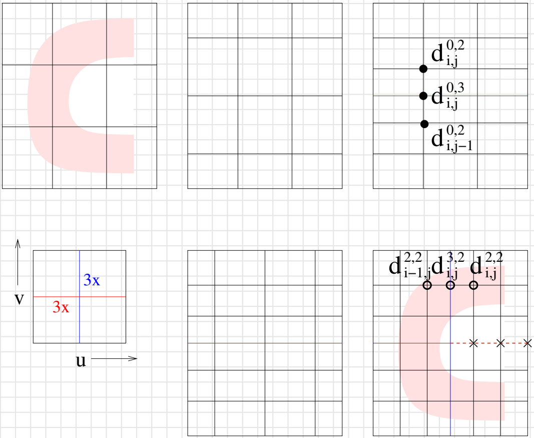
P^{IJ} and P^{JI} . . . the coupled continuity operators

Compatibility of $[b]_I$ and $[b]_J$ on interfaces $\mathcal{S}_0^{IJ}$

- to “stick” B-splines with *different number* of control vertices $n^A \neq n^B$,
- method: if $n^A < n^B$ increase $n^A \to \tilde{n}^A = n^B$ and compute the equivalent positions of the control points $[\tilde{b}]_A$



Spline Boxes – Knot insertion & C^1 continuity



$3\times$ insertion of $\hat{u} = 1/2$

$\Rightarrow 3\times$

$2\mathbf{d}^{0,3}_{i,j} = \mathbf{d}^{0,2}_{i,j-1} + \mathbf{d}^{0,2}_{i,j}$

$3\times$ insertion of $\hat{v} = 1/2$

$\Rightarrow 6\times$

$2\mathbf{d}^{2,3}_{i,j} = \mathbf{d}^{2,2}_{i-1,j} + \mathbf{d}^{2,2}_{i,j}$

...and decouple nodes X

$(4*9 + 4*3 + 1 + 3)*\ \vec{\mathrm{dof}}$ - $(6 + 4)*\ \vec{\mathrm{eqn}} =$ 42* free-$\vec{\mathrm{dof}}$

Design variables and gradients

Continuity and other design restrictions

- C^0 continuity (between attached spline boxes)
- C^1 continuity ??? ... not yet (complicated)
- other restrictions to positions of $\{b\}$

$$\Rightarrow \quad \text{matrix constraint op.} \quad B(\{b\}) = 0$$

Design variables γ_α

- define

$$\{d\}^\alpha \in \mathrm{Ker} B \quad \text{for } \alpha = 1, \ldots, \bar{\alpha} \leq \mathrm{card}(\mathrm{Ker} B) \ ,$$

- γ_α, $\alpha = 1, \ldots, \bar{\alpha}$ ... multipliers of $\{d\}^\alpha$
- varying the control points:

$$\{b\} = \{g\} + \sum_{\alpha=1}^{\bar{\alpha}} \gamma^\alpha \{d\}^\alpha$$

Design gradients

Admissible velocity field $\vec{\mathcal{V}}$

for a perturbation $\delta\gamma = (\delta\gamma^\alpha)_\alpha, \ \alpha \in I_{ad} \subset [1, \bar{\alpha}]$

$$\vec{\mathcal{V}} = \sum_{l=1}^{NSB} S^l\Big(\sum_{\alpha \in I_{ad}} \delta\gamma^\alpha \{d\}^\alpha, \{N\}, \overline{\Omega_0^l}\Big) \ ,$$

$\Rightarrow$ α-th basic velocity field $\vec{\mathcal{V}}^\alpha$

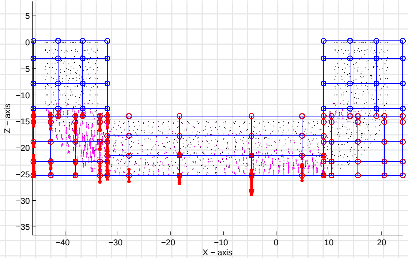
$$\vec{\mathcal{V}}^\alpha = \sum_{l=1}^{NSB} S^l(\{d\}^\alpha, \{N\}, \overline{\Omega_0^l}) \ ,$$

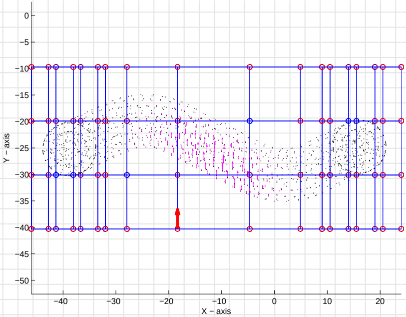
Design gradient

computed using the *domain method of the sensitivity analysis:*

$$\vec{\mathcal{V}}^\alpha \longrightarrow \frac{\mathrm{d}}{\mathrm{d}\,\gamma^\alpha}\Psi(\mathbf{u}) \quad \alpha \in I_{ad}$$

Basis velocity field $\vec{\mathcal{V}}^\alpha$ — example





$\mathrm{div}\mathcal{V}^\alpha \geq 0$???

Sensitivity analysis — more general case

Objective function — pressure loss and dissipation

$$\Phi(\mathbf{u},p) = \beta_0 \int_{\Gamma_{\rm in}} (p(x) - \bar{p})dS + \beta_1 \nu \frac{1}{2} \int_\Omega |\nabla \mathbf{u}|^2$$

Compute the adjoint state $(\mathbf{w}, q)$, $\mathbf{w} \in \mathbf{V}_0$ and for all $q \in L^2(\Omega)$

$$a_\Omega(\mathbf{v}, \mathbf{w}) + c_\Omega(\mathbf{v}, \mathbf{u}, \mathbf{w}) + c_\Omega(\mathbf{u}, \mathbf{v}, \mathbf{w}) + b_\Omega(\mathbf{v}, q) = -\delta_{\mathbf{u}}\Phi(\mathbf{u},p) \circ \mathbf{v}$$

$$b_\Omega(\mathbf{w}, \eta) = \delta_p \Phi(p) \circ \eta$$

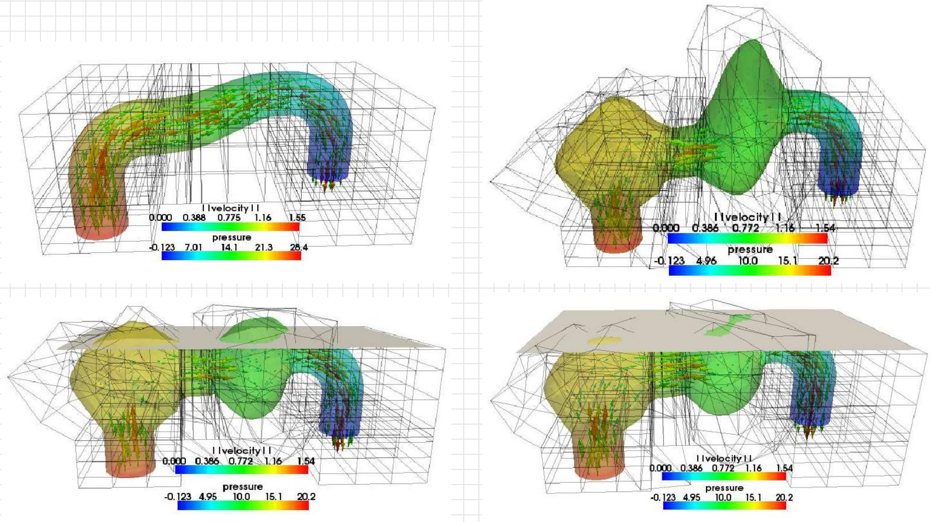
$$\forall \mathbf{v} \in \mathbf{V}_0, \quad \forall \eta \in L^2(\Omega)$$

Compute the design gradient: for all $\alpha \in I_{ad}$ using $\vec{\mathcal{V}}^\alpha$

evaluate the partial shape sensitivities $\delta_D()$ and compute

$$\frac{\mathrm{d}}{\mathrm{d}\,\gamma^\alpha}\Phi(\mathbf{u},p) = \delta_D \Phi(\mathbf{u},p) + \delta_D a_\Omega(\mathbf{u}, \mathbf{w}) + \delta_D c_\Omega(\mathbf{u}, \mathbf{u}, \mathbf{w}) - \delta_D b_\Omega(\mathbf{w}, p) + \delta_D b_\Omega(\mathbf{u}, q) \; .$$

Examples



Computations by Mgr. Zuzana Záhorová, (Dipl. thesis MFF UK 2010)

Sfepy — FEM code (Robert Cimrman)

Remarks

Current state:

- steady laminar incompressible flows
- stabilized FEM: P1+bubble in $\mathbf{u}$, P1 in p
- Newton method solver
- stab. formulation SUPG + PSPG (Matthies G., Lube G.) for low Reynolds numbers
- Spline-box parametrization — direct control:

  design $\longrightarrow$ *fluid* FE mesh
- Steepest descent (with linesearch), linear constraints...