The Sensitivity Equation Method: Characterisation and Optimisation of Flow Systems

D. Pelletier

École Polytechnique de Montréal

Canada Research Chair

J. Borggaard

ICAM, Virginia Tech

MOPTA 05

25–27 July, Windsor, ON, Canada

Theme of talk

Computational issues

- Complexity of PDEs modeling the system
- Cost of solving PDEs
- Many parameters ; not all are important
- Accuracy of numerics for optimisation
- Must assess uncertainty due to input data

Tools

- Adaptive FEM for accuracy & efficiency
- SEM : key parameters, optimisation, uncertainty

Outline

- CFD & sensitivities : Context
- Sensitivities : definition and uses
- Flow equations
- FEM & mesh adaptation
- SEM
- Application : Characterisation
- Application : Optimization
- Conclusions

Real Problems

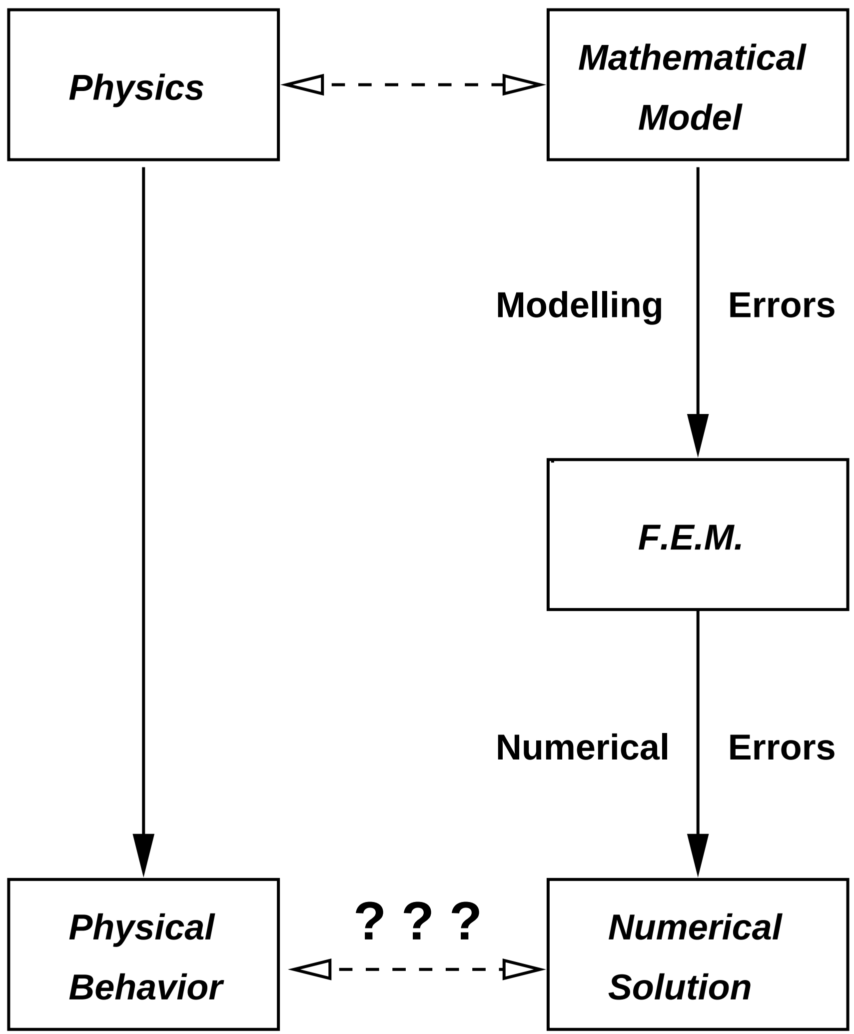
- High Reynolds, Péclet, Rayleigh numbers
- Extreme/Thin geometries
- Extreme/Thin features

1. Unknown kind
2. Unknown location
3. Unknown extent

- Input data uncertainty

- How to design under such conditions ?

Simulation Context



Issues :

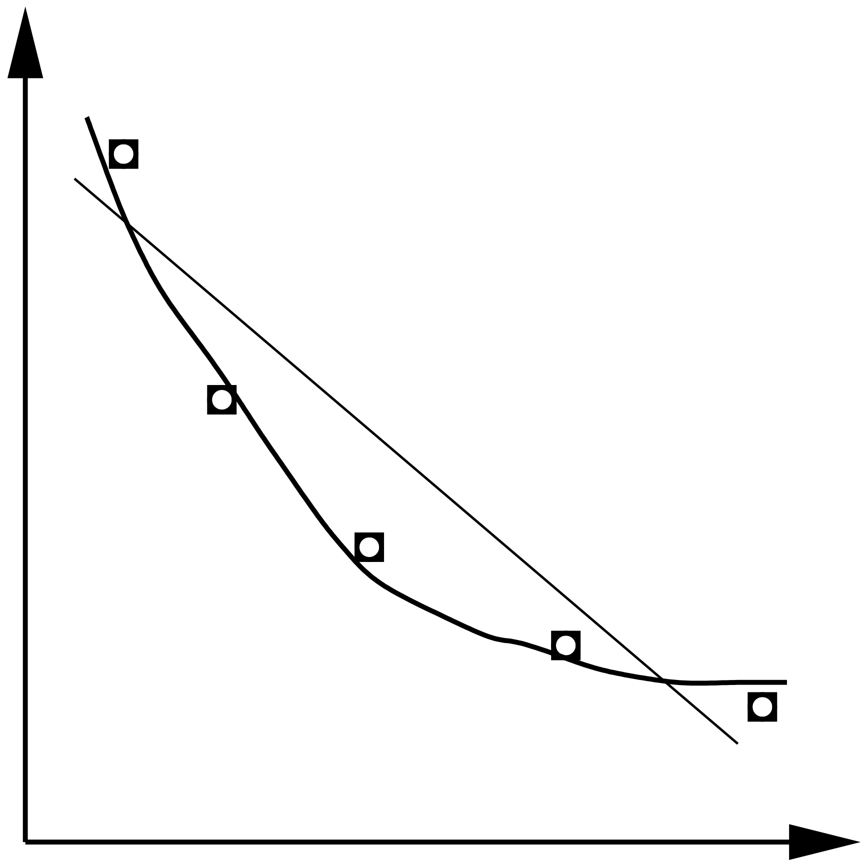
1. Numerical accuracy
2. Modelling errors
3. Input uncertainty
4. Experimental uncertainty

Tools

1, 2 = Mesh adaptation

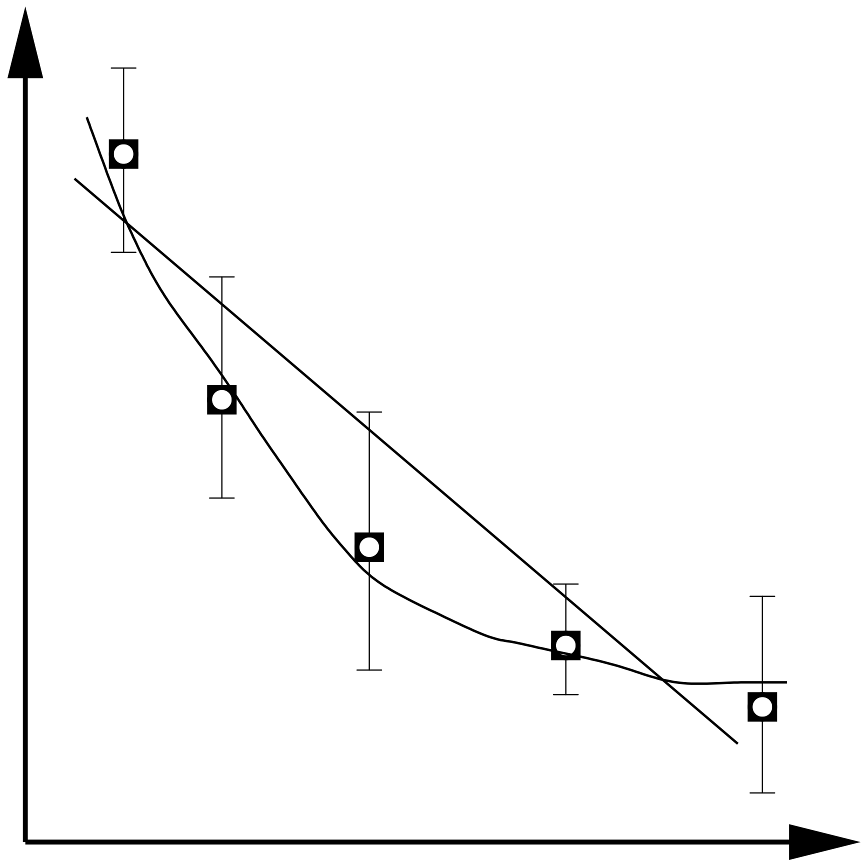
3, 4 = Sensitivities

Modeling & Simulation (1)



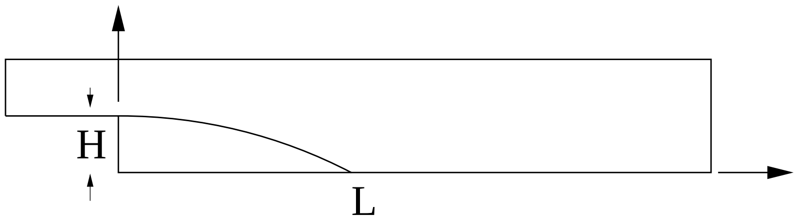
- **Situation :**
  - Linear model
  - Nonlinear model
  - Data points
- **Which is best ?**

Modeling & Simulation (2)

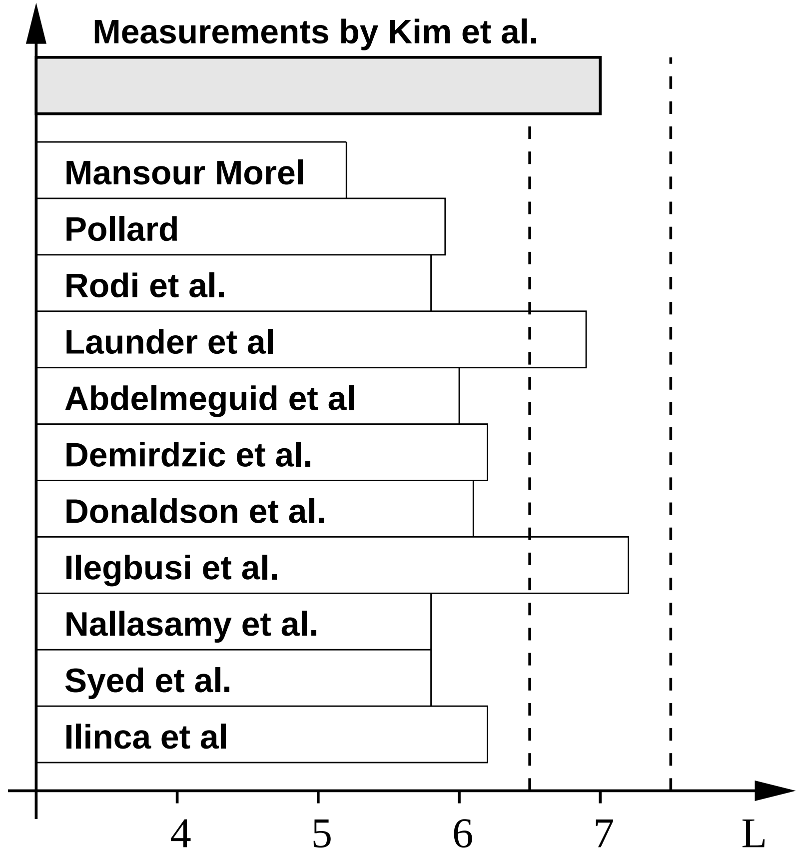


- **Situation :**
  - Linear model
  - Nonlinear model
  - Data points
  - Data uncertainty
- **Which is best ?**

Turbulent flow over step



- Same turbulence model
- Same code
- Except Donalson & Ilinca
- Different meshes



Navier-Stokes : laminar flow

- Free convection
- Variable properties = functions of T

$$\nabla \cdot \boldsymbol{u} = 0$$

$$\rho c_p \boldsymbol{u} \cdot \nabla T = \nabla \cdot (\kappa \nabla T) + q$$

$$\rho \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nabla \cdot \left[\mu \left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T \right) \right] - \rho \boldsymbol{g} \beta (T - T_0) + \boldsymbol{f}$$

Boundary conditions

$$\boldsymbol{u}(\boldsymbol{x_b}; a) = \hat{\mathbf{u}}(\boldsymbol{x}_b; a) \qquad \text{on } \Gamma_u$$

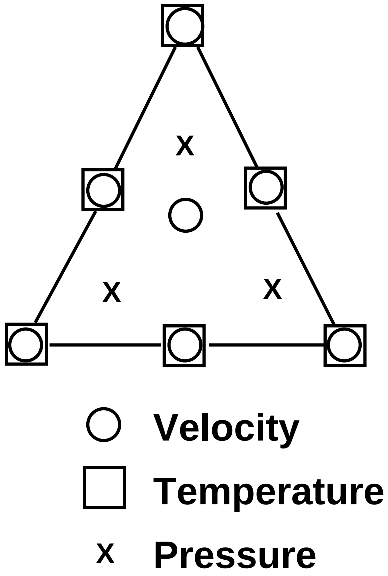
$$\boldsymbol{t}(\boldsymbol{x_b}; a) = \boldsymbol{\sigma} \cdot \hat{\boldsymbol{n}} = \boldsymbol{F}^N(\boldsymbol{x}_b; a) \qquad \text{on } \Gamma_t$$

$$\boldsymbol{\sigma} = \left[-p\boldsymbol{I} + \mu \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T - \frac{2}{3} \boldsymbol{I} \nabla \cdot \boldsymbol{u} \right) \right]$$

$$T(\boldsymbol{x_b}; a) = \hat{T}(\boldsymbol{x}_b; a) \qquad \text{on } \Gamma_T$$

$$\lambda \nabla T(\boldsymbol{x}_b; a) \cdot \hat{\boldsymbol{n}} = \hat{q}_w(\boldsymbol{x}_b; a) \qquad \text{on } \Gamma_q$$

FEM Solver



- Galerkin FEM + Stabilization
- Primitive variables = (U,V,p) & (T,k,ϵ)
- Augmented Lagrangian
- Newton's Method

Verification and Validation

Verification :

Are we solving the equations **Properly** ?

Validation :

Are we solving the **Proper** equations for the problem at hand ?

Verification

Verification involves :

- mathematics,
- numerical analysis,

For example :

- ? Is scheme/code implementation really $O(\delta x^2)$?
- ? $O(\delta x^2)$? on the case at hand ?

Validation

Validation means having

- the proper physics,
- the proper science,
- an appropriate engineering model.

Requirements :

- detailed experiments,
- high quality measurements.

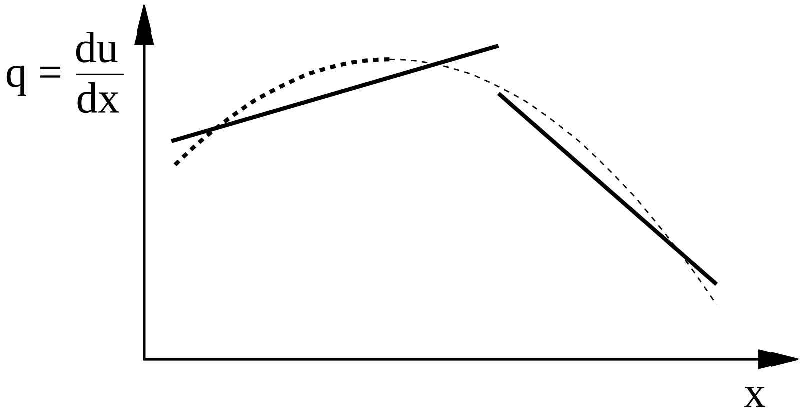
Adaptivity

- M_i : i^{th} adaptive mesh
- P_i : degree of interpolant

0. Initial Discretisation : (M_0, P_0)
1. FEM Solver : U_i
2. Error Estimation : $E_i(M_i, P_i, U_i)$
3. if ($E_i < \epsilon$) $\longrightarrow$ STOP
4. Adaptation $(M_{i+1}, P_{i+1}) = A(M_j, P_j, U_j)$

   Goto 1

Projection Error Estimator



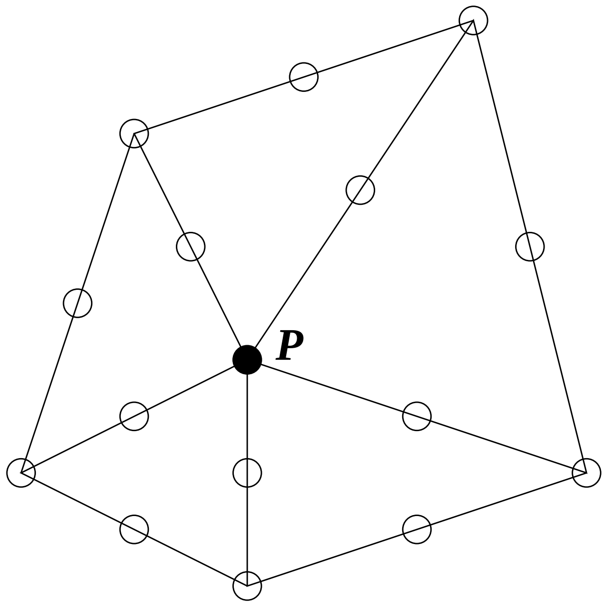
Error in derivative

$$e^2 = \int (q_{ex} - q_h)^2 dx$$

$$e^2 \simeq E^2 = \int (q^* - q_h)^2 dx$$

q^* an approximation of q_{ex}

Projection Error Estimator



$$Min \int_{\Omega_S} (q^* - q_h)^2 \, d\Omega$$

$$q^* = [1, x, y, x^2, xy, y^2][a1, \ldots, a6] = P\vec{a}$$

$$\left[\int_{\Omega_S} P^T P \, d\Omega\right] \{a\} = \left[\int_{\Omega_S} P^T q_h \, d\Omega\right]$$

Projection : Navier-Stokes

$$e_U^2 = \int (\tau^* - \tau_h) : (\tau^* - \tau_h)$$

$$e_P^2 = \int (p^* - p_h)^2$$

$$e_T^2 = \int (\nabla T^* - \nabla T_h) \cdot (\nabla T^* - \nabla T_h)$$

$$\tau = \mu \left[\nabla U + (\nabla U)^T \right]$$

New Mesh

- Target accuracy = $\bar{\eta}\%$
- Target error : $\bar{e} = \bar{\eta}\frac{\|U\|}{\sqrt{N_e}}$
- Equidistribution :
  - $e_i = cte \qquad , \qquad i = 1, .., N_e$
- Asymptotic convergence rate :
  - $e_i = Ch_i^k$
  - $\bar{e} = C\delta_i^k$
- New element size :

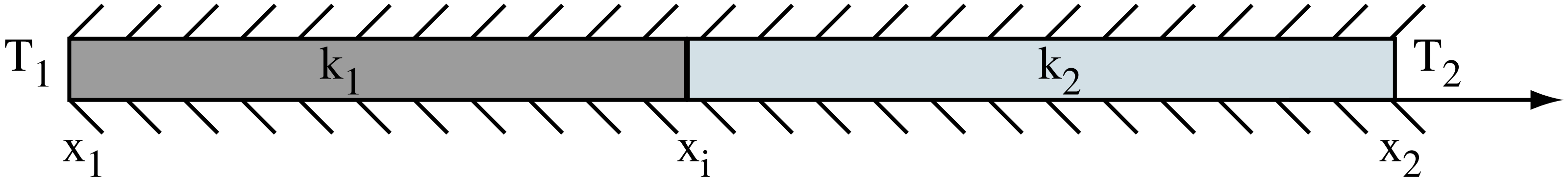
$$\delta_i = \left\{\bar{\eta}\frac{\|U\|}{\sqrt{N_e}\|e_i\|}\right\}^{\frac{1}{k}} h_i$$

Multiple fields

$$
\begin{aligned}
e_U &\longrightarrow \delta_U \\
e_P &\longrightarrow \delta_P \\
e_T &\longrightarrow \delta_T \\
e_k &\longrightarrow \delta_k \\
e_\epsilon &\longrightarrow \delta_\epsilon
\end{aligned}
$$

$$\delta = min\{\delta_i\}$$

Sensitivities



$$T = T(x; T_1, T_2, \kappa_1, \kappa_2, x_1, x_i, x_2)$$

Sensitivity with respect to a

$$s_T = \frac{\partial T}{\partial a}$$

$$a \in \{T_1, T_2, \kappa_1, \kappa_2, x_1, x_i, x_2\}$$

Sensitivities : uses

Gradient based optimization : $\min \; J(\boldsymbol{u}(\alpha), p(\alpha); \alpha)$

$$\frac{dJ}{d\alpha} = \frac{\partial J}{\partial \boldsymbol{u}} \underbrace{\frac{\partial \boldsymbol{u}}{\partial \alpha}}_{\boldsymbol{s}_u} + \frac{\partial J}{\partial p} \underbrace{\frac{\partial p}{\partial \alpha}}_{s_p} + \frac{\partial J}{\partial \alpha}$$

Fast nearby solution via Taylor series

$$C_p(x; \alpha_0 + \delta\alpha) = C_p(x; \alpha_0) + \left. \underbrace{\frac{\partial C_p}{\partial \alpha}} \right|_{\alpha_0} \delta\alpha$$

Uncertainty Analysis

Cascade input data uncertainty into CFD outputs

1st Order statistics : $$\sigma^2_{C_f} \approx \sum_{i=1}^{n} (\underbrace{\frac{\partial C_f}{\partial a_i}}_{s_F^{a_i}} \sigma_{a_i})^2$$

Worst case scenario : $$\delta C_f \approx \sum_{i=1}^{n} |\underbrace{\frac{\partial C_f}{\partial a_i}}_{s_F^{a_i}}||\delta a_i|$$

1st Order Sensitivity PDE

Continuity :

$$\boxed{\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0}$$

SEM : differentiate then discretize

$$\frac{\partial}{\partial a}\left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right] = 0 \qquad \longrightarrow \qquad \frac{\partial}{\partial x}\left(\frac{\partial u}{\partial a}\right) + \frac{\partial}{\partial y}\left(\frac{\partial v}{\partial a}\right) = 0$$

$$\boxed{\frac{\partial s_u}{\partial x} + \frac{\partial s_v}{\partial y} = 0}$$

1st Order Energy Sensitivity

$$\frac{\partial}{\partial a}\left[\rho c_p \boldsymbol{u}\cdot\nabla T = \nabla\cdot(\kappa\nabla T) + q_s\right]$$

general approach : all possible dependencies

$$T = T(x, y; a)$$

$$c_p = c_p(T(a); a)$$

$$\Downarrow$$

$$\left(\rho' c_p + \rho c_p'\right)\boldsymbol{u}\cdot\nabla T + \rho c_p\left(\boldsymbol{s}_u\cdot\nabla T + \boldsymbol{u}\cdot\nabla s_T\right)$$

$$= \nabla\cdot(\kappa'\nabla T + \kappa\nabla s_T) + q_s'$$

1st Order Momentum Sensitivity

$$\frac{\partial}{\partial a}\left[\ \rho \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla p + \nabla\cdot\left[\mu\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T\right)\right] - \rho\boldsymbol{g}\beta(T - T_0) + \boldsymbol{f}\ \right]$$

$$\Downarrow$$

$$\begin{aligned}\rho'\boldsymbol{u}\cdot\nabla\boldsymbol{u} + \rho\boldsymbol{s}_u\cdot\nabla\boldsymbol{u} + \rho\boldsymbol{u}\cdot\nabla\boldsymbol{s}_u = -\nabla s_p \\ +\nabla\cdot\left[\mu'\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T\right) + \mu\left(\nabla\boldsymbol{s}_u + (\nabla\boldsymbol{s}_u)^T\right)\right] \\ -\left(\rho'\boldsymbol{g}\beta + \rho\boldsymbol{g}'\beta + \rho\boldsymbol{g}\beta'\right)(T - T_0) - \rho\boldsymbol{g}\beta\left(s_T - T_0'\right) + \boldsymbol{f}'\end{aligned}$$

Newton linearization : 1 linear system of PDE / parameter

Variable fluid properties

$$\mu = \mu(\boldsymbol{u}(a), T(a); a)$$

$$\mu' = \frac{\partial \mu}{\partial a} + \frac{\partial \mu}{\partial u} s_u + \frac{\partial \mu}{\partial v} s_v + \frac{\partial \mu}{\partial T} s_T$$

$\rightarrow$ chain rule : all dependencies

Boundary conditions : Dirichlet

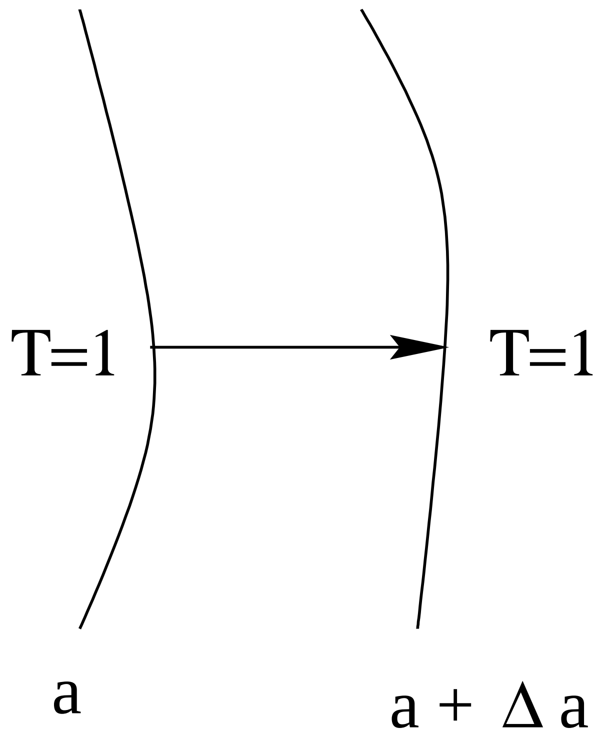
value parameter : $T = T_0$

$$s_T = \frac{\partial T_0}{\partial a}$$

shape parameter

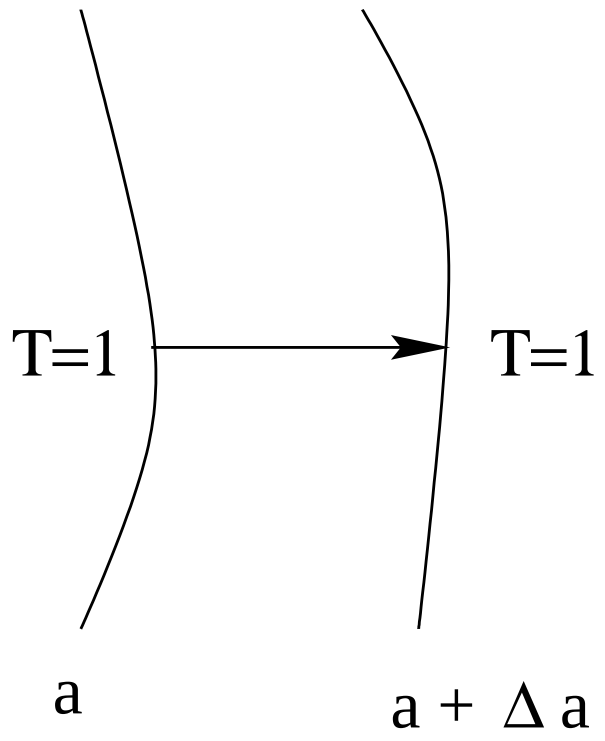
$$T(x_b(a), y_b(a); a) = T_0(x_b(a), y_b(a); a)$$

$$\frac{DT}{Da} = \frac{DT_0}{Da}$$



Boundary conditions : Dirichlet

$$
\begin{aligned}
\frac{DT}{Da} &= \frac{DT_0}{Da} \\
s_T &= -\frac{\partial T}{\partial x}\frac{\partial x_b}{\partial a} - \frac{\partial T}{\partial y}\frac{\partial y_b}{\partial a} + \frac{DT_0}{Da}
\end{aligned}
$$



- Provide $\frac{DT_0}{Da}$, $\frac{\partial x_b}{\partial a}$, and $\frac{\partial y_b}{\partial a}$
- Compute $\frac{\partial T}{\partial x}$ and $\frac{\partial T}{\partial y}$ from the flow solution

Boundary conditions : Neumann

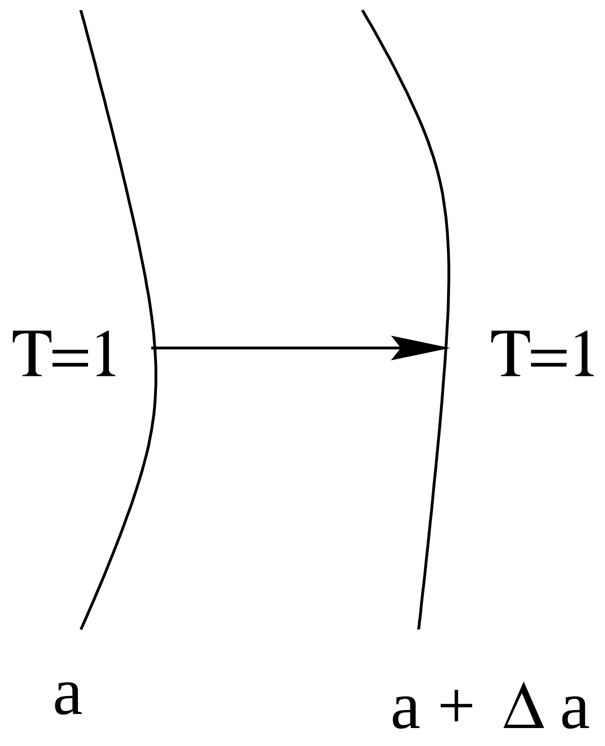
value parameter : $\kappa \nabla T \cdot \hat{\boldsymbol{n}} = q_0$

$$(\kappa' \nabla T + \kappa \nabla s_T) \cdot \hat{\boldsymbol{n}} = q_0'$$

shape parameter

$$\kappa \nabla T \cdot \hat{\boldsymbol{n}} = q_0(x_b(a), y_b(a); a)$$

$$\frac{D}{Da}(\kappa \nabla T) \cdot \hat{\boldsymbol{n}} + \kappa \nabla T \cdot \frac{D\hat{\boldsymbol{n}}}{Da} = \frac{Dq_0}{Da}$$



Boundary conditions : Neumann

<u>shape parameter</u>

$$\frac{D}{Da}\left(\kappa\nabla T\right)\cdot\hat{\boldsymbol{n}}+\kappa\nabla T\cdot\frac{D\hat{\boldsymbol{n}}}{Da}=\frac{Dq_0}{Da}$$

$$\left(\kappa'\nabla T+\kappa\nabla s_T\right)\cdot\hat{\boldsymbol{n}}=\frac{Dq_0}{Da}-\kappa\nabla T\cdot\frac{D\hat{\boldsymbol{n}}}{Da}$$

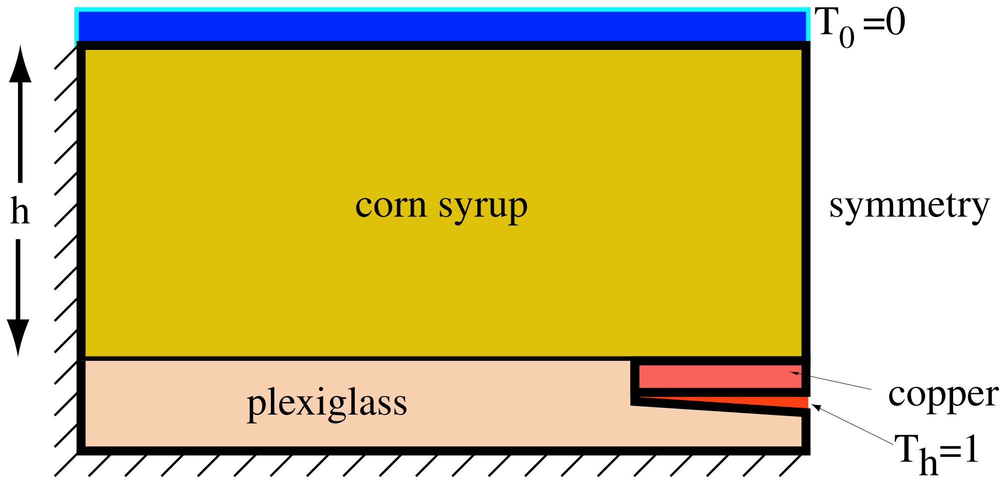
$$-\left[\frac{\partial}{\partial x}\left(\kappa\nabla T\right)\frac{\partial x_b}{\partial a}+\frac{\partial}{\partial y}\left(\kappa\nabla T\right)\frac{\partial y_b}{\partial a}\right]\cdot\hat{\boldsymbol{n}}$$

- Provide $\frac{Dq_0}{Da}$ and $\frac{D\hat{\boldsymbol{n}}}{Da}$
- Compute $\frac{\partial^2 T}{\partial x^2}, \frac{\partial^2 T}{\partial x \partial y}, \frac{\partial^2 T}{\partial x^2}$

Implementation

- General formulation
- FEM
- Numerical Jacobian
- Adaptivity : flow + sensitivities
- Projected flow gradients :
  - BC (shape parameters)
  - PDEs : $\rho c_p \boldsymbol{s}_u \cdot \nabla T$, $\nabla \cdot (\kappa' \nabla T)$

Free Convection in Corn Syrup



$$\mu = a_0 e^{a_1 e^{-T/a_2}}$$

$$\kappa = b_0 + b_1 T$$

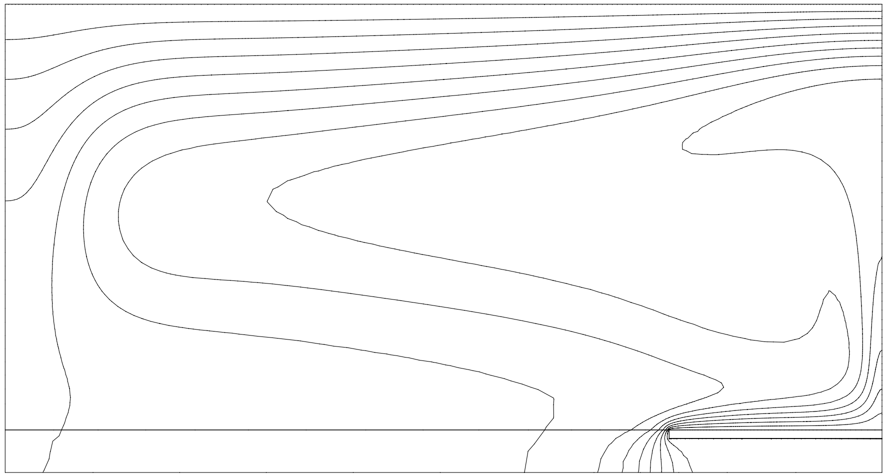
$$c_p = c_0 + c_1 T + c_2 T^2$$

A laboratory model for magma circulation in the earth's mantle

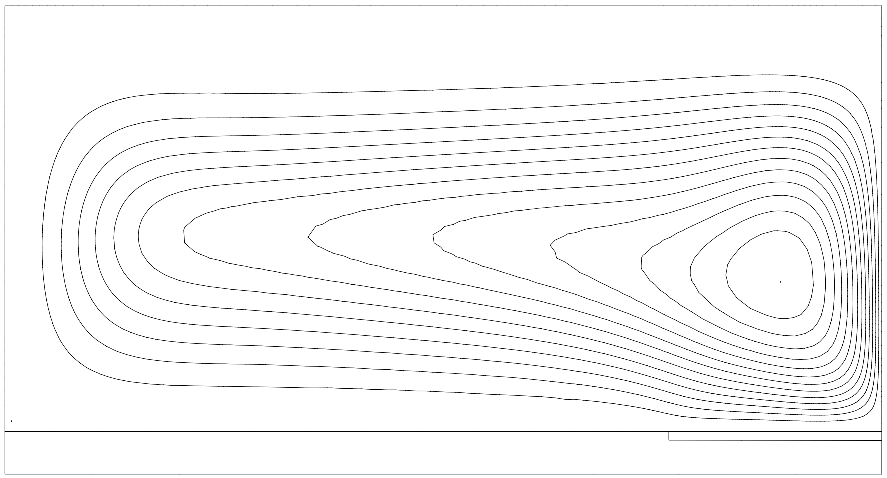
4 parameters :

h, T_h, a_0, and b_0

Free Convection in Corn Syrup

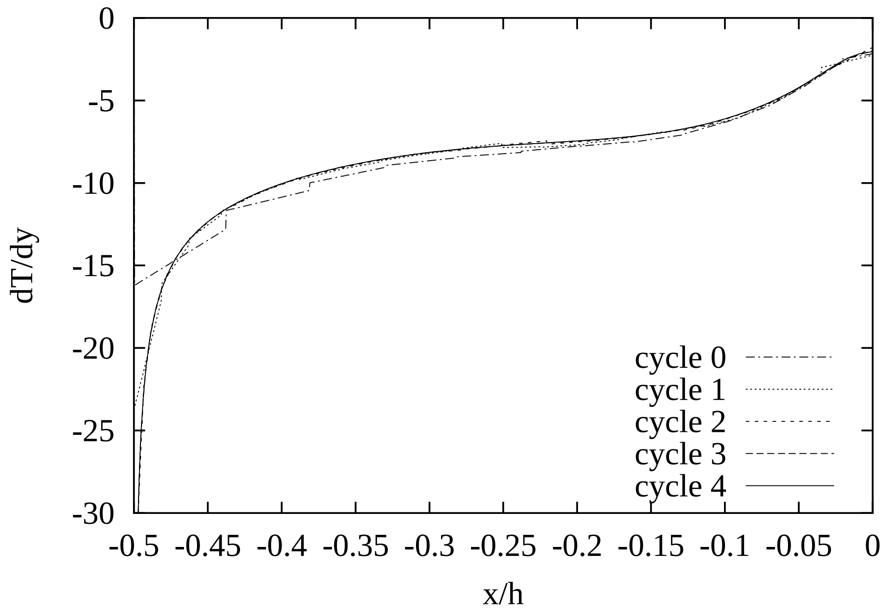


Temperature

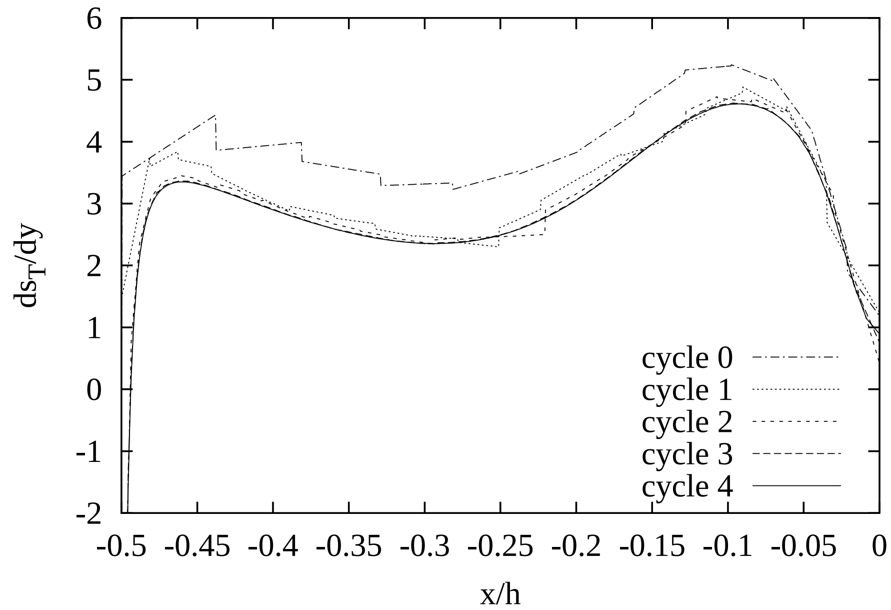


Streamlines

Grid convergence



$$\frac{\partial T}{\partial y}$$



$$\frac{\partial s_T}{\partial y}(b_0)$$

Must adapt mesh to flow & sensitivities

Scaled sensitivities

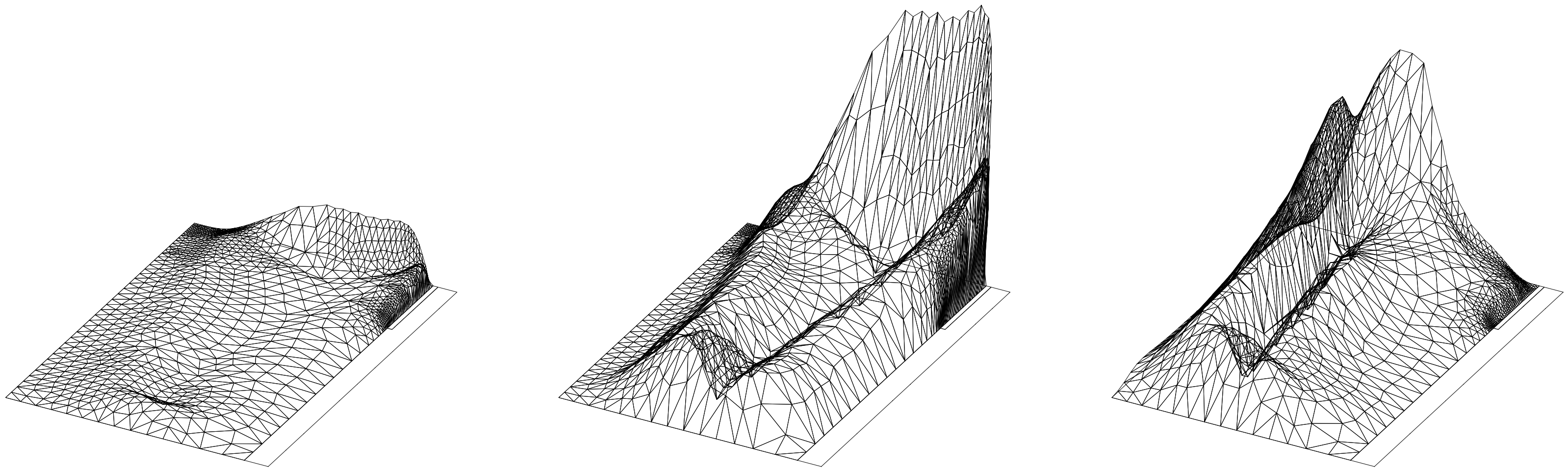
$$T(x; a_i + \Delta a_i) - T(x; a_i) = \frac{\partial T}{\partial a_i} \Delta a_i$$

$$\Delta T = \sum_i \underbrace{\left(\frac{\partial T}{\partial a_i} a_{i_0} \right)}_{\text{scaled sensitivity } \tilde{s}_{T_i}} \times \underbrace{\left(\frac{\Delta a_i}{a_{i_0}} \right)}_{\text{\% change in } a_i}$$

All $\dfrac{\partial T}{\partial a_i} a_{i_0}$ in $^\circ C \rightarrow$ Comparison more meaningful

Scaled Sensitivities : $\|\boldsymbol{s}_u\|$

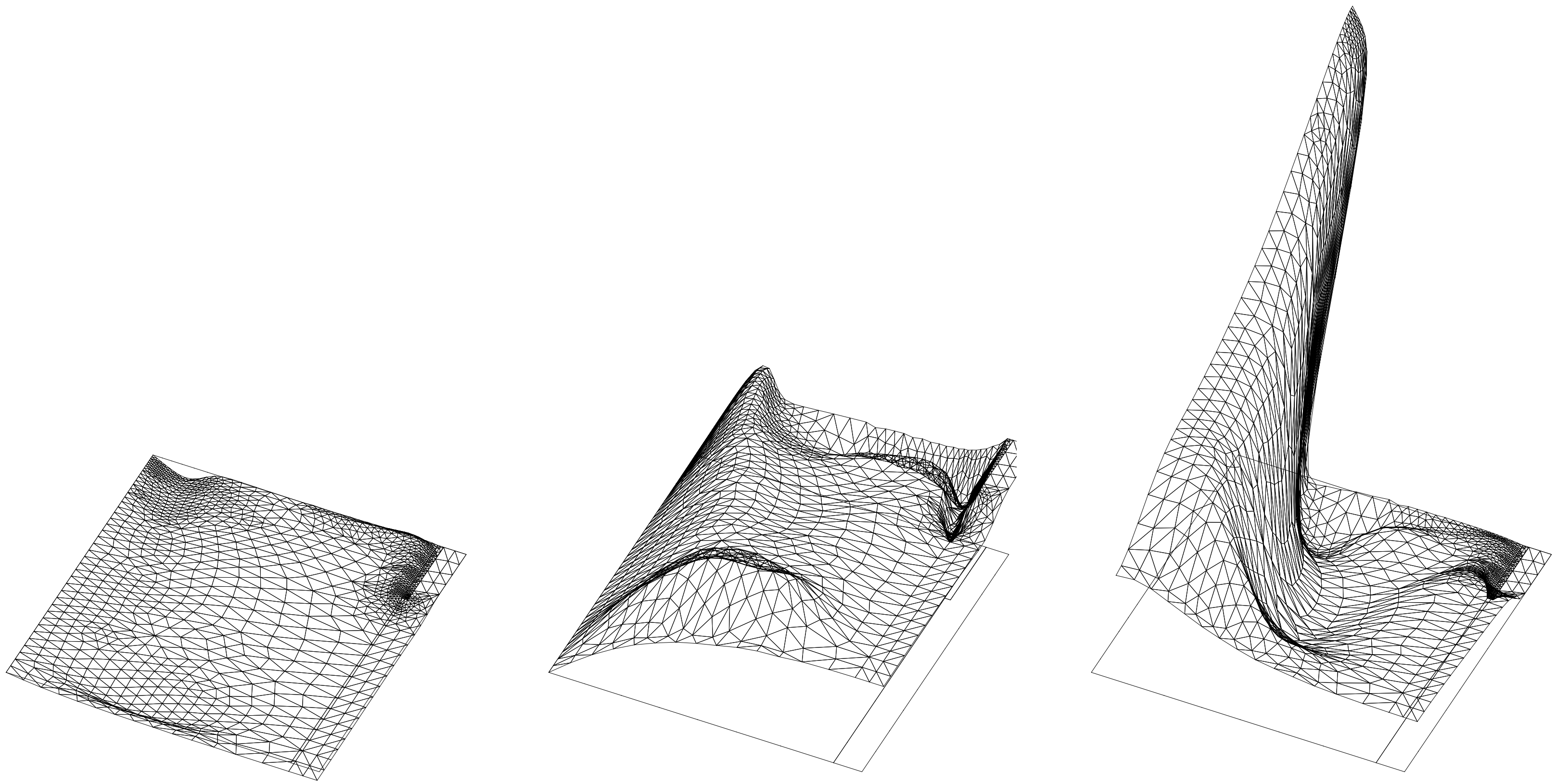
3 Scaled sensitivities of velocity



a_0 T_h h

Scaled Sensitivities : s_T

3 Scaled sensitivities of temperature



b_0 $\qquad$ T_h $\qquad$ h

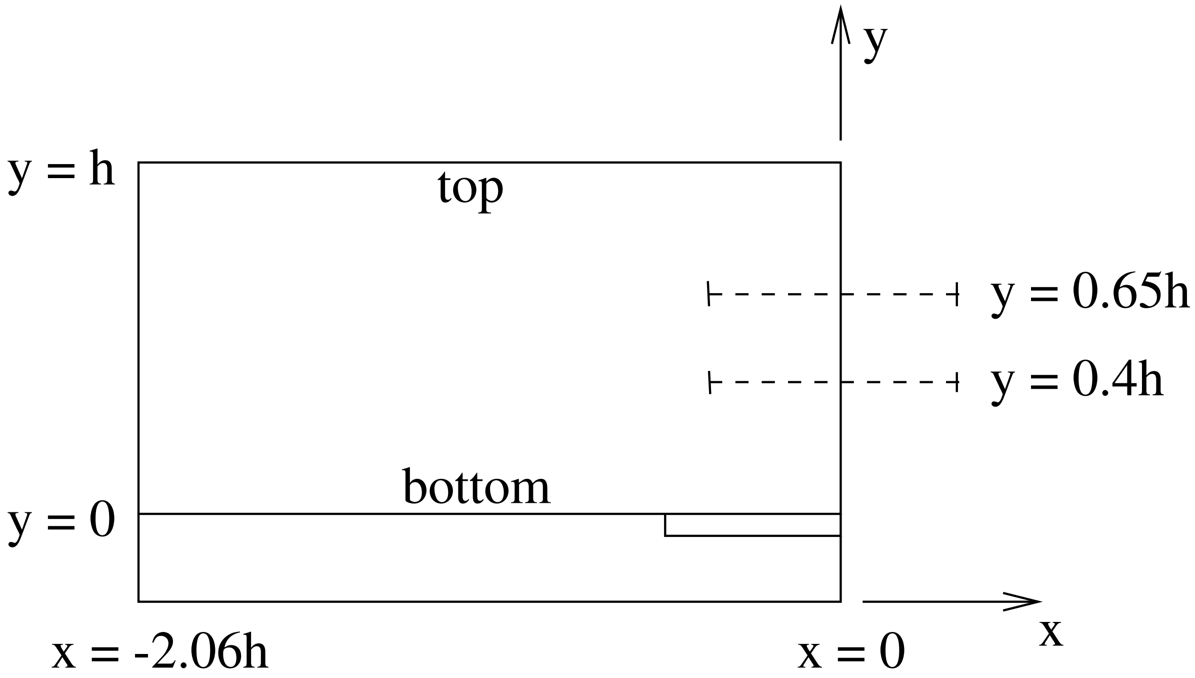
Uncertainty cascade

$$T(\boldsymbol{x};\boldsymbol{a}+\Delta\boldsymbol{a}) = T(\boldsymbol{x};\boldsymbol{a}) + \sum_{i=1}^{n} \frac{\partial T}{\partial a_i}(\boldsymbol{x};\boldsymbol{a})\Delta a_i$$

$$\Delta T = \sum_{i=1}^{n} \frac{\partial T}{\partial a_i}(\boldsymbol{x};\boldsymbol{a})\Delta a_i$$

$$|\Delta T| \leq \sum_{i=1}^{n} \left|\frac{\partial T}{\partial a_i}(\boldsymbol{x};\boldsymbol{a})\right| |\Delta a_i|$$

Cut planes



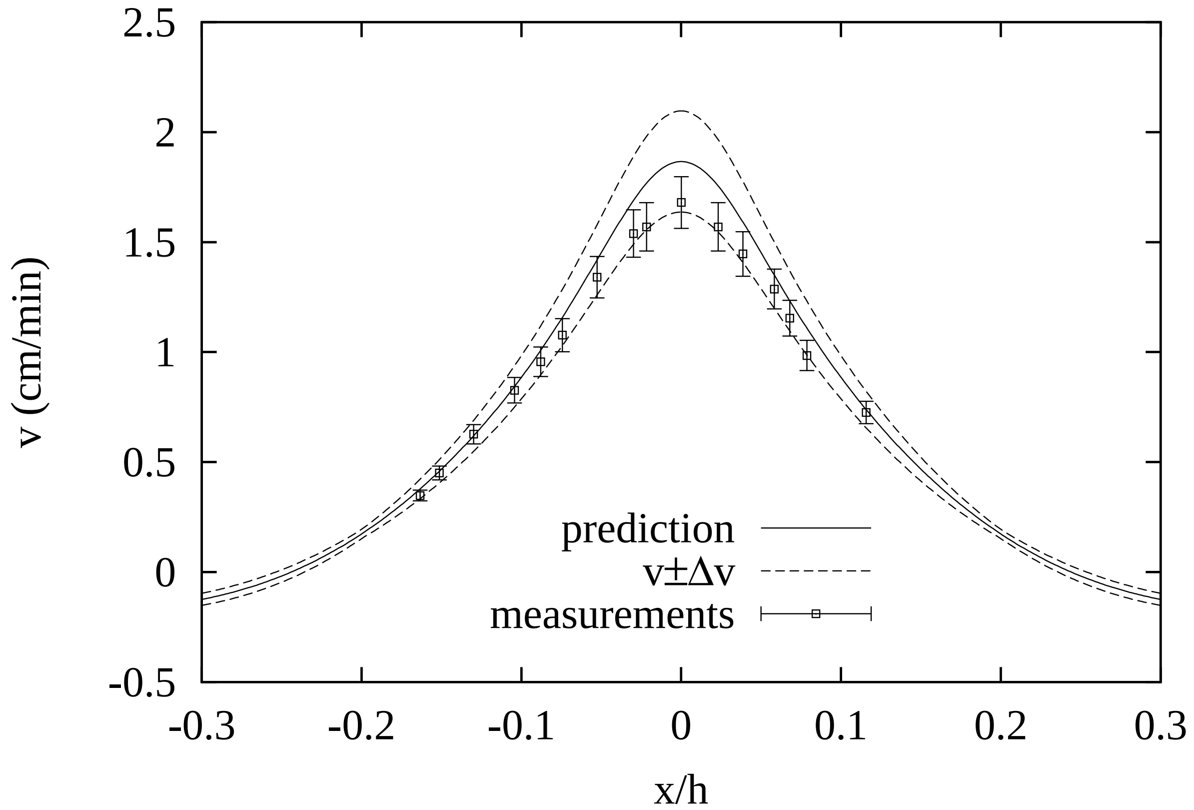
Vertical velocity at $y = 0.4h$

$$\Delta a_0 = 0.065 a_0$$

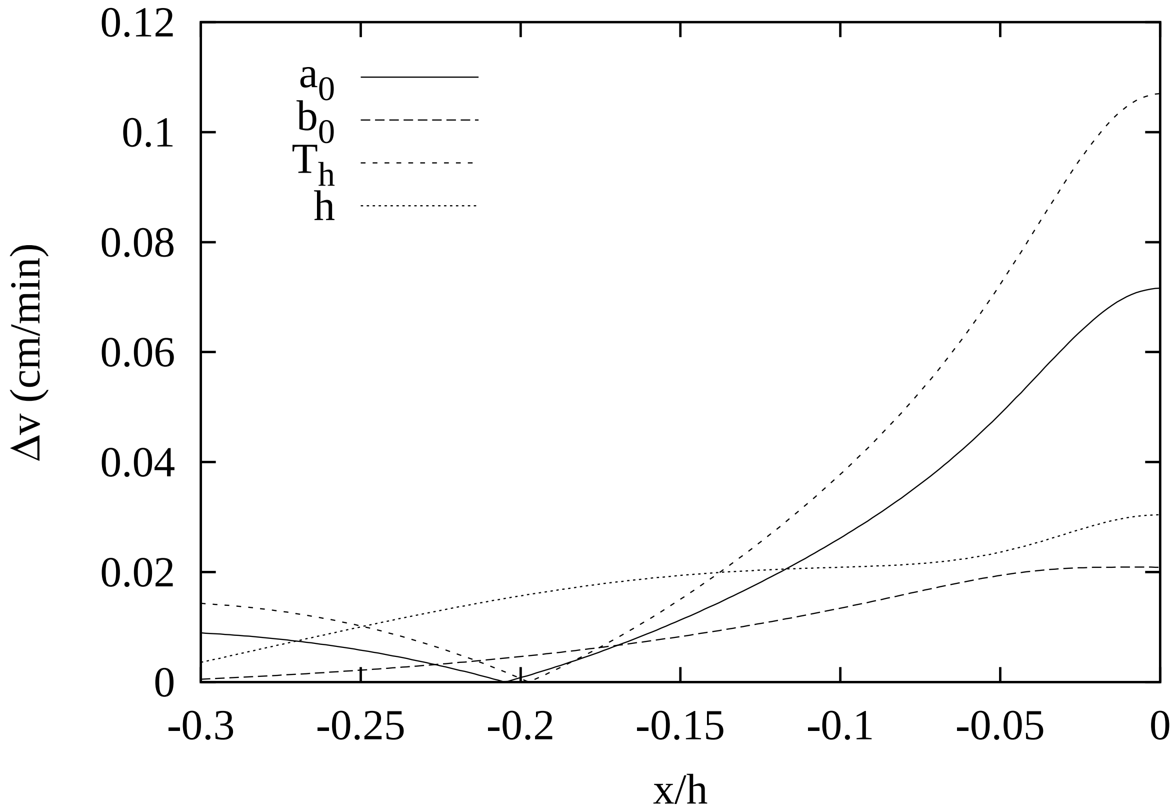
$$\Delta b_0 = 0.03 b_0$$

$$\Delta T_h = 0.02(T_h - T_0)$$

$$\Delta h = 0.02h$$

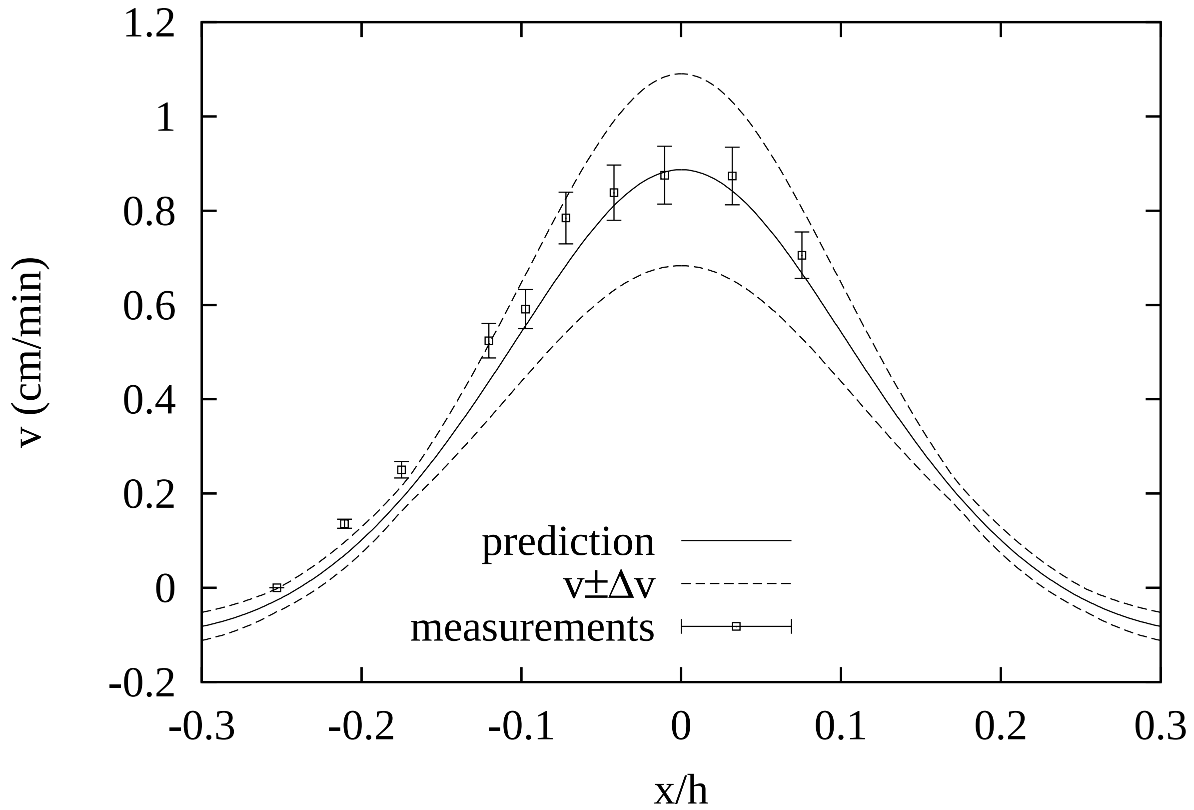


Contributions to Δv at $y = 0.4h$



Vertical velocity at $y = 0.65h$

$$\Delta a_0 = 0.065 a_0$$
$$\Delta b_0 = 0.03 b_0$$
$$\Delta T_h = 0.02(T_h - T_0)$$
$$\Delta h = 0.02h$$



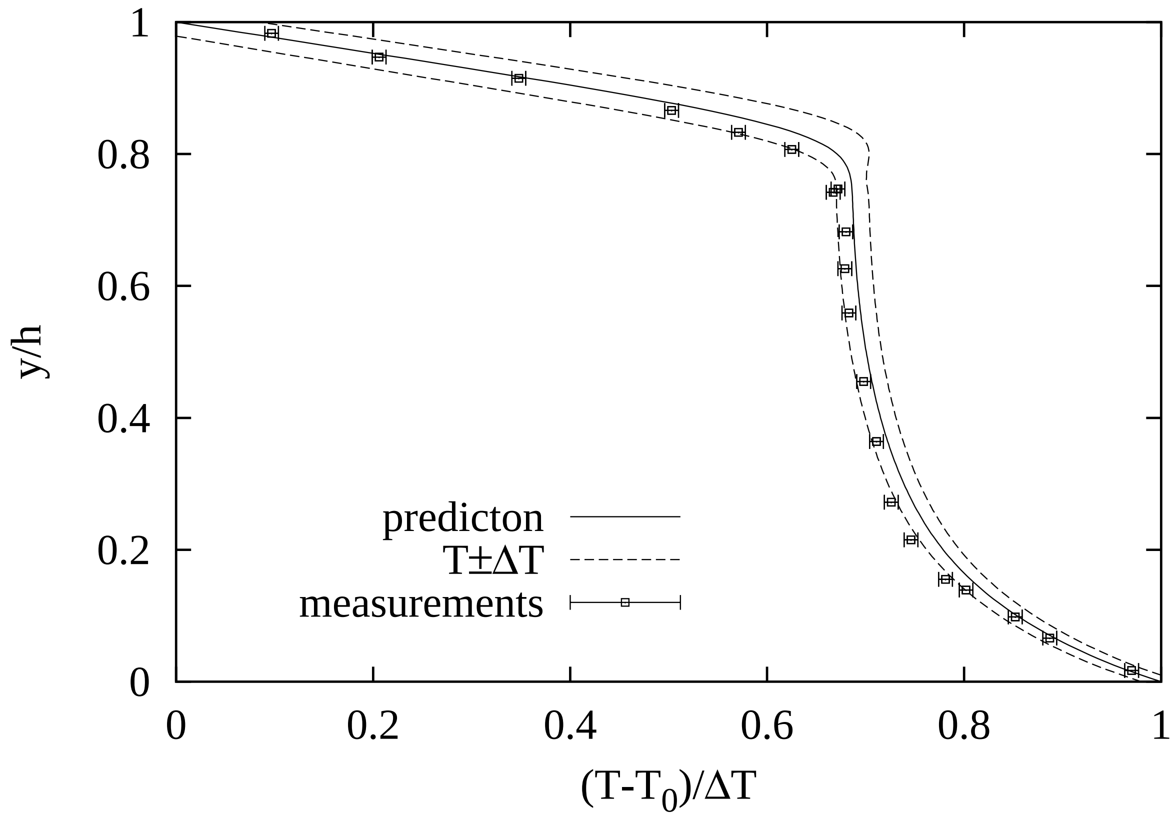
Temperature at $x = 0$

$$\Delta a_0 = 0.065 a_0$$

$$\Delta b_0 = 0.03 b_0$$

$$\Delta T_h = 0.02(T_h - T_0)$$

$$\Delta h = 0.02 h$$



Fast Nearby Solution

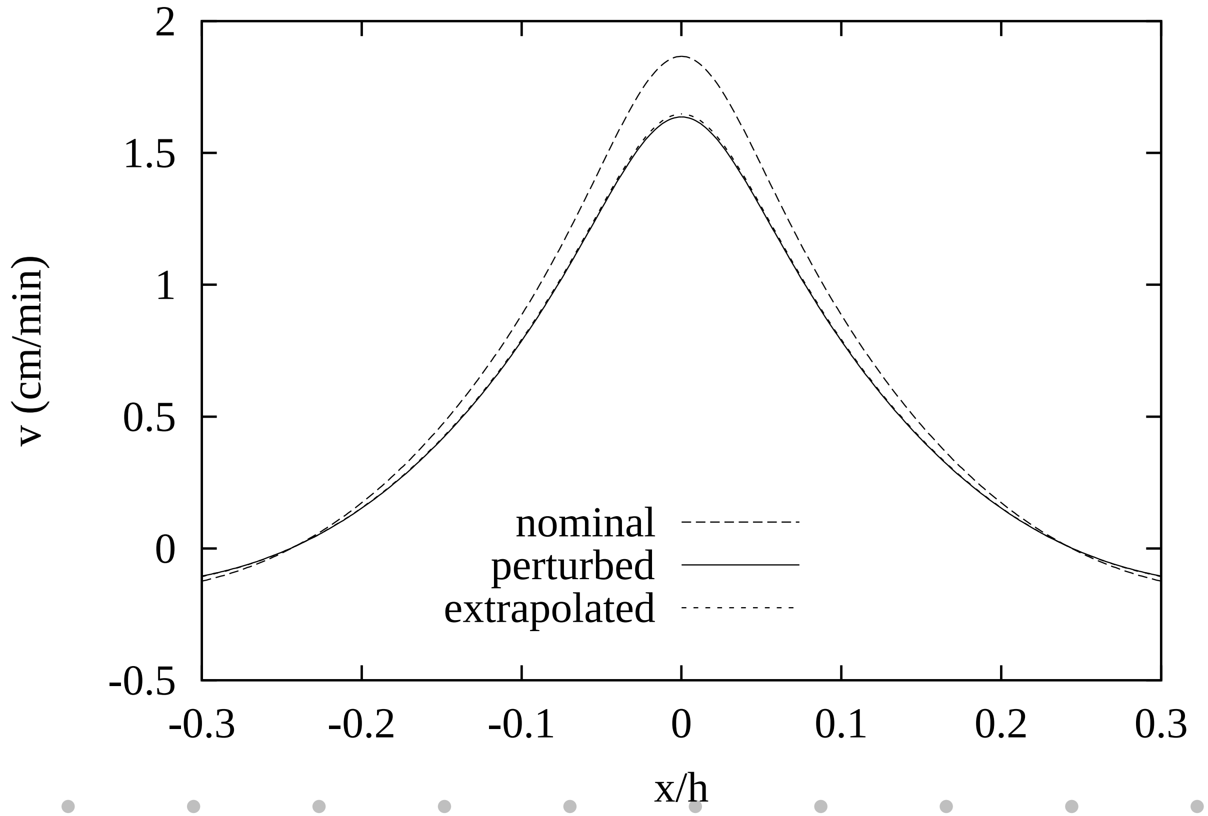
$$v(x; \boldsymbol{a} + \Delta \boldsymbol{a}) = v(x; \boldsymbol{a}) + \sum_{i=1}^{4} \frac{\partial v}{\partial a_i} \Delta a_i$$

$$\Delta a_0 = 0.065 a_0$$

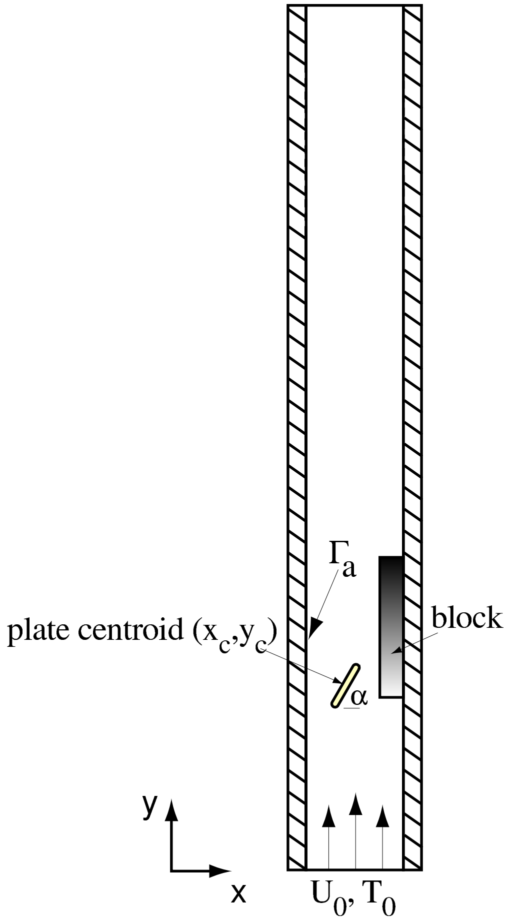
$$\Delta b_0 = 0.03 b_0$$

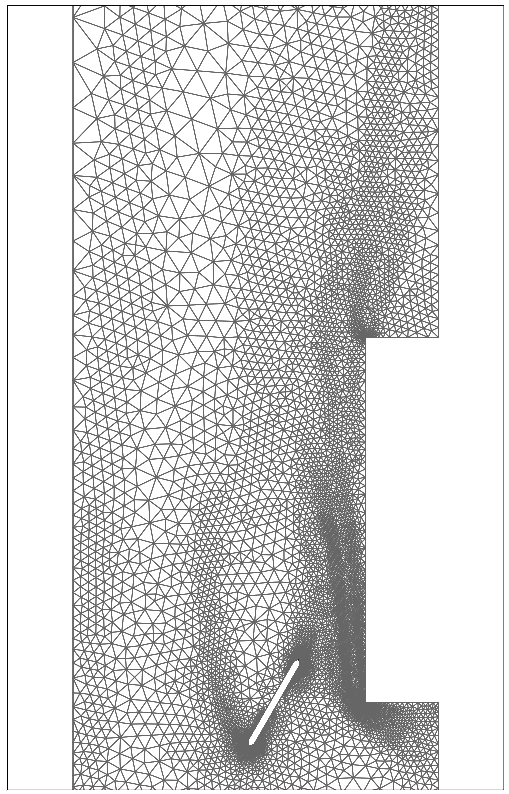
$$\Delta T_h = 0.02(T_h - T_0)$$

$$\Delta h = 0.02 h$$



A simple cooling system





A simple cooling system

Design parameter :

- plate angle α
- centroid coordinates (x_c, y_c)

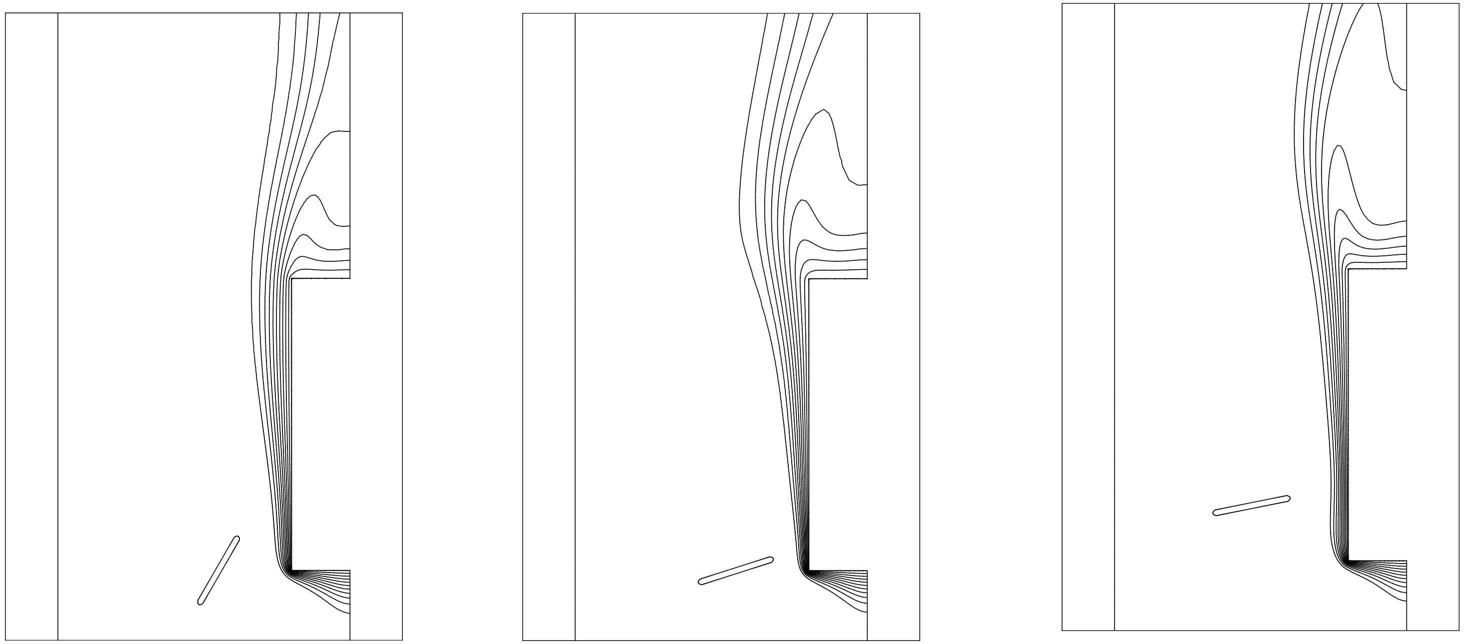
Objective function :

$$\mathcal{J}_1(\alpha, x_c, y_c) = \int_{\Gamma_b} \kappa \nabla T(\mathbf{x}; \alpha, x_c, y_c) \cdot \hat{\mathbf{n}} d\Gamma$$

Gradient of the objective function :

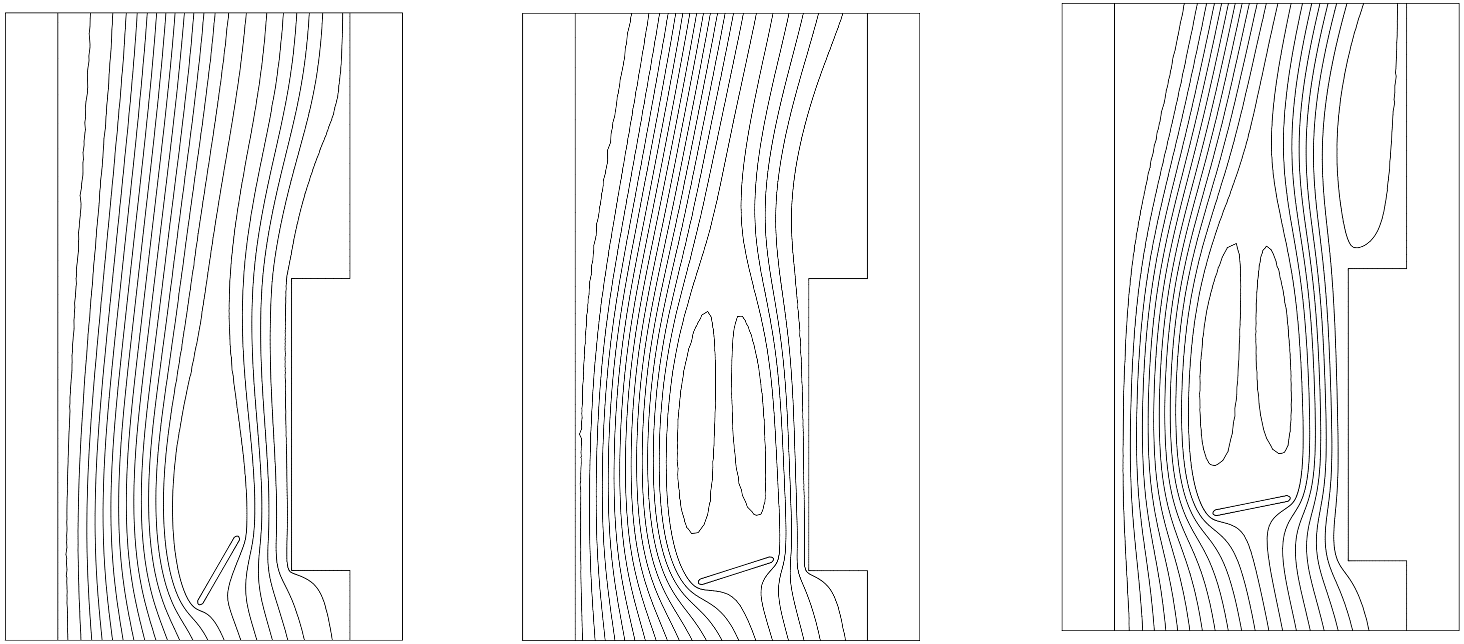
$$\frac{\partial}{\partial a} \mathcal{J}_1(\alpha, x_c, y_c) = \int_{\Gamma_b} \kappa \nabla s_T(\mathbf{x}; \alpha, x_c, y_c) \cdot \hat{\mathbf{n}} d\Gamma$$

Temperature fields



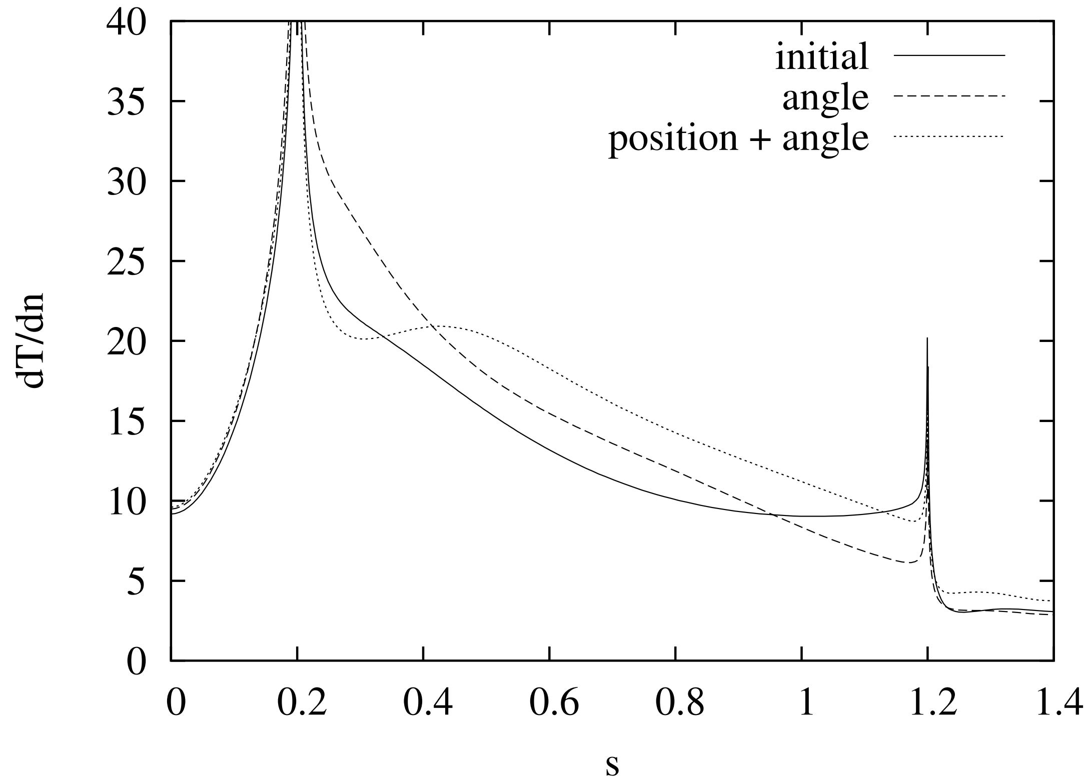
initial α (α, x_c, y_c)

Streamlines



initial α (α, x_c, y_c)

Heat Flux on thee block



Improvements : $\alpha = 10\% \quad (\alpha, x_c, y_c) = 15\%$

Optimisation path : α, x_c, y_c

It.	α	x_c	y_c	$\mathcal{J}$	$\frac{\partial}{\partial \alpha}\mathcal{J}$	$\frac{\partial}{\partial x_c}\mathcal{J}$	$\frac{\partial}{\partial y_c}\mathcal{J}$
0	17.7	0.550	1.000	0.07609	-0.00128	-0.04439	0.03101
1	17.5	0.433	1.082	0.07805	-0.00867	0.02027	0.01717
2	16.3	0.447	1.145	0.07890	-0.00843	0.01645	0.00658
3	14.8	0.454	1.158	0.07906	-0.00849	0.01140	0.02033
4	13.0	0.460	1.186	0.07931	-0.00770	0.01399	0.00168
5	11.4	0.467	1.190	**0.07946**	-0.00741	0.00846	0.01744
6	9.7	0.473	1.198	0.07945	-0.00590	0.00932	-0.00122

Maximum for non-zero gradient

Conclusion

- **Mesh adaptation provides**
  - cost effective solutions
  - accuracy for flow and sensitivities
- **General SEM formulation for complex flows**
  - identify important parameters
  - quantify flow uncertainty due to inputs
  - useful in optimal design
  - nearby flow evaluation : ***what if questions***