AE4233 Advanced design methods

MDO Practical session 5

- *Fmincon* and sensitivity analysis
- Surrogate handling in IDF
- Q3DAerosolver-EMWET load mapping

Dr. Ali Elham (a.elham@tudelft.nl)
Dr. Gianfranco La Rocca (g.larocca@tudelft.nl)
Ir. Durk Steenhuizen (d.steenhuizen@tudelft.nl)

April 13, 2014

FPP

TU Delft
Delft University of Technology

What you need to know to start an optimization with MATLAB

- FMINCON settings: options = optimset (...)

- Some options you HAVE TO define for your optimization:
  - Algorithm: SQP is recommended
  - DiffMinChange: not less than 0.01
  - DiffMaxChange: not higher than 0.1
  - TolFun
  - TolCon
  - TolX

Finding good values for them is a part of your assignment

TU Delft

What you need to know to start an optimization with MATLAB

- Some options are optional but recommended:
  - Display: Iter is recommended
  - PlotFun:

plotfunctions={@optimplotx @optimplotfval @optimplotconstrviolation};

See MATLAB help for “optimset” to find the available optimization options for fmincon

TU Delft

Sensitivity Analysis

- Gradient based algorithms need sensitivity of objective functions and constraints with respect to the design variables.
- Fmincon uses "finite difference" method for sensitivity analysis:

$$\frac{\partial f}{\partial x} = \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

The value of deltaX should be defined

TUDelft

Sensitivity Analysis

- Theoretically $\Delta x \to 0$
- Practical values in numerical analysis ~10^-6

- DiffminChange is the minimum value of Δx
- DiffMaxChange is the maximum value of Δx
- Matlab default value for DiffMinChange is 10^-6

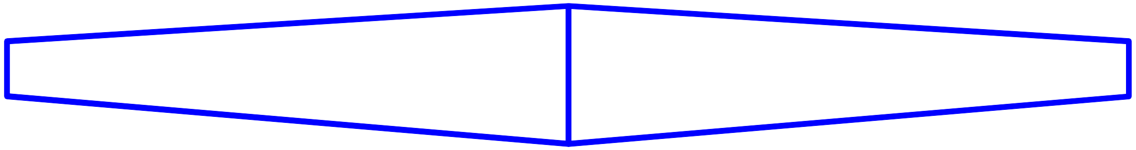
TU Delft

Sensitivity Analysis - Example

- Example: Sensitivity analysis of wing induced drag with respect to wing span using Q3D:

$$\left.\frac{\partial C_{Di}}{\partial b}\right|_{b=b_0} = \frac{C_{Di}(b_0+\Delta b) - C_{Di}(b_0)}{\Delta b}$$

- b0 = 29m



Sensitivity Analysis - Example

- For AOA = 2 deg → CDi = 0.0094
- Using Matlab default value for Δx :

$$f(x+\Delta x) = C_{Di}(b_0+\Delta b) = C_{Di}(29.000001)$$

- Running Q3D for new wing span of 29.000001m
  - For AOA = 2deg → CDi = 0.0094

$$\frac{\partial C_{Di}}{\partial b} = 0$$

Sensitivity Analysis - Example

- $\frac{\partial C_{Di}}{\partial b}=0$ means b=29m is the optimum wing span for minimum induced drag (other variables are fixed)

WRONG

- Use Δx = 0.1:
  - CDi(b=29.1) = 0.0093! Induced drag is reduced!

Q3D cannot see the effect of one micron difference in span on the induced drag, you need to change the span at least a few centimetres to have a change in Cdi

use diffminchange & diffmaxchange to definde deltaX

Effect of Scaling

- If you scale all the design variables: $x_{is} = \frac{x_i}{x_{i0}}$ → $b_s = \frac{b}{b_0} = \frac{b}{29}$
- Sensitivity @ b=29m: $b_s + \Delta b = 29/29 + 0.01*29 = 1.01*29 = 1.01 b_0$
- Δx is now percentage of the reference value… so $\Delta x = 0.01$ means 1% of the initial value of span: 29cm instead of 1cm for a 29m span

- If you do not scale your design variables for $\Delta x = 0.01$:
- $t/c + \Delta t/c = 0.14 + 0.01 = 0.15$ huge effect on objfun
- $W_{to} + \Delta W_{to} = 40000 + 0.01 = 40000.01$ No measurable effect on objfun

- For scaled design vector:

$$t/c + \Delta t/c = 0.14 + 0.01*0.14 = 0.1414$$
$$W_{to} + \Delta W_{to} = 40000 + 0.01*40000 = 40400$$

TUDelft

IDF Strategy

- From the second tutorial:

$$\min_{\mathbf{x}} \boxed{J = x_2^2 + x_3 + y_1 + e^{-y_2}}$$

$$y_1 = x_1^2 + x_2 + x_3 - 0.2y_2$$

$$y_2 = \sqrt{y_1} + x_1 + x_3$$

$$s.t.$$

$$\boxed{\begin{array}{l} y_1 / 3.16 - 1 \geq 0 \\ 1 - y_2 / 24 \geq 0 \end{array}}$$

$$-10 \leq x_1 \leq 10, 0 \leq x_2 \leq 10, 0 \leq x_3 \leq 10$$

In an IDF strategy both the objective function and the constraints are functions of surrogate variables y1* and y2*

TU Delft

Example DSM (IDF)

SO	$\mathbf{x}, y_2^*$	$\mathbf{x}, y_1^*$	$\mathbf{x}, y_1^*, y_2^*$
$y_1, r_1 = y_1^* - y_1$	$y_1 = x_1^2 + x_2 + x_3 - 0.2y_2^*$		
$y_2, r_2 = y_2^* - y_2$		$y_2 = \sqrt{y_1^*} + x_1 + x_3$	
J			$J = x_2^2 + x_3 + y_1^* + e^{-y_2^*}$

Optimizer

$$X = [x_1, x_2, x_3, y_1^*, y_2^*]$$

X

X

J

Objfun

$$J = x_2^2 + x_3 + y_1^* + e^{-y_2^*}$$

Constraints of the original problem (inequality constraints)

Constr.

$$\boxed{C_1 = 1 - y_1^* / 3.16 \quad C_2 = y_2^* / 24 - 1}$$

$$y_1 = x_1^2 + x_2 + x_3 - 0.2 y_2^*$$

$$y_2 = \sqrt{y_1^*} + x_1 + x_3$$

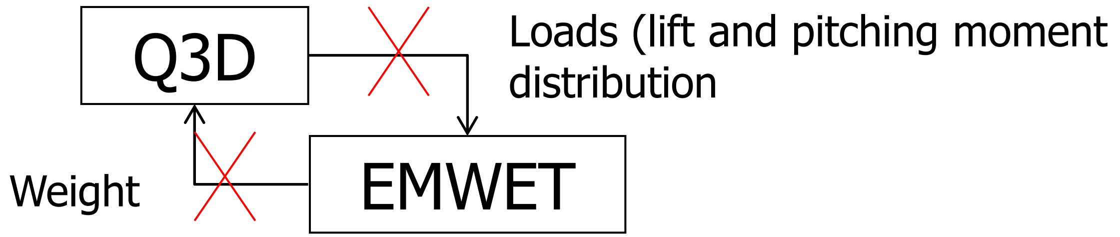
$$\boxed{Ceq_1 = y_1 - y_1^* \quad Ceq_2 = y_2 - y_2^*}$$

C_1, C_2, Ceq_1, Ceq_2

Consistency constraints on surrogate variables (equality constraints)

IDF approach for wing optimization

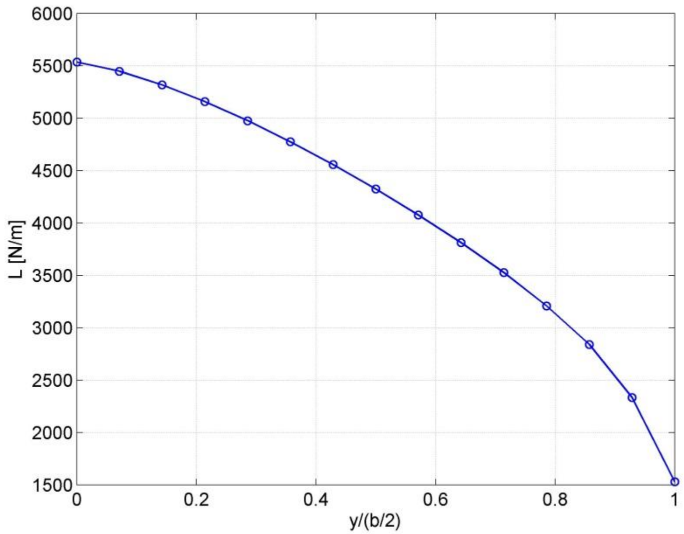
- Defining surrogate values for the coupling variables
- Loads are coupling variables:



In contrast to the weight, which is only one variable, for loads you need to define the spanwise distributions as surrogate variables

Load distribution using surrogate variables

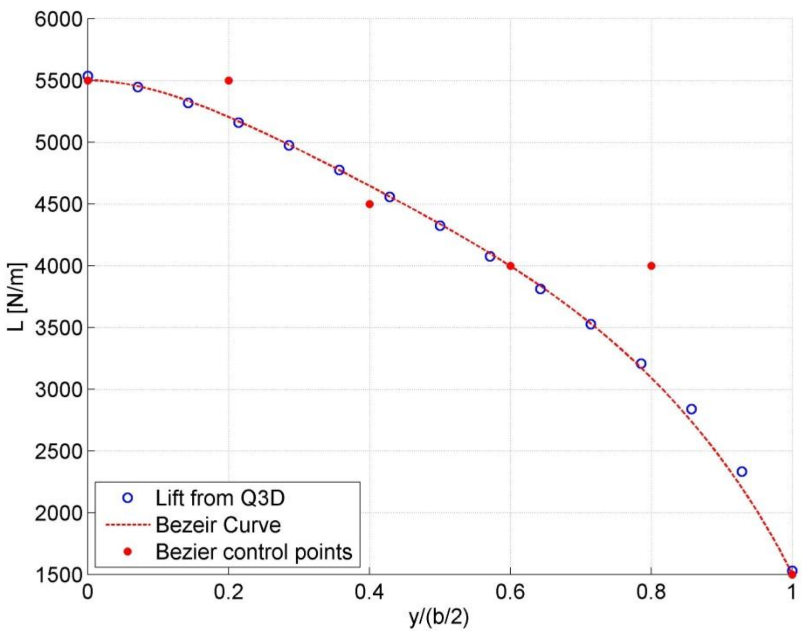
- Approach one:
  - Define loads at different points:
  - L = [L1 L2 … Li …Ln], Li is the load at yi
  - You need at least 15-20 points to have a nice load distribution



Load distribution using surrogate variables

- Approach two (optional):
  - Approximate the load distribution using a curve (Bezeir, B-Spline ,etc.) with 5-6 control points
  - Define the control point of the curve as design variables

A good approximation of the lift distribution is achieved using a Bezier curve with 6 control points



You can fix the spanwise position of the control points and define their weights (y positions) as design variables:

6 vs 15 design variables!

TUDelft

Making an automatic link between Q3D and EMWET

- Download the Matlab codes EMWET-Q3D.zip for connecting Q3D to EMWET from Blackboard

- Try to run the code and understand it

- Remove all the hard coded numbers in the code and use your design variables. Make an automatic link between the Q3D solver and EMWET.

TU Delft