A tiny introduction to MDO

Prof. Joseph Morlier

Thanks to materials provided by J. Martins, N. Bartoli, T. Lefebvre, S. Dubreuil and J. Mas Colomer

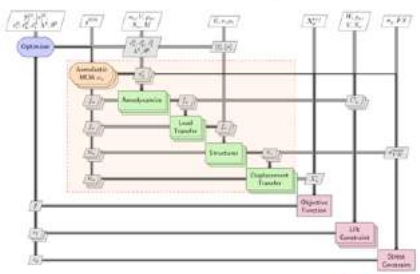
My Research Group (Joint research with ONERA on MDO)

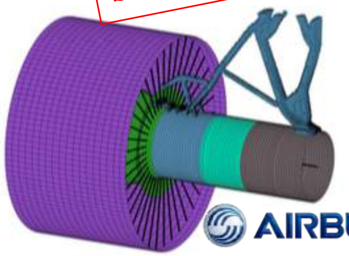
http://www.institut-clement-ader.org/pageperso.php?id=jmorlier

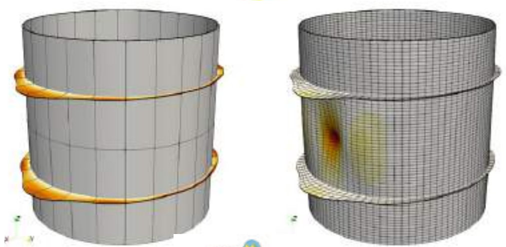
- 4 PhDs, 1 postdoc, 1 research assistant, 4 MsCs

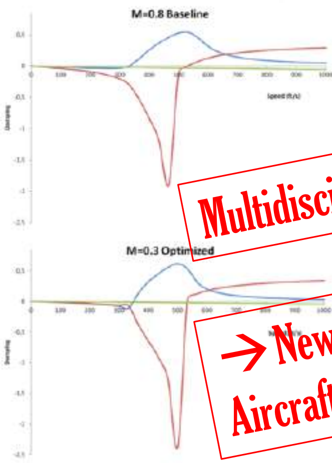
Structural Optimization

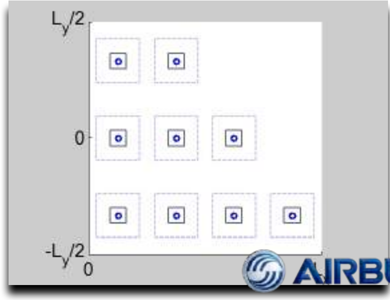
$$
\begin{aligned}
&\min_{\substack{a \in \mathbb{R}^{10} \\ c \in \Gamma^{10}}} \ w(a, c) \\
&s.t. \ \ s(a, c) \le 0 \\
&\qquad d(a, c) \le 0 \\
&\qquad \underline{a} \le a \le \overline{a}
\end{aligned}
$$





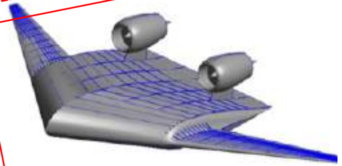


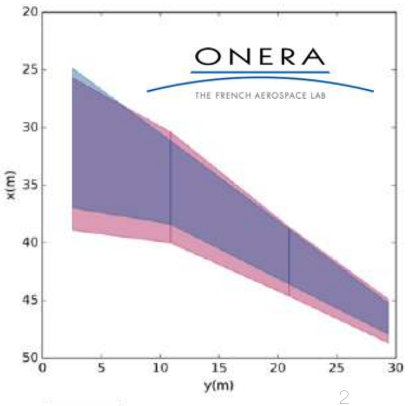




Multidisciplinary Design Optimization

→ New Aerostructures/ Aircraft Concept





Optimized planform (red) and baseline (blue).

CHAIR FOR ECO DESIGN OF AIRCRAFT

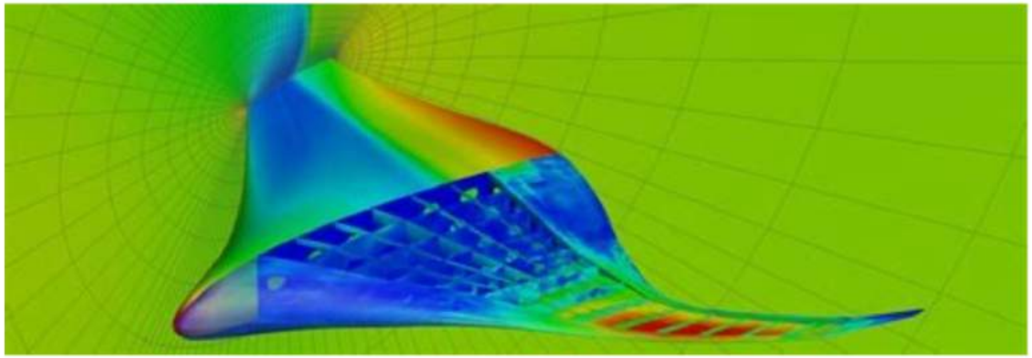
Outlines for today

multidisciplinary Design optimization

multidisciplinary optimization

1. MDA
2. MDO
3. Codesign

Popularization



http://mdolab.engin.umich.edu

Optimization [MDO] for connecting people?

Publié le 14 février 2019 Modifier l'article | Voir les stats

joseph morlier
Professor in Structural and Multidisciplinary Design Optimization, ... any idea?
2 articles

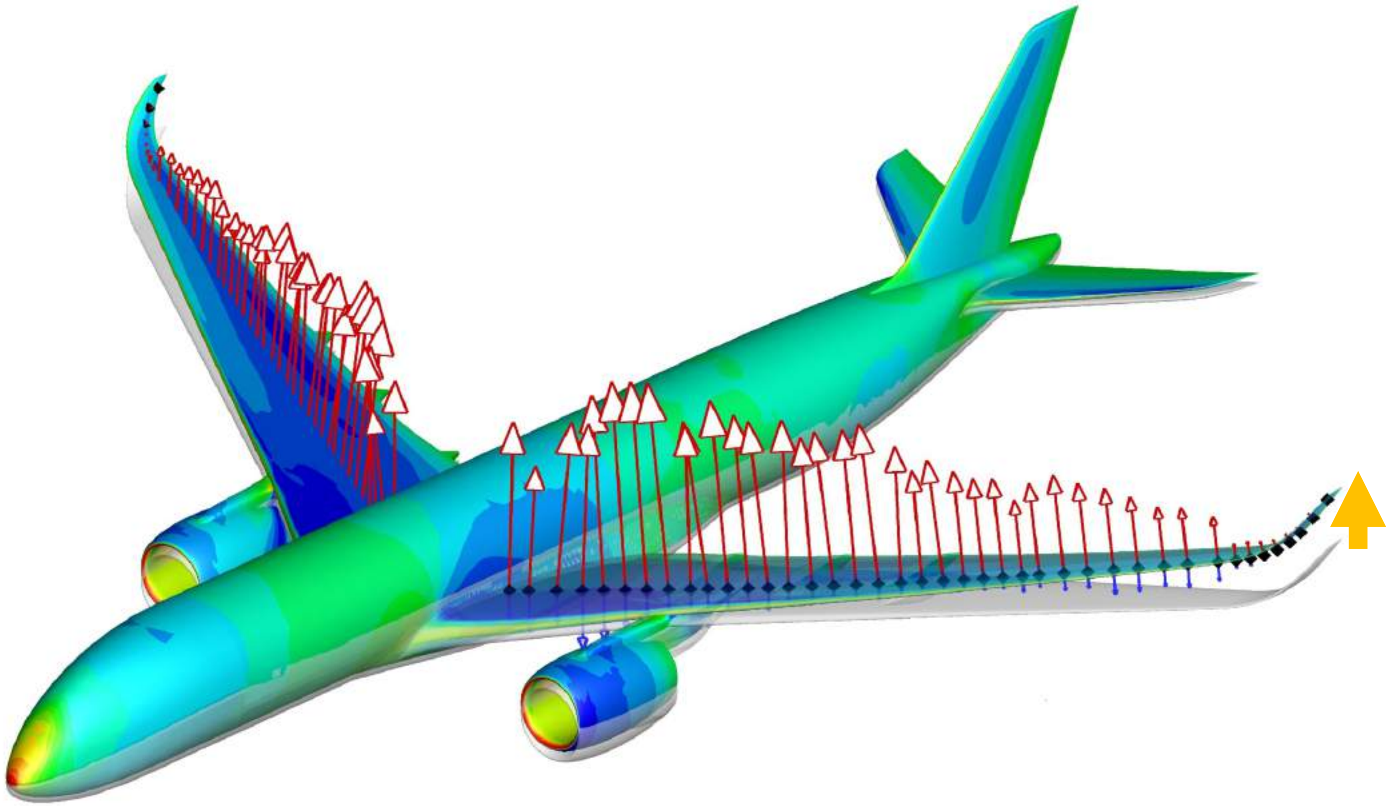
74 31 3 0

https://www.linkedin.com/pulse/loptimisation-multidisciplinaire-pour-connecter-les-humains-morlier/

Outlines for today

1. MDA
2. MDO
3. Codesign is MDO?

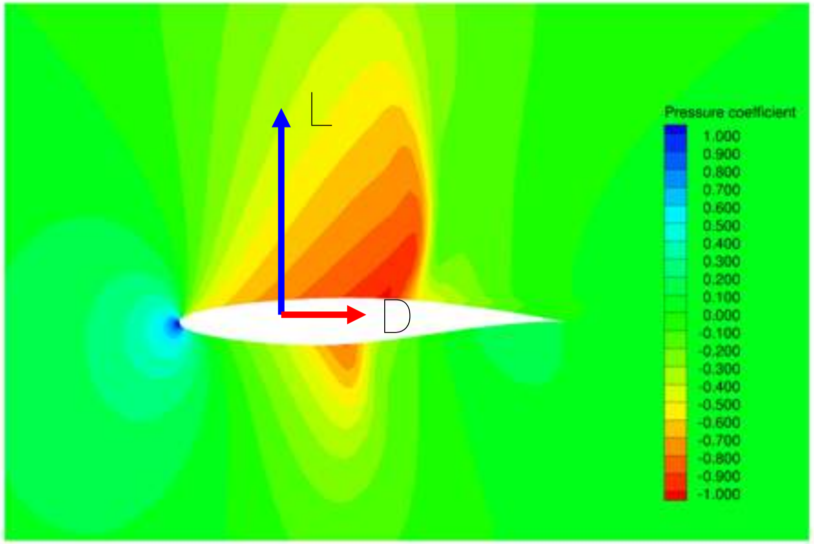
What is an MDA ? Static Aeroelasticity for example?



Source: DLR

But first, what is Disciplinary Optimization?

Example: Aerodynamics (L/D max)

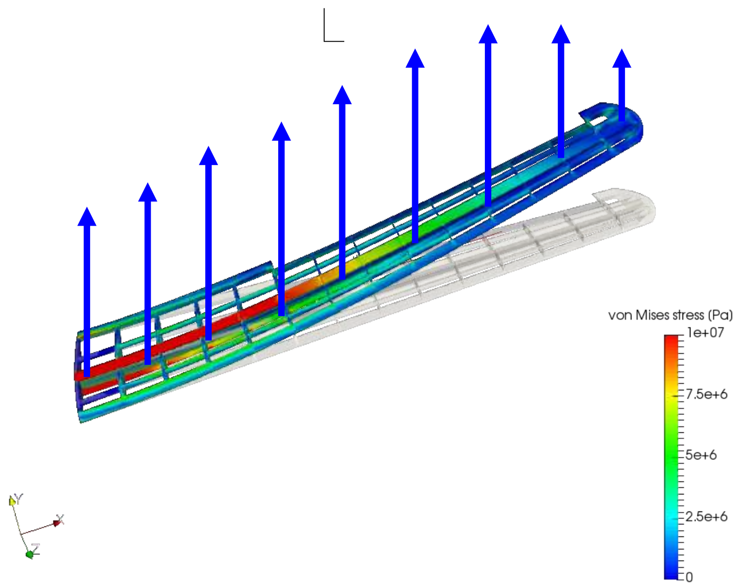


Source: NLR

Minimize D
w.r.t. shape, α
Subject to L = W

What is Disciplinary Optimization (2)?

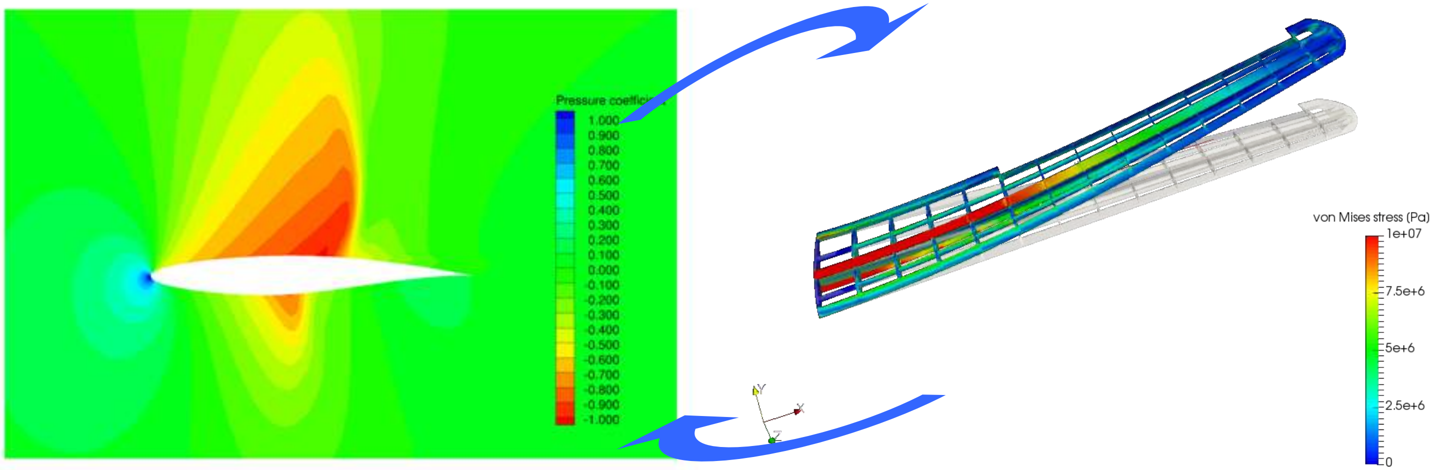
Another example: Structures



Source: simscale.com

Minimize Mass
w.r.t. thicknesses
Subject to $\sigma \leq \sigma y$

However… Disciplines are not isolated:



Structural deformation of wing → changes in the shape exposed to airflow

Changes in the shape exposed to airflow → changes in the aerodynamic loads

Then, how do we solve the complete system?

Nodal forces

Structure Analysis

Nodal displacements

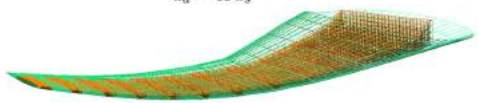
The solution of the complete system is the set of displacements and forces that “satisfy” this loop

Radial Basis Function displacement interpolation (same for y and z)

$$u_x = \sum_{i=1}^{N_s} \alpha_i^x \phi(\|\mathbf{x} - \mathbf{x_i}\|) + \gamma_0^x + \gamma_x^x x + \gamma_y^x y + \gamma_z^x z$$

Displacement interpolation matrix

$$u_a = H u_s$$



Loads using principle of virtual work

$$f_s = H^T f_a$$

Load Transfer

Rendall, T. C. S., & Allen, C. B. (2008). Unified fluid–structure interpolation and mesh motion using radial basis functions. International Journal for Numerical Methods in Engineering, 74(10), 1519-1559.

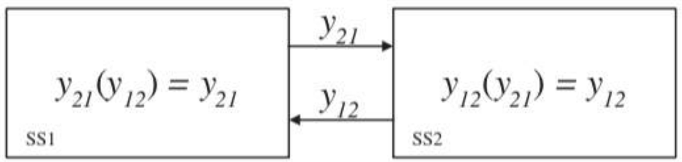
Forces on aero grid points

Aerodynamic Analysis

Aero grid displacements

Multi-Disciplinary Analysis

- Computation of the state variables at equilibrium for given x and z
  - Generally computed using a fixed-point algorithm (Jacobi or Gauss-Seidel)
  - Or a root-finding method (Newton-Raphson)



(Step 0) choose initial guess y_{12}^0, set $i = 0$

(Step 1) $i = i + 1$

(Step 2) $y_{21}^i = y_{21}(y_{12}^{i-1})$

(Step 3) $y_{12}^i = y_{12}(y_{21}^i)$

(Step 4) if $|y_{12}^i - y_{12}^{i-1}| < \varepsilon$ stop, otherwise go to **(Step 1)**

Check the default tolerence

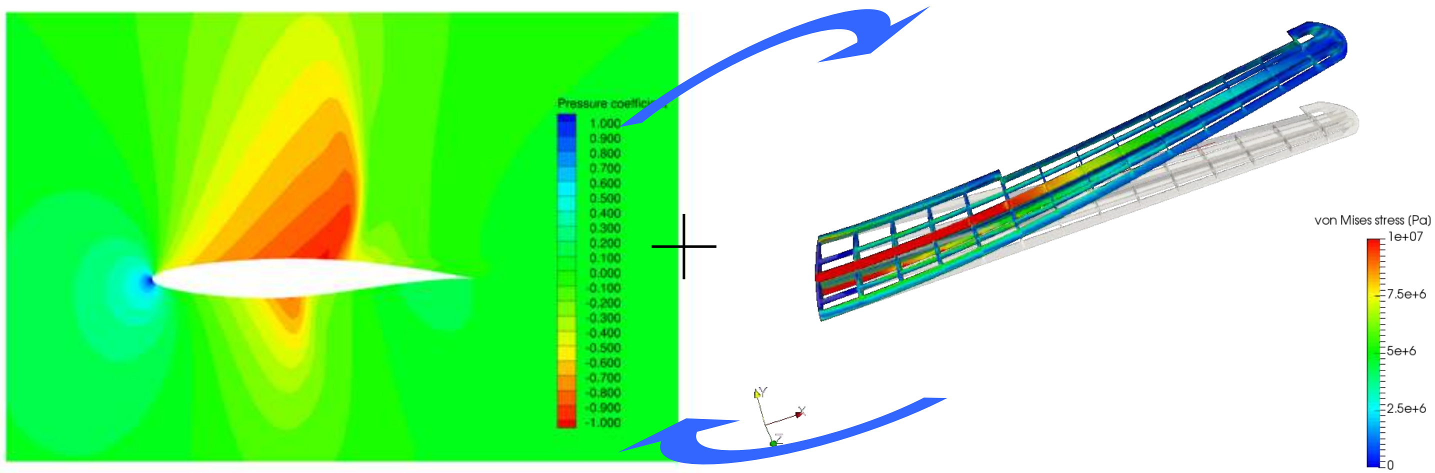
Examples of MDA

- file:///Recherche /Cours/optimstructures3A/SMO_Newcourse/cours JO/Day3/MDA_Intro/TutorialFPI/PROF/tutorialFPI.html

- https://github.com/nasa/NASTRAN-95

- https://github.com/mid2SUPAERO/aerostructures

→we need to analyze BOTH disciplines at the SAME TIME



Minimize D, or Mass, or a combination of D and Mass
w.r.t. shape, α, thicknesses
Subject to:

$$L = W$$

$$\sigma \leq \sigma y$$

In practice, how do we solve that problem?

One possible approach: MultiDisciplinary Feasible (MDF, probably the most intuitive one…)

Steps:

1. Start from a set of particular design variables: shape, α, thicknesses
2. Solve the complete system (with all the interactions) for these values
3. Evaluate objective function and constraints
4. From these values, the optimizer proposes a new set of design variables.

These steps are repeated until the optimum is reached.

Next: MDO …The big picture

Outlines for today

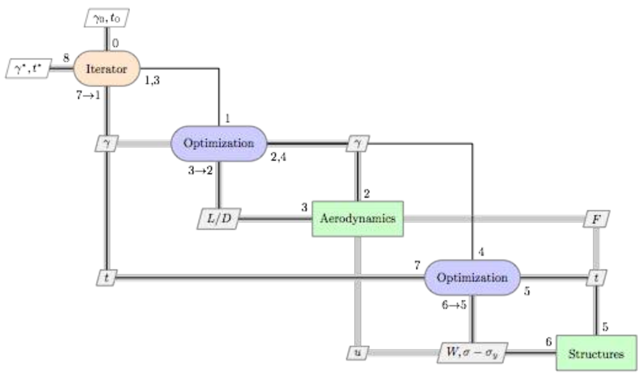
1. MDA

2. MDO

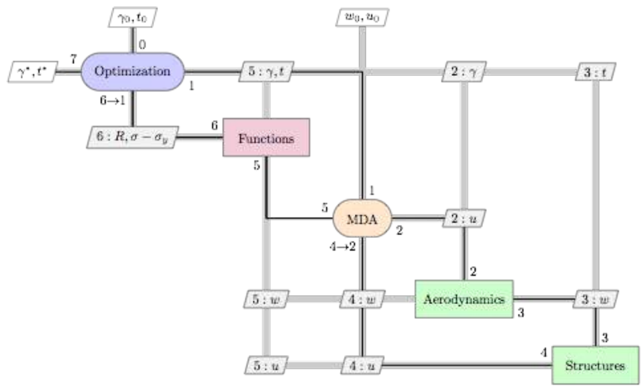
3. Codesign is MDO?

MDO optimizes all variables simultaneously, accounting for all the couplings

Sequential optimization



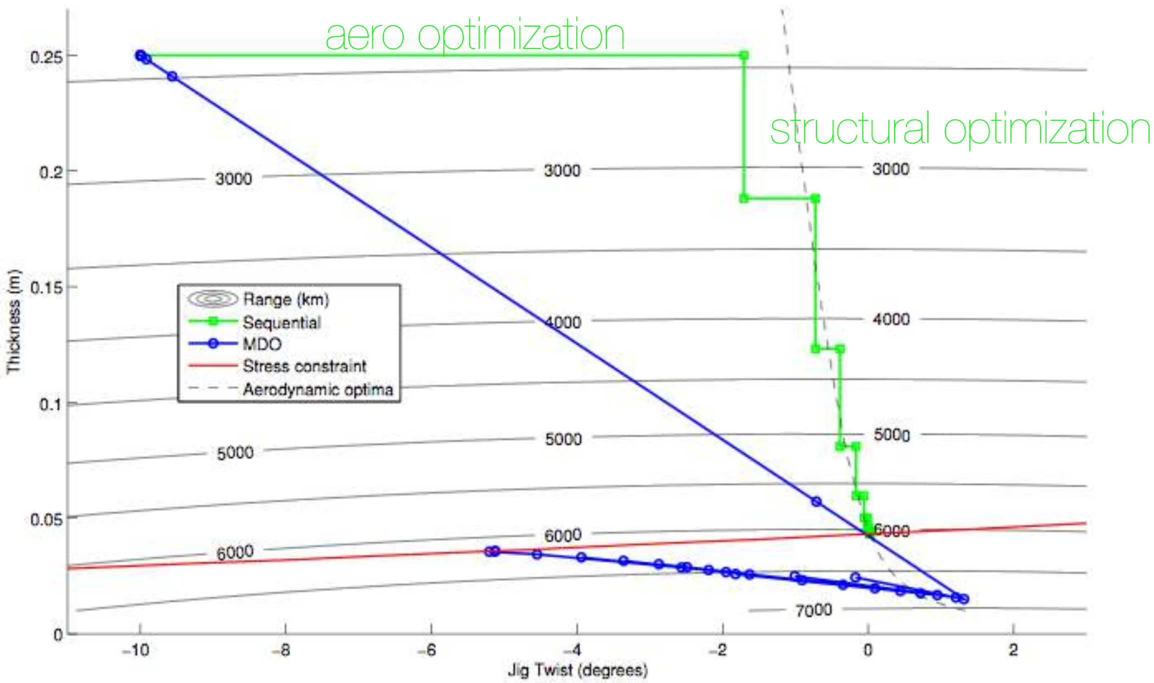
MDO



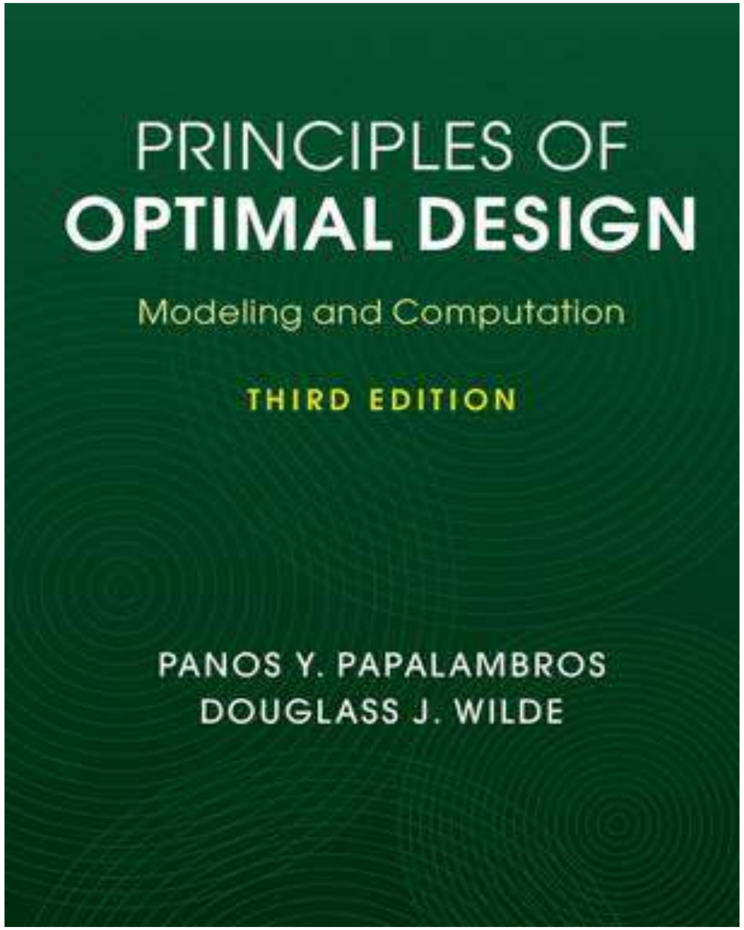
I. R. Chittick and J. R. R. A. Martins. An asymmetric suboptimization approach to aerostructural optimization. Optimization and Engineering, 10(1):133–152, Mar. 2009. doi:10.1007/s11081-008-9046-2.

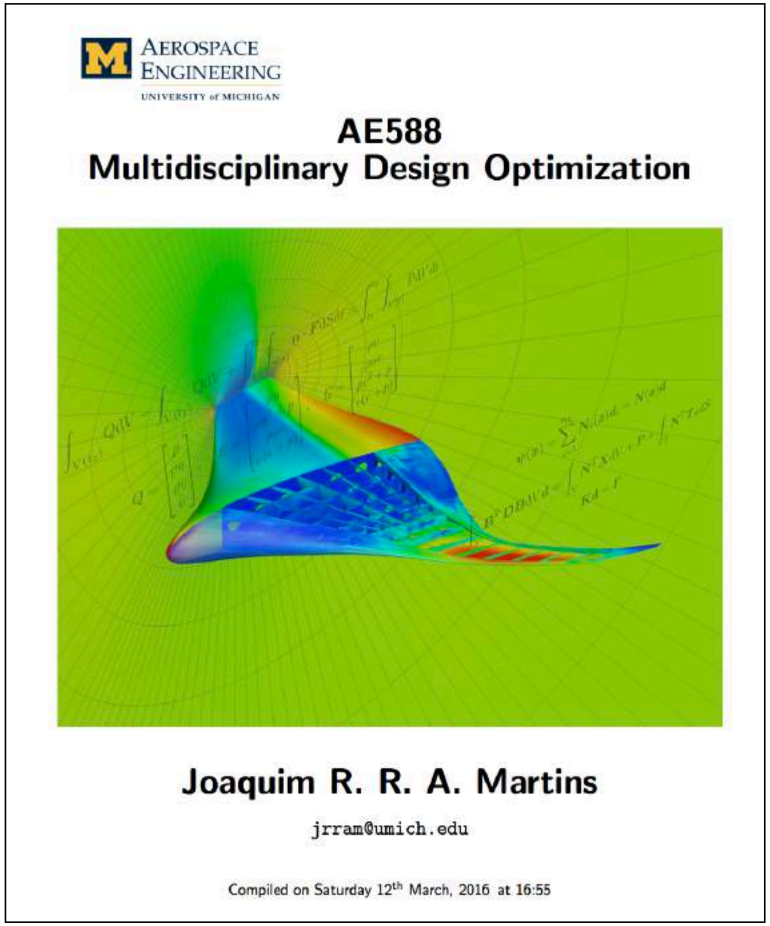
Sequential optimization fails to find the multidisciplinary optimum

Chittick, I. R., & Martins, J. R. (2008). Aero-structural optimization using adjoint coupled post-optimality sensitivities. Structural and Multidisciplinary Optimization, 36(1), 59-70.



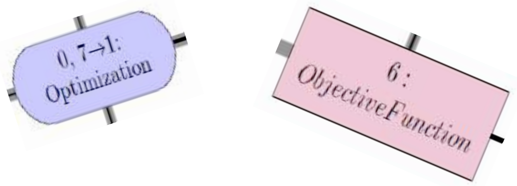
Good Starting Point (x0)



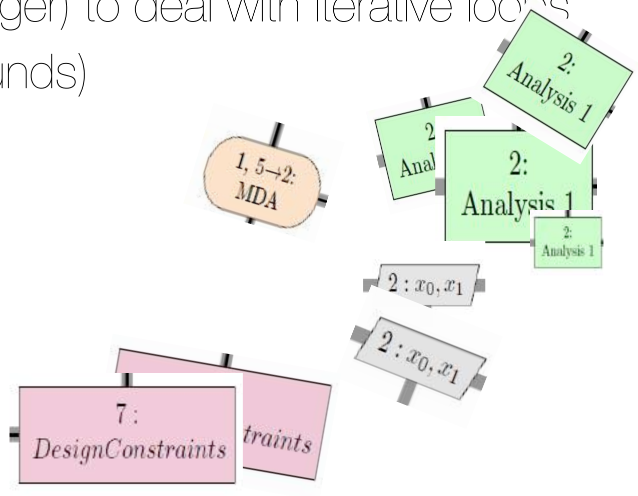


Assembling MDO systems

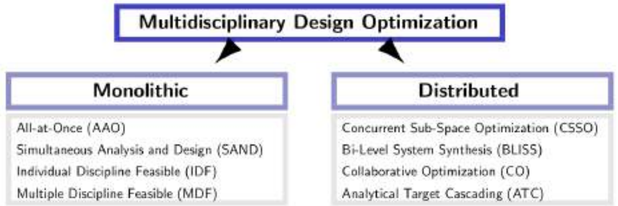
In order to assemble an MDO "architecture" we need a number of components:

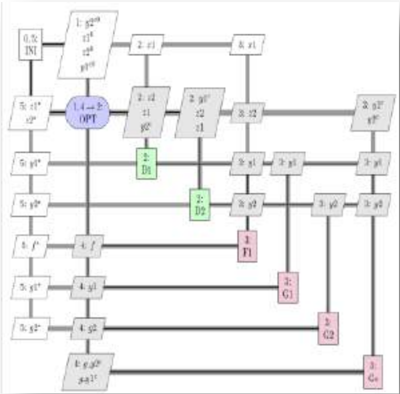


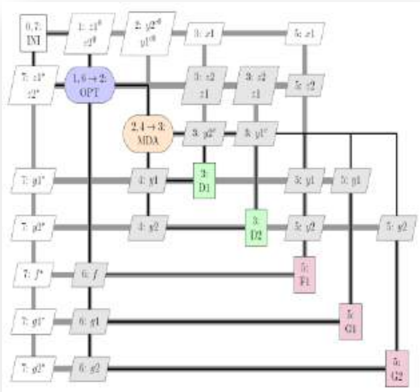
- One (or more) optimizers
- One (or more) objectives
- A number of disciplinary tools (or disciplines, or competences)
- Possibly some coordinator (or converger) to deal with iterative loops
- A bunch of design variables (with bounds)
- Some constraint specification

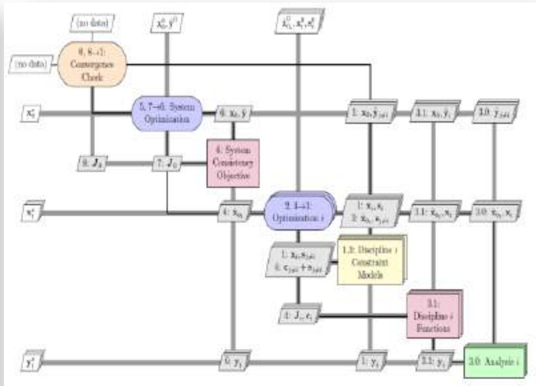


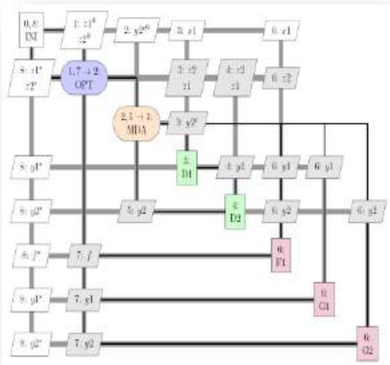
Assembling MDO systems

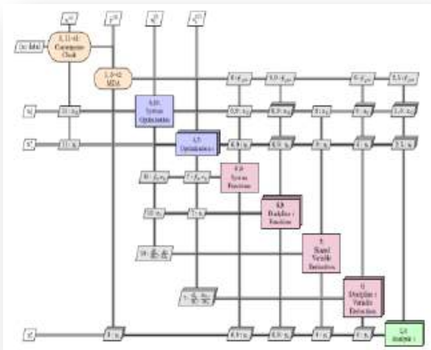


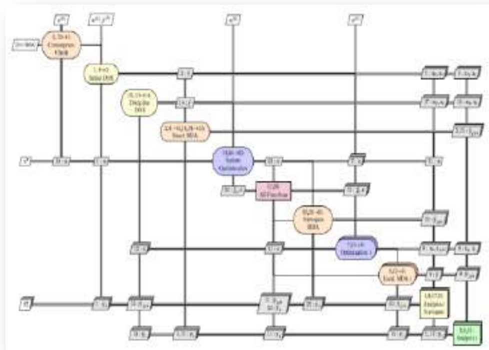












MDF Multidisciplinary Feasible approach—a complete analysis is performed at every optimization iteration. Also known as the All-in-One approach.

Illustrative example: the Sellar problem

2 disciplines involved

Variables: x_1, y_1, y_2, z_1, z_2

We'll see later what are the differences between these variables ...

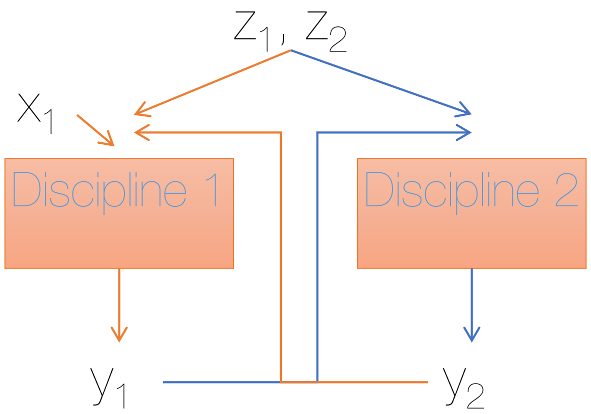
$$
\begin{gathered}
\text{minimize } x_1^2 + z_2 + y_1 + \exp(-y_2) \\
\text{with respect to } z, x \text{ or } (z_1, z_2, x_1) \\
\text{subject to:} \\
3.16 - y_1 \leqslant 0 \\
y_2 - 24 \leqslant 0 \\
-10 \leqslant z_1 \leqslant 10 \\
0 \leqslant z_2 \leqslant 10 \\
0 \leqslant x_1 \leqslant 10
\end{gathered}
$$

$$
\begin{gathered}
\text{Discipline 1}: y_1(z_1, z_2, x_1, y_2) = z_1^2 + x_1 + z_2 - 0.2y_2 \\
\text{Discipline 2}: y_2(z_1, z_2, y_2) = \sqrt{y_1} + z_1 + z_2
\end{gathered}
$$

Sellar, R. S., Batill, S. M., and Renaud, J. E., "Response Surface Based, Concurrent Subspace Optimization for Multidisciplinary System Design", 34th Aerospace Sciences Meeting and Exhibit, Aerospace Sciences Meetings, 1996.

Illustrative example: the Sellar problem

- Design variables: z_1, z_2, x_1 to minimize the objective
- Shared (or global) variables: z_1, z_2
- Local variable: x_1
- Coupling variables: y_1, y_2



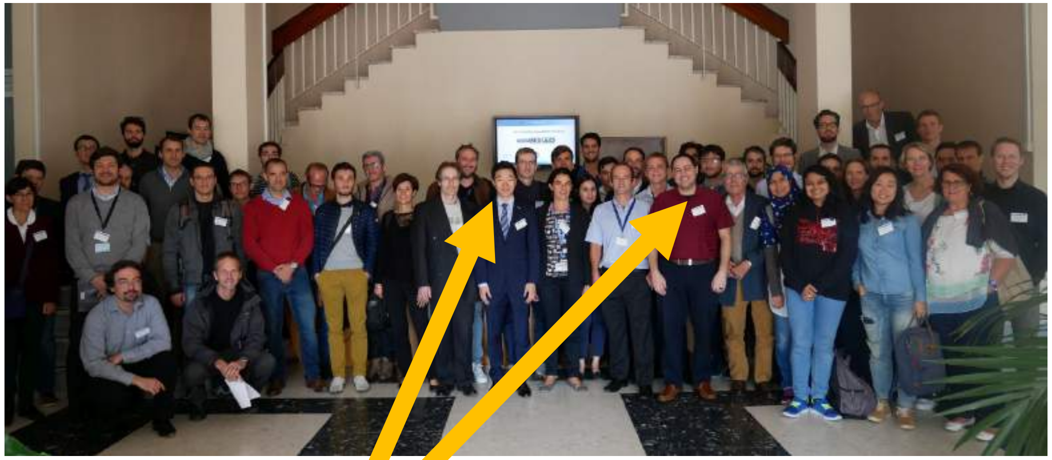
$$\begin{aligned} &\text{minimize } x_1^2 + z_2 + y1 + e^{-y_2} \\ &\text{with respect to } z_1, z_2, x_1 \\ &\text{subject to:} \\ &\frac{y_1}{3.16} - 1 \geq 0 \\ &1 - \frac{y_2}{24} \geq 0 \\ &-10 \leq z_1 \leq 10 \\ &0 \leq z_2 \leq 10 \\ &0 \leq x_1 \leq 10 \end{aligned}$$

$$\begin{aligned} &\text{Discipline 1: } y_1(z_1, z_2, x_1, y_2) = z_1^2 + x_1 + z_2 - 0.2y_2 \\ &\text{Discipline 2: } y_2(z_1, z_2, y_1) = \sqrt{y_1} + z_1 + z_2 \end{aligned}$$

Multidisciplinary analysis (MDA) consists in solution of the following equations

$R_1 = 0$ ➔ y_1 and y_2 solutions

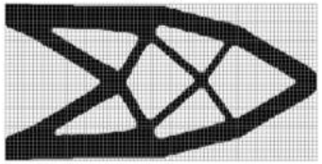
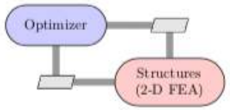
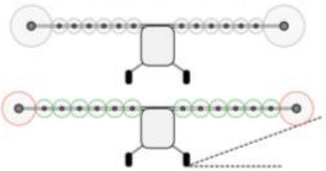
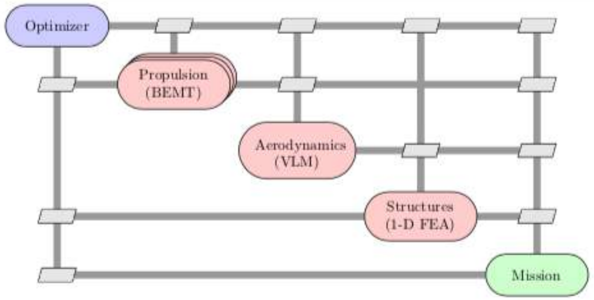
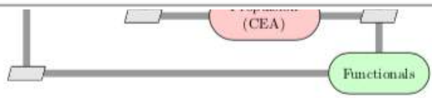
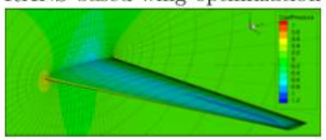
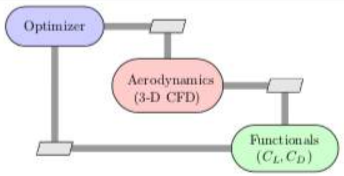
$R_2 = 0$



1st European OpenMDAO Workshop – Octobre 2017 - ONERA

- Originally developed by team at NASA Glenn
- Python-based, open source framework for coupling multiple models and optimization
- Facilitates collaboration between industry, academia, and government
- Provides a common platform for the development of new multidisciplinary analysis and design methods

J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, and B. A. Naylor, "OpenMDAO: An Open-Source Framework for Multidisciplinary Design, Analysis, and Optimization," Structural and Multidisciplinary Optimization, 2019.

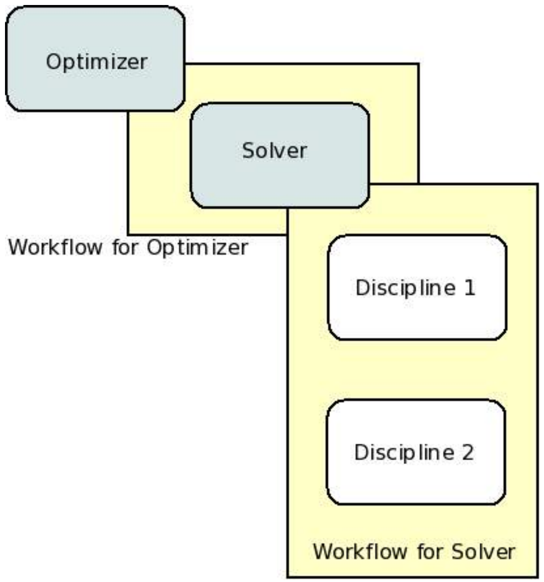
Problem	Model structure	Design variables	Objective	Constraints
Structural topology optimization 		Element densities	Compliance	Mass fraction
CTOL electric aircraft MDO 		Altitude prof., velocity prof., prop RPM profs., prop chord, prop twist, prop diam., wing twist, beam thickness	Range	Average speed, eqs. of motion, max. power, min. torque, ground clear., tip speed, wing failure
				
RANS-based wing optimization 		Shape	Drag coefficient	Lift coefficient

System is the base class in OpenMDAO

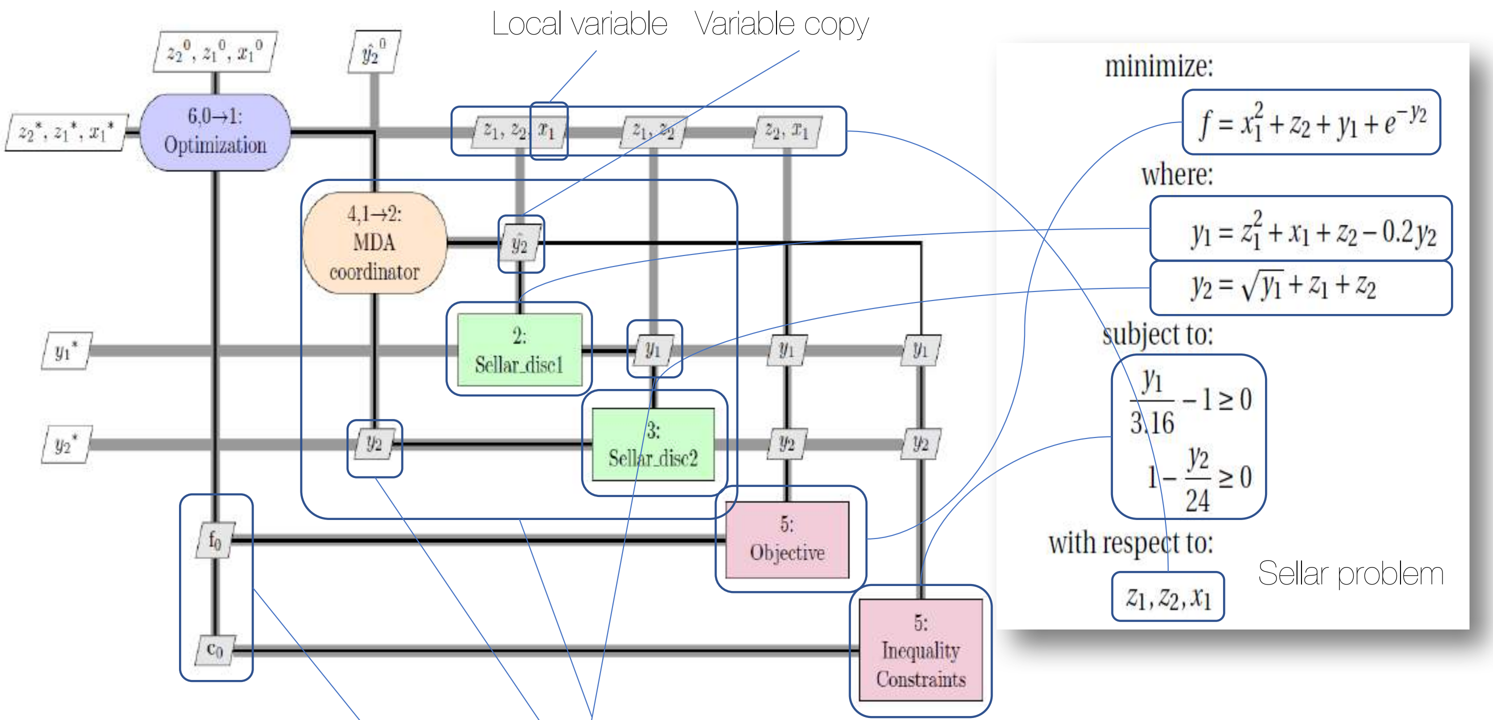
- System: base class for Component and Group classes. Represents a system of equations with variables:
  - params: input variables
  - unknowns: variables that are solved for. Can be explicitly defined (outputs) or implicitly defined (states)
  - resids: define the states implicitly
- The main System member functions are
  - solve_nonlinear: computes the unknowns for a give set of params.
  - apply_nonlinear: computes the residual values for a given state value

Multidisciplinary Feasible (MDF)

- The MDF architecture is the most intuitive for engineers
- The optimization problem formulation is identical to the single discipline case, except the disciplinary analysis is replaced by an MDA



MDF illustration on the Sellar problem:
MDF – Gauss-Seidel variant



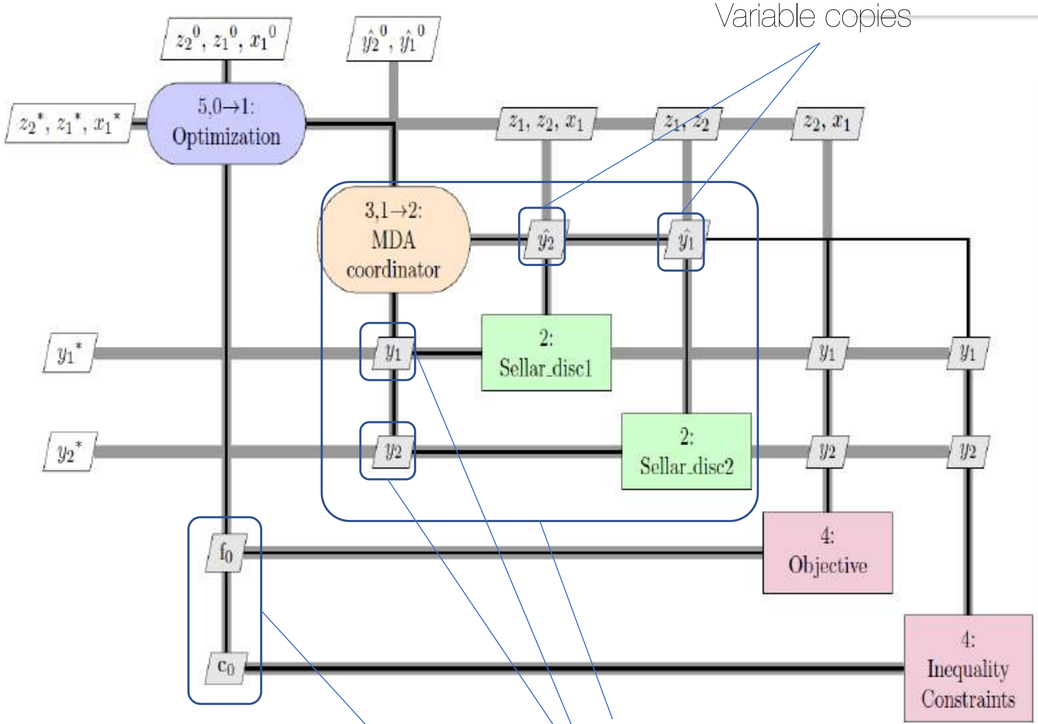
Coordinated MDA system with sequential discipline execution

Objective and constraints are sent to optimizer only after MDA coordinator has converged ($y_2 = \hat{y}_2$)

Coupling variables

MDF illustration on the Sellar problem:
MDF - Jacobi variant

Variable copies



minimize:

$$f = x_1^2 + z_2 + y_1 + e^{-y_2}$$

where:

$$y_1 = z_1^2 + x_1 + z_2 - 0.2 y_2$$

$$y_2 = \sqrt{y_1} + z_1 + z_2$$

subject to:

$$\frac{y_1}{3.16} - 1 \geq 0$$

$$1 - \frac{y_2}{24} \geq 0$$

with respect to:

$$z_1, z_2, x_1$$

Sellar problem

Coordinated MDA system with parallel discipline execution

Objective and constraints are sent to optimizer only after MDA coordinator has converged (y_1=^y_1 and y_2=^y_2)

Coupling variables

Multidisciplinary Feasible (MDF)

- Advantages:
  - Intuitive procedure/no specialized knowledge required → Easy to incorporate existing models
  - Always return **a system design that satisfies the consistency constraints**, even if the optimization process is terminated early – good from a pratical engineering point of view

- Disadvantages:
  - Intermediate results do not necessarely satisfy the optimization constraints
  - Cannot be parallelized
  - Developing **the MDA procedure with CSM/CFD might be time consuming***, if not already available

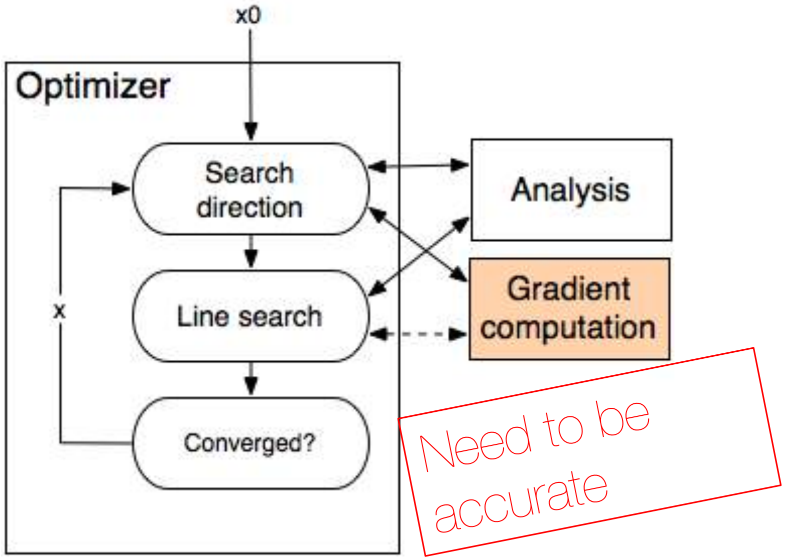
*** Automatic mapping, postprocessing etc…**

Gradients of the coupled system more challenging to compute see MAUD

Optimizer solver

- Requirements
- Problem to solve $\begin{cases} \min f(x) \\ wrt\ x \in R^d \\ st\ g_i(x) \leq 0 \text{ for } i = 1, \ldots m \end{cases}$

- Derivative Fee Optimizer (DFO)
- Evolutionnary Strategies (ES)
- **Surrogate based Optimizer (SBO)**
- . . .
- Gradient based Optimizer
- Computation of the derivatives of $f(x)$ and $g_i(x)$ to iterate and satisfy the KKT optimality conditions



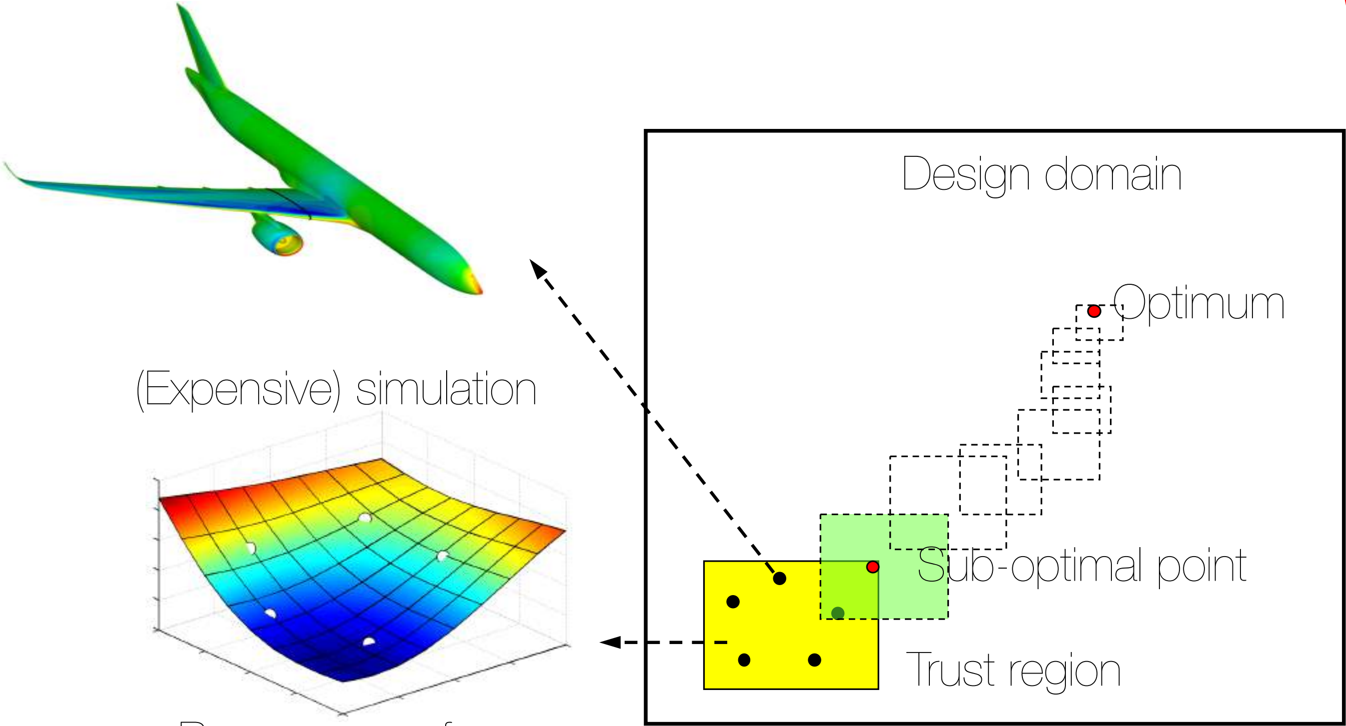
➔ Focus : computation of sensitivities (adjoint vs direct)

$$\frac{\partial f}{\partial x_i}, \frac{\partial g}{\partial x_i}, \frac{\partial h}{\partial x_i}$$

SURROGATE MODELING (learning for Optimizing)

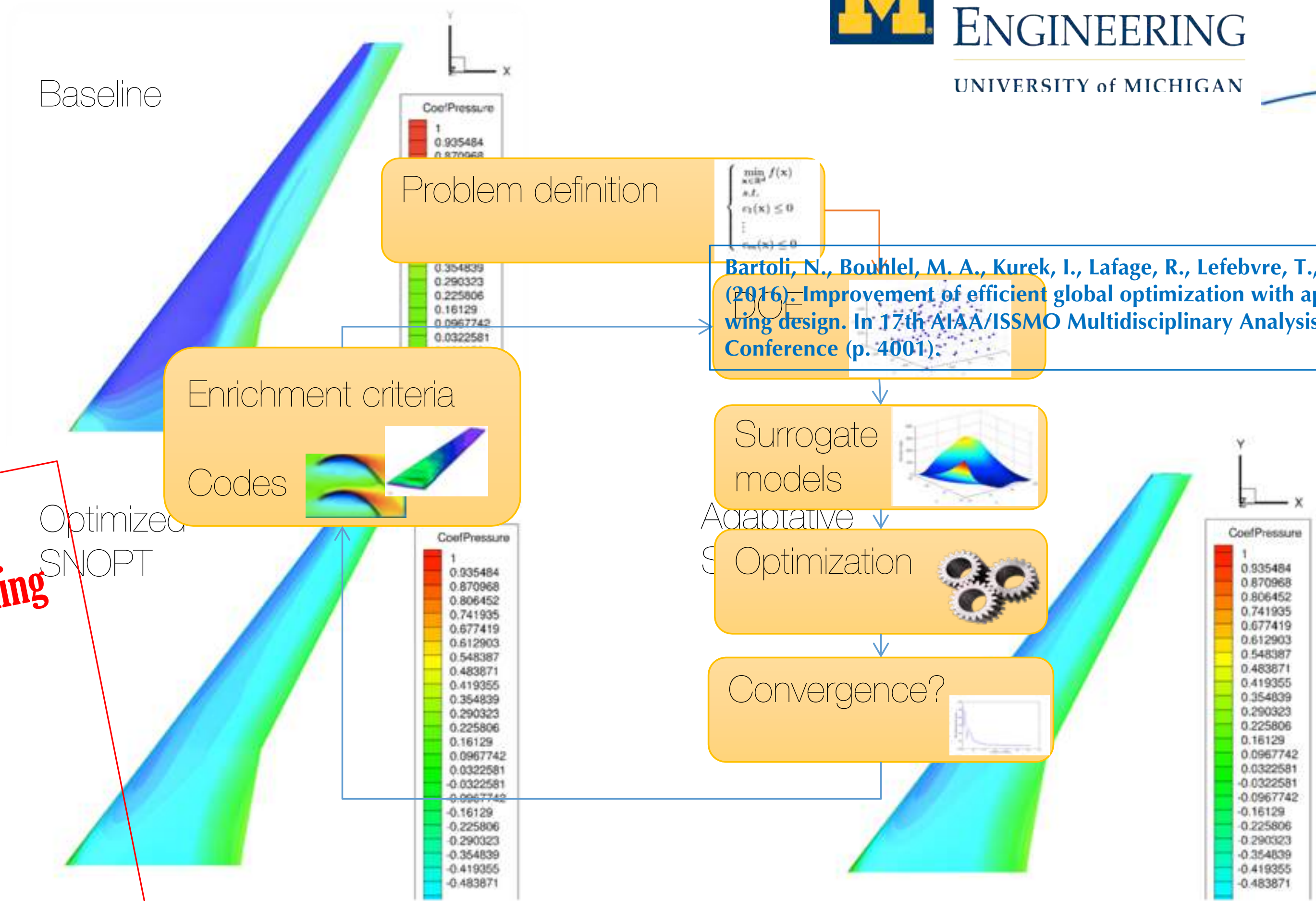
Jacobs, J. H., Etman, L. F. P., Van Keulen, F., & Rooda, J. E. (2004). Framework for sequential approximate optimization. Structural and Multidisciplinary Optimization, 27(5), 384-400.

But Why?



Response surfaces
Or metamodels, surrogate models etc…

SBO



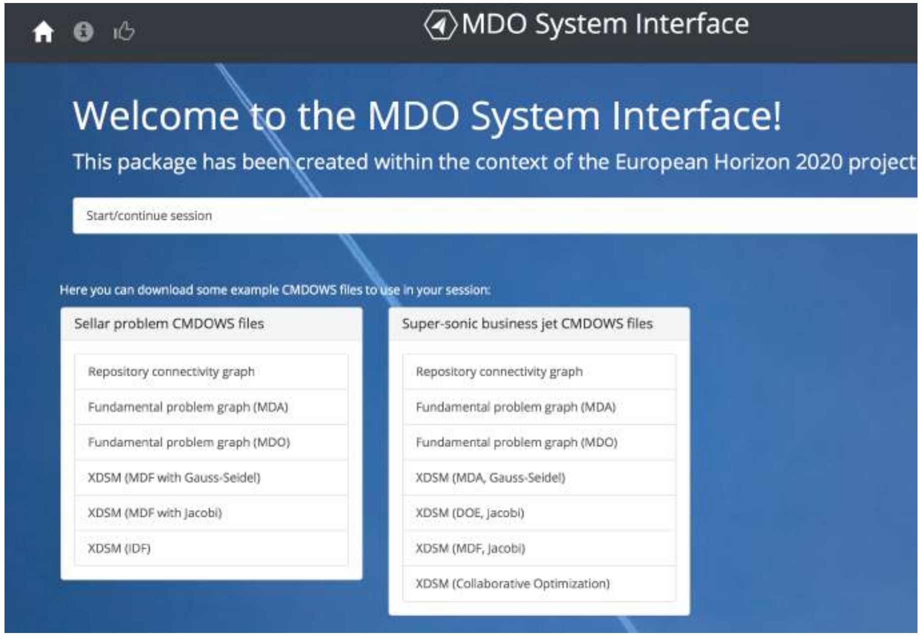
Bartoli, N., Bouhlel, M. A., Kurek, I., Lafage, R., Lefebvre, T., Morlier, J., & Regis, R. (2016). Improvement of efficient global optimization with application to aircraft wing design. In 17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference (p. 4001).

Use SMT for surrogate modeling !
No baseline…
No need of derivatives…

Example Sellar

AGILE AIRCRAFT 3RD GENERATION MDO FOR INNOVATIVE COLLABORATION OF HETEROGENEOUS TEAMS OF EXPERTS

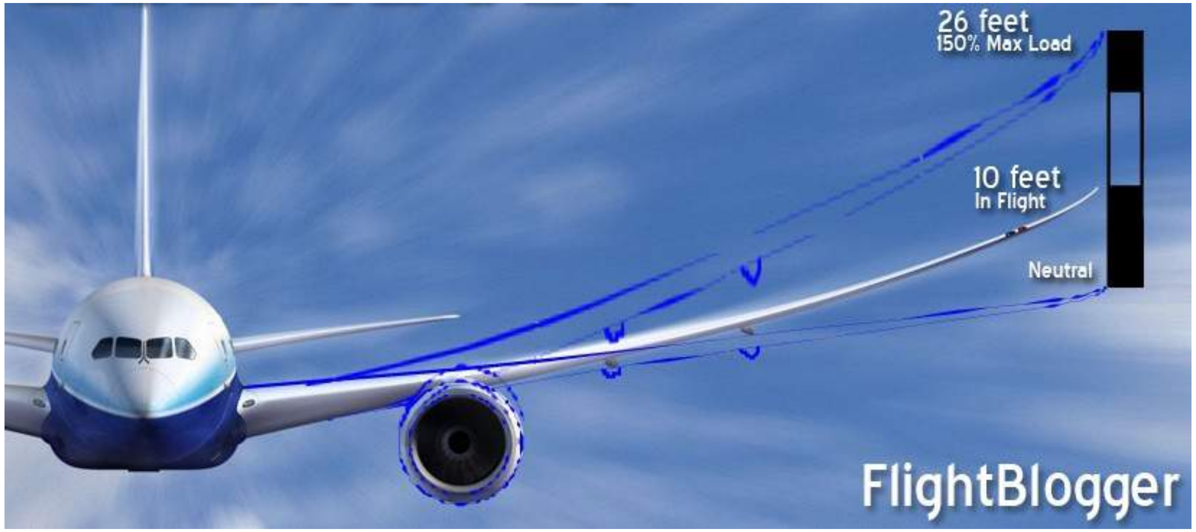
http://mdo-system-interface.agile-project.eu

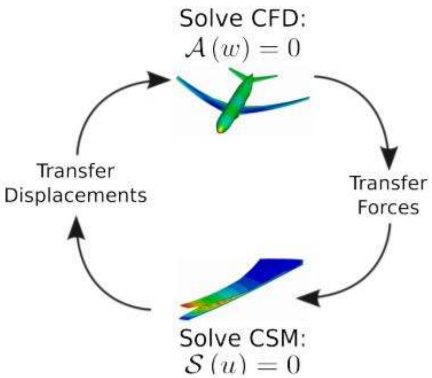


1. MDA
2. MDO

3. Codesign?

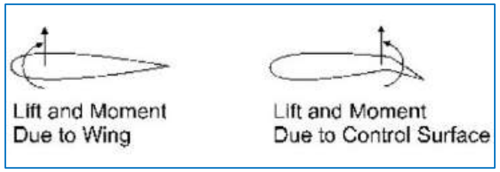
The importance of aerostructural coupling



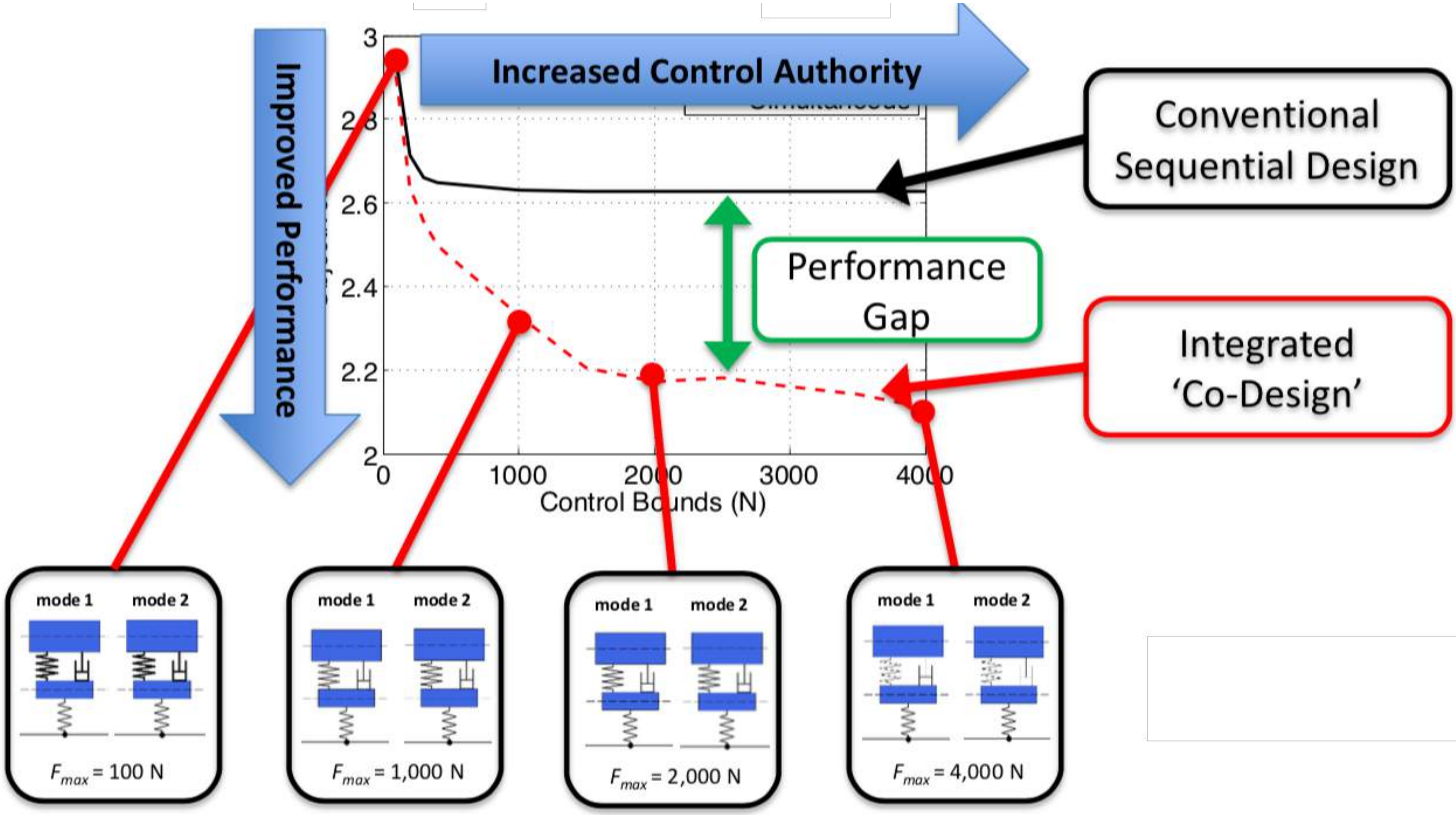


$\mathcal{A}$: Aerodynamic residuals
w: Aerodynamic states
$\mathcal{S}$: Structural residuals
u: Structural states

MDO + control law:

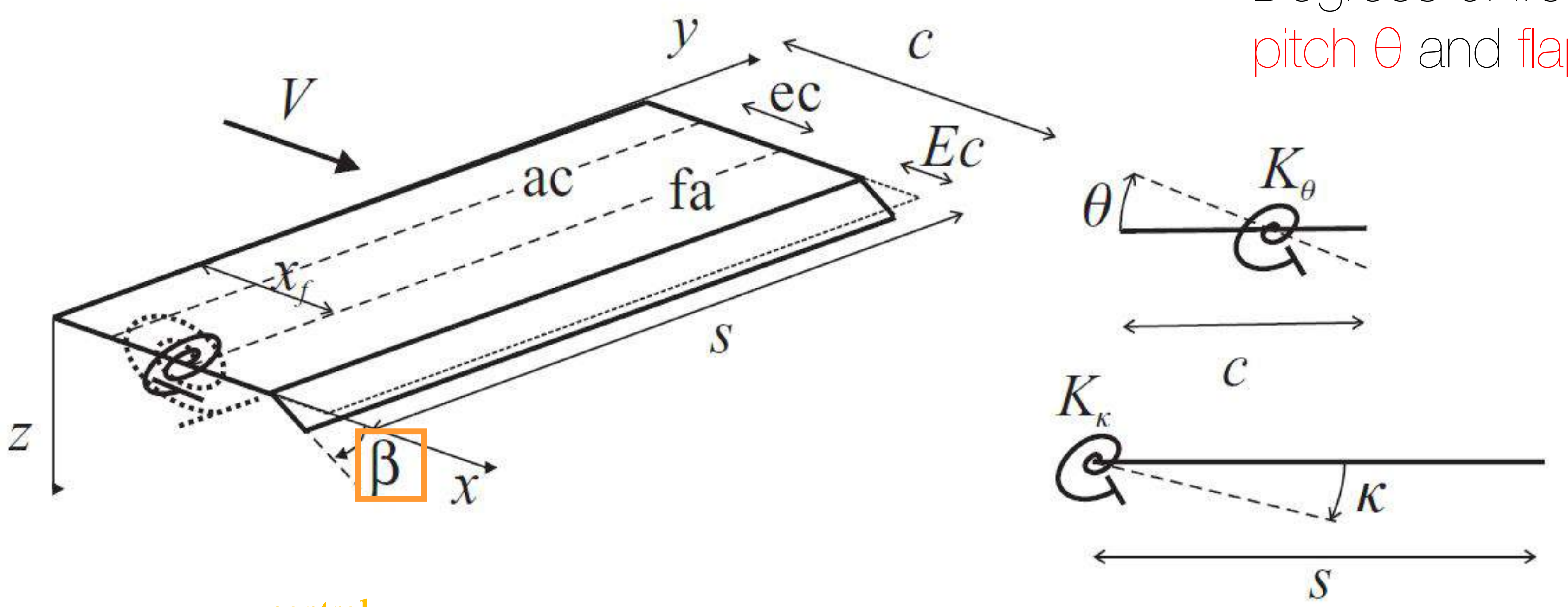


Co-Design: Integrated Physical and Control System Design *



Allison, J. T., Guo, T., & Han, Z. (2014). Co-design of an active suspension using simultaneous dynamic optimization. Journal of Mechanical Design, 136(8), 081003.
Deshmukh, A. P., & Allison, J. T. (2016). Multidisciplinary dynamic optimization of horizontal axis wind turbine design. Structural and Multidisciplinary Optimization, 53(1), 15-27.

A toy model* (G. Fillipi MsC)



Degrees of freedom:
pitch θ and flap κ

control
surface
angle

J. R. Wright, J. E. Cooper. Introduction to Aircraft Aeroelasticity and Loads. 2007.

(*monolithic)

State space modelling (*monolithic)

soved with [Direct Transcription Method](#)

$$\boxed{\begin{bmatrix} I_k & I_{k\vartheta} \\ I_{k\vartheta} & I_\vartheta \end{bmatrix}}\begin{Bmatrix}\ddot{k}\\ \ddot{\vartheta}\end{Bmatrix} + \rho V \boxed{\begin{bmatrix} \frac{cs^3 a_w}{6} & 0 \\ -\frac{c^2 s^2 e a_w}{4} & -\frac{c^3 s}{8} M_\vartheta \end{bmatrix}}\begin{Bmatrix}\dot{k}\\ \dot{\vartheta}\end{Bmatrix} + \left(\rho V^2 \boxed{\begin{bmatrix} 0 & \frac{cs^2 a_w}{4} \\ 0 & -\frac{c^2 s e a_w}{2} \end{bmatrix}} + \boxed{\begin{bmatrix} K_k & 0 \\ 0 & K_\vartheta \end{bmatrix}}\right)\begin{Bmatrix}K\\ \vartheta\end{Bmatrix} = \rho V^2 cs \begin{Bmatrix} -\frac{s a_c}{4} \\ \frac{c b_c}{2} \end{Bmatrix}\boxed{\beta} + \rho V cs \begin{Bmatrix} \frac{s}{4} \\ \frac{c}{2} \end{Bmatrix}\boxed{w_g}$$

structural inertia

aerodynamic damping

aerodynamic stiffness

structural stiffness

control surface angle

gust term

$$A\ddot{q} + \rho V B \dot{q} + (\rho V^2 C + E) q = g\beta + h w_g$$

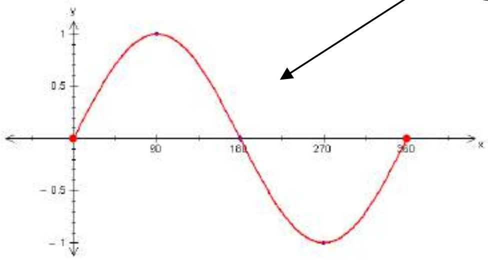
Objective function (multiObj → monoObj)

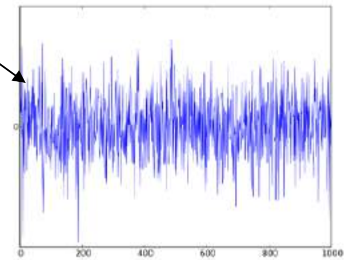
Normalized with r_i

handling + comfort + control cost

$$J = \int_0^{t_F} \left(r_1 z^2 + r_2 \ddot{z}^2 + r_3 u^2 \right) dt$$

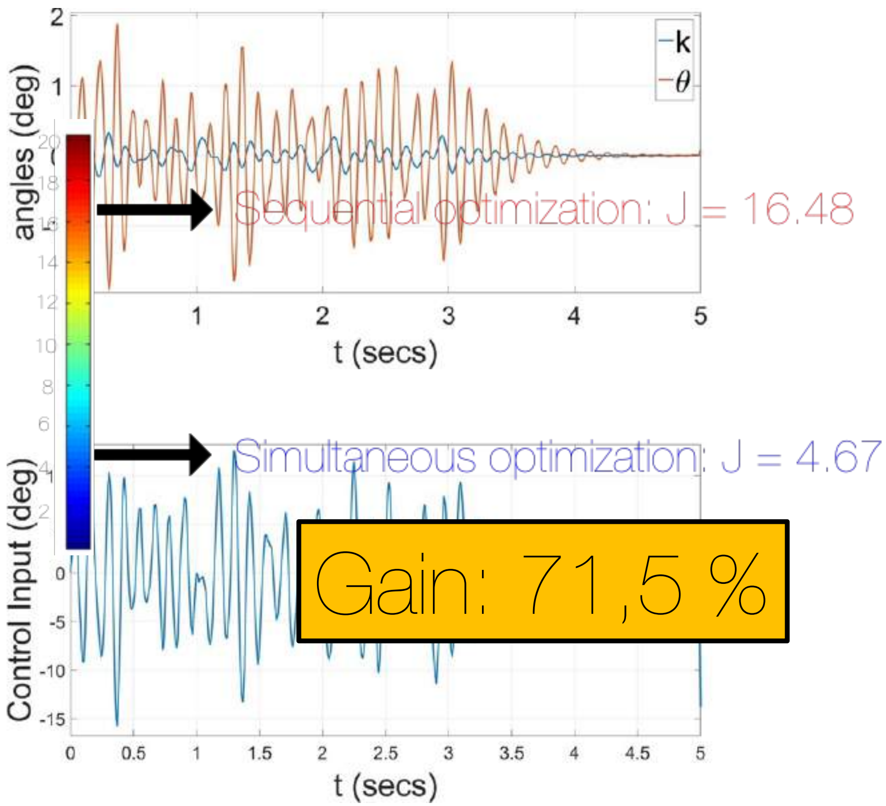
$$J_{\mathrm{tot}} = J_{gust} + J_{turb}$$





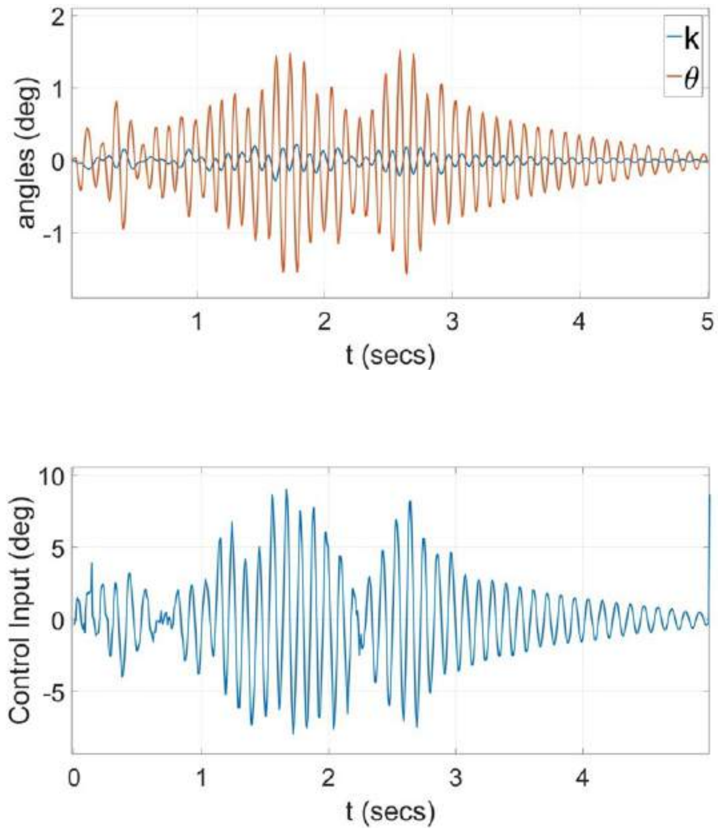
System response (Gust+Random)

Response to Random turbulence



sequential

Response to Random turbulence



vs simultaneous optimization

Conclusion

- In general, disciplines are not isolated in real world applications → coupled systems

- Optimizing each discipline separately can lead to underperforming results, as we are missing the interactions that will take place in the « real » operating conditions

- We can use the MultiDisciplinary Feasible approach to optimize the complete problem simply using openMDAO for example

- In the MDF, we solve the complete system for every set of variables proposed by the optimizer → One problem, One optimizer (to be tuned)

-Raw optimization : use SLSQP with mulltistart or Cobyla

ScipyOptimizer Options

Option	Default	Acceptable Values	Acceptable Types	Description
optimizer	SLSQP	['Nelder-Mead', 'Powell', 'CG', 'BFGS', 'Newton-CG', 'L-BFGS-B', 'TNC', 'COBYLA', 'SLSQP']	N/A	Name of optimizer to use
disp	True	N/A	N/A	Set to False to prevent printing of Scipy convergence messages
tol	1e-06	N/A	N/A	Tolerance for termination. For detailed control, use solver-specific options.
maxiter	200	N/A	N/A	Maximum number of iterations.

ScipyOptimizer Option Examples

optimizer

The "optimize" option lets you choose which optimizer to use. The ScipyOptimizer driver supports all of the optimizers in scipy.optimize except for 'dogleg' and 'trust-ncg'. Generally, the optimizers that you are most likely to use are "COBYLA" and "SLSQP", as these are the only ones that support constraints. Only SLSQP supports equality constraints, and SLSQP also uses gradients provide by OpenMDAO while COBYLA is gradient-free.

OpenSource tools

KADMOS => https://bitbucket.org/imcovangent/kadmos

CMDOWS => http://cmdows-repo.agile-project.eu

=> http://cmdows.agile-project.eu

RCE => http://rcenvironment.de/

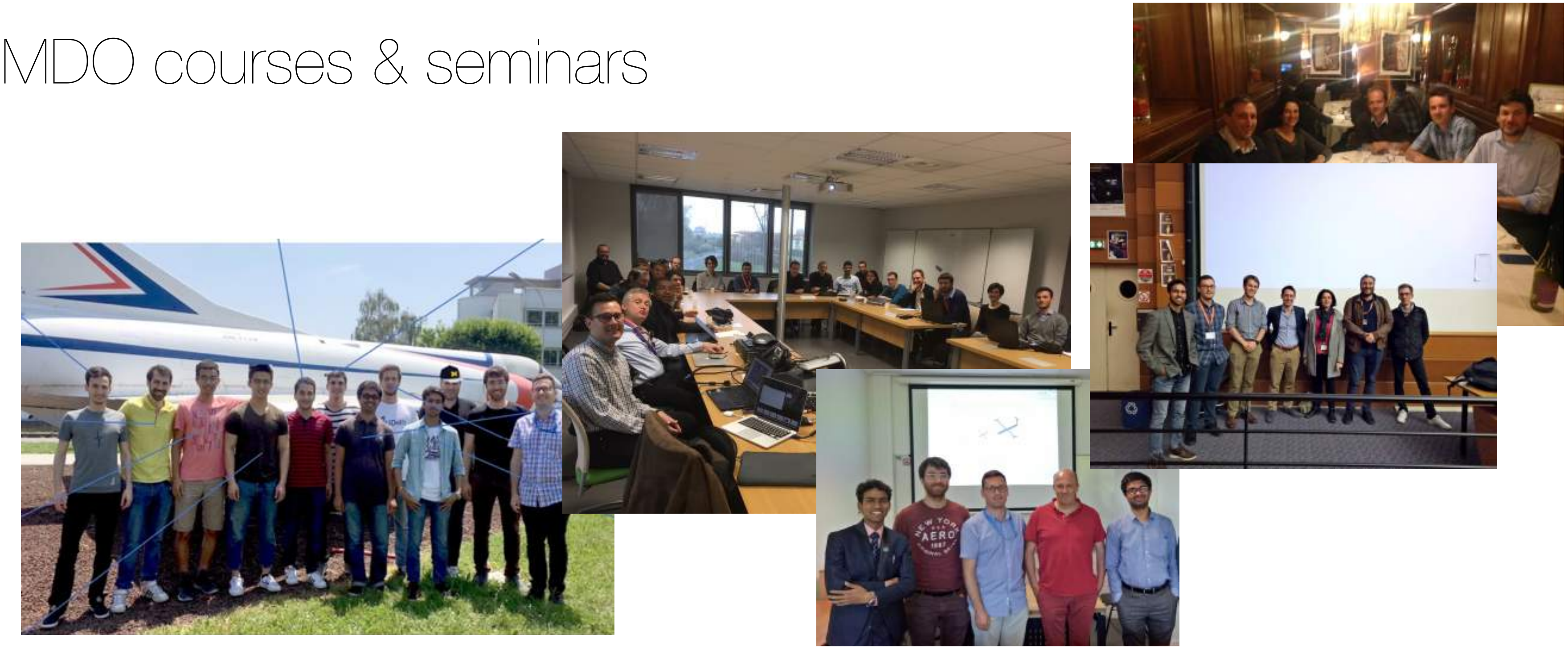
openMDAO => http://openmdao.org/

VISTOMS => https://www.agile-project.eu/files/VISTOMS_SellarProblem

=> https://www.agile-project.eu/files/VISTOMS_TUDWingDesign

http://www.agile-project.eu/

MDO courses & seminars



- NB: Since 2013 new course at SUPAERO : MDO [Structural&Multidisciplinary Design Optimization, 2*30H] (MsC level] with ONERA/AIRBUS. Since 2016 one MDO seminar per year (open to PhDs and researchers)

- Since 2017 we offer some fund to students to do research with us in order to be « PhD ready ». Part of this presentation has been made by SUPAERO MsC Students

→Mostafa Meliani (KTH), Mahfoud Herraz (ICA) already started a PhD

Please Visit :

https://github.com/SMTorg/SMT

https://github.com/mid2SUPAERO for student and research projects

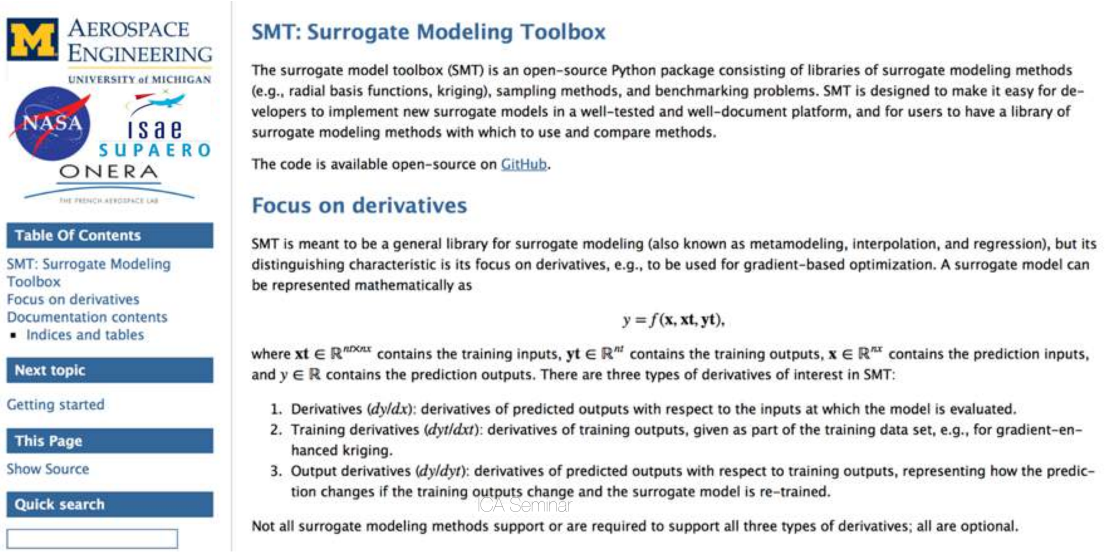
Thanks to My co-workers:

Joaquim Martins, Nathalie Bartoli, Thierry Lefevbre, Joan Mas Colomer

Emmanuel Benard, Claudia Bruni, John Hwang, Mohamed Bouhlel, Peter Schmolgruber, Youssef Diouane, Sylvain Dubreuil, Christian Gogu, Stephanie Lisy-Destrez, Miguel Charlotte and

PhDs: Pierre-Jean Barjhoux, Simone Coniglio, Alessandro Sgueglia, Laurent Beauregard

Surrogate modeling in HD, focus on derivatives



SMT: Surrogate Modeling Toolbox

The surrogate model toolbox (SMT) is an open-source Python package consisting of libraries of surrogate modeling methods (e.g., radial basis functions, kriging), sampling methods, and benchmarking problems. SMT is designed to make it easy for developers to implement new surrogate models in a well-tested and well-document platform, and for users to have a library of surrogate modeling methods with which to use and compare methods.

The code is available open-source on GitHub.

Focus on derivatives

SMT is meant to be a general library for surrogate modeling (also known as metamodeling, interpolation, and regression), but its distinguishing characteristic is its focus on derivatives, e.g., to be used for gradient-based optimization. A surrogate model can be represented mathematically as

$$y = f(\mathbf{x}, \mathbf{xt}, \mathbf{yt}),$$

where $\mathbf{xt} \in \mathbb{R}^{nt \times nx}$ contains the training inputs, $\mathbf{yt} \in \mathbb{R}^{nt}$ contains the training outputs, $\mathbf{x} \in \mathbb{R}^{nx}$ contains the prediction inputs, and $y \in \mathbb{R}$ contains the prediction outputs. There are three types of derivatives of interest in SMT:

1. Derivatives (dy/dx): derivatives of predicted outputs with respect to the inputs at which the model is evaluated.
2. Training derivatives (dyt/dxt): derivatives of training outputs, given as part of the training data set, e.g., for gradient-enhanced kriging.
3. Output derivatives (dy/dyt): derivatives of predicted outputs with respect to training outputs, representing how the prediction changes if the training outputs change and the surrogate model is re-trained.

Not all surrogate modeling methods support or are required to support all three types of derivatives; all are optional.

Thanks

Institut Supérieur de l'Aéronautique et de l'Espace

SUPAERO