

VDesign

A Formal Problem Optimization Software

Emmanuel Bignon¹ Jean Bignon² Eric Atienza³

¹École de Technologie Supérieure, Canada
emmanuel.bignon.1@ens.etsmtl.ca

²Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, F-38000 Grenoble, France
jean.bignon@grenoble-inp.fr

³Intel, Grenoble, France
eric.atienza@intel.com

May 9, 2018

- 1 An optimization software, why and for who?
- 2 Existing approaches
- 3 VDesign
- 4 Feature Overview

Actors in the preliminary design field

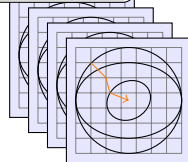
Industrial
Producer

Production process

Cost studies

Software
Engineer

Optimization
Expert



Physics Expert

$$m_n = \frac{u}{\chi v} (x_1 - x_0 - x_3 - x_5)(2x_2 + x_4 + 3x_5)$$

$$\Gamma_m = \frac{1}{\rho g} \Pi - v_v^2 * \sqrt{2\pi\sigma}$$

Characteristics of the models

Model types

Motor, Turbine, Electrical devices. . .

The easy part

- Continuous models
- Analytical (derivatives available)
- Small decision space

The no so easy part

- Non Linear
- Internal analytical problem resolution parameterized by the decision variables
 - Derivative less trivial

Real life models

Falcon 2000 (Dassault Aviation)

- 1000 equations
 - Analytic Model
- Gives the wing span, plane size, autonomy

Electrical Motor (Somfy)

- 600 equations
 - Analytic Model
- Gives the power output, weight, heating. . .

- 1 An optimization software, why and for who?
- 2 Existing approaches
- 3 VDesign
- 4 Feature Overview

The Standalone Software

Optimization Software

- Provides a formalism for the description of the model
- Solve the optimization

Product Designer

- Fit the problem into the software formalism

The Standalone Software

Method mastery

The software, oriented for the method allow fine tuning of the parameters for optimal use of the method.

Comparing methods

- Methods are adapted to specific problems (linear, nonlinear, constrained, unconstrained, ...)
- The user is not a specialist of those considerations
- He will test out several methods

Product designer increased workload

The problem has to be described in the formalism of each software

The Aggregating Software

Optimization Expert

- Designs an optimization method
- Implements the method following the Software API

Software

- Provide a unified problem description formalism
- Provide an API for optimization solver
- Drive the optimization process

Product Designer

- Fit the problem into the software formalism

The Aggregating Software

Optimization Expert

- Designs an optimization method
- Implements the method following the Software API

Software

- Provide a unified problem description formalism
- Provide an API for optimization solver
- Drive the optimization process

Product Designer

- Describe the physical model
- Describe the industrial constraints

The Aggregating Software

Product designer reduced workload

- One language to learn for all methods
- Easy comparison of methods

Product design update

- Black box
 - Update the function and re-compile it
 - Documentation should be maintained
- Domain Specific Language
 - Easier to read, accessibility without expertise.
 - The more advanced the language capabilities, the hardest to understand.

Preliminary Design Concerns

Confusion of concerns

In preliminary design, the optimization problem definition is divided in two parts:

- 1 Physical model definition (little to no variation)
- 2 Physico-economic constraints (possibly frequent and major variations)

The Preliminary Design, Role-Oriented Software

Physics Expert

Describe the physical model.

Product Sizing Engineer

Imposes physico-economic constraints

Software

Provide formalism and API to the different actors

Optimization Expert

- Designs an optimization method
- Implements the method following the Software API

- 1 An optimization software, why and for who?
- 2 Existing approaches
- 3 VDesign
- 4 Feature Overview

VDesign

A suite of tools

- Analytical Editor
- Sizing Table editor
- Optimization Trace
- Computation Server
- Optimization Method API

Capabilities

- Provide a formalism for the Physics Expert
- Provide an environment for the Product Sizing Engineer
- Exposes an API for the Optimization Expert

Modeling

Predim - C:\Users\Eliane\Workspaces\Wkspace-predim-Bobine

Fichier Aide

- Coil
 - Modèles
 - AirCoil
 - Optimisations

AirCoil

```

1 // Air coil
2
3 /* L : self inductance ( H ) */
4 L = (50.8E-9*a^2*n^2) / (3*a+9*b+10*c);
5 /* a : diameter of the coil; b : length (m); c : thickness (m); n : turn number */
6
7 /* Pj : Joules losses ( W ) */
8 Pj = rhocu*delta^2*Math.PI*a*b*c;
9 /* delta : current density (A.m²); rhocu : copper resistivity (ohm.m) */
10
11 /* I : intensity(A) */
12 I = delta*b*c/n;
13
14 /* Wmag : magnet energy ( J ) */
15 Wmag = 1/2 * L*I^2;
16
17 /* M : mass ( Kg ) */
18 M = mvcu*Math.PI*a*b*c;
19 /* mvcu : copper density ( Kg/m^3 ) */
20

```

1:0

Effacer

Stop Details

Modeling

Natural mathematical language formal grammar

- Description of the model through equations

$$\| y = a * \cos(x) + b * \sin(x);$$

- No programming language artifacts (var, param, func...)
- Explicit typing (real, matrix, complex) is optional
- Definition order not imposed
- Allows interpretation of the model
 - Computation of exact derivatives
- Code generation

Sizing

Predim - C:\Users\Eliane\Workspaces\Wkspce-predim-Bobine

Fichier Aide

Coil

- Modèles
 - AirCoil
- Optimisations
 - AirCoil-optim

AirCoil-optim

Propriétés générales

Projet : Coil
Modèle : AirCoil

Objective

- Bound constraint
- Value equality constraint
- Interval constraint

- Thickness: $0.005 < c < 0.1$
- Mass: $20 < M < 100$
- DiameterCoil: $0.1 < a < 0.5$
- Length: $0.1 < b < 1$
- rhocu: $\text{rhocu} = 1.724\text{E-}8$
- Density: $0.1\text{E}6 < \text{delta} < 10\text{E}6$
- TurnCoil: $10 < n < 5000$
- JoulesLoss: $70 < P_j < 90$
- LInductance: $L = 1\text{E-}3$
- Mass: M minimize

Nom	Valeur
a	3.0E-1
b	5.0E-1
c	1.0E-2
Wmag	Undefined
delta	1.0E6
mcu	8.8E3
l	Undefined
Pj	Undefined
rhocu	0
L	1.0E-3
M	Undefined
n	1.0E3

Effacer

Stop Details

Sizing

What we have

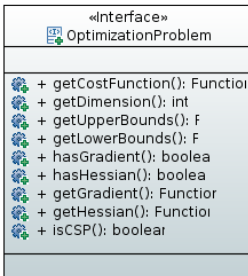
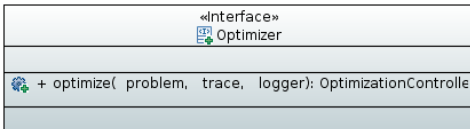
Model with named inputs and outputs.

Optimization Problem Definition Environment

- Meta-language (XML) for a standard description.
- Adapted edition tools
- Environment providing several optimization methods
 - Gradient based
 - Evolutionary (PSO, DE)
 - To be contributed

Optimization framework

$$\begin{aligned} & \min_{x \in \Delta} && f(x) \\ & \text{with} && c_E(x) = 0 \\ & \text{and} && c_I(x) \leq 0 \end{aligned}$$



Optimization framework

API

- Allows addition of methods with minimal workload
- One class (with one method) to implement: `OptimizationProblem`
- Requires comprehension of only 2 to 3 classes:

`OptimizationProblem` Accessors to the problem definition elements

`OptimizationController` Handle for the optimization convergence
result (success, point, value of objective)

`AlgorithmLogger` Trace of the process

Interoperability

Computation Server

- Originated from a request to run the optimization from an excel sheet.
- Allows a sequence of computation without stopping the environment (gain of bootstrap time)
- Socket-based, with textual communication language.
 - Network access
- Currently sequential version.

- 1 An optimization software, why and for who?
- 2 Existing approaches
- 3 VDesign
- 4 Feature Overview

Features

Modeling

- \cos , \sin , \tan , \cos^{-1} , \sin^{-1} , \tan^{-1} , atan2 , \cosh , \sinh , \tanh , \cosh^{-1} , \sinh^{-1} , \tanh^{-1}
- bessel_i^n , bessel_j^n , bessel_k^n , bessel_y^n
- abs , exp , log , pow , sqrt
- ceil , floor , round
- heaviside , if , min , max
- arg , mod , imag , real
- transpose , ld
- Complexes, Matrices
- integration (*integral*)
- ordinary differential equations (*ode*)
- implicit system (*implicit*)
- interpolation (*interp*)
- function optimization (*fmin*)
- find optimal point (*findMinimum*)
- indexed sum and product (*sum*, *prod*)
- black-box execution (*exec*)

Features

Sizing

- All type of constraints (lower bound, upper bound, interval, equality)
- Maximization and minimization.
- Selection of decision variables
- Definition of indicative initial point

Optimization

- Automatic differentiation of the model
- Implementations of PSO, Cobyla and VF13 optimization methods
- Logging and Reporting
- Export of optimized model, constraint definition and process trace.

Features

Symbolic Computation

API for computation of other arithmetics (i.e. interval)

Questions?