From linear to nonlinear black-box models : some more accurate models based on neural networks

A neural network metamodel of fan blades

patrice.kiener@inmodelia.com - manuel.henner@valeo.com
Tel : +33.9.53.45.07.38

NAFEMS France 1st congress – Paris – October 30th, 2010

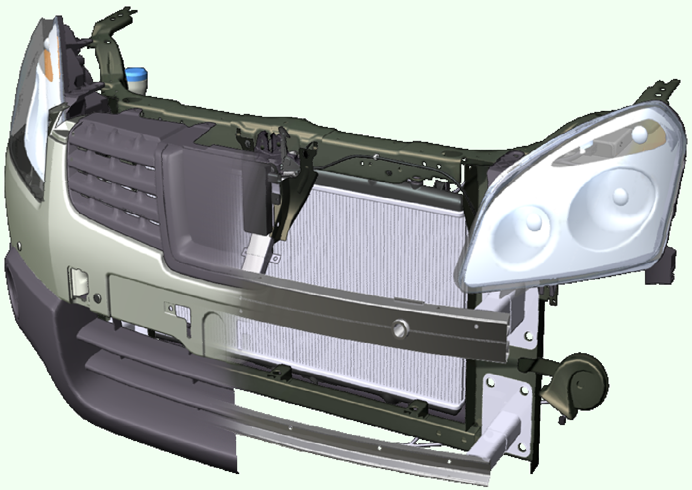
Industrial partners

INMODELIA

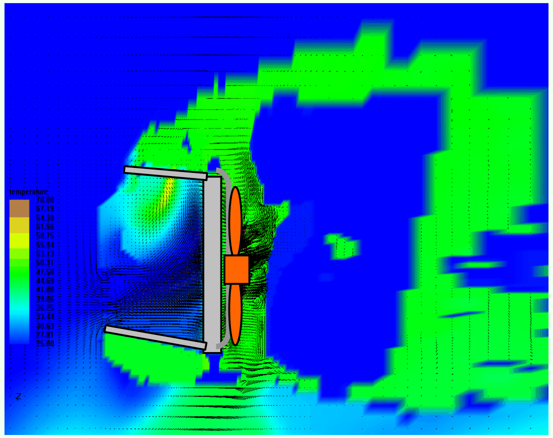
- **Editor and publisher of software dedicated to neural networks and design of experiments for nonlinear models and neural networks**
- **Consulting services in “black box” modelling**
- **Training courses in private companies and at the university (ESPCI)**

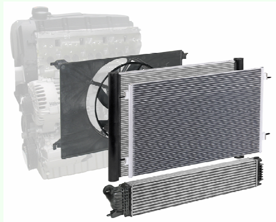
VALEO THERMIQUE MOTEUR

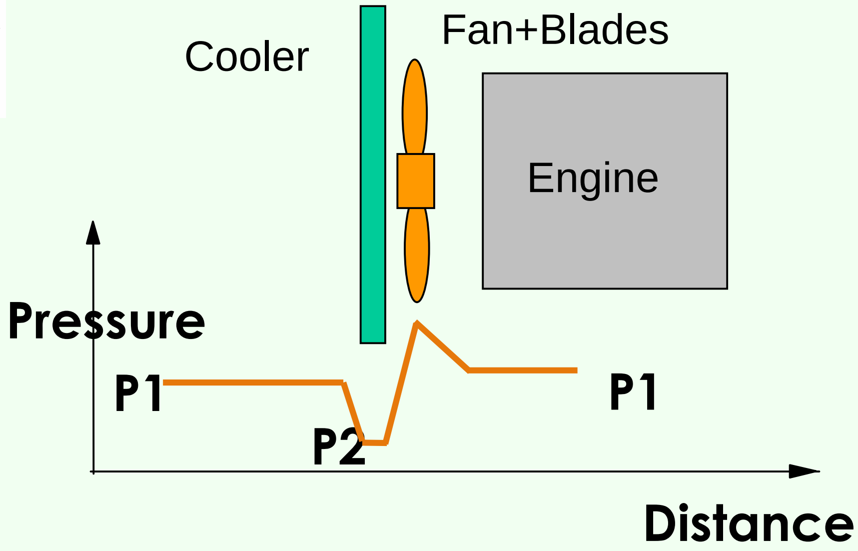
- **Automotive supplier of cooling modules**
  - **Front faces**
  - **Fans**
  - **Thermal exchangers**
- **System integrator with strong R&D**



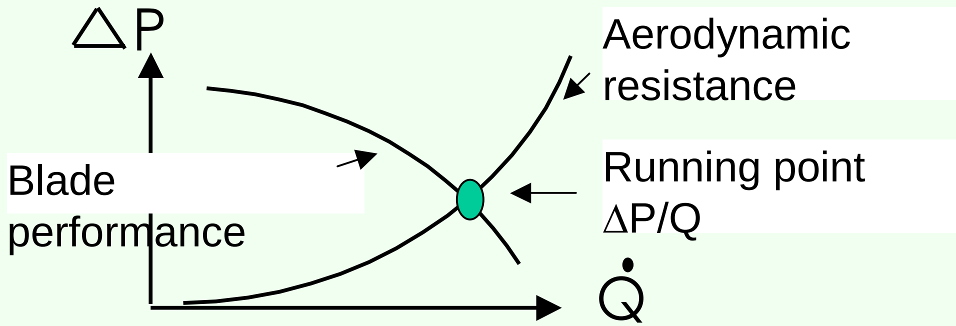
Objectives :
Modelling the fan function





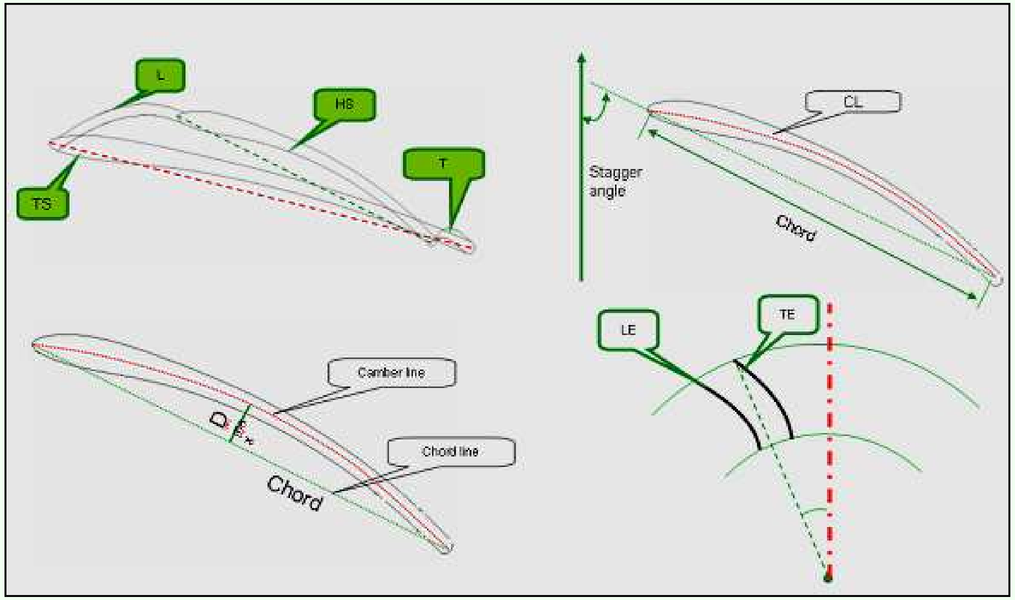


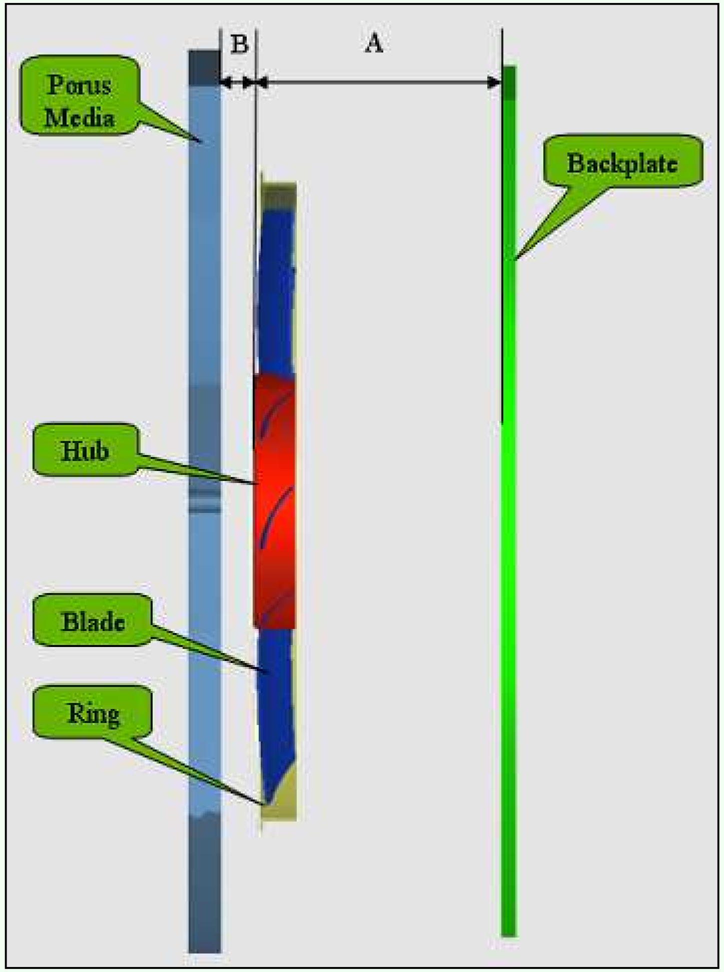
- **Nonlinear interconnected aerodynamic phenomenas**
- **Several geometric and physical parameters**



From 1 to 25 useful parameters

« Blade » parameters





« Module » parameters (distances)

Physical parameters
(air flow, blade rotation speed, …)

Global approach : metamodels

METAMODELS (Surrogate model) = Models of computer code

- **"Black box", 100 - 10.000 faster than original code**
  - **2nd order polynoms, if linear phenomena**
  - **Static nonlinear neural networks, if nonlinear phenomena**
  - **Other ? (PLS, curvilines components, kriging, Gaussian processes)**

ITERATIVE EXPERIMENTAL DESIGN for expensive experiments

- **Initial design is independent from the assumed model** ⟵ **Original**
- **Modelling**
  - **2nd order polynoms**
  - **Static neural networks**
- **Iterative approach : 2nd design**
  - **D-optimal design for neural networks** ⟵ **New**
- **Modelling**

The iterative approach is the best one when experiments are expensive

The curse of the dimension

There is no perfect initial design. Some designs are :

- **Unapplicable with models of large dimension**
- **Independent from the postulated model**
- **Or fully dependant from the postulated model**
- **Inadequate for nonlinear models**
- **etc…**

Number of inputs	NOLH design	Factorial design	D-optimal design poly2	NN-2hn	NN-3hn	NN-4hn
5	17	32	21	15	22	29
8	33	256	45	21	31	41
11	33	2048	78	27	40	53
16	65	65536	153	37	55	73
22	129	4194304	276	49	73	97
29	257	536870912	465	63	94	125

Model size (number of coefficients)

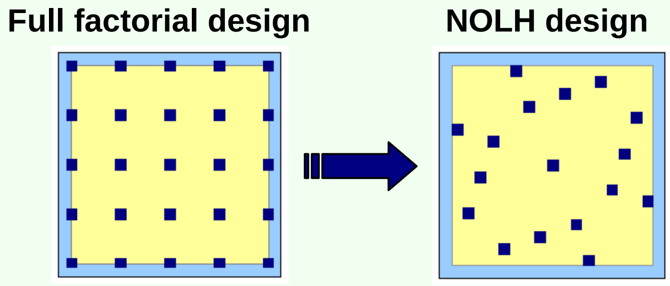
With large dimensions, a combined approach is required for the initial design

- **NOLH design from Cioppa (2002, 2007) said to be economical** new !
  combined to D-optimal design for polynoms
  or, an exception in low dimension, to the factorial design
  (middle of the faces, a fews corners or all corners)

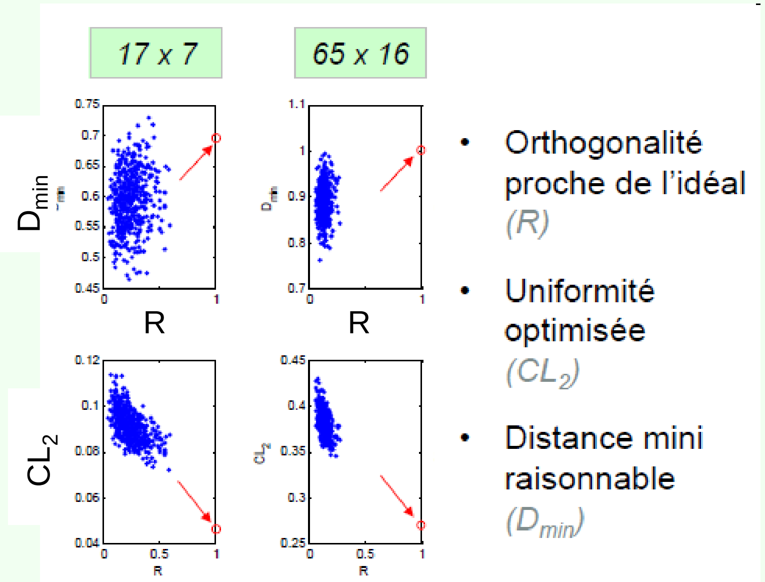
NOLH Designs (T. Cioppa, 2002 and 2007)

NOLH : NEARLY-ORTHOGONAL LATIN HYPERCUBE DESIGNS

	number of trials	*max. nb. of inputs*
- NOLH 17 x 7	17	7
- NOLH 33 x 11	33	11
- NOLH 65 x 16	65	16
- NOLH 129 x 22	129	22
- NOLH 257 x 29	257	29



SPACE FILLING PROPERTIES



NOLH designs are economical and have excellent space filling properties.

From all available latin hypercube designs, they are the best designs.

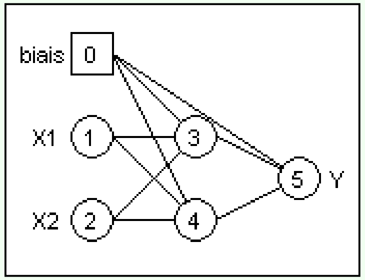
But, they are not sufficient … (cf. infra)

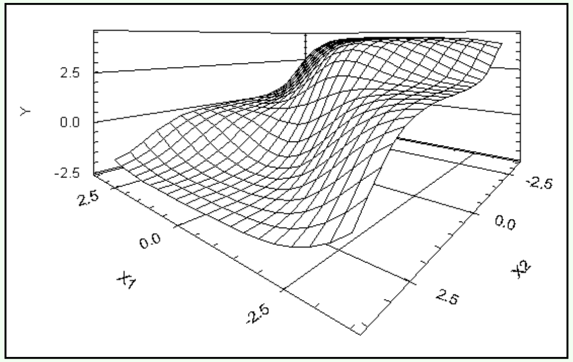
(Jean Leung, Renault – A study of the properties of LHS and NOLH designs – Expérimentique 2007)

Static neural networks

EXAMPLES of neural network with 2 inputs (allow a few graphs)

- 2 hidden neurons :
(9 coefficients)

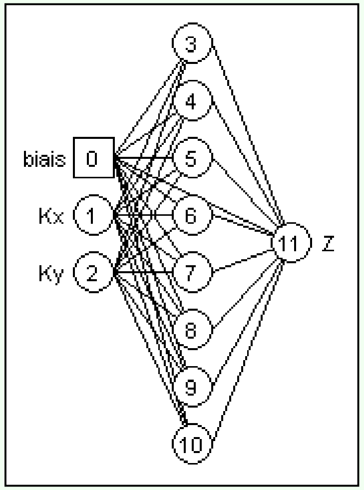


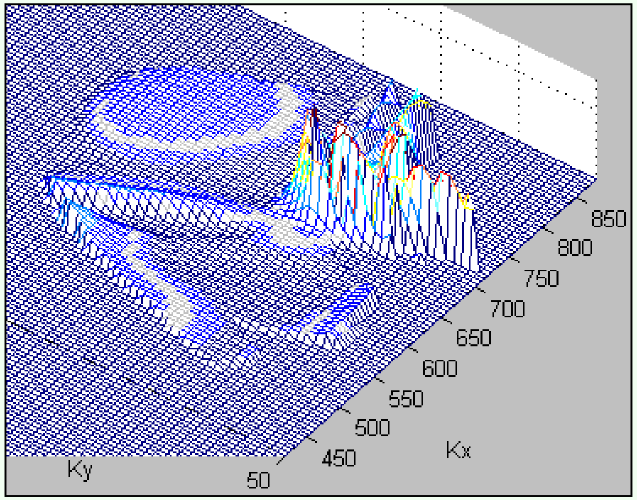


$$Y = \theta_1 + \theta_2 th(\theta_4 + \theta_5 X_1 + \theta_6 X_2) + \theta_3 th(\theta_7 + \theta_8 X_1 + \theta_9 X_2)$$

- 8 hidden neurons :
(33 coefficients)
overtraining ?

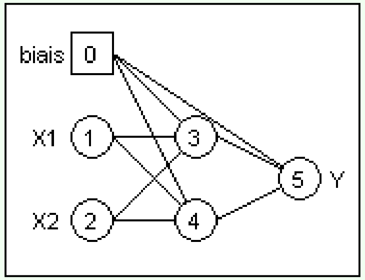
$$Z = \sum_{i=0}^{8} b_i \left(th \sum_{j=0}^{2} a_{ij} X_j \right)$$





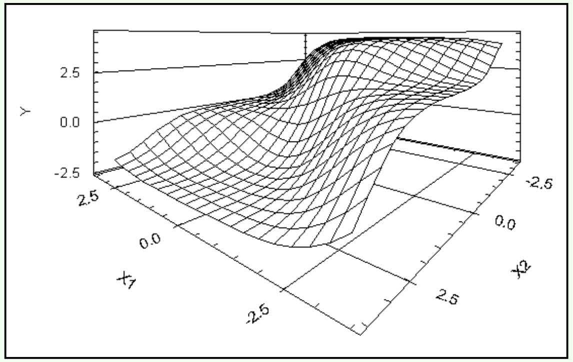
One can use NN models with 10-25 inputs if the inputs are independant.
One will use as many NN as there are outputs (allows some reverse calculation)

Optimal design of experiments for neural networks



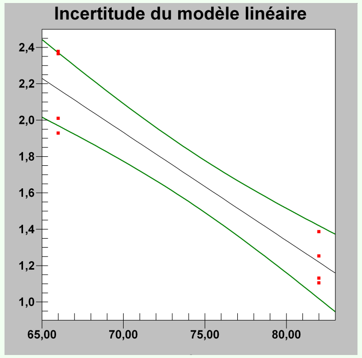
They are a PARTICULAR CASE of "design of experiments for nonlinear models"

NN response surface

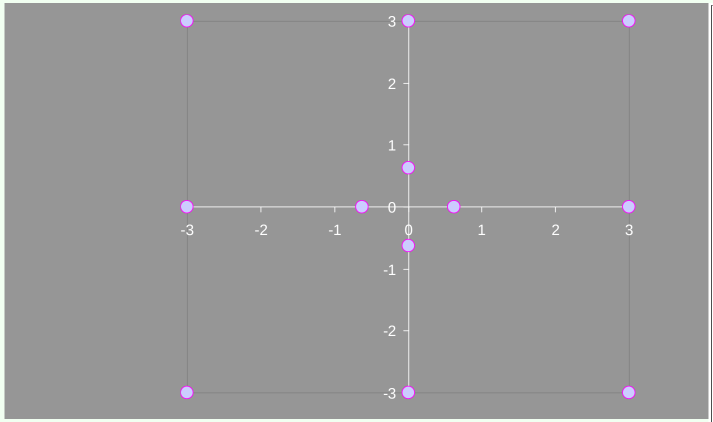


- Software NEURO PEX
- The optimal points are located where the model is the most uncertain
- D-optimality, G-optimality
- The points are at the peripheral and in the area with large inflexions (on the uncertainties)

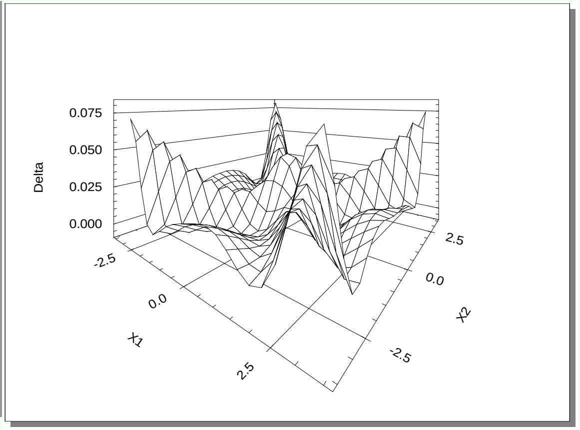
Remind : D-optimal points and uncertainties of the linear model



NN D- and G-optimal points



NN uncertainty surface

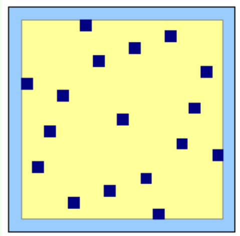


Experimental strategies

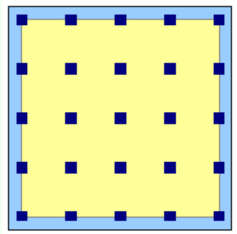
Test of several strategies on a model with 5 parameters

- **PEX-1 with 5 parameters**
  - **1 exchanger parameter**
  - **1 distance for the module**
  - **1 foot-blade parameter**
  - **1 head-blade parameter**
  - **air flow**

NOLH design
Training Poly2/NN
Test on factorial design



Neuro Pex :
additional points
Training NN
Test



Etc…

Use the best strategy on larger models :

- **PEX-2 with 11 parameters**
  - **The 5 above parameters**
  - **5 additional blade parameters**
  - **1 exchanger parameter**

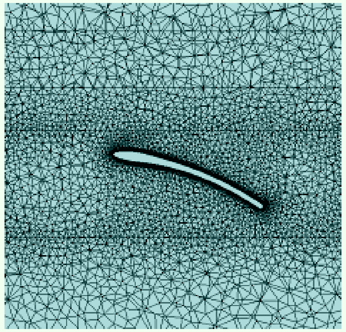
- **PEX-3 with 16 parameters**
  - **The previous 11 parameters**
  - **4 module parameters**
  - **1 physical parameter**

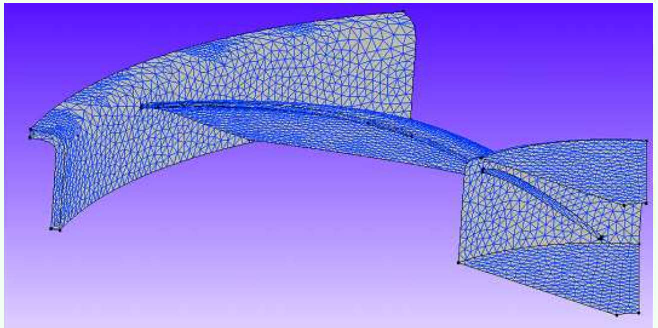
Ponctual computation : numerical simulation

1st step :

MESH BLADE - MODULE

- **Parameterised geometry**
- **Automatic mesh**



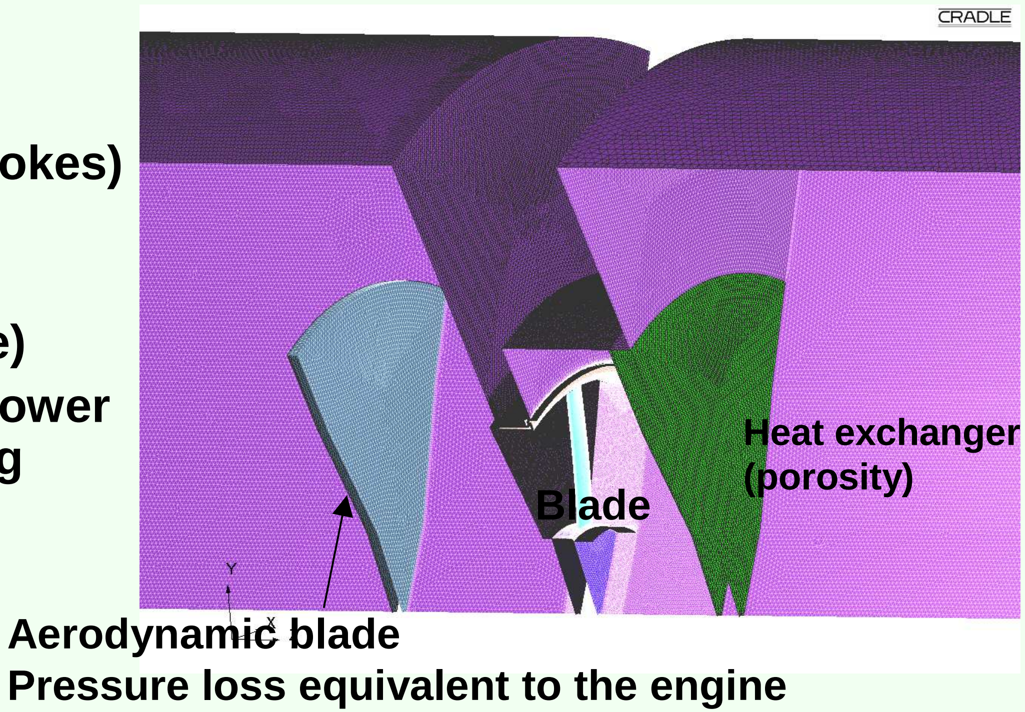


2nd step :

FOR 1 GIVEN CONFIGURATION

- **RANS Simulation (Navier-Stokes)**
- **Code CFD SC-Tetra**
- **~ 100 hours CPU per run**
  (~ 6 hours with parallel code)
- **Cost estimated to 50 times lower than manufacturing + testing a real blade**

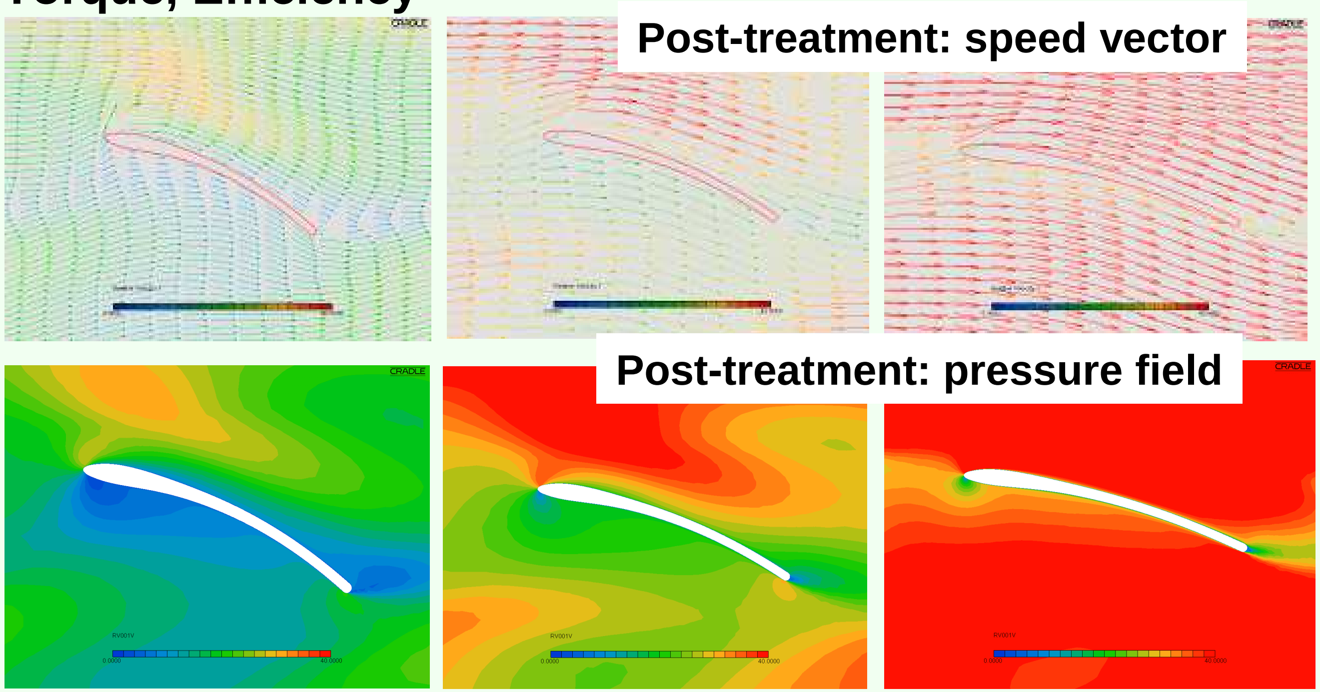
Simulation space



Outputs issued from numerical simulation

AERODYNAMIC FIELDS FOR 1 GIVEN CONFIGURATION

- Air flow, Aerodynamic
- Pressure, Torque, Efficiency



3rd step :

The outputs produced by numerical simulation are then introduced into :

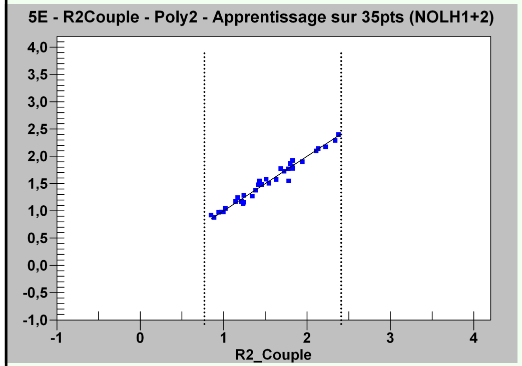
- 2^{nd} order polynoms
- Neural network with 2 or 3 hidden neurons

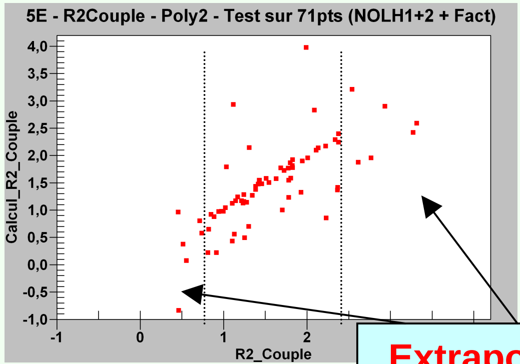
NOLH designs are not sufficient

NOLH designs are said to be good designs for estimating a polynom.
In practice, it is not fully true (a necessary but not sufficient condition)

Torque - 5 inputs – 2nd order polynom

Training NOLH1+2 (33pts) **Test 71pts**

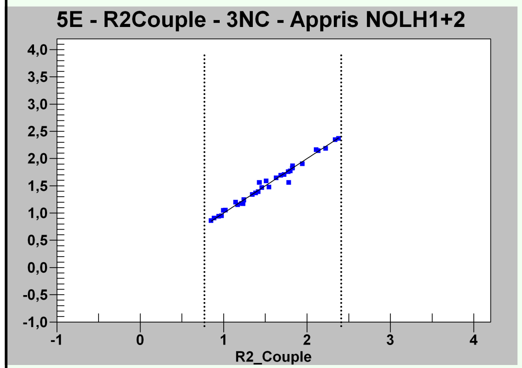


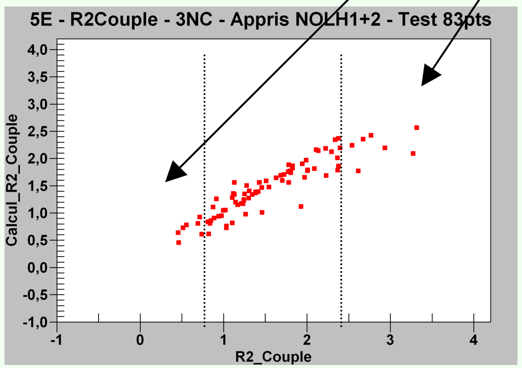


Extrapolated points

Torque – 5 inputs - NN with 3 hidden neurons

Training NOLH1+2 (33pts) **Test 83pts**



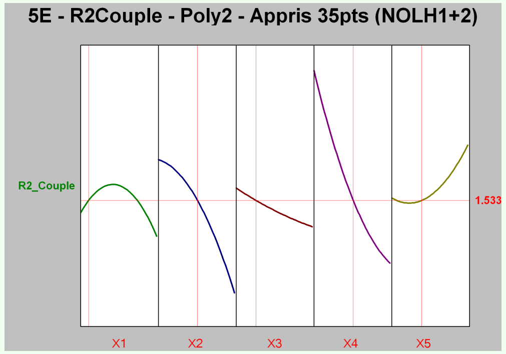


The models trained exclusively with the NOLH designs poorly extrapolate :

- **Average distance from the center of the cube to the points of the NOLH design :**
  D_{moy} = ~ 0,63
- **Distance from the centre of the cube to the corners of the cube :**
  D = 1

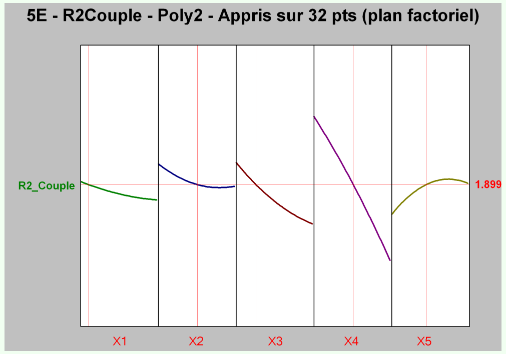
NOLH designs are excellent to fill the experimental space but not sufficient to cover the whole experimental space

And can create model errors

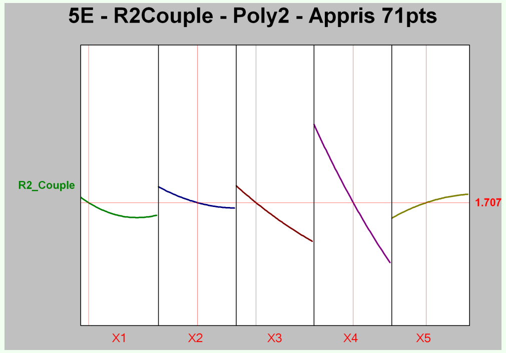


Example : Model trained on the NOLH design
(+ complementary part) = 33 points (+ 2 repeat)

=> 3 out of 5 inputs have wrong first partial derivatives



Example : Model trained on the 2-levels factorial design
(32 points)



Example : Model trained on the NOLH + factorial design
(71 points (33+2+32)), The most “true” model

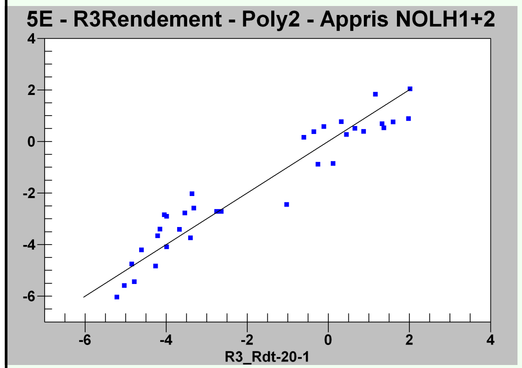
NOLH designs must be completed by points located at the domain peripheral (corners, faces, lines)
=> Suggestion : the middle of the faces

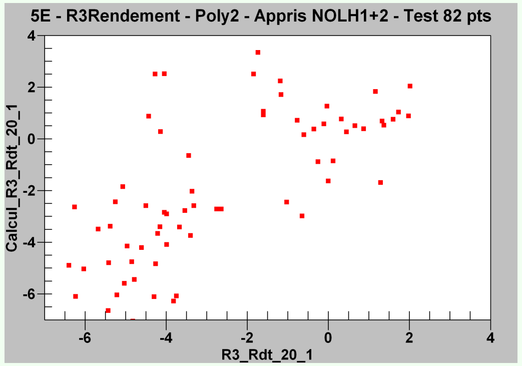
Neural model of the efficiency (1)

The efficiency is very poorly described by a polynom and very well described by a neural network.

Efficiency - 5 inputs – 2nd order polynom

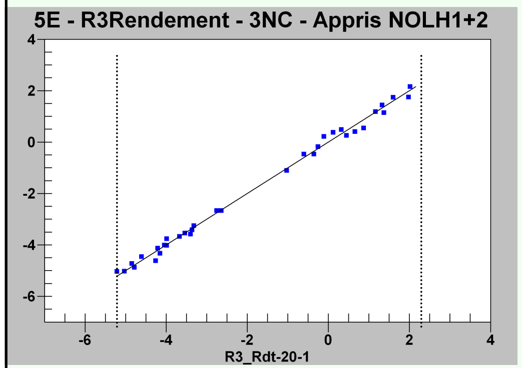
Training NOLH1+2 (33pts) | Test 82pts

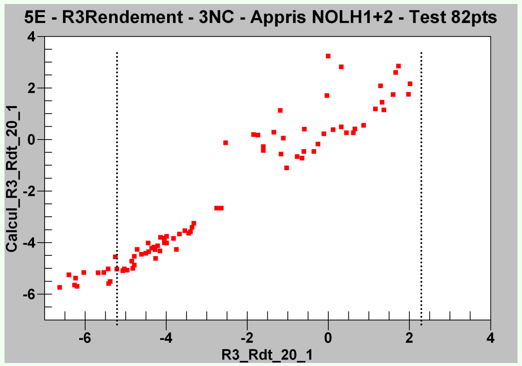




Efficiency - 5 inputs - NN with 3 hidden neurons

Training NOLH1+2 (33pts) | Test 82pts





Experimental strategy :

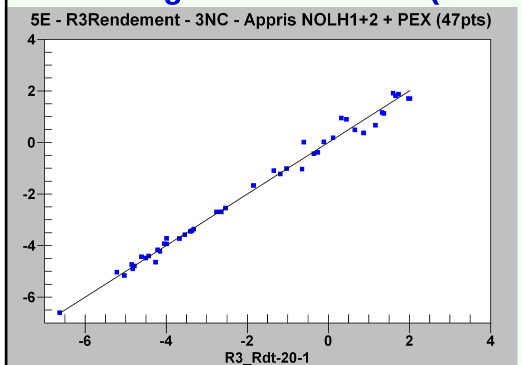
- **Training on NOLH 1+2 design (NOLH-2 complement to NOLH-&)**
- **Polynomial model (very bad)**
- **Neural network with 3 hidden neuron**
- **Calculation (NeuroPex) of 12 D-optimal points to be added**
- **Test of the neural network on those new D-optimal points**
  => Prediction is not perfect, but…
  => NeuroPex saw the critical areas

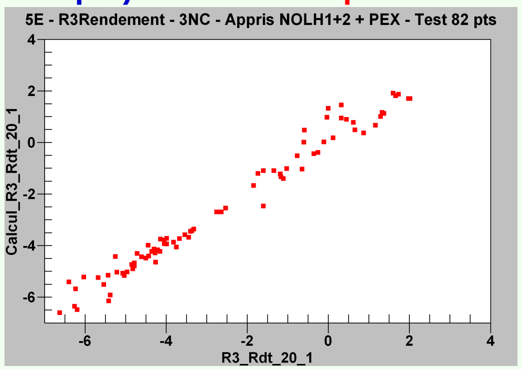
Neural model of the efficiency (2)

The neural network returns good results after the second iteration.

Efficiency - 5 inputs - NN with 3 hidden neurons – Step 2

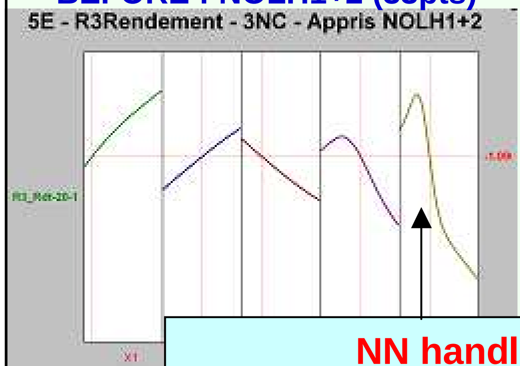
Training NOLH1+2+PEX (33+2+12=47pts) **Test 82pts**

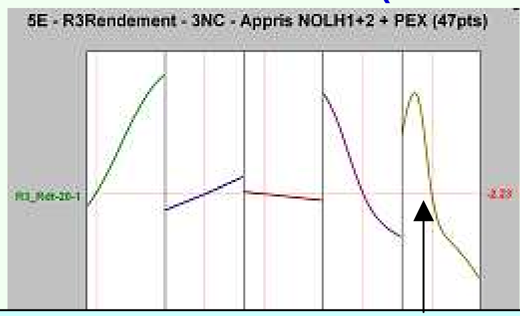




Efficiency – Partial derivatives of NN with 3 hidden neurons

BEFORE : NOLH1+2 (33pts) **AFTER : NOLH1+2+PEX (33+2+12=47pts)**

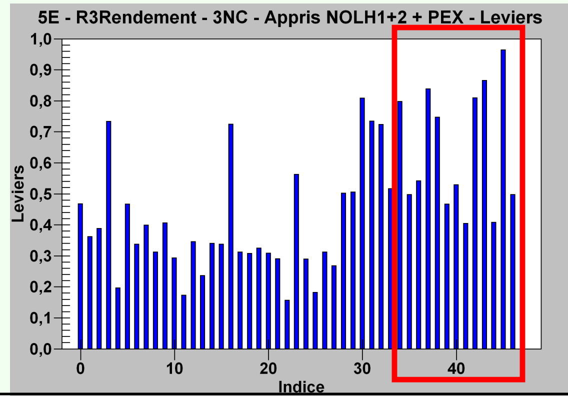




NN handles concav + convex curves

Experimental strategy :

- **Training on design made of NOLH 1+2 + PEX 12pts => 47points**
- **Test of the NN with 3 hn – Step 2 on all points, including the factorial design + a few other points**
- **NeuroPex saw the critical areas => 12 points with strong leverages**



Conclusion (1)

- Some complex aerodynamic phenomena, linked to the engine cooling, have been studied through numerical simulation and neural networks, with the help of new designs of experiments called NOLH.

- A good experimental strategy is primordial to minimize the number of trials. The choice of the initial design is a key element if one wish to study simultaneously polynoms and neural networks, which are the natural choice for large dimensions (Ex : 16 inputs : 55 coeff. vs 153 coeff.)

- The new NOLH designs and some points at the peripheral of the experimental domain among the corners and the middle of the faces are a good initial design.

- The additional points proposed by Neuro Pex help to return robust models and justify the iterative strategy.

- The resulting neural networks of the simulated physical outputs (pressure drop, blade torque, efficiency) have confidence intervals lower than 5%, validated with additional runs.

Conclusion (2)

- A first approach on reduced model with 5 input parameters has helped us validate the various options of our iterative strategy.

- The design with 11 parameters returns some equivalent results, although the validation is more difficult, due to to the cost of the validation. We are actively preparing a model with 16 parameters.

- The obtained metamodels can be used :
  - for fast evaluation during pre-project phases
  - for optimization purpose
  - for linking the code to other applications (modelling at a larger scale)
  - for embedded code (thanks to the small size of the model)

The most recent advances in numerical simulation,
design of experiments and neural networks
offer some new industrial applications and objectives.

Thank you for your attention

patrice.kiener@inmodelia.com - manuel.henner@valeo.com
Tel : +33.9.53.45.07.38