A shallow dive into Hexaly Black-box Optimization Solver

Emeline Tenaud

etenaud@hexaly.com

JFRO 2024

Hexaly

A quick introduction

Hexaly

The company

Software company specializing in Mathematical Optimization, Operations Research, and Decision Science

- **Powerful optimization solver & platform** used by Amazon, FedEx, Starbucks, …
- **Turnkey custom optimization and planning applications** for Air Liquide, Toyota, …

20 years of experience

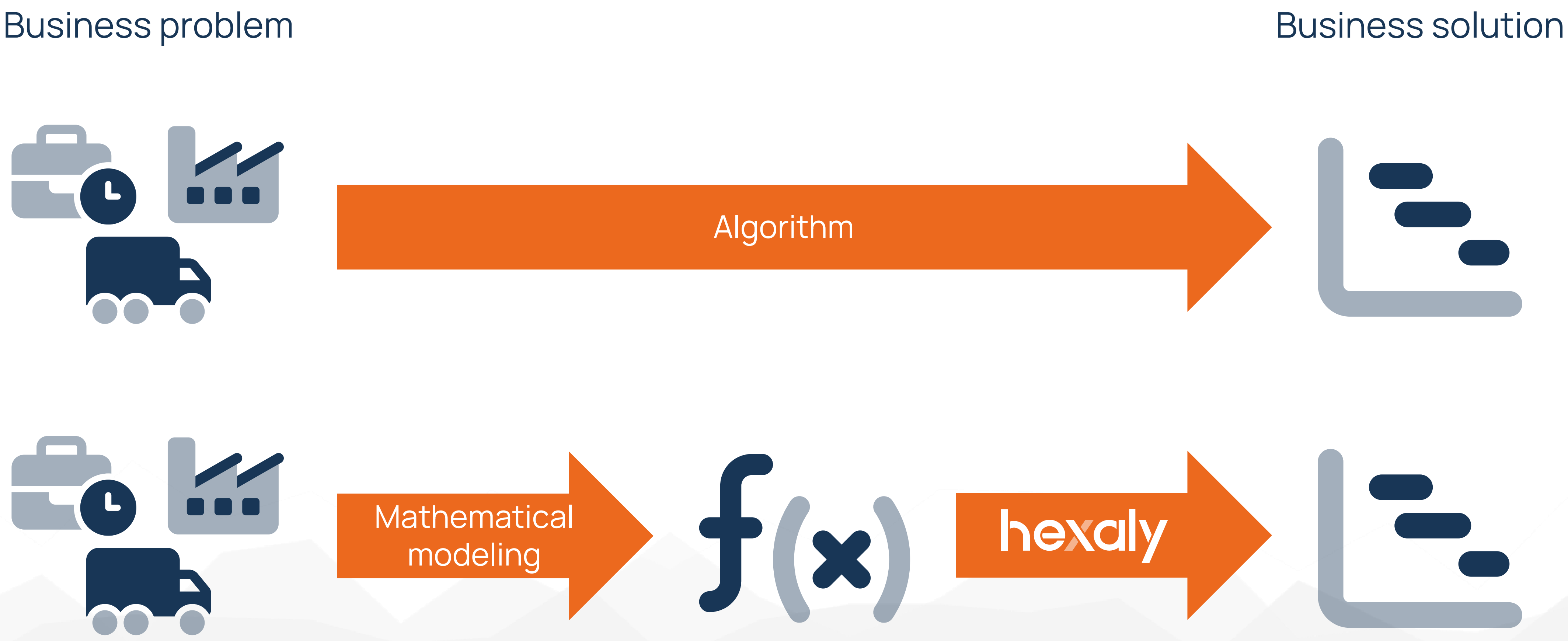
25 R&D engineers

200 clients,
400 applications,
20,000 users in 25 countries

Offices in Brooklyn, NY, and Paris, France

Hexaly Optimizer

A generic optimization solver



Hexaly Optimizer

Architecture

Hexaly Model

- Python
- Java
- C#
- C++
- Hexaly Modeler

Solutions ⇅ Bounds

Hexaly Optimizer

Proprietary algorithms

- A myriad of **primal and dual methods** hybridized into a model-and-run, global solver
- Computes **solutions and bounds**, provides optimality or inconsistency proofs

Black-box Optimization Solver
› Surrogate modeling

Black-box Optimization
Optimize costly functions

Black-box functions

Hexaly Optimizer comprises a **black-box solver**, specialized for **expensive function evaluations**

Many use cases:

Design optimization

Engines

Fluid dynamics

Machine learning

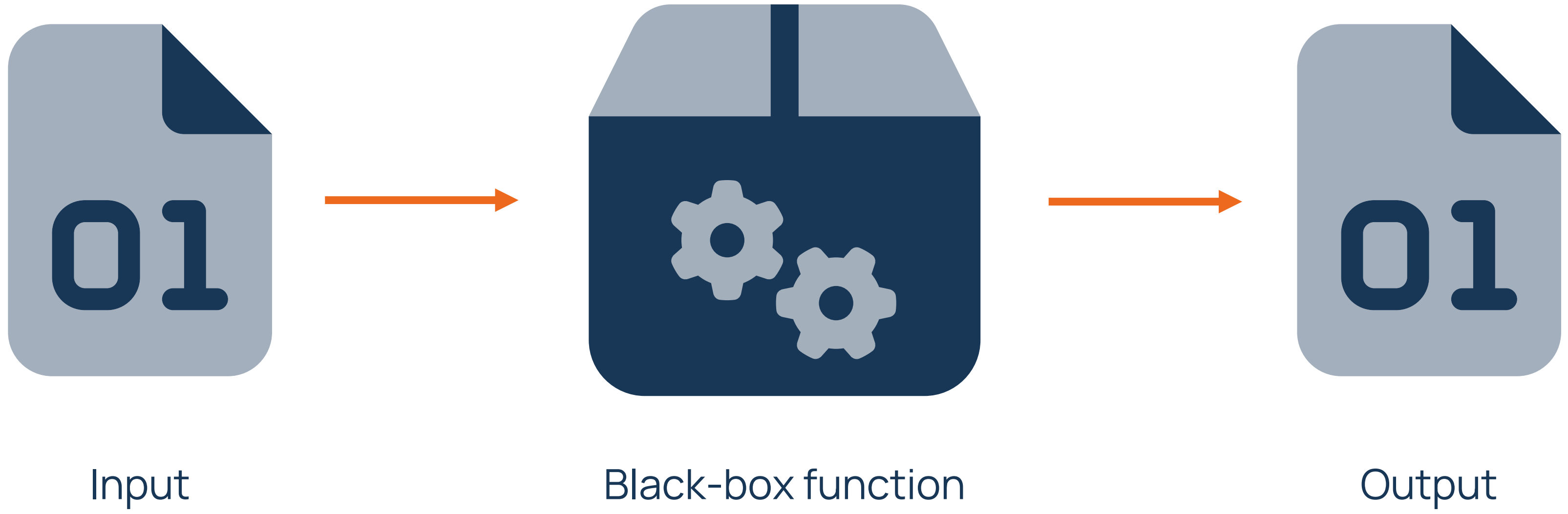
Optimize model parameters

Finance

Portfolio optimization

Black-box functions

$$f : x \mapsto f(x)$$



$$[BB]:\ \min f(x)$$

$$st \begin{cases} c_j(x) \leq 0 & \forall j \in [\![1, m]\!] \\ x_i \in \left[x_i^L, x_i^U\right] & \forall\, i \in [\![1, d]\!] \end{cases}$$

Black-box functions

- A black-box function: the **analytical formula is not available**
    - Simulator evaluates points and returns values
- An evaluation can last from **a few seconds to several hours**
- Limited **evaluation budget**
    - › Each point to be evaluated must be chosen carefully

Ideal properties of a black-box function:

- **As little noise as possible**: same point, same evaluation
- Kind of continuity

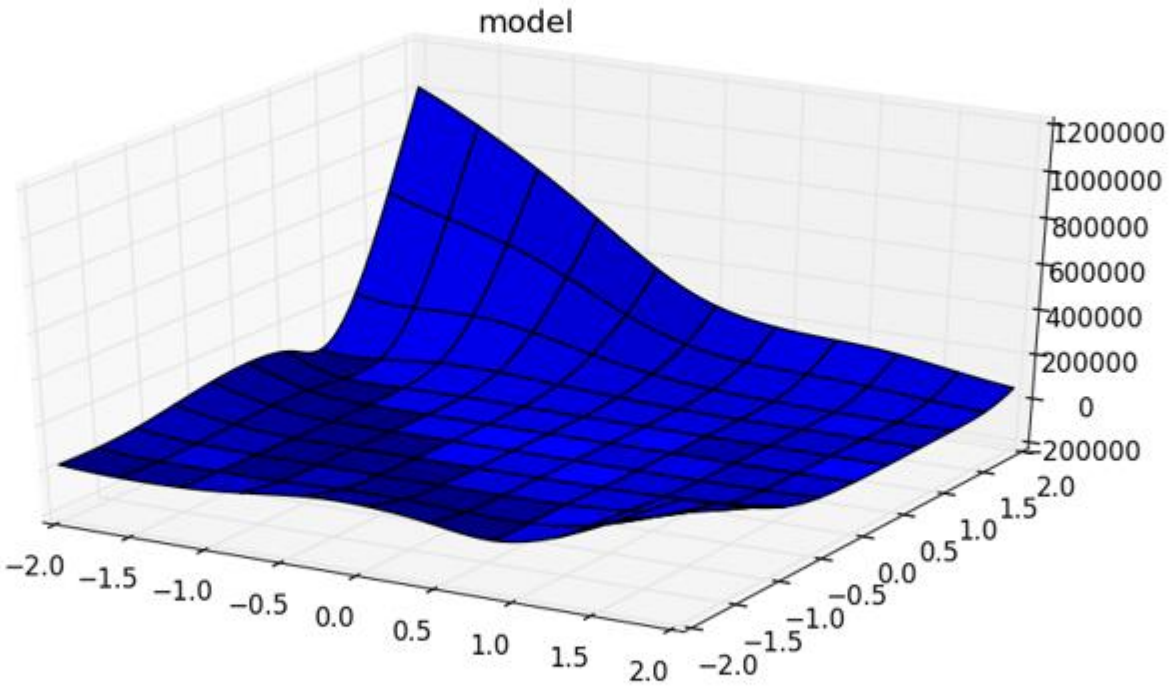
Hexaly Black-box Solver

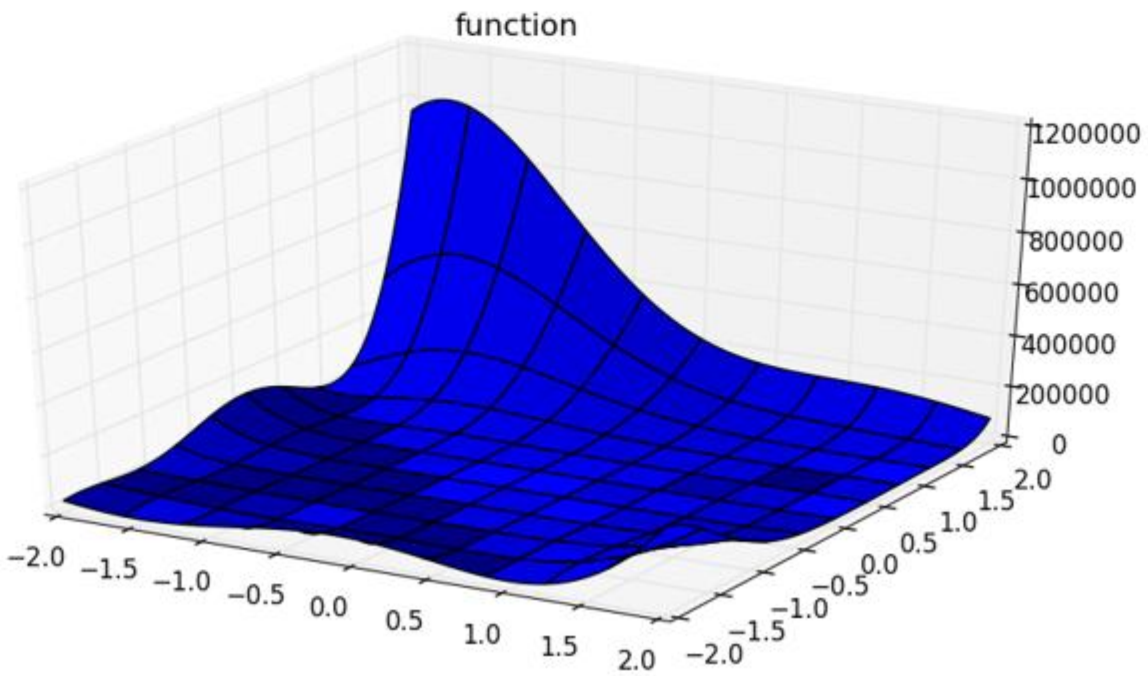
Surrogate modeling

Hexaly Black-box Optimizer

In a nutshell

Hexaly Optimizer tries to learn the profile of the actual black-box function thanks to previously evaluated points





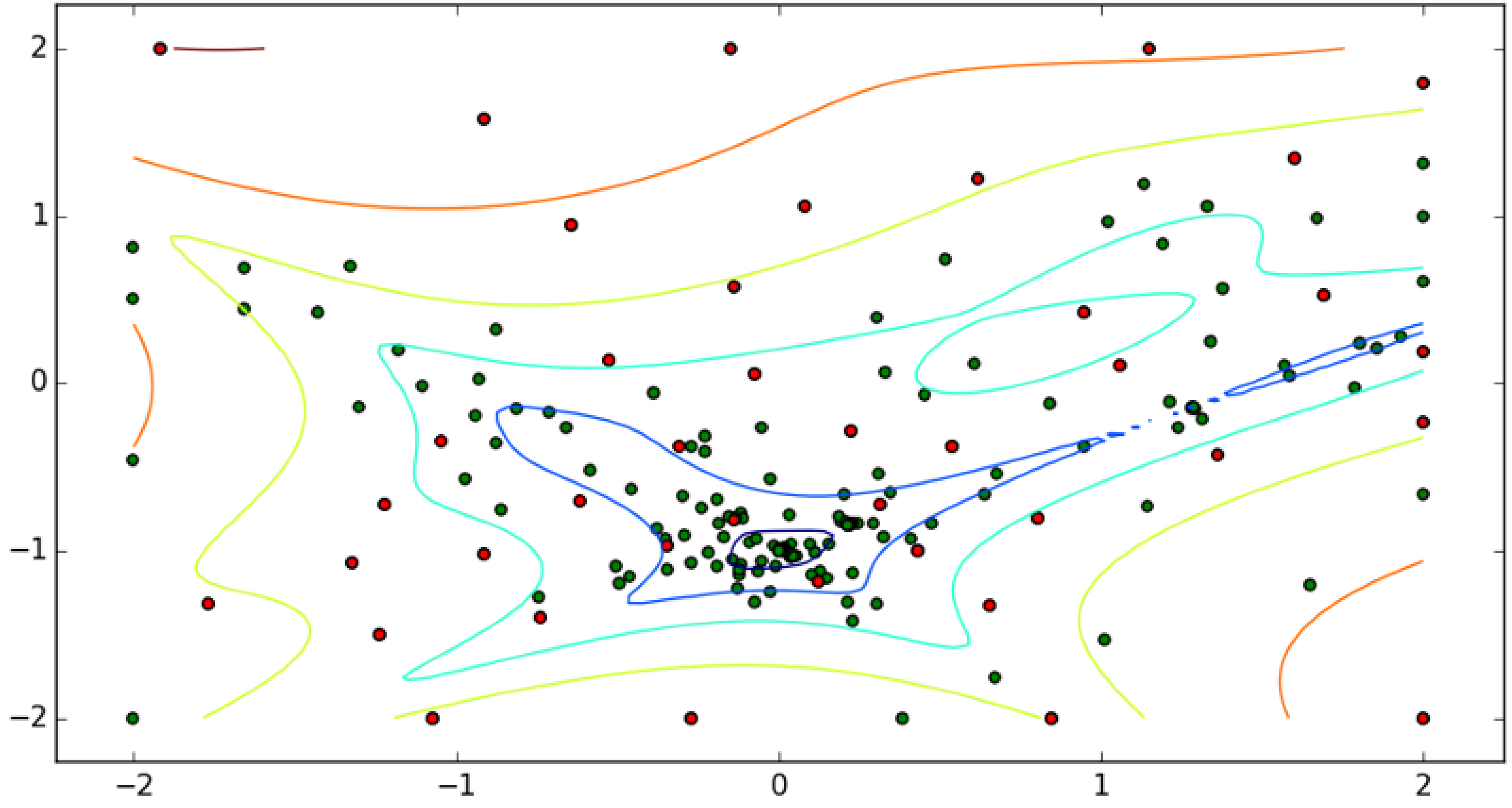
It optimizes the surrogate model to identify and evaluate new promising points

[1] A. Costa, G. Nannicini. *RBFOpt : an open-source library for black-box optimization with costly function evaluations*. Optimization Online 2014-09-4538, 2014.

Hexaly Black-box Optimizer

In a nutshell

From time to time, it explores new areas to find promising regions and escape local optima



Surrogate Modeling

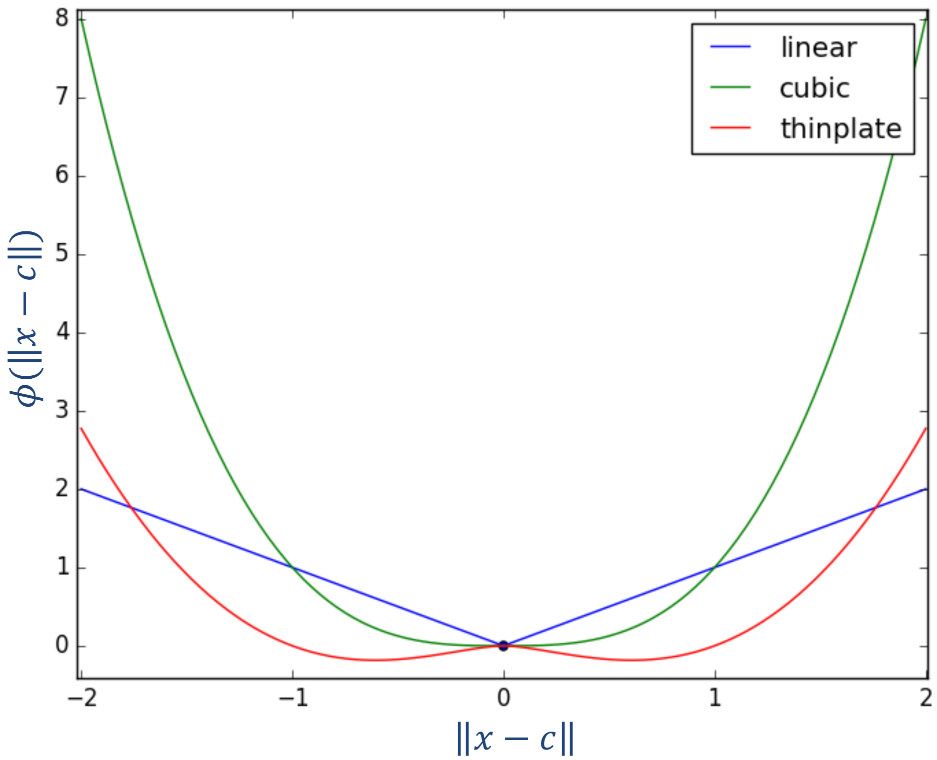
Radial Basis Function (RBF)

Surrogate modeling = Approximate the heavy function f by a surrogate s

$$\min f(x) \Rightarrow \min s(x)$$

Radial Basis Function (RBF)[2]

RBF	$\phi(r)$
Cubic	r^3
Linear	r
Thin Plate Spline	$r^2 \, log \, r$
Multiquadric	$\sqrt{r^2 + \gamma^2}$
Gaussian	$e^{-\gamma r^2}$



[2] H.M. Gutmann. *A radial basis function method for global optimization*. Journal of Global Optimization 19, 201-227, 2001.

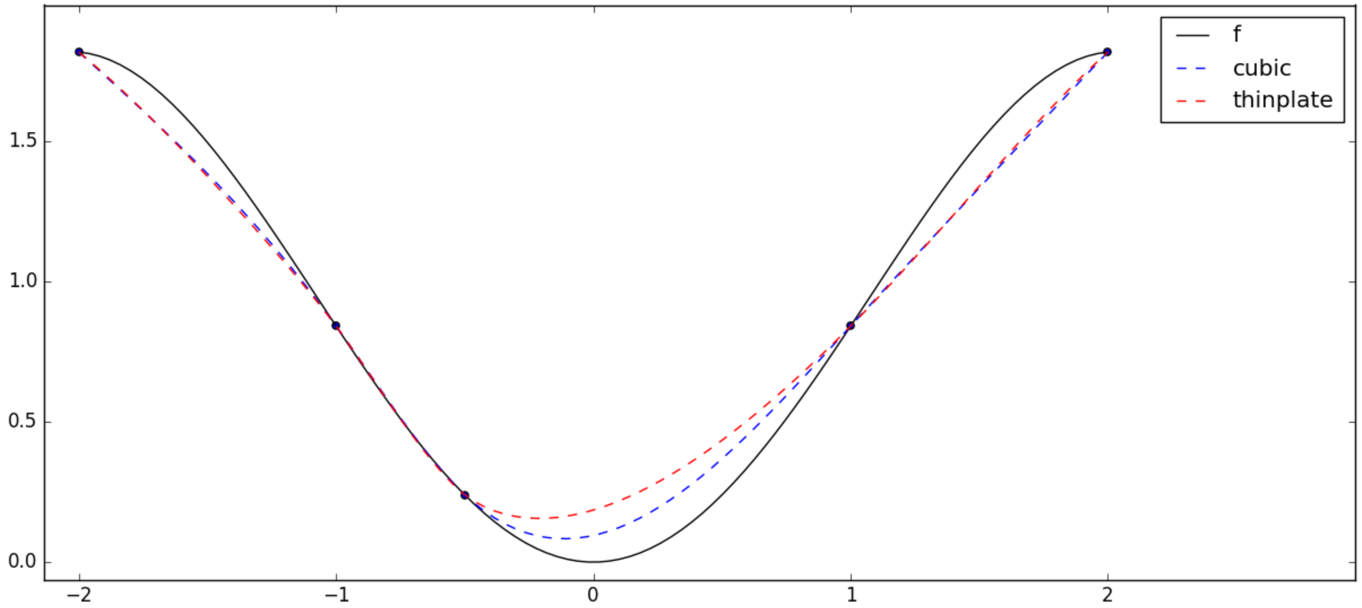
Surrogate Modeling

Radial Basis Function (RBF)

Surrogate modeling = Approximate the heavy function f by a surrogate s

$$\min f(x) \Rightarrow \min s(x)$$

$$s_k(x) = \sum_{i=0}^{k} \lambda_i \phi(||x - \boldsymbol{x_i}||) + p(x)$$



Build the surrogate: find λ_i and p coefficients thanks to the previously evaluated points $\boldsymbol{x_i}$

Hexaly Black-box Optimizer

Complete resolution method

Initialisation

Selection & evaluation of $n + 1$ points (random, LHS)

Iteration

1. Choice of the best surrogate (leave-one-out cross-validation)
2. Generation of a candidate point & evaluation

Generation of a candidate point

Alternate between :

- **Exploitation**
  - **Optimizing** the surrogate $\min s_k(x)$ -> via solving a Hexaly sub-model
  - **Neighborhood** for **integer** problems
- **Exploration**
  - Get a point as **far away** as possible from previously evaluated points (Hostile Brothers)

$$\min_x HB_k(x) = \max_{i \in [\![1,k]\!]} \frac{1 + \log(1 + |f(x_i) - f(x^*)|)}{\sqrt{\sum_{j \in J} \left(x_j - c_{ij}\right)^2}}$$

    -> via solving a Hexaly sub-model

  - Randomly

Hexaly Black-box Optimizer

Complete resolution method

Initialisation

Selection & evaluation of $n+1$ points (random, LHS)

Iteration

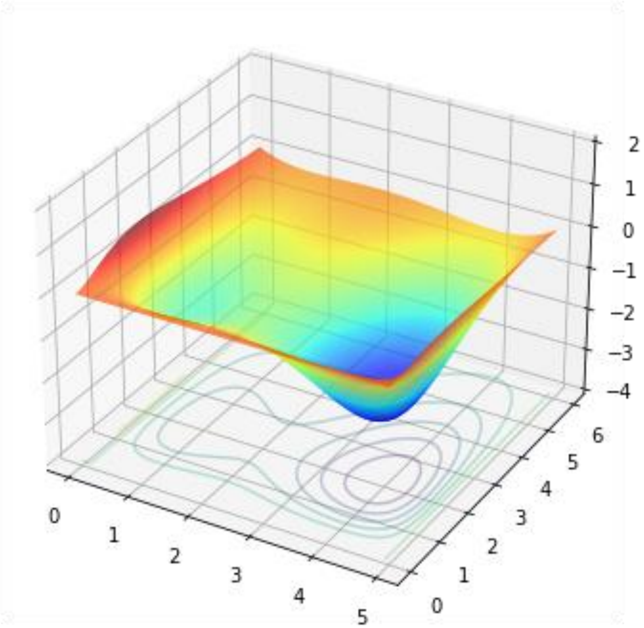
1. Choice of the best surrogate (leave-one-out cross-validation)
2. Generation of a candidate point & evaluation

$\Rightarrow$ **Stop** when the evaluation limit is reached

Example: Hosaki Function

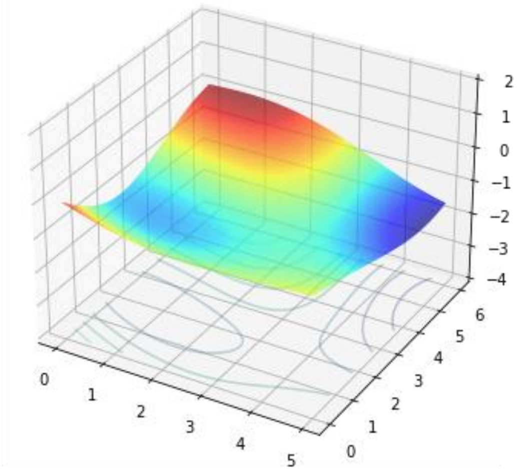
$$\min\left(1 - 8x_1 + 7x_1^2 - \frac{7}{3}x_1^3 + \frac{1}{4}x_1^4\right)x_2^2\, e^{-x_2}$$

$$Sc\begin{cases}0 \leq x_1 \leq 5\\ 0 \leq x_2 \leq 6\end{cases}$$

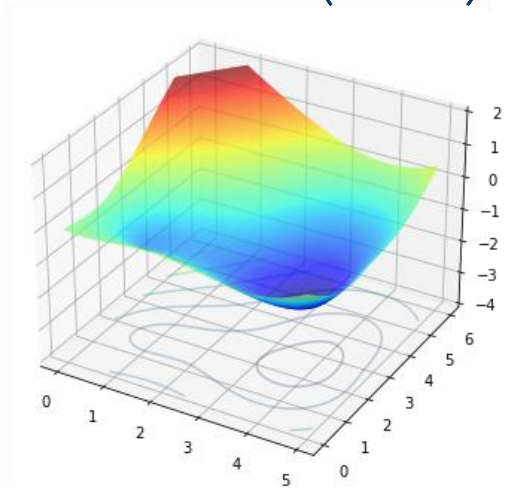


Real function

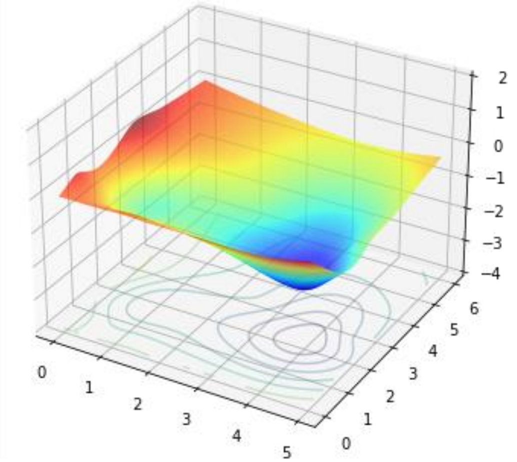
1 iteration (Gaussian)



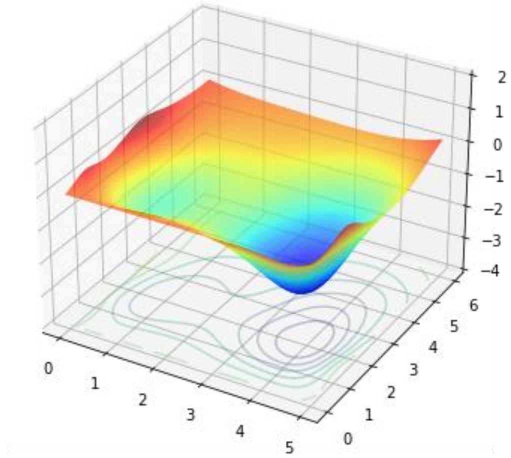
20 iterations (Cubic)



45 iterations (ThinPlate)



70 iterations (Multiquadric)



Constraints integration

Analytical and black-box

Constraints

Analytical

$$[BB]:\ \min f(x)$$

$$st \begin{cases} c_j(x) \leq 0 & \forall j \in [\![1, m]\!] \\ x_i \in \left[x_i^L, x_i^U\right] & \forall\, i \in [\![1, d]\!] \end{cases}$$

C_a: analytical

C_b: black-box

- Analytical formula available :

$$\forall\, c \in C_a, c(x) \leq 0$$

  › **Using Hexaly operators**

- All the analytical constraints must be met to evaluate a point

Constraints

Black-box

- Each constraint is **approximated by a surrogate**[3] :

$$\forall c \in C_{bb}, c(x) \leq 0 \Rightarrow s(x) \leq 0$$

  - Surrogate fitted and chosen at each iteration

- Add an **adaptive margin criterion**:

$$\forall\, c \in C_{bb}, c(x) \leq 0 \Rightarrow s(x) + \varepsilon \leq 0$$

  - Move away from the bounds to **increase the chances of finding a feasible point**
  - Adapted at each iteration, depending on the status of the point previously evaluated :
    - Reduce margin if $C_{\text{feas}} \geq T_{\text{feas}}$: $\varepsilon = \frac{1}{2}\varepsilon$
    - Increase margin if $C_{\text{infeas}} \geq T_{\text{infeas}}$: $\varepsilon = \min(2\varepsilon, \varepsilon_{\max})$

[3] R.G. Regis. *Constrained optimization by radial basis function interpolation for high-dimensional expensive black-box problems with infeasible initial points*. Engineering Optimization, 46(2):218–243, 2014.

Multi-objective optimization
Hierarchical

Multi-objective optimization

Lexicographic

The objectives are handled **hierarchically**, in a **lexicographic order**

```
model.minimize(obj1)

model.minimize(obj2)
```

Better on the first objective ⇒ Better overall

- When considering two solutions A and B, A is considered better than B overall if:
  - A is better than B on objectives 1 and 2
  - A is better than B on objective 1, although B is better than A on objective 2
  - A and B are equivalent on objective 1, and A is better than B on objective 2

Multi-objective optimization

Black-box Optimization

Handled in the **same way** in the black-box solver:

- **Lexicographic** order: objectives should be prioritized
- Each black-box objective is **approximated by a surrogate**

Pareto-efficient solutions: don't prioritize one objective over the others, but rather have a **tradeoff between all objectives**

- No built-in options in Hexaly Optimizer for Pareto optimization
- Still possible to build Pareto fronts, but **you should do it manually**

Black-box modeling
Using Hexaly Optimizer

Black-box functions

Where can black-box functions be used in a Hexaly Optimizer model?

A call to a black-box function can be used in a Hexaly Optimizer model as

- One or several **objective functions**
- One or several **constraints**
- It is possible to **mix analytical and black-box constraints and objectives** in the same model

API available in C++, C#, Java, Python, HXM:

- Function that returns **one or multiple values of the same type** (integer or double)

```
func <- doubleExternalFunction(...) / intArrayExternalFunction(...);
surrogateParams = func.context.enableSurrogateModeling();
callValues <- call(func, ...);
```

Hexaly Black-box

Modeling

Control the resolution time thanks to the black-box function evaluation limit:

- Number of evaluations is a good stopping criterion for black-box
- Objective threshold can be useful to save computation time

```
function param() {
    surrogateParams.evaluationLimit = 20;
}
```

Hexaly Black-box

Modeling

Inject known initial points:

- Each initial point sent provides valuable information and saves computation time

```
function param() {
    evaluationPoint = surrogateParams.createEvaluationPoint();
    for [i in 0...nbArguments]
        evaluationPoint.addArgument(pointArguments[i]);
    evaluationPoint.returnValue = pointValue;
}
```

Set an initial solution (not mandatory)

Implementation
Challenges

Implementation Challenges

Generic solver

- Surrogate modeling: easy to use with our solver

Parameters configuration

- Tuned with internal benchmarks (customers and academic instances)
- Advanced parameters

Challenges

- **Parallelization**
- **Scalability**
- **Memory**
- **Reproducibility**
- **Integration** (simulation platforms)

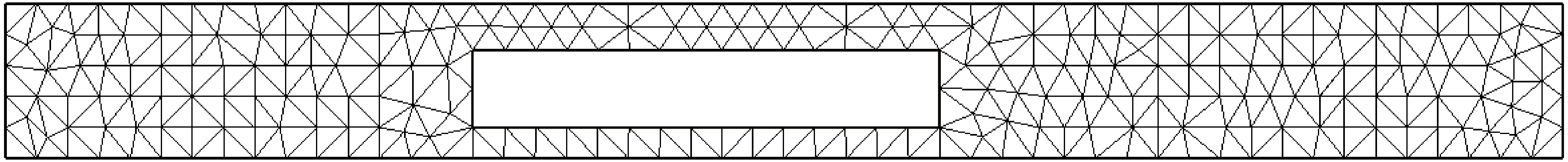
Black-box

Some use cases

Shape Optimization

A simple use case

- Rectangular beam in two dimensions, clamped on the left and with a rectangular hole
- Goal is to determine the hole position to minimize compliance of the structure & maximizing its stiffness (obtained by a simulator)
- **Payoff between an increase and a decrease in the size of the hole**:
  - Increasing the size of the hole reduces the mass of the beam
  - Decreasing its size increases the beam’s stiffness



Rectangular beam in two dimensions, with a rectangular hole
(picture generated by FreeFem++[4])

[4] F. Hecht. *New development in FreeFem++*. Journal of numerical mathematics, 2012, vol. 20, no 3-4, p. 251-266.

Shape Optimization

A simple use case

Decisions

- The **hole's dimensions and the position** of its bottom left corner

Constraints (analytical)

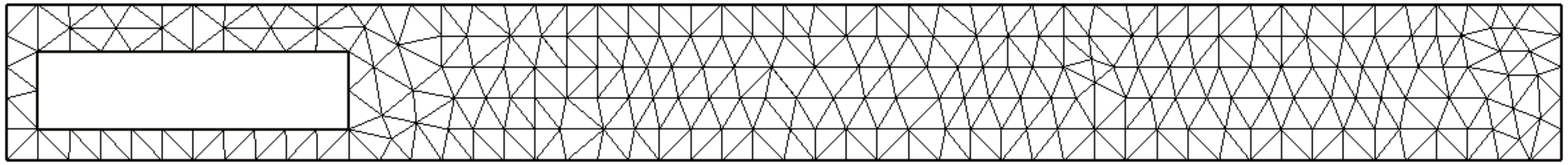
- The size of the hole is constrained to be between a minimal and a maximal value
- The hole must not be too close to the beam's boundary

Objective (simulation)

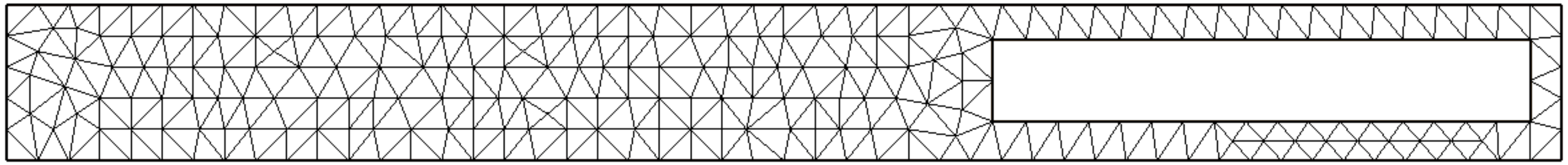
- Minimizing the compliance of the structure
  - Obtained by a simulation (scalar product of the displacement field of the system and the volumic force applied to it)

Shape Optimization

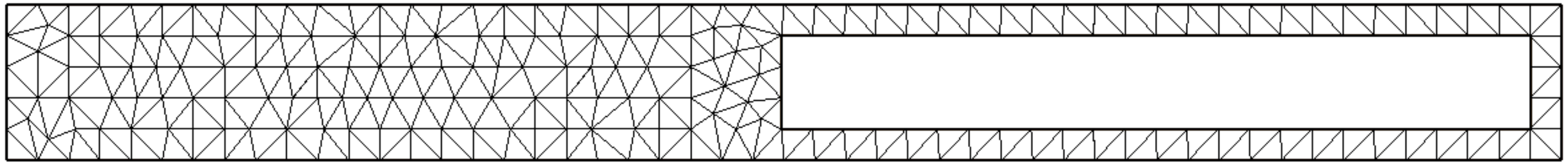
A simple use case



1 iteration



10 iterations



50 iterations

After 50 iterations, a very good quality solution is found by Hexaly

https://www.hexaly.com/tutorial/shape-optimization-through-simulation-optimization-with-localsolver

Automated Design Optimization Application

Ongoing project

- **Computer Architecture Simulation**
- **Different simulators** can interact with one another
- Generate a **user-friendly** web application, to easily model and optimize design optimization problems
- Goal: provide a **low-code** application, **easy to use for non-optimization experts**

hexaly

Design Optimization

Name: * Cantilevered Beam

Design parameters

Name * h1
Values / Configuration elements
Possible values * { 0.1, 0.26, 0.35, 0.5, 0.65, 0.75, 0.9, 1.0 }

Name * H
Values / Configuration elements
Possible values * [3.0, 7.0]

Name * b1
Values / Configuration elements
Possible values * [2.0, 12.0]

Name * b2
Values / Configuration elements
Possible values * [3.0, 7.0]

Constraints

Name * Bending Stress
Function arguments * h1 H b1 b2
Maximum value * 5000

Name * Deflection
Function arguments * h1 H b1 b2
Maixmum value * 0.1

Objectives

Name * Beam Volume
Direction * Minimization
Function arguments * h1 H b1 b2

Optimization parameters

Solver * Hexaly
Evaluation limit 50

Launch optimization

Optimal fertilization of agricultural parcels

Veolia x Hexaly

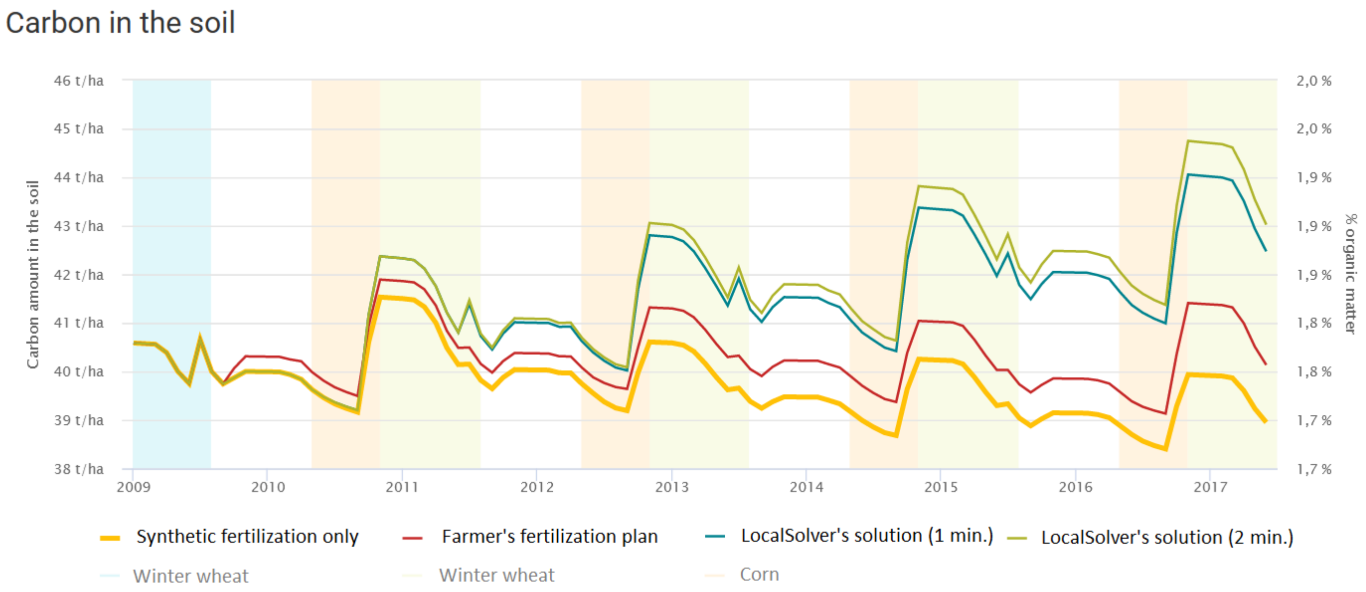
- **Simulation optimization tool** to help farmers transition toward smart and sustainable agriculture via organic fertilization
  - More sustainable alternative to synthetic fertilizers: **organic soil amendment and fertilization via compost**
  - **25-year** time horizon
- Decisions:
  - **Quantity of synthetic and organic fertilizer spread** onto the crops each month
- Objectives and constraints:
  - Evaluated by **calling a complex soil simulation software** that generates **predictions about the soil's evolution** over the years
  - From 1 to 10 seconds per simulation call

› Optimal fertilization plans in **minutes**

Optimal fertilization of agricultural parcels

Veolia x Hexaly

Solutions that both meet their goals in the short term and are sustainable in the long term



Advantages of using organic fertilizers by simulating the evolution of the carbon amount in the soil over eight years

Fertilization strategy resulting in 8% more carbon in the soil than the farmer plan, and 10% more than synthetic fertilization only, making the soil more fertile in the long term

A shallow dive into Hexaly Black-box Optimization Solver

Emeline Tenaud

etenaud@hexaly.com

JFRO 2024

References

1. A. Costa, G. Nannicini. *RBFOpt : an open-source library for black-box optimization with costly function evaluations*. Optimization Online 2014-09-4538, 2014.
2. H.M. Gutmann. *A radial basis function method for global optimization*. Journal of Global Optimization 19, 201-227, 2001.
3. R.G. Regis. *Constrained optimization by radial basis function interpolation for high-dimensional expensive black-box problems with infeasible initial points*. Engineering Optimization, 46(2):218–243, 2014.
4. F. Hecht. *New development in FreeFem++*. Journal of numerical mathematics, 2012, vol. 20, no 3-4, p. 251-266.