Bayesian optimization of variable-size design space problems

Julien Pelamatti[1]
in collaboration with Loïc Brevault[2], Mathieu Balesdent[2], El-Ghazali Talbi[3], Yannick Guerin[4]

[1] EDF R&D [2] ONERA/DTIS [3] Inria Lille-Nord Europe
[4] CNES, direction des lanceurs

1st of April 2021
UQSay seminars

Disclaimer

The presented work originates from a PhD thesis and a collaboration between ONERA, CNES and the University of Lille.

- Supervisors : Loïc Brevault & Mathieu Balesdent
- Research director : El-Ghazali Talbi
- Industrial supervisor : Yannick Guerin

More details can be found in the thesis manuscript:
Mixed-variable Bayesian optimization : application to aerospace system design

Overview

1. Context & problem definition
2. Mixed-variable Bayesian optimization
3. Variable-size design space problems optimization

Context: Complex system design optimization

- Variable-Size Design Space Problem
  - Continuous variables (*e.g.*, structure sizing, propellant mass, combustion chamber pressure)
  - Discrete variables (*e.g.*, number of structural reinforcements, type of material, type of propellant)
  - Dimensional variables (*e.g.*, architectural choices, presence of wings)

Context: Complex system design optimization

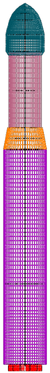
- Variable-Size Design Space Problem
  - Continuous variables (*e.g.*, structure sizing, propellant mass, combustion chamber pressure)
  - Discrete variables (*e.g.*, number of structural reinforcements, type of material, type of propellant)
  - Dimensional variables (*e.g.*, architectural choices, presence of wings)
- Multiple constraints
  - Mission requirements
  - Safety requirements

Context: Complex system design optimization

- Variable-Size Design Space Problem
  - Continuous variables (*e.g.*, structure sizing, propellant mass, combustion chamber pressure)
  - Discrete variables (*e.g.*, number of structural reinforcements, type of material, type of propellant)
  - Dimensional variables (*e.g.*, architectural choices, presence of wings)
- Multiple constraints
  - Mission requirements
  - Safety requirements
- Computationally intensive objective and constraint functions
  - Finite Element Models
  - Multi-Disciplinary Analyses

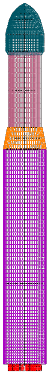
Context: Variable-Size Design Space Problem

- Presence of dimensional variables leads to dynamically varying optimization problems



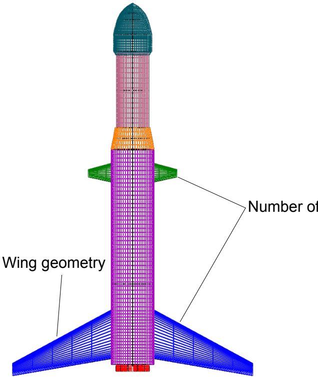
Context: Variable-Size Design Space Problem

- Presence of dimensional variables leads to dynamically varying optimization problems
- Dimensional variables variations can influence:



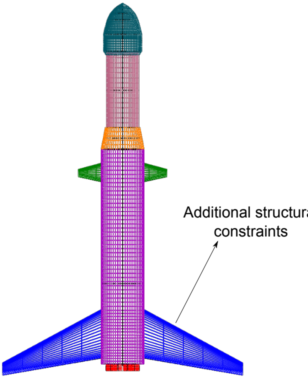
Context: Variable-Size Design Space Problem

- Presence of dimensional variables leads to dynamically varying optimization problems
- Dimensional variables variations can influence:
  - The number and type of design variables
    - Presence of lifting surfaces
    - Type of propulsion (*i.e.*, solid, liquid)



Context: Variable-Size Design Space Problem

- Presence of dimensional variables leads to dynamically varying optimization problems
- Dimensional variables variations can influence:
  - The number and type of design variables
    - Presence of lifting surfaces
    - Type of propulsion (*i.e.*, solid, liquid)
  - The number and the nature of the constraints



Problem statement

Variable-Size Design Space Problem (VSDSP)

$$
\begin{aligned}
\min \quad & f(\mathbf{x}, \mathbf{z}, \mathbf{w}) && f : \mathbb{R}^{n_x(\mathbf{w})} \times \prod_{d=1}^{n_z(\mathbf{w})} F_{z_d} \times F_w \to F_f \subseteq \mathbb{R} \\
\text{w.r.t.} \quad & \mathbf{x} \in F_x(\mathbf{w}) \subseteq \mathbb{R}^{n_x(\mathbf{w})} && \text{Continuous variables} \\
& \mathbf{z} \in \prod_{d=1}^{n_z(\mathbf{w})} F_{z_d} && \text{Discrete variables} \\
& \mathbf{w} \in F_w && \text{Dimensional variables} \\
\text{s.t.} \quad & \mathbf{g}(\mathbf{x}, \mathbf{z}, \mathbf{w}) \leq 0 \\
& g_i : F_{x_i}(\mathbf{w}) \times \prod_{d=1}^{n_{z_i}(\mathbf{w})} F_{z_{d_i}} \times F_w \to F_{g_i} \subseteq \mathbb{R} \\
& \text{for} \quad i = 1, \ldots, n_g(\mathbf{w})
\end{aligned}
$$

Context: Existing approaches

- Redefinition of the optimization problem through high-level variables (*e.g.*, thrust) Frank 2016

Context: Existing approaches

- Redefinition of the optimization problem through high-level variables (*e.g.*, thrust) Frank 2016
- Hierarchical decomposition of optimization problems Venter 2004, Roy 2017

Context: Existing approaches

- Redefinition of the optimization problem through high-level variables (*e.g.*, thrust) Frank 2016
- Hierarchical decomposition of optimization problems Venter 2004, Roy 2017
- Heuristic algorithms
  - Hidden gene Genetic Algorithm Abdelkhalik 2013
  - Hidden gene Differential Evolution Abdelkhalik 2013
  - Structured-chromosome evolutionary algorithm Nyew 2015

Context: Existing approaches

- Redefinition of the optimization problem through high-level variables (*e.g.*, thrust) Frank 2016
- Hierarchical decomposition of optimization problems Venter 2004, Roy 2017
- Heuristic algorithms
  - Hidden gene Genetic Algorithm Abdelkhalik 2013
  - Hidden gene Differential Evolution Abdelkhalik 2013
  - Structured-chromosome evolutionary algorithm Nyew 2015
- Mesh-search algorithms Audet 2000, Abramson 2007, Abramson 2009

Context: Existing approaches

- Redefinition of the optimization problem through high-level variables (*e.g.*, thrust) Frank 2016
- Hierarchical decomposition of optimization problems Venter 2004, Roy 2017
- Heuristic algorithms
  - Hidden gene Genetic Algorithm Abdelkhalik 2013
  - Hidden gene Differential Evolution Abdelkhalik 2013
  - Structured-chromosome evolutionary algorithm Nyew 2015
- Mesh-search algorithms Audet 2000, Abramson 2007, Abramson 2009

✗ Large number of function evaluations
✗ Inefficient constraint handling

⇒ **Not viable for computationally intensive design problems**

Context

Proposed approach:

Surrogate Model Based Design Optimization (SMBDO)

more specifically

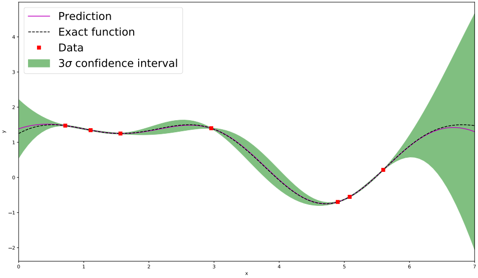
Bayesian Optimization (BO) Jones 1998

✓ Relying on computationally cheap surrogate models of the problem functions
✓ Problem functions evaluated at the most promising locations of the search space
✓ Fast convergence towards the optimum neighborhood

Bayesian Optimization

Prediction of the modeled function at an unmapped location $\mathbf{x}^*$ computed with the help of a Gaussian Process (GP) Y.

$$\begin{aligned} \text{Prediction:} \quad \hat{y}(\mathbf{x}^*) &= \mu + \boldsymbol{\psi}^T(\mathbf{x}^*)\mathbf{K}^{-1}(\mathbf{y} - \mathbf{1}\mu) \\ \text{Variance:} \quad \hat{s}^2(\mathbf{x}^*) &= k(\mathbf{x}^*, \mathbf{x}^*) - \boldsymbol{\psi}^T(\mathbf{x}^*)\mathbf{K}^{-1}\boldsymbol{\psi}(\mathbf{x}^*) \end{aligned}$$



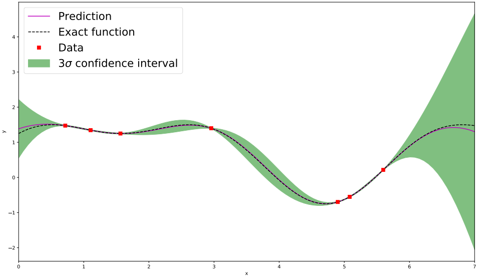
Bayesian Optimization

Prediction of the modeled function at an unmapped location $\mathbf{x}^*$ computed with the help of a Gaussian Process (GP) Y.

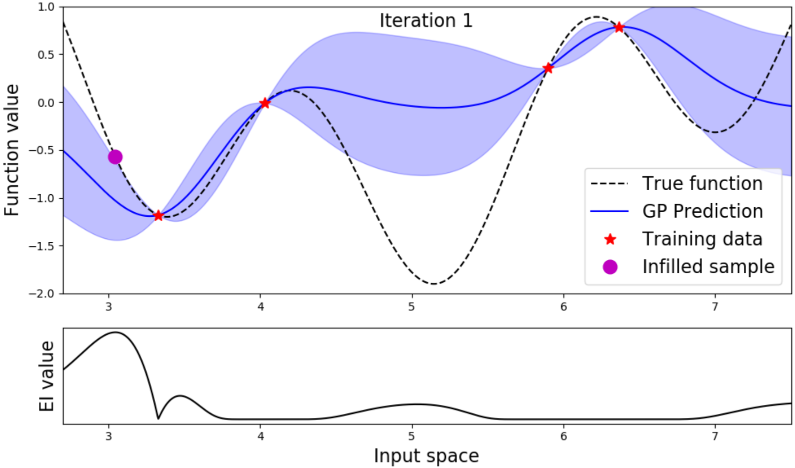
$$\begin{aligned} \text{Prediction:} \quad \hat{y}(\mathbf{x}^*) &= \mu + \boldsymbol{\psi}^T(\mathbf{x}^*)\mathbf{K}^{-1}(\mathbf{y} - \mathbf{1}\mu) \\ \text{Variance:} \quad \hat{s}^2(\mathbf{x}^*) &= k(\mathbf{x}^*, \mathbf{x}^*) - \boldsymbol{\psi}^T(\mathbf{x}^*)\mathbf{K}^{-1}\boldsymbol{\psi}(\mathbf{x}^*) \end{aligned}$$

- Defined through a parameterized covariance kernel $k(\cdot,\cdot)$

$$\begin{aligned} \mathbf{K}_{i,j} &= k\left(\mathbf{x}^i, \mathbf{x}^j\right) \\ \boldsymbol{\psi}_i &= k\left(\mathbf{x}^i, \mathbf{x}^*\right) \end{aligned}$$



Bayesian Optimization



Bayesian Optimization

- Original formulation of BO defined for purely continuous optimization problems.

Bayesian Optimization

- Original formulation of BO defined for purely continuous optimization problems.

- In order to adapt the algorithm for variable-size design space problems, it is necessary to:
  1. **Redefine the covariance kernel**
  2. **Redefine the acquisition function definition and optimization**

Bayesian Optimization

- Original formulation of BO defined for purely continuous optimization problems.

- In order to adapt the algorithm for variable-size design space problems, it is necessary to:
    1. **Redefine the covariance kernel**
    2. **Redefine the acquisition function definition and optimization**

- A few discrete variable GP kernel parameterizations exist
  Never applied within the context of BO

Approach layout

Followed approach

1. **Modeling of mixed continuous/discrete functions**
2. **Bayesian Optimization of constrained mixed continuous/discrete problems**
3. **Bayesian Optimization of variable-size design space problems**

Part 1: Mixed-variable modeling

Mixed-variable functions

$$f(\mathbf{x}, \mathbf{z}) \qquad f : \mathbb{R}^{n_x} \times \prod_{d=1}^{n_z} F_{z_d} \rightarrow F_f \subseteq \mathbb{R}$$

where $\mathbf{x} \in F_x \subseteq \mathbb{R}^{n_x}$

$$\mathbf{z} \in \prod_{d=1}^{n_z} F_{z_d}$$

Mixed-variable modeling

- Each discrete variable z_s is characterized by finite number of possible values: **Level**
- Discrete level combination: **Category**

Mixed-variable modeling

- Each discrete variable z_s is characterized by finite number of possible values: **Level**
- Discrete level combination: **Category**

Levels:

material = {steel, aluminum, titanium}

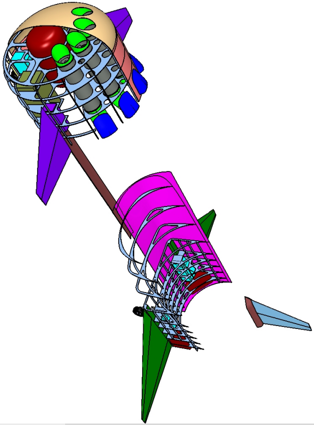
propulsion type = {solid, liquid, hybrid}

number of boosters = {2,4}

$$\Downarrow$$

Category:

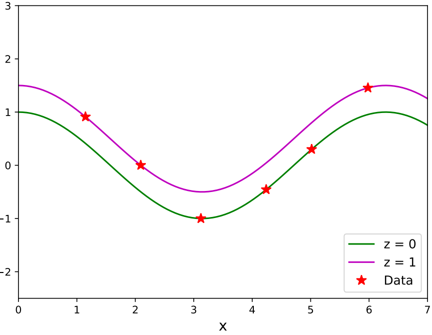
{steel, hybrid propulsion, 2 boosters }



Mixed-variable modeling

Example: $f(x,z) = \cos(x) + 0.5 \cdot z$

with $z = \{0,1\}$, $\quad x \in [0,7]$

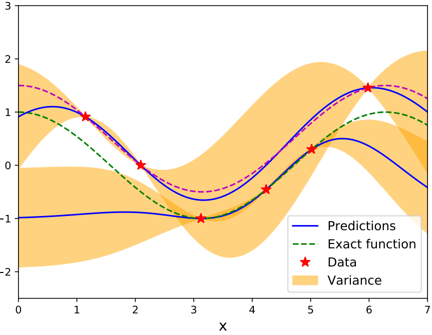


Mixed-variable modeling

Example: $f(x,z) = \cos(x) + 0.5 \cdot z$

with $z = \{0,1\}$, $\quad x \in [0,7]$

Separate surrogate modeling

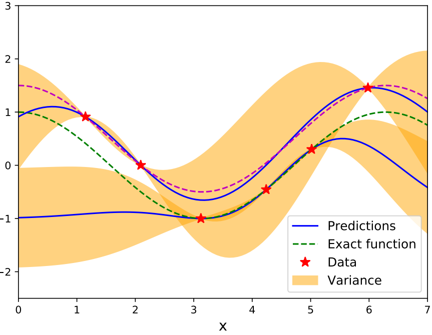


Mixed-variable modeling

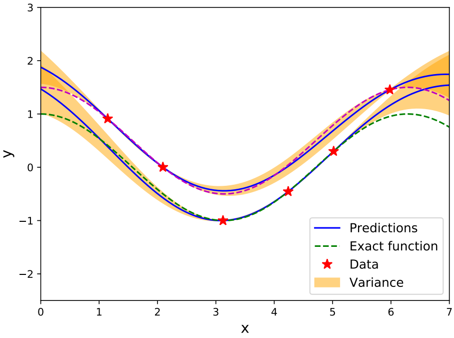
Example: $f(x,z) = \cos(x) + 0.5 \cdot z$

with $z = \{0,1\}, \quad x \in [0,7]$

Separate surrogate modeling



Mixed surrogate modeling



Mixed-variable kernel

- Complex kernels can be composed through sums and products of valid kernels following the RKHS formalism

Mixed-variable kernel

- Complex kernels can be composed through sums and products of valid kernels following the RKHS formalism
- A mixed-variable kernel can be obtained by combining purely continuous and discrete kernels [Roustant 2018]:

$$k\left((\mathbf{x},\mathbf{z}),(\mathbf{x}',\mathbf{z}')\right) = k_c(\mathbf{x},\mathbf{x}') * k_d(\mathbf{z},\mathbf{z}')$$

Mixed-variable kernel

- Complex kernels can be composed through sums and products of valid kernels following the RKHS formalism

- A mixed-variable kernel can be obtained by combining purely continuous and discrete kernels [Roustant 2018] :

$$k\left((\mathbf{x},\mathbf{z}),(\mathbf{x}',\mathbf{z}')\right) = k_c(\mathbf{x},\mathbf{x}') * k_d(\mathbf{z},\mathbf{z}')$$

- Common continuous kernels, such as the *squared-exponential*, can be used:

$$k_c(\mathbf{x}^i,\mathbf{x}^j) = \exp\left(-\sum_{i=1}^{n_x}\theta_i|x_i - x_i'|^2\right)$$

Discrete kernel parameterizations

- Decomposing the discrete kernel into a product of uni-dimensional ones [Pelamatti 2018]:

$$k_d(\mathbf{z},\mathbf{z}') = \prod_{i=1}^{n_z} k_{d_i}(z_i, z_i')$$

Discrete kernel parameterizations

- Decomposing the discrete kernel into a product of uni-dimensional ones [Pelamatti 2018]:

$$k_d(\mathbf{z},\mathbf{z}') = \prod_{i=1}^{n_z} k_{d_i}(z_i, z_i')$$

A valid kernel can be constructed through the RKHS formalism

$$k(z,z') := \langle \phi(z), \phi(z') \rangle_{\mathscr{H}}$$

- $\phi(z) : F_z \rightarrow \mathscr{H}$ Is a mapping function
- $\langle \cdot, \cdot \rangle_{\mathscr{H}}$ Is an inner product on $\mathscr{H}$

Discrete kernel parameterizations

- Decomposing the discrete kernel into a product of uni-dimensional ones [Pelamatti 2018]:

$$k_d(\mathbf{z}, \mathbf{z}') = \prod_{i=1}^{n_z} k_{d_i}(z_i, z_i')$$

A valid kernel can be constructed through the RKHS formalism

$$k(z, z') := \langle \phi(z), \phi(z') \rangle_{\mathscr{H}}$$

- $\phi(z) : F_z \rightarrow \mathscr{H}$ Is a mapping function
- $\langle \cdot, \cdot \rangle_{\mathscr{H}}$ Is an inner product on $\mathscr{H}$

$\rightarrow$ Different parameterizations can be considered

Discrete kernel parameterizations

Example: Compound symmetry kernel

$$k(z,z') = \delta(z,z') = \begin{cases} 1 & \text{if} \quad z = z' \\ 0 & \text{if} \quad z \neq z' \end{cases}$$

mapping:

$$z \in \{z_1, z_2, z_3, z_4\} \rightarrow \begin{cases} \phi(z = z_1) = [1,0,0,0] \\ \phi(z = z_2) = [0,1,0,0] \\ \phi(z = z_3) = [0,0,1,0] \\ \phi(z = z_4) = [0,0,0,1] \end{cases}$$

Inner product:

$$\langle \phi(z), \phi(z') \rangle = \phi(z)^T \phi(z') = \delta_z(z,z')$$

Discrete kernel parameterizations

Alternatively:

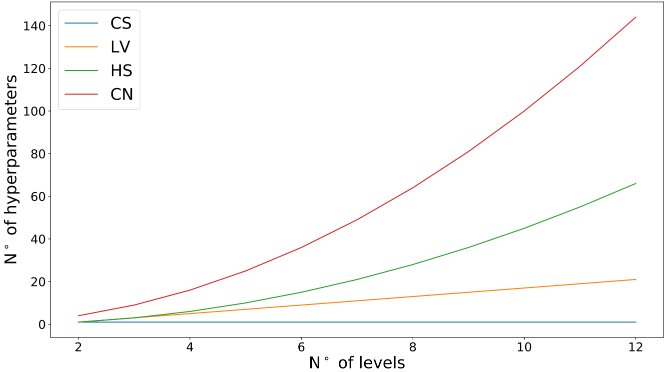
- Finite number of categories
- Discrete kernel represented by a matrix $\mathbf{T}$.
    - Symmetric
    - Positive-definite
- Each element $\mathbf{T}_{i,j}$ contains the correlation between two categories of the problem $k_d(\mathbf{z}^i, \mathbf{z}^j)$
- Size of $\mathbf{T}$ proportional to the number of categories
- $\rightarrow$ Different parameterizations can be considered

"Overview and comparison of gaussian process-based surrogate models for mixed continuous and discrete variables: application on aerospace design problems"

Mixed-variable modeling considerations

Hyperparameter scaling

- The number of hyperparameters characterizing each kernel varies as a function of the number of levels
- Trade-off between modeling complexity and available data



Part 2: Mixed-variable problems Bayesian optimization

$$
\begin{aligned}
\text{min} \quad & f(\mathbf{x}, \mathbf{z}) & & f : F_x \times F_z \rightarrow F_f \subseteq \mathbb{R} \\
\text{w.r.t.} \quad & \mathbf{x} \in F_x \subseteq \mathbb{R}^{n_x} \\
& \mathbf{z} \in F_z \\
\text{s.t.} \quad & \mathbf{g}(\mathbf{x}, \mathbf{z}) \leq 0 \\
& g_i : F_{x_i} \times F_{z_i} \rightarrow F_{g_i} \subseteq \mathbb{R} \quad \text{for} \quad & & i = 1, \ldots, n_g
\end{aligned}
$$

Mixed-variable Bayesian Optimization

Extension of continuous Bayesian Optimization to the mixed-variable case

Popular acquisition functions can be used under the condition that the GP prediction is normally distributed

- Objective function: $Y(\mathbf{x}^*, \mathbf{z}^*) \sim \mathcal{N}\left(\hat{y}(\mathbf{x}^*, \mathbf{z}^*), \hat{s}^2(\mathbf{x}^*, \mathbf{z}^*)\right)$
- Constraints: $G(\mathbf{x}^*, \mathbf{z}^*) \sim \mathcal{N}\left(\hat{g}(\mathbf{x}^*, \mathbf{z}^*), \hat{s}_g^2(\mathbf{x}^*, \mathbf{z}^*)\right)$

Mixed-variable Bayesian Optimization

Extension of continuous Bayesian Optimization to the mixed-variable case

Popular acquisition functions can be used under the condition that the GP prediction is normally distributed

- Objective function: $Y(\mathbf{x}^*, \mathbf{z}^*) \sim \mathcal{N}\left(\hat{y}(\mathbf{x}^*, \mathbf{z}^*), \hat{s}^2(\mathbf{x}^*, \mathbf{z}^*)\right)$
- Constraints: $G(\mathbf{x}^*, \mathbf{z}^*) \sim \mathcal{N}\left(\hat{g}(\mathbf{x}^*, \mathbf{z}^*), \hat{s}_g^2(\mathbf{x}^*, \mathbf{z}^*)\right)$

Criteria validity can be ensured in the mixed search space through proper kernel parameterization

Acquisition functions

Objective function:

- Expected Improvement:

$$EI(\mathbf{x}^*, \mathbf{z}^*) = \mathbb{E}\left[\max\left(y_{min} - Y(\mathbf{x}^*, \mathbf{z}^*), 0\right)\right]$$

Constraints : Schonlau 2011

- Probability of Feasibility:

$$PoF(\mathbf{x}^*, \mathbf{z}^*) = \prod_{i=1}^{n_g} \mathbb{P}(G_i(\mathbf{x}^*, \mathbf{z}^*) \leq 0)$$

- Expected Violation:

$$EV_i(\mathbf{x}^*, \mathbf{z}^*) = \mathbb{E}\left[\max\left(0 - G_i(\mathbf{x}^*, \mathbf{z}^*), 0\right)\right]$$

Acquisition functions

Objective function:

- Expected Improvement:

$$EI(\mathbf{x}^*, \mathbf{z}^*) = \mathbb{E}\left[\max\left(y_{min} - Y(\mathbf{x}^*, \mathbf{z}^*), 0\right)\right]$$

Constraints : Schonlau 2011

- Probability of Feasibility:

$$PoF(\mathbf{x}^*, \mathbf{z}^*) = \prod_{i=1}^{n_g} \mathbb{P}(G_i(\mathbf{x}^*, \mathbf{z}^*) \leq 0)$$

- Expected Violation:

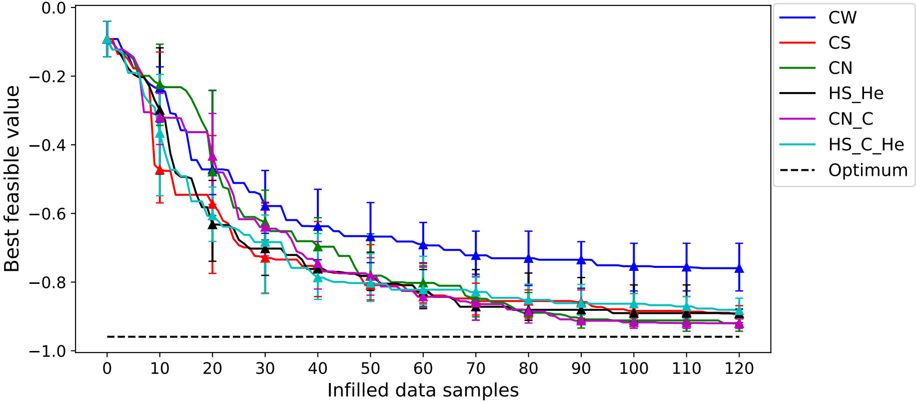
$$EV_i(\mathbf{x}^*, \mathbf{z}^*) = \mathbb{E}\left[\max\left(0 - G_i(\mathbf{x}^*, \mathbf{z}^*), 0\right)\right]$$

Acquisition function must be optimized in the mixed search space

Optimization performance comparison

Augmented Branin function:

- 10 continuous variables
- 2 discrete variables (4 categories)
- 1 constraint



Variable-size design space optimization problem

Part 3: Variable-size design space problems

$$
\begin{aligned}
\min \quad & f(\mathbf{x},\mathbf{z},\mathbf{w}) && f: \mathbb{R}^{n_x(\mathbf{w})} \times \prod_{d=1}^{n_z(\mathbf{w})} F_{z_d} \times F_w \rightarrow F_f \subseteq \mathbb{R} \\
\text{w.r.t.} \quad & \mathbf{x} \in F_x(\mathbf{w}) \subseteq \mathbb{R}^{n_x(\mathbf{w})} && \text{Continuous variables} \\
& \mathbf{z} \in \prod_{d=1}^{n_z(\mathbf{w})} F_{z_d} && \text{Discrete variables} \\
& \mathbf{w} \in F_w && \text{Dimensional variables} \\
\text{s.t.} \quad & \mathbf{g}(\mathbf{x},\mathbf{z},\mathbf{w}) \leq 0 \\
& g_i : F_{x_i}(\mathbf{w}) \times \prod_{d=1}^{n_{z_i}(\mathbf{w})} F_{z_{d_i}} \times F_w \rightarrow F_{g_i} \subseteq \mathbb{R} \\
& \text{for} \qquad i = 1, \ldots, n_g(\mathbf{w})
\end{aligned}
$$

Variable-size design space problems

Two alternative solutions are explored:

Variable-size design space problems

Two alternative solutions are explored:

Approach 1:
Separate optimization of several fixed-sized sub-problems:

- Budget allocation as a function of the surrogate model information
- Discarding of worst performing sub-problems

Variable-size design space problems

Two alternative solutions are explored:

Approach 1:
Separate optimization of several fixed-sized sub-problems:

- Budget allocation as a function of the surrogate model information
- Discarding of worst performing sub-problems

Approach 2:
Direct Bayesian optimization in the variable-size design space:

- Definition of a covariance kernel in the variable-size design space

Approach 1:

Strategy for the Optimization of Mixed Variable-Size design space Problems (SOMVSP)

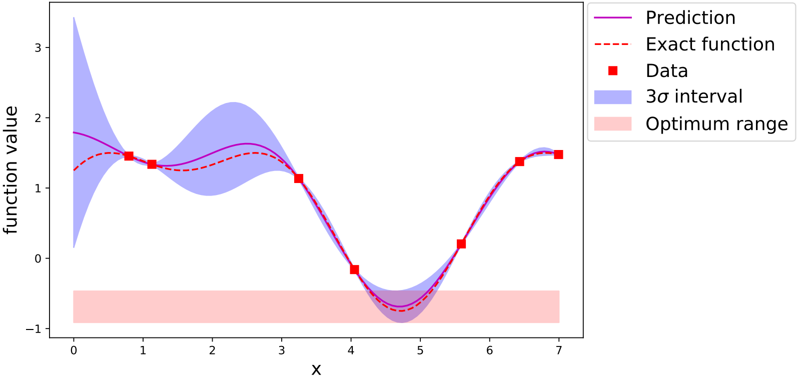
$$\begin{aligned}
\min \quad & f(\mathbf{x},\mathbf{z},\mathbf{w}_q) \qquad f:\mathbb{R}^{n_x(\mathbf{w}_q)} \times \prod_{d=1}^{n_z(\mathbf{w}_q)} F_{z_d} \times F_w \rightarrow F_f \subseteq \mathbb{R} \\
\text{w.r.t.} \quad & \mathbf{x} \in F_x(\mathbf{w}_q) \subseteq \mathbb{R}^{n_x(\mathbf{w}_q)} \\
& \mathbf{z} \in \prod_{d=1}^{n_z(\mathbf{w}_q)} F_{z_d} \\
\text{s.t.} \quad & \mathbf{g}(\mathbf{x},\mathbf{z},\mathbf{w}_q) \leq 0 \\
& g_i : F_{x_i}(\mathbf{w}_q) \times \prod_{d=1}^{n_{z_i}(\mathbf{w}_q)} F_{z_{d_i}} \times F_w \rightarrow F_{g_i} \subseteq \mathbb{R} \\
& \quad \text{for} \quad i = 1, \ldots, n_g(\mathbf{w}_q)
\end{aligned}$$

Separate optimization for every combinatorial value $\mathbf{w}_q$ of $\mathbf{w}$

SOMVSP

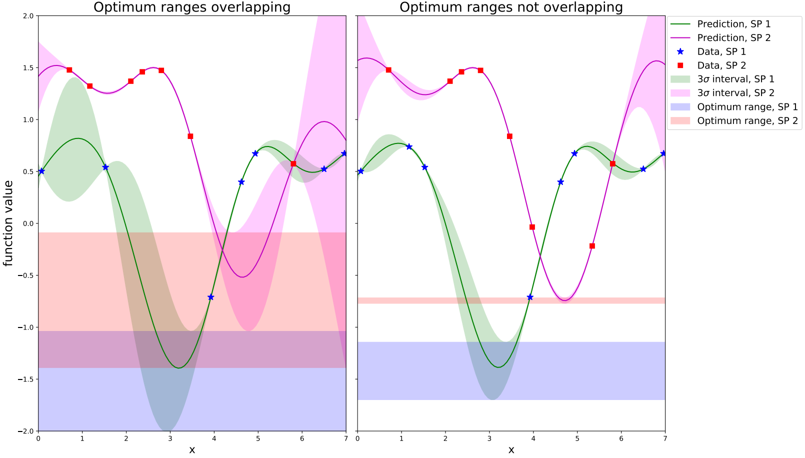
Step 1 : Predicted optimum range computation

For each sub-problem, the range between the best-case and worst-case feasible optimum is computed



- Optimization of $(\hat{y} - a\sigma)$ and $(\hat{y} + a\sigma)$

Step 2 : Sub-problem discarding



- A sub-problem can be discarded if the optimum ranges do not overlap

SOMVSP

Step 3 : Computational budget allocation

- Each remaining sub-problem is allocated a computational budget as a function of the predicted feasible optimum and its total dimension

SOMVSP

Step 3 : Computational budget allocation

- Each remaining sub-problem is allocated a computational budget as a function of the predicted feasible optimum and its total dimension

Step 4 : Optimization of remaining sub-problems

- Mixed-variable BO of each remaining sub-problem
- Number of infilled data samples proportional to the allocated computational budget

SOMVSP

Step 3 : Computational budget allocation

- Each remaining sub-problem is allocated a computational budget as a function of the predicted feasible optimum and its total dimension

Step 4 : Optimization of remaining sub-problems

- Mixed-variable BO of each remaining sub-problem
- Number of infilled data samples proportional to the allocated computational budget

↻ Iterated until exhaustion of the computational budget

Variable-size design space BO

Approach 2:
Direct BO of the variable-size design space problem

Necessity of defining a kernel in the variable-size design space:

$$k\left((\mathbf{x},\mathbf{z},\mathbf{w}),(\mathbf{x}',\mathbf{z}',\mathbf{w}')\right)$$

Computes the covariance between samples characterized by partially different sets of variables

Variable-size design space BO

Approach 2:
Direct BO of the variable-size design space problem

Necessity of defining a kernel in the variable-size design space:

$$k\left((\mathbf{x},\mathbf{z},\mathbf{w}),(\mathbf{x}',\mathbf{z}',\mathbf{w}')\right)$$

Computes the covariance between samples characterized by partially different sets of variables

2 proposed alternatives:

1. Sub-Problem-Wise decomposition
2. Dimensional Variable-Wise decomposition

Variable-size design space BO

Sub-Problem-Wise (SPW) decomposition kernel
Based on grouping the training samples as a function of the sub-problem they belong to

The kernel is decomposed in:

- Between sub-problems covariance
  - With respect to dimensional variables
- Within sub-problems covariance
  - With respect to continuous and discrete variables

$$k\left((\mathbf{x},\mathbf{z},w),(\mathbf{x}',\mathbf{z}',w')\right) = \sum_{q=1}^{N_p} k_{x_q}(\mathbf{x}_q,\mathbf{x}'_q)\cdot k_{z_q}(\mathbf{z}_q,\mathbf{z}'_q)\cdot \delta_q(w,w') + k_w(w,w')$$

where

$$\delta_q(w,w') = \begin{cases} 1 & \text{if} \quad w = w' = q \\ 0 & \text{else} \end{cases}$$

Variable-size design space BO

$$K = \begin{bmatrix} W_1 + B_{1,1} & B_{1,2} & \cdots & B_{1,N_p} \\ B_{2,1} & W_2 + B_{2,2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & B_{N_p-1,N_p} \\ B_{N_p,1} & \cdots & B_{N_p,N_p-1} & W_{N_p} + B_{N_p,N_p} \end{bmatrix}$$

W_q represents the within sub-problem covariance:

$$W_q = k_{x_q}(\mathbf{x}_q, \mathbf{x}'_q) \cdot k_{z_q}(\mathbf{z}_q, \mathbf{z}'_q)$$

$B_{q,p}$ represents the covariance between the sub-problems q and p:

$$B_{q,p} = k_w(w = q, w' = p)$$

Variable-size design space BO

Dimensional Variable-Wise (DVW) decomposition kernel
Based on grouping the variables as a function of the dimensional variable they depend on

A SPW approach can be relied on for each dimensional variable:

$$k\left((\mathbf{x},\mathbf{z},\mathbf{w}),(\mathbf{x}',\mathbf{z}',\mathbf{w}')\right) =$$
$$\prod_{d=1}^{n_w}\left(\sum_{l=1}^{l_{w_d}} k_{x_{d_l}}(\mathbf{x}_{d_l},\mathbf{x}'_{d_l})\cdot k_{z_{d_l}}(\mathbf{z}_{d_l},\mathbf{z}'_{d_l})\cdot\delta_l(w_d,w'_d) + k_{w_d}(w_d,w'_d)\right)$$

- n_w number of dimensional variables
- l_{w_d} number of levels associated to w_d

Variable-size design space BO

- Both SPW and DVW kernels are constructed through sums and product of kernels

- Their validity is ensured as long as the single continuous and discrete kernels are valid

Variable-size design space BO

Acquisition function optimization

Negligible infill criterion cost $\rightarrow$ the acquisition function can be separately optimized for each sub-problem:

$$\{\mathbf{x}^n, \mathbf{z}^n, \mathbf{w}^n\} = \text{argmax} \left\{ \begin{array}{ll} \text{argmax} & (EI(\mathbf{x}, \mathbf{z}, \mathbf{w}_q)) \\ \\ \text{s.t.} & EV\left(g_c(\mathbf{x}, \mathbf{z}, \mathbf{w}_q)\right) < t_c \quad \text{for} \quad c = 1, \ldots, n_g(\mathbf{w}_q) \\ \\ \text{w.r.t.} & \mathbf{x} \in F_x(\mathbf{w}_q) \subseteq \mathbb{R}^{n_x(\mathbf{w}_q)} \\ & \mathbf{z} \in \prod_{d=1}^{n_z(\mathbf{w}_q)} F_{z_d} \end{array} \right\}$$

$$\text{for} \quad q = 1, \ldots, N_p$$

- ✓ Simpler implementation
- ✓ Straightforward handling of constraints

Optimization performance comparison

Test-cases

Several analytical and engineering related test-cases of various complexity [Pelamatti 2020] :

Analytical test-case

- 5 continuous variables
- 4 discrete variables
- 648 equivalent continuous problems

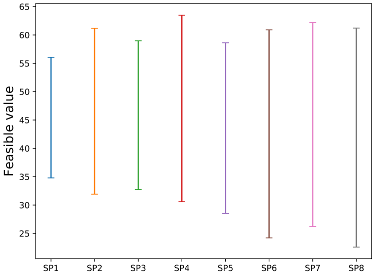
Engineering related test-case

- 14 continuous variables
- 12 discrete variables
- 19 constraints
- $\sim$ 30000 equivalent continuous problems

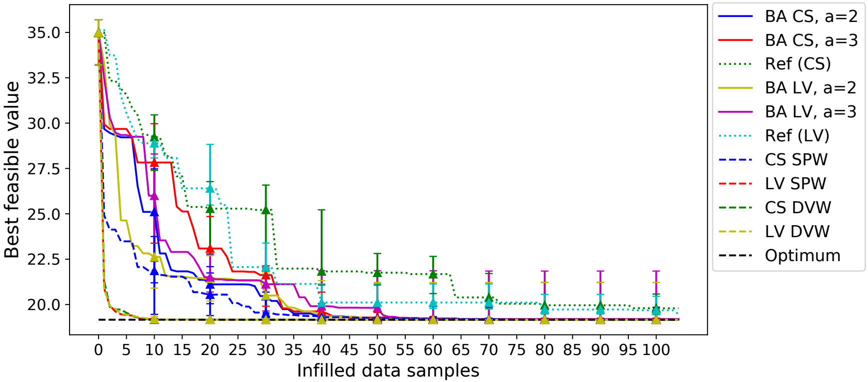
Applications

Variable-size design space Goldstein function

- 5 continuous variables
- 4 discrete variables
- 2 dimensional variables
- 8 sub-problems
- 648 equivalent continuous problems
- 1 constraint
- Initial data set size: 104 samples
- Number of infilled samples: 104

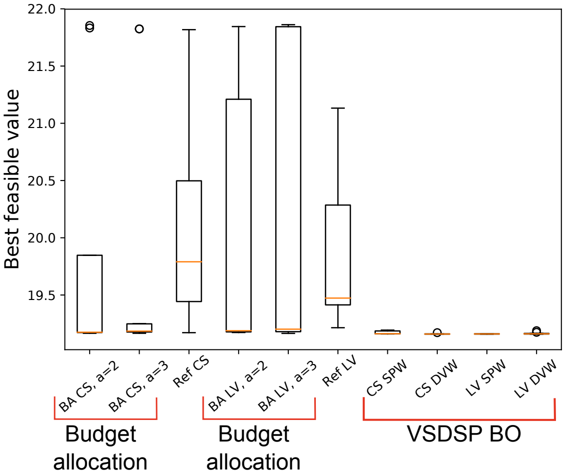


Applications



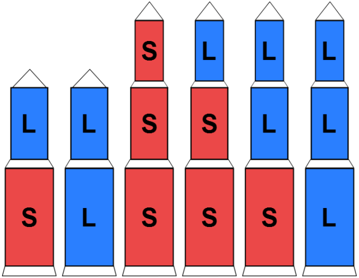
- Both proposed algorithms provide faster convergence w.r.t., the reference approach
- Direct BO yields better results if compared to the budget allocation

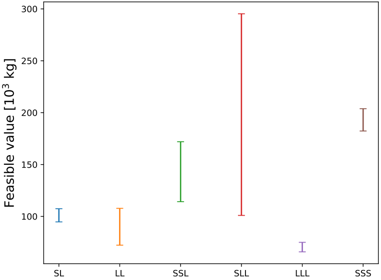
Applications



- Budget allocation characterized by a larger result variance because of premature sub-problem discarding

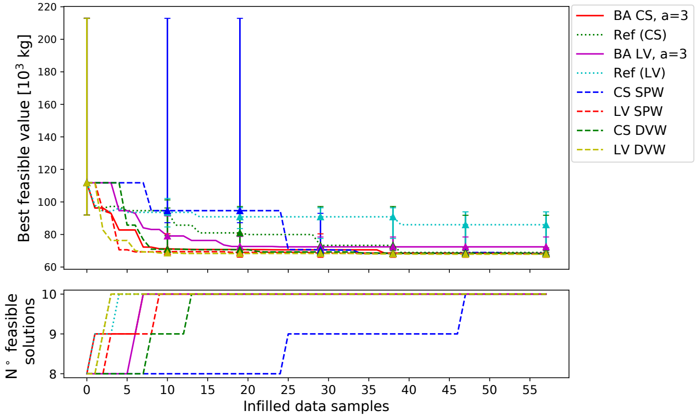
Applications: Multi-stage launch-vehicle architecture





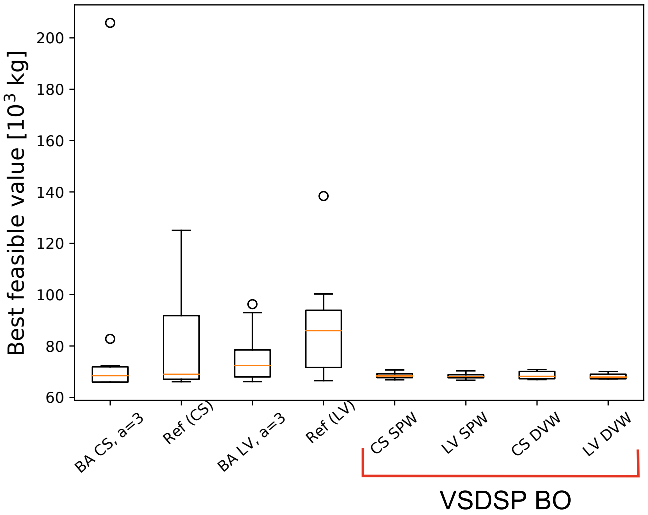
Sub-problem	SL	LL	SSS	SSL	SLL	LLL
N° continuous variables	6	4	12	10	8	6
N° discrete variables	5	3	10	8	6	4
N° discrete categories	48	32	27648	1152	192	64
N° constraints	8	3	19	14	9	4

Applications



- Both proposed algorithms provide faster convergence w.r.t., the reference approach
- Both proposed algorithms behave similarly

Applications



- Budget allocation characterized by a larger result variance

VSDSP optimization synthesis

- Two proposed approaches for the optimization of variable-size design space problems

VSDSP optimization synthesis

- Two proposed approaches for the optimization of variable-size design space problems
- Considerably faster convergence if compared to the independent optimization of each sub-problem

VSDSP optimization synthesis

- Two proposed approaches for the optimization of variable-size design space problems
- Considerably faster convergence if compared to the independent optimization of each sub-problem
- The variable-size design space BO provides faster convergence and more robust results if compared to the SOMVSP

VSDSP optimization synthesis

- Two proposed approaches for the optimization of variable-size design space problems
- Considerably faster convergence if compared to the independent optimization of each sub-problem
- The variable-size design space BO provides faster convergence and more robust results if compared to the SOMVSP
- Black-box optimization algorithm

VSDSP optimization synthesis

- Two proposed approaches for the optimization of variable-size design space problems
- Considerably faster convergence if compared to the independent optimization of each sub-problem
- The variable-size design space BO provides faster convergence and more robust results if compared to the SOMVSP
- Black-box optimization algorithm
- Promising for the preliminary design of computationally intensive variable-size design space problems

VSDSP optimization synthesis

- Two proposed approaches for the optimization of variable-size design space problems
- Considerably faster convergence if compared to the independent optimization of each sub-problem
- The variable-size design space BO provides faster convergence and more robust results if compared to the SOMVSP
- Black-box optimization algorithm
- Promising for the preliminary design of computationally intensive variable-size design space problems

✗ Curse of dimension & Hyperparameter scaling

Perspectives

- Sensitivity analysis to identify most influential variables -> How?

Perspectives

- Sensitivity analysis to identify most influential variables -> How?
- Application to multi-fidelity problems

Perspectives

- Sensitivity analysis to identify most influential variables -> How?
- Application to multi-fidelity problems
- Extension of proposed methods to multi-objective optimization

Perspectives

- Sensitivity analysis to identify most influential variables -> How?

- Application to multi-fidelity problems
- Extension of proposed methods to multi-objective optimization
- Integration of the proposed methods within MDO framework

Perspectives

- Sensitivity analysis to identify most influential variables -> How?
- Application to multi-fidelity problems
- Extension of proposed methods to multi-objective optimization
- Integration of the proposed methods within MDO framework
- Coupling of the proposed methods with local optimization algorithms for solution refinement


"That's all Folks!"

References

- D. R. Jones, M. Schonlau, and W. J. Welch. *Efficient Global Optimization of Expensive Black-Box Functions*. Journal of Global Optimization, 1998.
- O. Roustant, E. Padonou, Y. Deville, A. Clément, G. Perrin, J. Giorla, and H. P. Wynn. *Group kernels for Gaussian process metamodels with categorical inputs*. working paper or preprint, arXiv:1802.02368, 2018.
- Q. Zhou, P. Z. G. Qian, and S. Zhou. *A Simple Approach to Emulation for Computer Models With Qualitative and Quantitative Factors*. Technometrics, 2011.
- M. Schonlau, W. J. Welch, and D. R. Jones. *Global versus local search in constrained optimization of computer models*. Lecture Notes-Monograph Series, 1998.
- C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. MIT Press, 2006.
- J. Pinheiro and D. M. Bates. *Unconstrained parametrizations for variance-covariance matrices*. Statistics and Computing, 1996
- J. Pelamatti, L. Brevault, M. Balesdent, E.-G. Talbi, and Y. Guerin. *Overview and comparison of Gaussian process-based surrogate models for mixed continuous and discrete variables, application on aerospace design problems* (Publication pending). High-performance simulation based optimization, Springer series on Computational Intelligence, 2018.
- J. Pelamatti, L. Brevault, M. Balesdent, E.-G. Talbi, and Y. Guerin. *Efficient global optimization of constrained mixed variable problems* , Journal of Global Optimization, 2018.
- Y. Zhang, S. Tao, W. Chen et D. W. Apley, *A Latent Variable Approach to Gaussian Process Modeling with Qualitative and Quantitative Factors*, pp. 1-36 (2018).
- J. Pelamatti, L. Brevault, M. Balesdent, E.-G. Talbi, and Y. Guerin. *Bayesian optimization of variable-size design space problems*, Optimization & Engineering, 2020.
- Hutter, F. *Automated configuration of algorithms for solving hard computational problems*, Doctoral dissertation, University of British Columbia, 2009.