PUBO$_i$: a tunable benchmark with variable importance

Sara Tari, Sébastien Verel, and Mahmoud Omidvar

Laboratoire d'Informatique, Signal et Image de la Côte d'opale (LISIC)
Université du Littoral Côte d'Opale, Calais, France
http://www-lisic.univ-littoral.fr/~verel/

EvoCOP conference,
April, 20, 2022

Motivation

How to design efficient optimization algorithm
according to the properties of the instance ?

Benchmark oriented design :

- Create set of diverse instances with relevant properties
- Train, and test algorithms (components, and parameters) :
  Machine/Human Learning approach
- Improve the understanding of optimization algorithms :
  Using fitness landscape analysis, or other techniques

Another benchmark ?

Another benchmark ?

Yes !

Benchmarking is an open ending research field :
we corroborate (or not) research hypothesis using benchmark

PUBO$_i$ benchmark :
Polynomial Unconstrained Binary Optimization
with variable importance

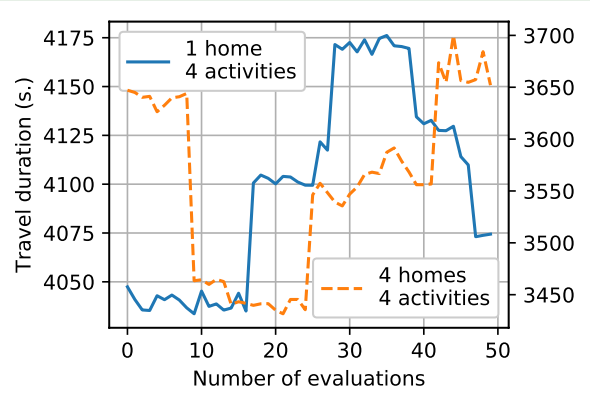
- Based on Walsh functions,
  orthogonal basis of pseudo-boolean functions
- With variable importance,
  non "isotropic", real-like property, local search operator
- A bridge to a larger research community,
  Quantum Computing

Variable importance

Intuitively, in real-world problem, some variables are more impactful on the objective value than others.

Example : Traffic light problem (Lepretre *et al.* 2019 [5])

Importance degree of variable i : $\delta_i = |f(mutate_i(x)) - f(x)|$
Estimation based on random walk on fitness landscape



Important crossroads in red

Huge effort in combinatorial benchmarking

- Classic benchmarks :
  flow shop, job shop and scheduling problems (Taillard [7])
  QAPLib : artificial, and real-world (few), and real-like
  Dimacs instances [], nk-landscapes, etc.
- Black-Box Optimization Benchmarking (BBOB) [4]
  COCO plateform : single, multiobjective, mixed integer problems
- IOHprofiler [1]
  23 real-valued (continuous),
  25 academic pseudo-boolean problems

Each benchmark have is own relevance :

Compare algorithms on typical problems, compare difficulty of instances, real-world problems, etc.

Benchmark for quantum computers

Quantum computers

Available architectures, and/or algorithms allow to solve Quadratic Unconstrained Binary Optimization problems (QUBO/UBQP) :

$$\mathcal{H}(s) = \sum_{i=1}^{n} J_i s_i + \sum_{i,j=1}^{n} J_{ij} s_i s_j \qquad \text{with } s \in \{-1, 1\}^n$$

See other evoCOP talks

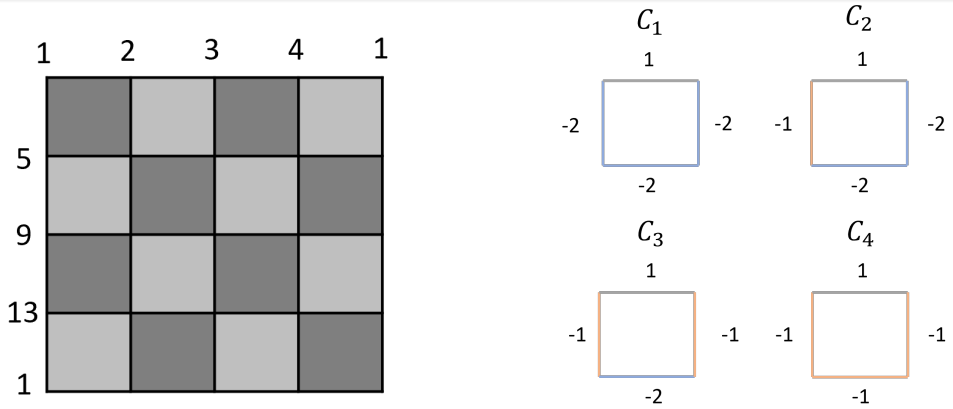
Chook generator

Tile Planting instances (Perrera *et al.* 2020, PRE 2018 [6, 3])

QUBO with tunable degree of difficulty, and know global optimum

$$\mathcal{H}(s) = \sum_{\ell=1}^{m} \mathcal{H}_\ell(s)$$

Variables of $\mathcal{H}_\ell$: a Tile ; $\mathcal{H}_\ell \in C_i$ according to prob. p_i



C_i : quadratic, i frustrated states (loc. opt.), and 1111 as optimum

Proportion p_1 *vs.* p_j tunes the difficulty, but :

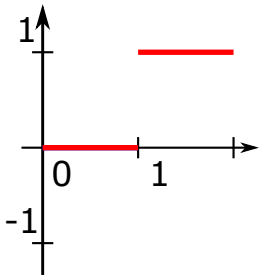
- Nearly isotropic, limited interdependence between subproblems
- Polynomial time algorithm can solve instances [2]

Walsh functions, QUBO/UBQP, PUBO

- Space pseudo-boolean function is a vector space, $\{f : \{0,1\}^n \to \mathbb{R}\}$
- Basis : multi-linear functions, $x_{k_1} \dots x_{k_\ell}$ [Baptista, Poloczek, BOCS, ICML 2018]

Multi-linear :

$n = 1$, $\psi_1(x) = x$

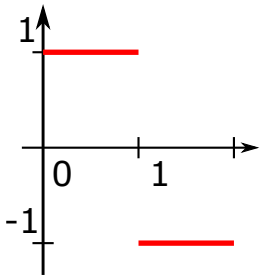


Orthonormal : No

x	ψ_0	ψ_1
0	1	0
1	1	1

Walsh :

$n = 1$, $\varphi_1(x) = (-1)^x$



Orthonormal : Yes

x	φ_0	φ_1
0	1	1
1	1	-1

Extension to dimension n using tensorial product :

$$\psi_{k_1 \dots k_\ell}(x) = x_{k_1} \dots x_{k_\ell} \qquad \varphi_{k_1 \dots k_\ell}(x) = (-1)^{x_{k_1}} \dots (-1)^{x_{k_\ell}}$$

Variable interaction in Walsh functions

Walsh functions

$\varphi_k(s) = \prod_{i:k_i=1} s_i$ with $s_i = (-1)^{x_i}$

order : number of 1 in the binary representation of k (degree)

Example of order 2

$$w_0^{(0)} + \sum_{i=0}^{n} w_i^{(1)} s_i + \sum_{i<j} w_{i,j}^{(2)} s_i s_j$$

Why Walsh functions ?

Explicit algebraic model (not black-box), easy to interpret

- Interdependence between the variables (non-zero terms)
- Intensity of interaction $|w_{i,j}|$
- Neutrality levels (plateaus) : integer for w_k

PUBO$_i$ principle

As Tile Planting, sum of m sub-functions (sub-problems, clauses) :

$$\forall x \in \{0,1\}^n, \quad f(x) = \sum_{i=1}^{m} f_i(x)$$

the class of f_i is selected at random, probabilities p_j of class C_j

Originality : selection of variables

$f_i(s) = -s_0 s_1 - s_1 s_2 - s_2 s_3 + s_3 s_0$ depends on 4 variables
How to select variable in each sub-function ?

Notice that, derivative of f : $\nabla_j f_i(x) = \sum_{j \in i} \nabla_j f_i(x)$
i.e. x_j is impactful
when the variable x_j is more frequent in sub-functions.

Variable importance in PUBO$_i$

Importance classes

k disjoint classes of importance $c_i \subset X = \{x_1, \ldots, x_n\}$ such that $\cup_k c_k = X$, and $c_i \cap c_j = \emptyset \ \forall \{i, j\}$.

n_i : number of variables in class c_i

Degree of importance

$d_i \in \mathbb{R}^+$: degree of importance of class c_i

Probability of selecting a variable $x \in c_i$ in each sub-function :

$$p_{c_i} = \frac{d_i}{\sum_{j=1}^{k} d_j}$$

Co-appearance of important variables

Independent co-appearance

If random selection is independent, only bias by degree of importance :

$$P(x_{i_1} \in c_{i_1}, \ldots, x_{i_a} \in c_{i_a}) = p_{c_{i_1}} \ldots p_{c_{i_a}}$$

then every sub-functions are similar, the problem is "isotropic".
In r-w problems, important variables should not be randomly distributed

Co-appearance parametrisation

Suppose there is only 2 classes of importance : c_0, and c_1

$p_i^{(a)}$: prob. of having i var. of class 1 in the same sub-function of arity a

Arity $a = 1$, $f_k(x_{i_1})$

$p_0^{(1)} + p_1^{(1)} = 1$. Thus, $p_0^{(1)} = p_{c_0}$, and $p_1^{(1)} = 1 - p_{c_0}$.

Co-appearance of important variables (2)

Arity $a = 2$, $f_k(x_{i_1}, x_{i_2})$

Prob. to select c_0 should remain the same, same marginal probability :

$$\begin{cases} p_0^{(2)} + p_1^{(2)} = p_0^{(1)} \\ p_1^{(2)} + p_2^{(2)} = p_1^{(1)} \end{cases}$$

By setting $p_0^{(2)} = p'_{c_0} p_0^{(1)}$:

$$\begin{cases} p_0^{(2)} = p'_{c_0} p_{c_0} \\ p_1^{(2)} = (1 - p'_{c_0}) p_{c_0} \\ p_2^{(2)} = (1 - p'_{c_0})(1 - p_{c_0}) + p'_{c_0} - p_{c_0} \end{cases}$$

$p'_{c_0} = \alpha p_{c_0}$: independence degree of co-appearance of the same class

Arity a, $f_k(x_{i_1}, x_{i_2}, \ldots, x_{i_a})$

$\forall i,\ p_i^{(a)} + p_{i+1}^{(a)} = p_i^{(a-1)}$

By setting $p_0^{(a)} = p'_{c_0} p_0^{(a-1)}$: $p_0^{(a)} = (p'_{c_0})^{a-1} p_{c_0}$, etc.

more than 2 classes, inclusion/exclusion principle : $x_i \in c_0$, or $x_i \notin c_0$,...

Instances set of PUBO$_i$ benchmark

	Description	Experimental values
n	Problem dimension	$[1000, 5000]$
m	Number of sub-functions	$[0.01, 0.2] \times \frac{n(n-1)}{2}$
$\mathcal{C}$	Portfolio of sub-functions	Tile Planting
p_i	Probabilities of sub-function class	$[0, 1]$
k	Number of class of variable importance	2
n_i	Num. of var. in each class of importance	$n_0 = 0.25n$, $n_1 = n - n_0$
d_i	Degree of importance of each class	$d_0 \in [1, 10]$, $d_1 = 1$
α	Prob. of importance class co-appearance	$[1, 1/(p_{c_0} - 1)]$

Design of experiments

Factorial design is poor (coverage, and size),
1000 instances using Latin Hypercube Sampling (LHS) design.

Reject samples which do not respect constraints (avoid scaling bias)

Sources

Code of generator, of design, and instances :

`https://gitlab.com/verel/pubo-importance-benchmark`

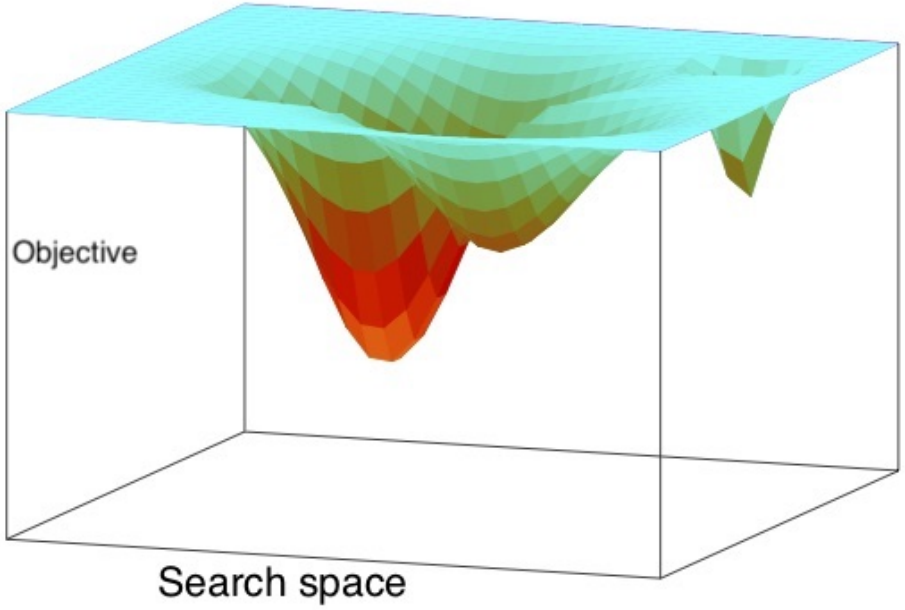
Methodology

- We do not use algorithm performance (to avoid bias analysis),
- Fitness landscape analysis
- Contrast benchmark parameters, in particular related to importance, with basic features of fitness landscape
- Using Generalized Additive Model (GAM) :
  $y = f(x) + \epsilon = \sum_{j=1}^{d} \beta_j B_j(x) + \epsilon$
  where B_j basis functions (splines for example)

Fitness landscape

Fitness landscape (Wright 1920)

- $\mathcal{S}$: set of candidate solutions, search space
- $f : \mathcal{S} \rightarrow \mathbb{R}$: objective function
- $\mathcal{N} : \mathcal{S} \rightarrow 2^{\mathcal{S}}$, neighborhood relation between solutions



- Geometry of the fitness landscape :
  Features/metrics are correlated to algorithm performance

Neighborhood : 1-bit mutation

Three main features of fitness landscapes

Features estimation based on random walk :
$(x_1, x_2, \ldots, x_\ell)$ s.t. $x_{t+1} \in \mathcal{N}(x_t)$

Ruggedness

Local "non-regularity"

metric :
autocorr. length

$$\rho(n) = \frac{\mathrm{E}[(f(x_t)-\bar{f})(f(x_{t+n})-\bar{f})]}{\mathrm{var}(f(x_t))}$$

Neutrality

Random mutation, plateaus

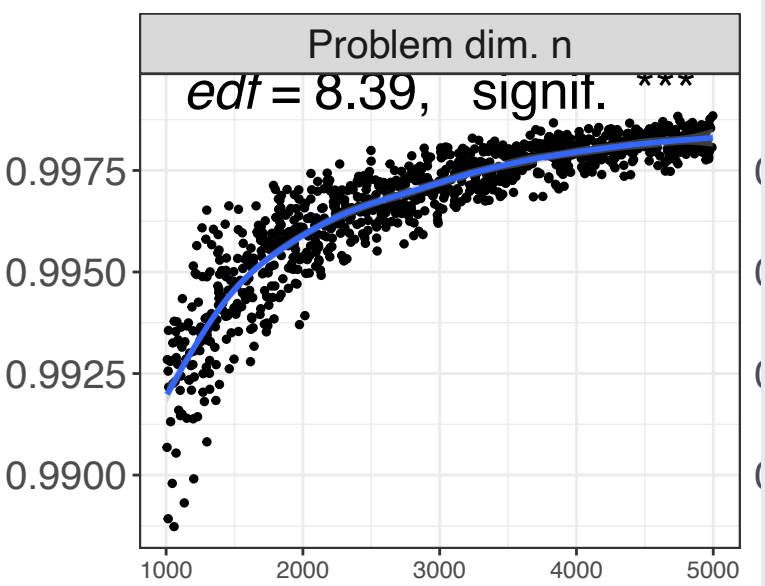
metric :
neutral rate

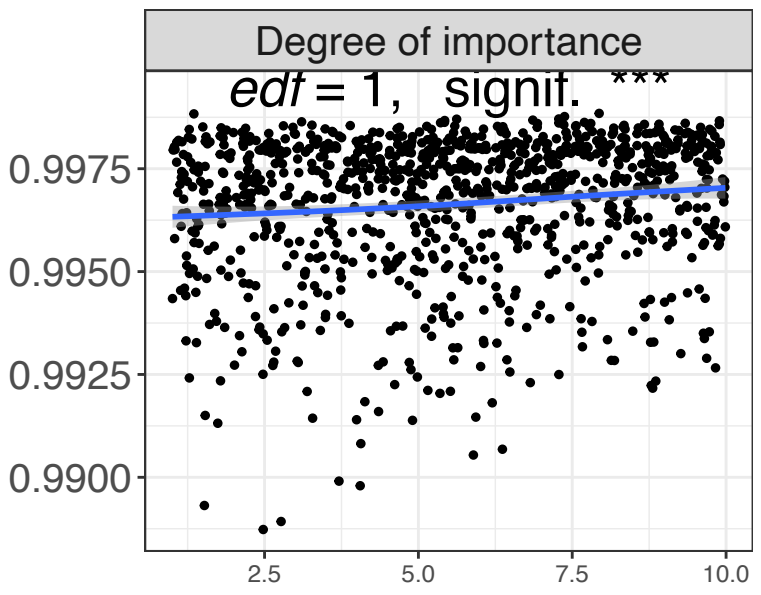
$$\frac{\sharp\{(x_t,x_{t+1}) \,:\, f(x_t)=f(x_{t+1}),\ t\in\{1,\ell-1\}\}}{\ell-1}$$

Multimodality : Adaptive walk length ℓ

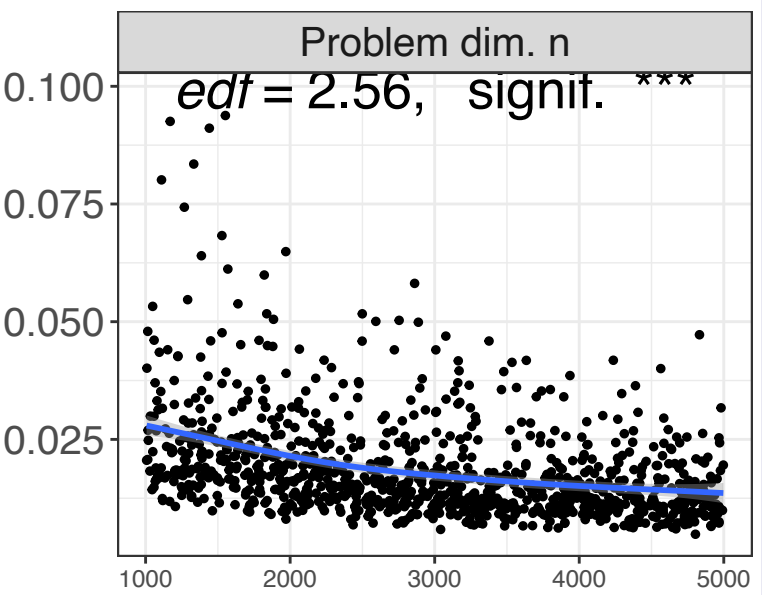
$(x_1, x_2, \ldots, x_\ell)$ s.t. $x_{t+1} \in \mathcal{N}(x_t)$, and $f(x_{i+1}) < f(x_i)$

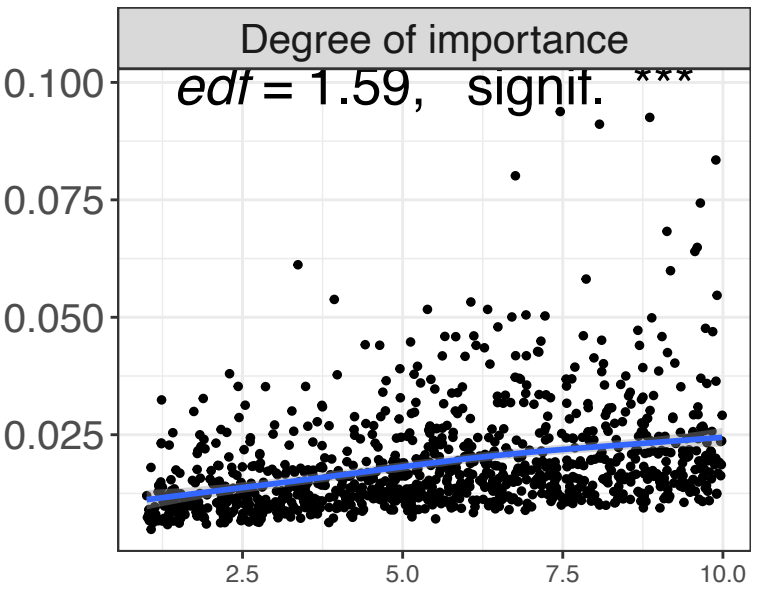
Autocorrelation



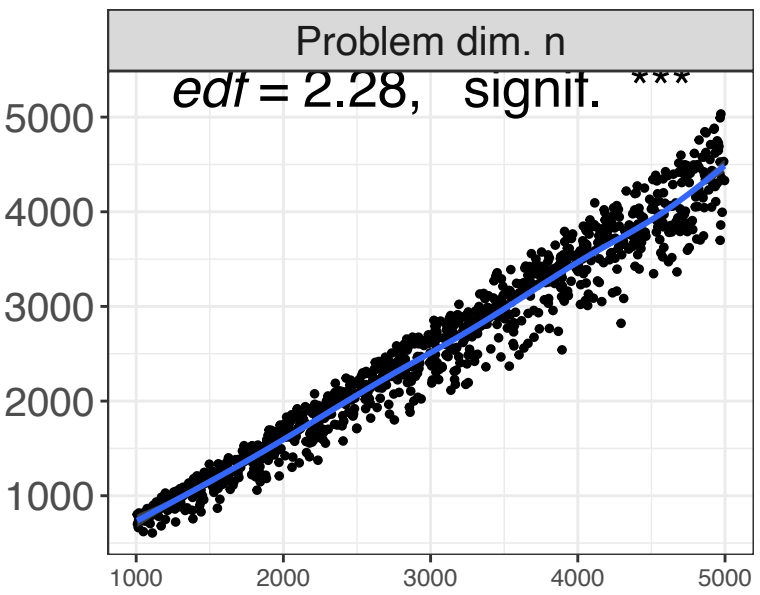


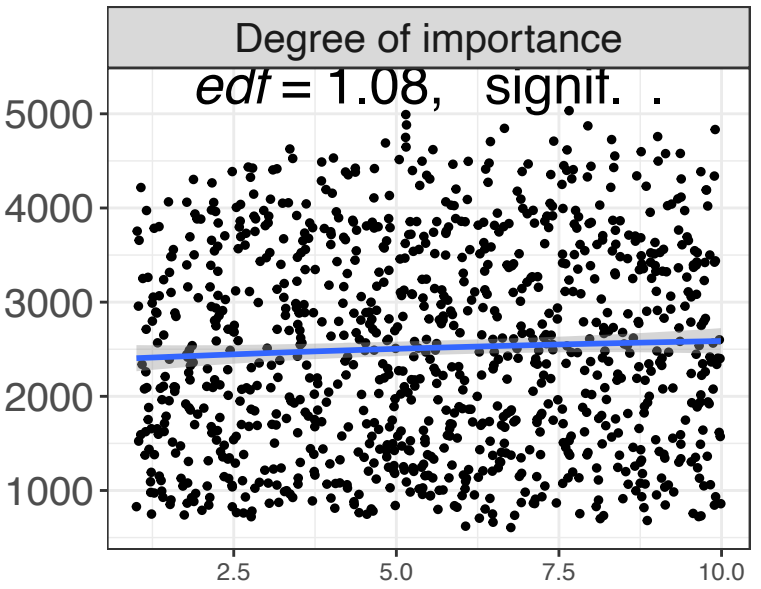
Neutrality


Problem dim. n
edf = 2.56, signif. ***
0.100
0.075
0.050
0.025
1000
2000
3000
4000
5000


Degree of importance
edf = 1.59, signif. ***
0.100
0.075
0.050
0.025
2.5
5.0
7.5
10.0

Length of adaptive walk





Summary of results

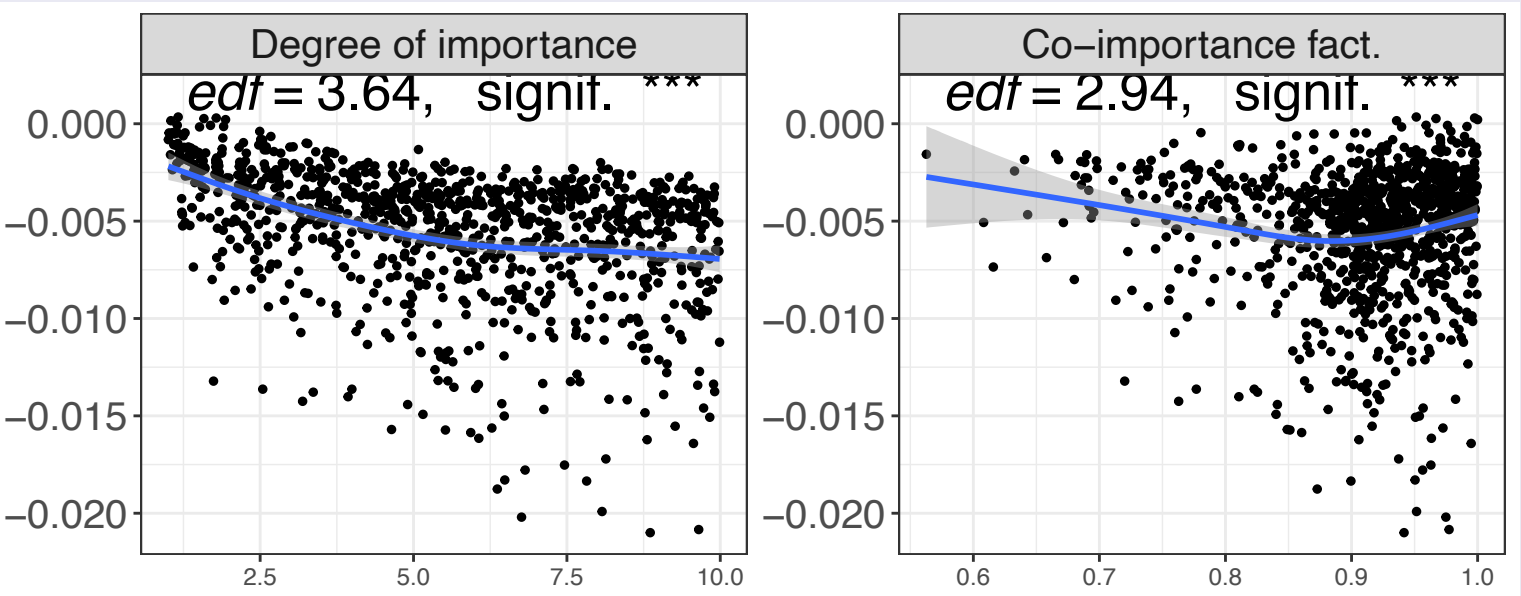
Param.	Autocor.	Neutrality	Adapt. length
n	8.39 (***)	2.56 (***)	2.28 (***)
m	1 (***)	6.78 (***)	2.81 (***)
d_0	1 (***)	1.59 (***)	1.08 (.)
α	2.56 (**)	2.68 (***)	3.33 (*)
p_1	1 (***)	2.19 (*)	1 (-)
p_2	1 (-)	1 (-)	1 (-)
p_3	1 (-)	1 (-)	2.96 (-)
p_4	1 (*)	1.71 (***)	1 (-)

- Main parameters tune the difficulty
- p_i not very impactful
- Adaptive length less less correlated

Important *vs.* non-important variables

Autocor., and neutrality can be computed individually on each variable : features difference between important, and non-important variables.

Autocorrelation

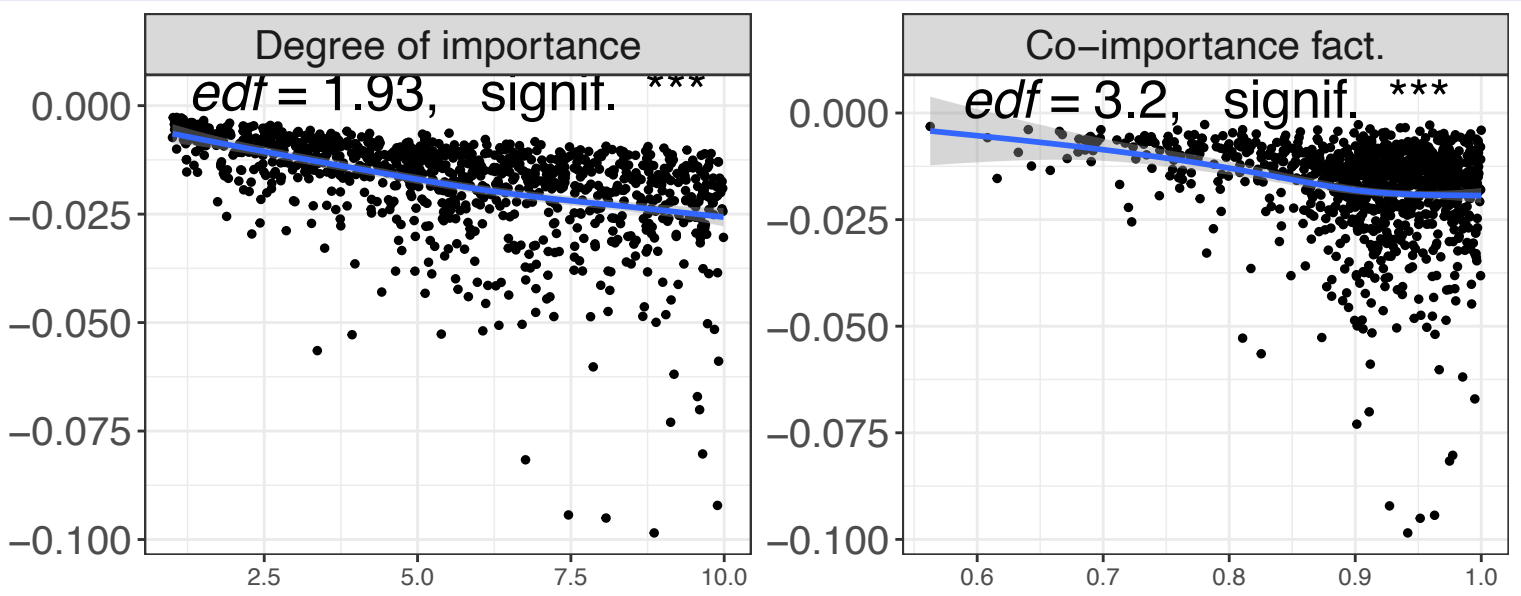


Subspace of important is more rugged, and less flat

Important *vs.* non-important variables

Autocor., and neutrality can be computed individually on each variable : features difference between important, and non-important variables.

Neutrality



Subspace of important is more rugged, and less flat

Summary of important *vs.* non-important

Param.	Autocor.	Neutrality
n	4.13 (***)	2.47 (***)
m	2.64 (***)	6.7 (***)
d_0	3.64 (***)	1.93 (***)
α	2.94 (***)	3.02 (***)
p_1	2.02 (.)	2.21 (*)
p_2	1 (-)	1 (-)
p_3	2.67 (-)	1.9 (-)
p_4	1 (*)	1.78 (**)

- Main parameters tune the difference
- p_i not very impactful

Conclusions

Summary

- Except for p_i (portfolio) parameters, all of PUBO$_i$ parameters significant impact on ruggedness, multimodality and neutrality levels of landscapes.
- Variable importance parameters could have the same impact on landscape than classical parameters
- Non isotropic landscapes where the features of landscapes are different for the subspace of important variables.

Consequences

Difference between important and non-important variables :

- Should be considered in the design of EA, and LS,
- Local search operator, and neighborhood should be designed according to the variable importance
- Large set of instances to learn/test new ideas

Perspectives

- Compare real-world problems to PUBO$_i$ instances,
- Study other possible benchmark parameters :
  number of importance classes, the portfolio, etc.
- New analysis should be conducted :
  Local Optima Network
- Design new fitness landscape tools for anisotropy
- Train, test, and understand new optimization algorithms :
  quantum or classical
- Extend the generator to other type of optimization problems :
  continuous, etc.

Sources

Code of generator, of design, and instances :

`https://gitlab.com/verel/pubo-importance-benchmark`

Carola Doerr, Furong Ye, Naama Horesh, Hao Wang, Ofer M. Shir, and Thomas BÃ¤ck.
Benchmarking discrete optimization heuristics with iohprofiler.
Applied Soft Computing, 88 :106027, 2020.

Anna Galluccio, Martin Loebl, and Jan Vondrák.
Optimization via enumeration : a new algorithm for the max cut problem.
Mathematical Programming, 90(2) :273–290, 2001.

Firas Hamze, Darryl C Jacob, Andrew J Ochoa, Dilina Perera, Wenlong Wang, and Helmut G Katzgraber.
From near to eternity : spin-glass planting, tiling puzzles, and constraint-satisfaction problems.
Physical Review E, 97(4) :043303, 2018.

Nikolaus Hansen, Anne Auger, Raymond Ros, Steffen Finck, and Petr Pošík.

Comparing results of 31 algorithms from the black-box optimization benchmarking bbob-2009.
In *GECCO*, pages 1689–1696, 2010.

Florian Leprêtre, Cyril Fonlupt, Sébastien Verel, Virginie Marion, Rolando Armas, Hernán Aguirre, and Kiyoshi Tanaka.
Fitness landscapes analysis and adaptive algorithms design for traffic lights optimization on sialac benchmark.
Applied Soft Computing, 85 :105869, 2019.

Dilina Perera, Inimfon Akpabio, Firas Hamze, Salvatore Mandra, Nathan Rose, Maliheh Aramon, and Helmut G Katzgraber.
Chook–a comprehensive suite for generating binary optimization problems with planted solutions.
arXiv preprint arXiv :2005.14344, 2020.

Eric Taillard.
Benchmarks for basic scheduling problems.

european journal of operational research, 64(2) :278–285, 1993.