



**HAL**  
open science

# Surrogate-assisted Multi-objective Combinatorial Optimization based on Decomposition and Walsh Basis

Geoffrey Pruvost, Bilel Derbel, Arnaud Liefoghe, Sébastien Verel, Qingfu Zhang

► **To cite this version:**

Geoffrey Pruvost, Bilel Derbel, Arnaud Liefoghe, Sébastien Verel, Qingfu Zhang. Surrogate-assisted Multi-objective Combinatorial Optimization based on Decomposition and Walsh Basis. GECCO '20 - Genetic and Evolutionary Computation Conference, Jul 2020, Cancun, Mexico. 10.1145/3377930.3390149 . hal-02898952

**HAL Id: hal-02898952**

**<https://inria.hal.science/hal-02898952v1>**

Submitted on 15 Jul 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Surrogate-assisted Multi-objective Combinatorial Optimization based on Decomposition and Walsh Basis

Geoffrey Pruvost  
Univ. Lille, CNRS, Inria, CRISTAL  
Lille, France  
geoffrey.pruvost@univ-lille.fr

Bilel Derbel  
Univ. Lille, CNRS, Inria, CRISTAL  
Lille, France  
bilel.derbel@univ-lille.fr

Arnaud Liefooghe  
JFLI, CNRS, University of Tokyo  
Tokyo, Japan  
arnaud.liefooghe@univ-lille.fr

Sébastien Verel  
Univ. Littoral Côte d'Opale, UR 4491,  
LISIC, F-62100 Calais, France  
verel@univ-littoral.fr

Qingfu Zhang  
City University Hong Kong  
Kowloon Tong, Hong Kong  
qingfu.zhang@cityu.edu.hk

## ABSTRACT

We consider the design and analysis of surrogate-assisted algorithms for expensive multi-objective combinatorial optimization. Focusing on pseudo-boolean functions, we leverage existing techniques based on Walsh basis to operate under the decomposition framework of MOEA/D. We investigate two design components for the cheap generation of a promising pool of offspring and the actual selection of one solution for expensive evaluation. We propose different variants, ranging from a filtering approach that selects the most promising solution at each iteration by using the constructed Walsh surrogates to discriminate between a pool of offspring generated by variation, to a substitution approach that selects a solution to evaluate by optimizing the Walsh surrogates in a multi-objective manner. Considering bi-objective NK landscapes as benchmark problems offering different degree of non-linearity, we conduct a comprehensive empirical analysis including the properties of the achievable approximation sets, the anytime performance, and the impact of the order used to train the Walsh surrogates. Our empirical findings show that, although our surrogate-assisted design is effective, the optimal integration of Walsh models within a multi-objective evolutionary search process gives rise to particular questions for which different trade-off answers can be obtained.

## CCS CONCEPTS

• **Theory of computation** → *Discrete optimization*.

## KEYWORDS

Multi-objective optimization, discrete surrogates, decomposition

### ACM Reference Format:

Geoffrey Pruvost, Bilel Derbel, Arnaud Liefooghe, Sébastien Verel, and Qingfu Zhang. 2020. Surrogate-assisted Multi-objective Combinatorial Optimization based on Decomposition and Walsh Basis. In *Genetic and Evolutionary*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

GECCO '20, July 8–12, 2020, Cancún, Mexico

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-7128-5/20/07...\$15.00

<https://doi.org/10.1145/3377930.3390149>

Computation Conference (GECCO '20), July 8–12, 2020, Cancún, Mexico. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3377930.3390149>

## 1 INTRODUCTION

### 1.1 Context and Goal

Evolutionary algorithms (EAs) have been proved to be effective in solving complex black-box optimization problems coming from various application fields, e.g., in multi-disciplinary engineering design, complex system tuning, etc. In such cases, evaluating the quality of a candidate solution is the result of an (external) simulation process. Unfortunately, it is also often the case that evaluating a solution implies heavy and CPU-intensive computations, hence restricting drastically the overall budget that one can afford to few hundreds or few thousands of solution evaluations. Surrogates, or meta-models, can then be used to assist the optimization process, by providing a cheap alternative to the expensive evaluation, or by providing information about the regions of the search space that are worth to be explored. The goal is to optimize the use of the overall affordable budget in order to find a solution with the highest possible quality. In this paper, we are specifically interested in developing surrogate-assisted EAs dealing with expensive *multi-objective combinatorial* optimization problems (MCOPs).

On the one hand, a MCOP consists in at least two objective functions being mutually conflicting. Our goal is to compute a set of solutions providing different optimal trade-offs among the objectives. Hence, in addition to the expensive nature of the problem, other challenges related to the target approximation set have to be addressed. It is worth noticing that there exist a number of surrogate-assisted multi-objective approaches dealing with continuous domains. In fact, leveraging multi-objective evolutionary algorithms (MOEAs) so that they can operate in a combined manner with well-established surrogates techniques, in particular those from expensive numerical single-objective optimization, has attracted a lot of attention [6, 13, 21, 22]. However, there is still a lack of general guidelines that can inform in a comprehensive manner about the best options that should be chosen for an efficient design.

On the other hand, compared to the literature dealing specifically with continuous variables, relatively few studies exist w.r.t. surrogate-assisted techniques for combinatorial optimization problems (COPs), i.e., discrete variables. This can be attributed to the fact that most well-established surrogate techniques find their

foundations in the continuous domain, and have then been developed under such an assumption. Hopefully, there exist a number of recent attempts to develop meta-models for single-objective COPs [1, 2, 10, 12, 20]. In particular, we are interested in binary variables, where a candidate solution is a bit-string which, e.g., model the presence or absence of a component when dealing with expensive complex system design [15].

Putting the pieces together, our main motivation stems from the lack of investigations on the smooth and flexible combination of the state-of-the-art techniques in *both* combinatorial surrogates and expensive multi-objective optimization. Consequently, our goal is to contribute to pushing a step forward the setting up of the first surrogate-assisted MOEAs for MCOPs, and to provide some insights about the challenges and the key ingredients for a successful design of such algorithms. Despite its novelty, one can find a number of work related to the subject, that we shall rely on in the rest of this paper, and which are highlighted briefly in the following.

## 1.2 Related Work and Contribution Overview

Independently of the problem being discrete or continuous, and single- or multi-objective, surrogates require to build a model using a sample of solutions, for which the true (expensive) evaluation is already known. The model can be used to help sampling new promising solution(s) that are then evaluated using the real (costly) objective(s). The model is trained again by incorporating the newly evaluated solution(s), and so on.

*Discrete Surrogates.* We are aware of four models proposed for single-objective expensive COPs, namely, Radial basis function (RBF) [12], Kriging [20], Bayesian [1], and Walsh [17] models. For example, Kriging [20] is based on a ‘standard’ Gaussian process, where in the context of a discrete binary representation, the Euclidean distance is replaced by the Hamming distance. This is then embedded within the well-known Numerical Efficient Global Optimizer (EGO) framework, where the solution optimizing some acquisition function, e.g., the Expected Improvement (EI), is selected for true evaluation. Bayesian Optimization of Combinatorial Structures (BOCS) [1] is another recent state-of-art surrogate-assisted algorithm using a multi-linear polynomial model for binary variables. It was shown that BOCS is able to outperform Kriging, which is itself better than RBF, for dimensions up to 25 variables. These first three models can be viewed as extensions of statistical and machine learning numerical models, which raise some issues about their adaptation to combinatorial spaces.

Interestingly, the model recently proposed in [17] uses a Walsh basis [19], which can be employed to decompose any function of the Hilbert space. Hence, they can be naturally used as basis for the space of pseudo-boolean problems. The accuracy of a Walsh-based surrogate when fitting various single-objective pseudo-boolean landscapes was demonstrated in [17]. It was then extended to a single-objective surrogate-assisted EA in [10] by using an efficient gray-box solver [5] to compute the solution optimizing the underlying single-objective trained model. In particular, it was shown to be competitive compared against BOCS, Kriging, and RBF when scaling up to a dimensionality of 100. In our work, we hence choose to work with Walsh functions to fit the different objectives of a MCOP. However, since the aim is to find a whole approximation set, and

not a single solution, we shall adopt a different surrogate-assisted framework that fits the multi-objective setting.

*Multi-objective Numerical Surrogates.* Turning to the literature on numerical surrogate-assisted MOEAs, one can find a variety of specialized approaches; see, e.g., [6, 22] to cite a few. In fact, in addition to the number of existing numerical models, there exists a range of MOEAs based on seemingly different principles, hence exposing as much design components as challenges when combining them with surrogates. Conducting a complete review of the literature on the subject is out of the scope of this paper. The interested reader is referred to, e.g., [3] and the references therein. However, our contribution on leveraging discrete surrogates to multi-objective optimization is guided by the lessons learnt from the past in the numerical optimization literature. In particular, we continue the recent efforts made by the MOEA community to classify the existing approaches and to provide a unified view of the underlying challenges. In this respect, one can identify two major interdependent research lines: (i) approaches focusing on the design and optimization of accurate acquisition functions, e.g., the EI in a multi-objective setting, and (ii) approaches focusing on the category of MOEAs that are better suited to work under an expensive setting.

*Our Methodology.* Since the discrete Walsh surrogate does not come with an uncertainty measure of the predicted quality of a solution, we use the underlying model as an estimate of the true objective functions. Other existing discrete surrogates such as Kriging do provide such an information and could also be good alternatives. However, the fitting superiority of Walsh surrogates justifies such a design choice. We also consider decomposition-based MOEAs, represented by the state-of-the-art MOEA/D algorithm [14], as our core evolutionary framework. In fact, besides being at the heart of the most recent state-of-the-art numerical surrogate-assisted MOEAs, such as MOEA/D-EGO [22] and K-RVEA [6], decomposition has been proved to be a flexible framework for leveraging single-objective techniques to multi-objective optimization.

*Contribution Overview.* In this paper, we contribute to the design and analysis of surrogate-assisted MOEAs for pseudo-boolean functions by leveraging Walsh models within the framework of decomposition-based algorithms. More precisely,

- We adopt a coarse-grained design exposing few algorithmic components that can be configured in a plug-and-play manner. In addition to demonstrating the effectiveness of the proposed design, our goal is in fact to highlight what makes an approach better than another. Accordingly, we are able to consider both a filtering approach based on pre-screening a pool of offspring generated by standard variation operators, as well as a substitution approach that consists in optimizing the trained surrogate models in lieu of the true objective functions. Besides, we investigate the role of the strategy used to select a solution for the (expensive) true evaluation and study its impact on approximation quality.
- Extending benchmarking effort from the single-objective setting, we analyze the designed algorithms using as benchmark problems a number two-objective NK landscapes with different degrees of non-linearity. This allows us to shed lights into the accuracy of the considered Walsh surrogate model, and to contribute to a better understanding of the strengths and weaknesses of such a model.

- We provide a systematic analysis of the considered design components and parameters, including an anytime analysis of algorithm performance under different budgets. Although we provide evidence on the superiority of the proposed algorithms compared against not using any surrogate, we found that different design choices can imply algorithms with different anytime profiles or different characteristics of the computed approximation set. In particular, this confirms the complexity of designing an optimal surrogate-assisted MOEA, and provides more insights on the key challenges to be addressed in the future.

*Outline.* In Section 2, we provide the necessary background related to decomposition and Walsh basis. In Section 3, we describe the proposed framework. In Section 4, we report our main empirical findings. In Section 5, we finally conclude the paper.

## 2 BACKGROUND

### 2.1 Multi-objective Optimization and Decomposition

*Multi-objective Combinatorial Optimization.* A MCOP can be defined by a decision space  $X$ , and a vector function  $F = (f_1, f_2, \dots, f_m)$ , where  $m$  is the number of objectives, to be maximized. In this paper, we consider unconstrained pseudo-boolean multi-objective problems, i.e., a vector function  $F: \{0, 1\}^n \mapsto \mathbb{R}^m$ , where  $n$  is the problem size. Let  $Z = F(X) \subseteq \mathbb{R}^m$  be the set of outcome vectors in the *objective space*. An objective vector  $z \in Z$  is *dominated* by a vector  $z' \in Z$  iff  $\forall i \in \{1, \dots, m\}, z_i \leq z'_i$  and  $\exists i \in \{1, \dots, m\}$  s.t.  $z_i < z'_i$ . A solution  $x \in X$  is dominated by a solution  $x' \in X$  iff  $F(x)$  is dominated by  $F(x')$ . A solution is *Pareto optimal* if there does not exist any other solution dominating it. The set of all Pareto optimal solutions is the *Pareto set*. Its mapping in the objective space is the *Pareto front*. Computing the Pareto set is NP-hard for a wide range of MCOPs. Our goal is to design an algorithm providing a high-quality approximation set.

*Decomposition-based MOEAs.* We rely on the state-of-the-art algorithm in the wider class of aggregation-based MOEAs, namely MOEA/D [14]. The basic concept behind aggregation (or decomposition) is to transform the objective vector into a scalar value using some scalarizing function configured with a weight vector. We rely on the Chebyshev function, to be minimized, which is defined as:

$$g(x | \omega, F) = \max_{i \in \{1, \dots, M\}} \omega_i \cdot |z_i^* - f_i(x)|$$

where  $x \in X$ ,  $\omega = (\omega_1, \dots, \omega_M)$  is a positive weight vector, and  $z^*$  is a reference point such that  $\forall x \in X, i \in \{1, \dots, M\}, z_i^* > f_i(x)$ .

MOEA/D relies on evolving a population of individuals by structuring them using a set of evenly distributed weigh vectors. Given a set  $\mathcal{W}_\mu := \{\omega^1, \dots, \omega^\mu\}$ , the original MOP is broken into  $\mu$  single-objective sub-problems. Each sub-problem is defined using the scalarizing function  $g(x | \omega^i, F)$  parameterized by a different weigh vector  $\omega^i$ . A solution  $x^i$  is then assigned to each sub-problem and is intended to optimize the corresponding scalar value. The so-implied population, denoted  $\mathcal{P}_\mu := \{x^1, \dots, x^\mu\}$ , is evolved following standard evolutionary selection, replacement, and variation (mutation, crossover, etc) mechanisms in order to optimize the different sub-problems. The most distinguishable feature of MOEA/D is that the

population is evolved simultaneously and cooperatively. The solutions underlying different sub-problems can be used to generate a new offspring, and the newly generated solution can for instance replace the solutions underlying different sub-problems. The guiding principle relies on the fact that the solutions w.r.t different sub-problems can help optimizing other sub-problems and enhancing the overall population dynamics to approach the Pareto front. In its initial design, MOEA/D uses a parameter to define how much cooperation between sub-problems, namely the neighborhood of sub-problems, i.e., the  $T$ -neighborhood. In our work, we consider that  $T = \mu$  meaning that all defined sub-problems are neighbors. This is basically motivated by the expensive setting we are considering and the fact that we would like to push the population as fast as possible towards the Pareto front using a very restricted budget.

### 2.2 Walsh Surrogates

*Mathematical Foundations.* Walsh functions [19] constitute an enumerable set of functions  $\phi_k: [0, 1] \rightarrow \{-1, 1\}$  which composes a normal and orthogonal basis of the Hilbert space  $L^2([0, 1])$ . Like the trigonometric functions of the Fourier basis, they can be used to decompose any function of the Hilbert space under some conditions [19], and were used since the late seventies in the theory of evolutionary computation [4].

Given a single pseudo-boolean function  $f: \{0, 1\}^n \mapsto \mathbb{R}$ , Walsh functions are defined as follows [17]. For any integer  $\ell \in \llbracket 0, 2^n - 1 \rrbracket$  with the binary representation  $\ell = \sum_i \ell_i 2^i$  and  $\ell_i \in \{0, 1\}$ , the Walsh function  $\phi_\ell: \{0, 1\}^n \rightarrow \{-1, 1\}$  is defined for any binary string  $x = (x_1, \dots, x_i, \dots, x_n) \in \{0, 1\}^n$  as:

$$\phi_\ell(x) = (-1)^{\sum_{i=0}^{n-1} \ell_i x_i} \quad (1)$$

The *order* of a Walsh function  $\phi_\ell$ , denoted by  $o(\phi_\ell)$ , is defined by the number of binary digits equals to 1 in the binary representation of  $\ell$ . For example, the function of order 0 is  $\phi_0$ , the functions of order 1 are  $\phi_{2^p}$  for all integers  $p \geq 0$ , the functions of order 2 are  $\phi_{2^p + 2^{p'}}$  for all pairs of integers  $p \neq p' \geq 0$ , and so on. The so-defined (finite) set of discrete functions is a normal orthogonal basis for the space of pseudo-boolean functions, i.e.,  $\forall \ell, \ell' \in \llbracket 0, 2^n - 1 \rrbracket, \frac{1}{2^n} \sum_{x \in \{0, 1\}^n} \phi_\ell(x) \cdot \phi_{\ell'}(x) = \delta_{\ell \ell'}$ . Therefore, any pseudo-boolean function  $f: \{0, 1\}^n \rightarrow \mathbb{R}$  can be written as:

$$\forall x \in \{0, 1\}^n, f(x) = \sum_{\ell=0}^{2^n-1} w_\ell \cdot \phi_\ell(x) \quad (2)$$

$$\text{s.t. } w_\ell = \frac{1}{2^n} \sum_{x \in \{0, 1\}^n} f(x) \cdot \phi_\ell(x) \quad (3)$$

*From a Walsh basis to a Surrogate Model.* An important remark is that the functions  $\phi_k$  are problem independent, and hence uniquely defined irrespective to any black-box function  $f$ , as given in Eq. (1). However, the values of the coefficients  $w_k$  in Eq. (2) depend on the considered function  $f$ , as given in Eq. (3). Although these coefficients are unknown and their number in general exponential in  $n$ , there might exist a significantly large number of zero coefficients. Hence, the idea developed in [10, 17] is to approximate  $f$  using solely the Walsh functions up to a constant order  $d \ll n$  and using an estimate  $\tilde{w}_k$  of the (unknown) coefficient  $w_k$ . More formally,  $f$  can be approximated by the following model:

$$\forall x \in \{0, 1\}^n, \tilde{f}(x|d) = \sum_{k: o(\phi_k) \leq d} \tilde{w}_k \cdot \phi_k(x) \quad (4)$$

For example, a second order approximation can be rewritten as:

$$\forall x \in \{0, 1\}^n, \tilde{f}(x|2) = \tilde{w}_0 + \sum_{i=1}^n \tilde{w}_i \cdot (-1)^{x_i} + \sum_{i < j < n} \tilde{w}_{ij} \cdot (-1)^{x_i + x_j}$$

where  $\tilde{w}_0$  is the zero order estimate coefficient, and  $\tilde{w}_i$  and  $\tilde{w}_{ij}$  are first- and second-order estimated coefficients, respectively. Intuitively, the larger the order  $d$  and the better the quality of the estimate coefficients  $\tilde{w}_k$ , the more accurate the expansion.

To summarize, given a sample of solutions for which we know the true  $f$ -value, a surrogate model can be constructed by computing the estimate coefficients  $\tilde{w}$  for some value of  $d$ . This turns out to be a standard optimization problem in linear regression methods, since Eq. (4) can be interpreted as a linear model whose predictors are the Walsh functions' values. In particular, sparse techniques can be used to minimize the number of non-zero coefficients when the number of predictors is large [7]. Following the work in [10, 17], we shall use the Lasso algorithm [16] to fit the model. This is a method of choice, since the number of Walsh functions of up to a fixed order  $d$  might actually be greater than the number of solutions for which we know the true objective value in expensive optimization.

### 3 THE PROPOSED S-CMO FRAMEWORK

*General Description.* As illustrated in Algorithm 1, we follow a simple design with few configurable components. The algorithm *Surrogate-assisted Combinatorial Multiobjective Optimization* (S-CMO) is initialized with  $\mu$  randomly-generated solutions which are evaluated and mapped to the different sub-problems. They constitute both the initial population and the initial training data set. Thereafter, the algorithm runs in multiple generations (the **for** loop, line 6) where the  $\mu$  sub-problems are considered iteratively.

At each iteration,  $m$  Walsh surrogate models are trained for each objective function (line 7). In other words, for every  $i \in \{1, \dots, m\}$ ,  $\tilde{f}_i$  is trained using all solutions evaluated so far, stored in  $\mathcal{D}$ , and the corresponding (estimate) coefficients (Eq. (4)) are computed following a sparse linear regression methodology as described above. Notice that the maximum order of the computed Walsh expansion is provided as a user-defined parameter. After training, a set  $\mathcal{S}_\lambda$  of  $\lambda$  offspring is generated (line 8) and one solution  $x'$  is selected (line 9) in order to be evaluated using the true objective functions  $F$ .

Once a new solution is selected and evaluated, it can enter the population following a standard replacement mechanism (the **for** loop, line 13). More precisely, if the newly evaluated solution improves any sub-problem with respect to its scalarizing function, it simply replaces the corresponding solution in the current population. This replacement mechanism might look relatively aggressive compared to the existing literature dealing with inexpensive problems. Nevertheless, given the restricted global budget implied by the cost of evaluation, any improving solution is to be considered for entering the population. Finally, the training set  $\mathcal{D}$  is updated with the newly evaluated solution, and a new iteration starts. Let us comment that all solutions evaluated so far are included for training, which is also an important design choice, and other alternatives may be considered as in previous numerical algorithms, e.g., [22].

*Pool Generation.* The first component of the proposed framework is the pool generation procedure (GENERATEPOOL, line 8). Its

---

#### Algorithm 1: High level template of S-CMO

---

**Input:**  $\mathcal{W}_\mu := \{\omega^1, \dots, \omega^\mu\}$ : weight vectors;  $g(\cdot | \omega, \cdot)$ : a scalarizing function to be minimized;  $d$ : maximum order of Walsh functions;

- 1  $\mathcal{P}_\mu \leftarrow \{x^1, \dots, x^\mu\}$ : initial population of size  $\mu$ ;
- 2  $\mathcal{D} \leftarrow \{(x^1, F(x^1)), \dots, (x^\mu, F(x^\mu))\}$ : (initial) training data ;
- 3  $\mathcal{EP} \leftarrow$  : (optional) initialize external archive ;
- 4  $z^* \leftarrow$  initialize reference point;
- 5 **while** *Global budget is not exhausted* **do**
- 6     **for**  $i \in \{1, \dots, \mu\}$  **do**
- 7         /\* Train Walsh models \*/  
 $\tilde{\mathcal{F}} := (\tilde{f}_1, \dots, \tilde{f}_m) \leftarrow \text{TRAINWALSHMODELS}(\mathcal{D}, d)$ ;
- 7         /\* Generate  $\lambda$  offspring with an evolutionary mechanism \*/
- 8          $\mathcal{S}_\lambda \leftarrow \text{GENERATEPOOL}(\mathcal{P}_\mu, \lambda, \tilde{\mathcal{F}})$  ;
- 8         /\* Select a solution for true evaluation \*/
- 9          $x' \leftarrow \text{SELECTFOREVALUATION}(\mathcal{S}_\lambda, i, \tilde{\mathcal{F}})$  ;
- 10          $F(x') \leftarrow$  evaluate  $x'$  ;
- 11          $\mathcal{EP} \leftarrow$  (optional) update external archive using  $x'$  ;
- 12          $z^* \leftarrow$  update reference point using  $F(x')$  ;
- 12         /\* Update Population (replacement) \*/
- 13         **for**  $j \in \{1, \dots, \mu\}$  **do**
- 14             **if**  $g(x' | \omega^j, F) < g(x^j | \omega^j, F)$  **then**
- 15                  $x^j \leftarrow x'$  ;
- 15             /\* Update training data \*/
- 16              $\mathcal{D} \leftarrow \mathcal{D} \cup \{(x', F(x'))\}$ ;

---

goal is to generate a whole set of solutions using some specialized mechanisms. Both the size of the generated pool and the way it is generated are important. In fact, in standard (non surrogate-assisted) MOEA/D for instance, only one single offspring is created and evaluated immediately. The idea here is to increase in a first stage the number of eventually promising solutions which are *not* evaluated using the true expensive objective functions. Only in a second stage, one solution can be selected on the basis of the cheap model predictions. We consider two alternative strategies depending on whether the trained surrogate model is used or not to help constructing the pool, namely, FILTER and SUBSTITUTE.

- (1) The FILTER strategy consists in generating  $\lambda$  offspring in an independent manner using standard variation operators. Given the stochastic nature of such operators, the motivation is to artificially increase the probability that one more promising solution is generated. We use a one-point crossover followed by a bit-flip mutation, to generate  $\lambda$  offspring independently.
- (2) The SUBSTITUTE strategy consists in running an algorithm to optimize the multi-objective problem defined by the trained surrogate models  $\tilde{\mathcal{F}}$ . The idea is to use an optimization engine to intensify the search of promising solutions, at a cheap cost, assuming the constructed models are accurate enough to approximate the original function  $F$ . We choose to rely on a decomposition-based algorithm that solves  $\tilde{\mathcal{F}}$  and that follows seemingly the same template than Algorithm 1.

The hypothesis behind these two strategies can be formulated w.r.t. the prediction accuracy of the constructed models. A Walsh surrogate does not provide any measure of uncertainty with respect to the output predictions. Hence, optimizing the trained models very aggressively can be misleading, and creating some offspring from the current population might actually be safer. Similarly, increasing the pool size can bias the subsequent selection for expensive evaluation step towards non-efficient regions. Such considerations are actually to be studied in more details in our empirical analysis.

*Selection for Expensive Evaluation.* The second component deals with the selection of one offspring to be evaluated from the set  $S_\lambda$  (line 9). For each offspring solution  $x \in S_\lambda$ , we attribute a scalar value of the constructed Walsh surrogates  $\tilde{\mathcal{F}}$ , which is to be understood as an indicator of how promising  $x$  is. Then, we consider two alternative strategies to select the offspring, namely LOCAL and GLOBAL, defined as follows:

- (1) The LOCAL strategy chooses the solution that has the best scalar  $g$ -value with respect to sub-problem  $i$  considered in the current iteration, i.e., underlying the **for** loop, line 6. More formally, the selected offspring  $x'$  is given by:

$$x' := \operatorname{argmin}_{x \in S_\lambda} g(x \mid \omega^i, \tilde{\mathcal{F}}) \quad (5)$$

- (2) The GLOBAL strategy chooses the solution that has the best scalar  $g$ -value with respect to any sub-problem, independently of the current iteration. The selected offspring  $x'$  is given by:

$$x' := \operatorname{argmin}_{x \in S_\lambda} \min_{1 \leq \ell \leq \mu} g(x \mid \omega^\ell, \tilde{\mathcal{F}}) \quad (6)$$

The GLOBAL strategy assumes that the range of the objective values is the same. If such an assumption does not hold, normalization should be considered. The hypothesis guiding the design of these two strategies can be formulated in terms of the expected convergence behavior towards the Pareto front. In fact, the LOCAL strategy aims at intensifying the search evenly among sub-problems since it is ‘biased’ by one different sub-problem at each iteration. Hence, the computational effort is likely to be more evenly distributed w.r.t. the different sub-problems. By contrast, the GLOBAL strategy does not imply any specific control of which sub-problem should be preferred at each iteration. Hence, it can be thought as having a more opportunistic intensification/diversification behavior.

*Further Discussion.* Combining the pool generation and the selection components, we end up with four variants. Besides, our pool generation strategies are consistent with the *filtering* and *substitution* classes of existing numerical surrogate-assisted algorithms [3]. It was shown that such algorithms expose different performance profiles, while being able to outperform other classes of algorithms leveraging single-objective techniques such as EGO [8]. Although there is no evidence that the same holds in our setting, the proposed S-CMO framework and underlying variants aim at making a first step towards a better understanding of what makes a surrogate-assisted algorithm successful when dealing with MCOPs. In the reminder of the paper, we shall in fact conduct a comprehensive analysis informing about the relative performance profile of the designed variants and the impact of their different parameters, in tight relation with the accuracy of the Walsh surrogates.

## 4 EXPERIMENTAL RESULTS AND ANALYSIS

### 4.1 Experimental Setup

*Multi-objective NK Landscapes.* We consider multi-objective NK landscapes as a problem-independent model of multi-objective multi-modal combinatorial optimization problems [18]. This is in line with previous works on single-objective surrogates [10], where NK landscapes are used for benchmarking. Solutions are binary strings of size  $n$  and the objective vector, to be maximized, is defined as  $F: \{0, 1\}^n \mapsto [0, 1]^m$ . An objective value  $f_i(x)$  of a solution  $x = (x_1, \dots, x_n)$  is defined as [9]:

$$f_i(x) = \frac{1}{n} \sum_{j=1}^n c_j^i(x_j, x_{j_1}, \dots, x_{j_k}) \quad (7)$$

where the  $c_j^i: \{0, 1\}^{k+1} \rightarrow [0, 1]$  are component functions, and  $k$  is a parameter specifying the number of epistatic interactions. The component function  $c_j^i$  assigns a real-valued contribution for every combination of  $x_j$  and its  $k$  epistatic interactions  $\{x_{j_1}, \dots, x_{j_k}\}$ . The parameter  $k$  defines the degree of non-linearity of the problem, and hence the *ruggedness* of the landscape. By increasing this value, problems can be gradually tuned from smooth to rugged. We use the same ruggedness degree and interactions for all the objectives. Component values are generated uniformly at random in  $[0, 1]$ .

We consider bi-objective NK landscapes with this setting:  $n \in \{25, 50\}$  and  $k \in \{0, 1, 2\}$ . This allows us to study the designed algorithms on different problem size and landscape difficulty.

*Competing Algorithms and Parameter Setting.* Since we are not aware of any existing surrogate-assisted MOEAs for combinatorial problems, we consider a non surrogate-assisted MOEA as a baseline algorithm extracted from the S-CMO framework described in Algorithm 1, referred as MOEA/D for simplicity. More precisely, we deactivate all instructions related to model training (lines 2, 7, and 16). The GENERATEPOOL and the SELECTFOREVALUATION procedures in lines 8 and 9, are deactivated and replaced by a procedure that creates one offspring solution by variation, and evaluates it immediately. For the 5 competing algorithms (i.e., MOEA/D and S-CMO variants), we use a one-point crossover followed by a uniform bit-flip mutation with a rate of  $1/n$ . For the SUBSTITUTE variants, we use the standard MOEA/D described in the previous paragraph as the optimization engine in the pool generation procedure.

The number of weight vectors is set to  $\mu = 50$ , with  $\omega^i = ((i-1)/(\mu-1), 1 - (i-1)/(\mu-1))$ ,  $\forall i \in \llbracket 1, 50 \rrbracket$ . This choice is consistent with other numerical expensive approaches [3] and is accurate w.r.t. the size of considered NK landscapes [18]. For the FILTER-based variants, the size of the offspring pool is set in the range  $\lambda \in \{2, 4, 8, 16, 32\}$ . For the SUBSTITUTE-based variants, more configuration aspects are to be considered regarding the initial population and the stopping condition of the embedded MOEA/D optimization engine. The number of weight vectors in the multi-objective optimization engine is set with the value of  $\mu$  of the surrogate-assisted MOEA/D. Accordingly, we simply initialize each call to the inner multi-objective optimization engine with the current population of the outer surrogate-assisted algorithm being executed, and we let it run for  $g$  generations with the trained model  $\tilde{\mathcal{F}}$ , where  $g$  is a parameter in the range  $g \in \{1, 2, 5, 10, 20, 30, 40, 50\}$ . The S-CMO variants require a maximum order  $d$  as input for constructing and

**Table 1: Ranks and average hypervolume value (between brackets) of the different algorithms after 500 and 1500 evaluations (a lower rank is better). The parameter values for the S-CMO variants are those corresponding to the best average hypervolume. For each budget and instance, a rank  $c$  indicates that the corresponding algorithm was found to be significantly outperformed by  $c$  other strategies w.r.t. a Wilcoxon statistical test at a significance level of 0.05. Ranks in bold correspond to approaches that are not significantly outperformed by any other, and the underlined values corresponds to the best approach in average.**

		$n = 25$			$n = 50$			avg. rank
		$k = 0$	$k = 1$	$k = 2$	$k = 0$	$k = 1$	$k = 2$	
budget = 500	MOEA/D	1(7.71)	2(12.49)	2(10.47)	2(5.59)	3(5.9)	1(6.81)	1.83
	FILTER LOCAL	1(7.83)	3(12.74)	1(10.71)	2(5.58)	3(6)	1(6.8)	1.83
	FILTER GLOBAL	<b>0(8.04)</b>	<b>0(13.36)</b>	1(11.04)	1(5.95)	1(6.41)	1(7.1)	0.66
	SUBSTITUTE LOCAL	3(7.38)	<b>0(13.18)</b>	1(10.82)	2(5.75)	1(6.51)	1(7.15)	1.33
	SUBSTITUTE GLOBAL	1(7.47)	<b>0(13.35)</b>	<b>0(11.62)</b>	<b>0(6.65)</b>	<b>0(7.16)</b>	<b>0(7.93)</b>	0.16
budget = 1500	MOEA/D	2(8.07)	2(13.39)	1(11.58)	2(6.27)	3(6.73)	2(7.88)	2.00
	FILTER LOCAL	2(8.12)	2(13.32)	2(11.39)	2(6.16)	3(6.68)	3(7.72)	2.33
	FILTER GLOBAL	<b>0(8.3)</b>	<b>0(1.36)</b>	1(11.18)	1(6.51)	1(7.06)	1(8.11)	0.66
	SUBSTITUTE LOCAL	2(8.06)	<b>0(13.58)</b>	1(11.71)	2(6.29)	1(6.99)	1(8.16)	1.16
	SUBSTITUTE GLOBAL	<b>0(8.31)</b>	2(13.44)	<b>0(12.32)</b>	<b>0(6.85)</b>	<b>0(7.38)</b>	<b>0(8.54)</b>	0.33

training the Walsh surrogates. This is set in the range  $d \in \{1, 2, 3\}$ , allowing to consider gradually complex Walsh models, i.e., linear, quadratic, and cubic. It is important to remark that the *exact* Walsh expansion of a single-objective NK landscape has an order of  $k + 1$ .

*Performance Evaluation.* Given the large number of configurations ( $1 + (5 \times 2 + 8 \times 2) \times 3 = 79$ ), and in order to keep our experiments manageable, every configuration is executed 10 independent times for each of the  $2 \times 3 = 6$  instances for a total of 4 740 runs using an overall budget of 1 500 objective evaluations per run. Besides, we consider to analyze approximation quality at different intermediate budget values to render the anytime performance. Due to lack of space, we shall however report our findings on a representative set of experimental data. For performance assessment, we consider the external archive, storing all non-dominated points found so far during the search process. We use the hypervolume [23] to assess the quality of the different competing algorithms, where the reference point is set to the nadir point estimated over all runs.

## 4.2 Overall Hypervolume Performance

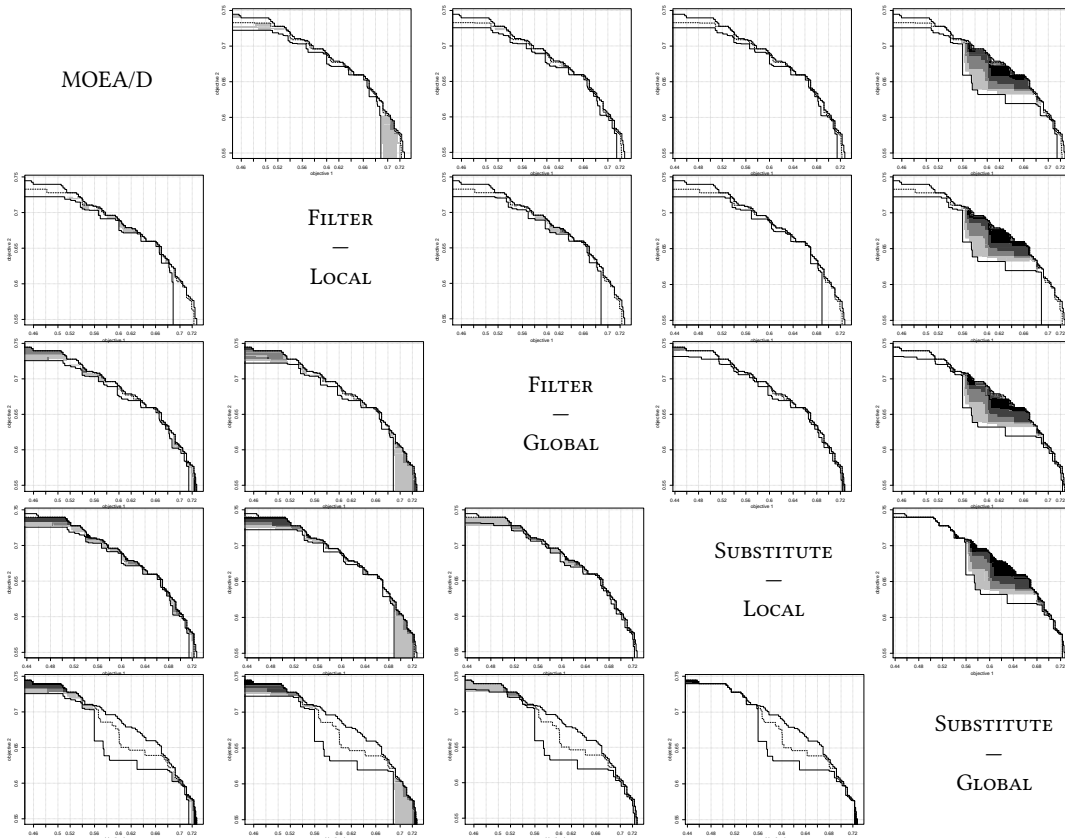
We start by providing an overview of the performance of the different algorithms using two different budgets. As a first step, we rank in Table 1 the different competing algorithms when considering the parameter setting corresponding to the best average hypervolume value for each budget, each instance and each algorithm. In fact, our goal here is to draw an overall tendency with respect to our two main design components, even though their best parameter setting might be different across instances and budgets. For clarity, this is analyzed later in the paper. Overall, we can see that the best performing variant is based on the combination of the SUBSTITUTE pool generation strategy and the GLOBAL selection strategy, with few exceptions for instance size  $n = 25$ , where a FILTER strategy is sometimes a better choice. Interestingly, the second best performing variant, which is based on the FILTER pool generation strategy, shares the same GLOBAL selection strategy and is overall

slightly better than the SUBSTITUTE strategy combined with the LOCAL strategy. When combined with the LOCAL strategy, the FILTER strategy provides comparable hypervolume values than the non surrogate-assisted MOEA/D algorithm. This indicates that although optimizing the pool generation using a multi-objective engine to optimize the trained Walsh surrogates is of high importance, the combined effect of the selection strategy is also crucially important.

## 4.3 Impact of the Selection Strategy

To better elicit the combined effect of the pool generation and selection, we consider in Fig. 1 their empirical attainment functions (EAFs) [11]. The EAF provides the probability, estimated from several runs, that an arbitrary objective vector is dominated by, or equivalent to a solution obtained by a single run of the algorithm. The difference between the EAFs for two different algorithms enables to identify the regions of the objective space where one algorithm outperforms the another. The magnitude of the difference in favor of one algorithm is plotted within a gray-colored graduation. We can see that the combination of the SUBSTITUTE and GLOBAL strategies, which was found to be the best at first, has a tendency to better approach the borders of the Pareto front while performing poorly in the central area. Since such a property has an impact on the range of the hypervolume values, this explains for instance why the GLOBAL strategy was found to achieve better performance in Table 1. Interestingly, the combination of the SUBSTITUTE and LOCAL strategies, which was found to be the second best combination according to the hypervolume value in Table 1, dominates significantly the middle area compared against the SUBSTITUTE and GLOBAL combination, while behaving relatively good at the borders. Notice also that the SUBSTITUTE and LOCAL combination outperforms the other approaches consistently in all regions.

We recommend the combination of the SUBSTITUTE and LOCAL strategies as a competitive S-CMO variant providing an approximation set exposing well-balanced properties in terms of spread and distance to the front. This means that a good optimization

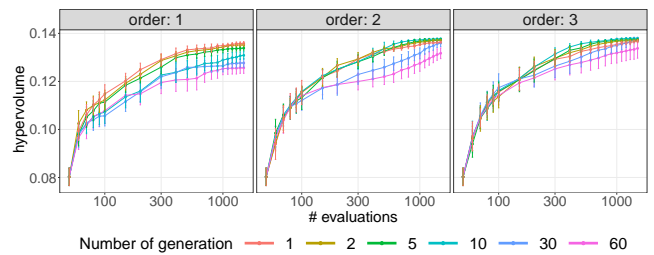


**Figure 1: EAF differences for the FILTER ( $\lambda = 4$ ) and the SUBSTITUTE ( $g = 10$ ) strategies, for the NK landscape with  $n = 25$  and  $k = 1$  at 1 500 function evaluations. The plot in row  $i$  and column  $j$  shows the areas where algorithm  $i$  improves over algorithm  $j$ .  $x$ - and  $y$ -axis are respectively objective 1 and objective 2.**

engine and a good balance over the different sub-problems should be adopted when searching for a promising solution to evaluate.

#### 4.4 Impact of the Search Engine Effort

The amount of effort required for generating a pool of offspring can be viewed as controlled by the parameters  $\lambda$  and  $g$  respectively for the FILTER and SUBSTITUTE strategies. Due to lack of space, we omit the impact of  $\lambda$  from our analysis, and focus on the setting of parameter  $g$ , that is the stopping condition of the inner multi-objective optimizer in the SUBSTITUTE strategy, which was found to perform better overall. This is reported in Fig. 2, in combination with the LOCAL strategy, for instance  $n = 25$  and  $k = 1$ , and for different values of the maximum order  $d$  used by the Walsh surrogates. As discussed in the previous section, notice that the LOCAL strategy removes the biases when interpreting the hypervolume values. The first noticeable observation is that increasing the value of  $g$  does not necessarily help preparing a more effective pool for the selection step. This means that optimizing the Walsh functions aggressively can be misleading, which can partly be attributed to the accuracy error when computing and training the models. In other words, optimizing a model that does not fit accurately the true objective function can lead in wasting expensive function evaluations in unuseful regions. Overall, and with respect to the multi-objective



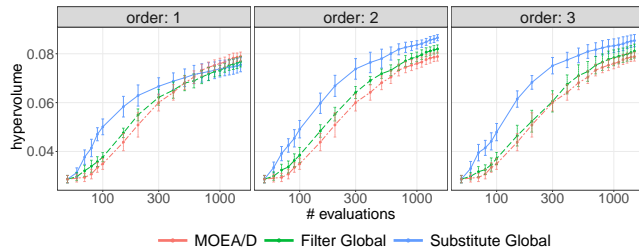
**Figure 2: Profile of the SUBSTITUTE-LOCAL variant for different  $g$  values ( $n = 25, k = 1$ ).**

optimizer that we are using, a relatively small value of  $g$  of up to 10 is recommended. More importantly, Fig. 2 provides a first hint on the impact of the Walsh order  $d$ , since different relative performances are obtained for different values of  $d$ .

#### 4.5 Impact of the Walsh Order

In Fig. 3, we show the relative performance of the different competing algorithms for different values of  $d$  and for the hardest instance with  $n = 50$  and  $k = 2$ . We can clearly see that when using Walsh surrogates of the smallest order  $d = 1$ , only the SUBSTITUTE-based S-CMO variants are able to substantially outperform the non

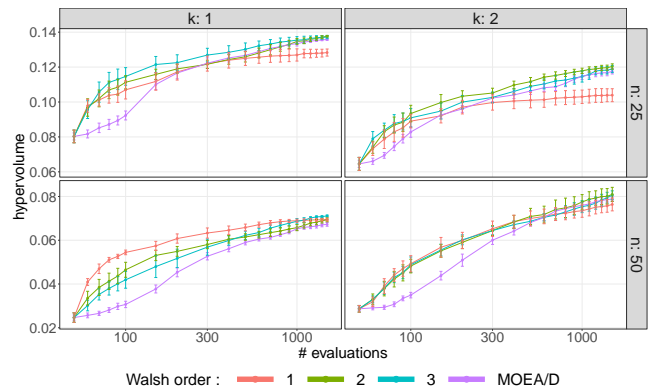




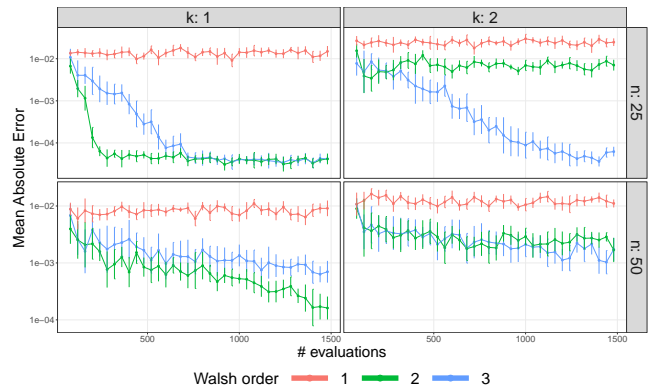
**Figure 3: Profile of the SUBSTITUTE ( $g = 10$ ) and FILTER ( $\lambda = 4$ ) variants, GLOBAL strategy ( $n = 50, k = 2$ ).**

surrogate-assisted algorithm MOEA/D, and only for smallest budgets. As the order increases, the difference in performance becomes more pronounced, and the anytime profile of the best performing SUBSTITUTE-based variant is clearly better than the non surrogate-assisted one. Notice that for  $k = 2$ , the *exact* Walsh expansion has order  $k + 1 = 3$ , and hence one would expect that using a *trained* Walsh surrogate for  $d = 3$  is the best option. This is actually *not* true, since some strategies show better performance for a smaller order. The impact of the order choice is further detailed in Figs. 4–5.

In Fig. 4, we show the relative anytime performance of the recommended SUBSTITUTE-LOCAL variant for different order values. We can clearly see that for  $d \geq k + 1$ , that is when the order used for training is larger than the a priori perfect (exact) one, we do *not* observe a systematic increase of the anytime behavior, independently of the considered instance size  $n$  and ruggedness  $k$ . Accordingly, in Fig. 5, we show the accuracy of the trained Walsh surrogate measured by the mean absolute error in the fitting phase. We can see that overestimating the order value (when choosing  $d = 3$  for an instance with  $k = 1$ ), has a tendency to decrease the model accuracy. This is attributed to the increase of the number of Walsh predictors, which is in  $O(n^d)$ , making it more difficult for the Lasso regression to find accurate coefficients estimates by using a restricted number of solutions for training. For the hardest instance ( $n = 50, k = 2$ ), we see that a sub-optimal order  $d = 2$  is able to provide a similar accuracy than  $d = 3$ , while using  $d = 1$  decreases accuracy. Interestingly, depending both on the instance size and the instance difficulty, the accuracy of the model (as depicted in Fig. 5) is not necessary perfectly correlated with the relative hypervolume values (as shown in Fig. 4) especially for small budgets. This means that when the budget is very restricted, an underestimated order is a viable alternative. Furthermore, the accuracy of the model seems to be impacted by different issues, namely, the power of the regression technique, the size of the training data, and the number of non-zero coefficients in the exact Walsh decomposition which is bounded, for NK landscape, by  $n(2^{k+1} - 1) + 1$ . Overall, given that  $k$  is only used for benchmarking purposes – the exact order of the Walsh decomposition of a black-box problem is a priori unknown – we found that an order which is too small is to be typically avoided (unless the budget is extremely restricted), and an order of two ( $d = 2$ ), related to a quadratic approximation, seems to be a reasonable choice. Nonetheless, further investigations on using other machine learning techniques to estimate the Walsh coefficients, or to change the order dynamically, are to be further considered.



**Figure 4: Profile of the SUBSTITUTE-LOCAL variant ( $g = 10$ ) w.r.t. the Walsh order.**



**Figure 5: Mean absolute error of the Walsh model for objective function  $f_1$  using SUBSTITUTE-LOCAL ( $g = 10$ ).**

## 5 CONCLUSION

In this paper, we proposed and analyzed a framework for discrete surrogate-assisted multi-objective optimization using decomposition and Walsh basis as core components. In particular, we showed that configuring both decomposition components and Walsh surrogates can impact the behavior of the multi-objective optimization process in several aspects, depending on instance properties and available budget. Accordingly, one major future research direction would be to address adaptive landscape-aware techniques that can accommodate different settings in an automatic manner and to study the computational budget of decomposition components.

Since we are dealing with an expensive setting, online approaches that avoid prohibitive offline solution pre-sampling, and which are able to adaptively change their behavior while the search process is running, seem a viable option to investigate. One idea is to acquire a feedback on the model accuracy and on the evolution of solutions at the different decomposed sub-problems. This would allow us to adjust the parameters and components of both the Walsh surrogates and the decomposition framework, interdependently.

## ACKNOWLEDGMENTS

This work was supported by the French national research agency (ANR-16-CE23-0013-01) and the Research Grants Council of Hong Kong (RGC Project No. A-CityU101/16).

## REFERENCES

- [1] Ricardo Baptista and Matthias Poloczek. 2018. Bayesian Optimization of Combinatorial Structures. arXiv:1806.08838
- [2] Thomas Bartz-Beielstein and Martin Zaefferer. 2017. Model-based methods for continuous and discrete global optimization. *Applied Soft Computing* 55 (2017), 154 – 167.
- [3] Nicolas Berveglieri, Bilel Derbel, Arnaud Liefoghe, Hernán Aguirre, and Kiyoshi Tanaka. 2019. Surrogate-Assisted Multiobjective Optimization Based on Decomposition: A Comprehensive Comparative Analysis. In *Proceedings of the Genetic and Evolutionary Computation Conference on - GECCO '19*. ACM Press, Prague, Czech Republic, 507–515.
- [4] Albert Donally Bethke. 1980. *Genetic algorithms as function optimizers*. Ph.D. Dissertation. University of Michigan.
- [5] Francisco Chicano, Darrell Whitley, and Andrew M. Sutton. 2014. Efficient Identification of Improving Moves in a Ball for Pseudo-Boolean Problems. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO '14)*. Association for Computing Machinery, New York, NY, USA, 437–444. <https://doi.org/10.1145/2576768.2598304>
- [6] T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, and K. Sindhya. 2018. A Surrogate-Assisted Reference Vector Guided Evolutionary Algorithm for Computationally Expensive Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation* 22, 1 (2018), 129–142.
- [7] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. 2015. *Statistical Learning with Sparsity: The Lasso and Generalizations* (1 ed.). Chapman and Hall/CRC.
- [8] Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* 13, 4 (1998), 455–492.
- [9] S. A. Kauffman. 1993. *The Origins of Order*. Oxford University Press.
- [10] Florian Leprière, Sébastien Verel, Cyril Fonlupt, and Virginie Marion. 2019. Walsh Functions as Surrogate Model for Pseudo-Boolean Optimization Problems. In *Proceedings of the Genetic and Evolutionary Computation Conference on - GECCO '19*. ACM Press, Prague, Czech Republic, 303–311.
- [11] Manuel López-Ibáñez, Luis Paquete, and Thomas Stützle. 2010. Exploratory Analysis of Stochastic Local Search Algorithms in Biobjective Optimization. In *Experimental Methods for the Analysis of Optimization Algorithms*, Thomas Bartz-Beielstein, Marco Chiarandini, Luis Paquete, and Mike Preuss (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 209–222.
- [12] Alberto Moraglio and Ahmed Kattan. 2011. Geometric Generalisation of Surrogate Model Based Optimisation to Combinatorial Spaces. In *Evolutionary Computation in Combinatorial Optimization*, Peter Merz and Jin-Kao Hao (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 142–154.
- [13] L. M. Pavelski, M. R. Delgado, C. P. d. Almeida, R. A. Gonçalves, and S. M. Venske. 2014. ELMOEA/D-DE: Extreme Learning Surrogate Models in Multi-objective Optimization Based on Decomposition and Differential Evolution. In *2014 Brazilian Conference on Intelligent Systems*. 318–323.
- [14] Qingfu Zhang and Hui Li. 2007. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation* 11, 6 (2007), 712–731.
- [15] Juan P. Romero, Angel Ibeas, Jose L. Moura, Juan Benavente, and Borja Alonso. 2012. A Simulation-optimization Approach to Design Efficient Systems of Bike-sharing. *Procedia - Social and Behavioral Sciences* 54 (2012), 646 – 655.
- [16] Robert Tibshirani. 1996. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58, 1 (1996), 267–288.
- [17] Sébastien Verel, Bilel Derbel, Arnaud Liefoghe, Hernán Aguirre, and Kiyoshi Tanaka. 2018. A Surrogate Model Based on Walsh Decomposition for Pseudo-Boolean Functions. In *Parallel Problem Solving from Nature PPSN XV*, Anne Auger, Carlos M. Fonseca, Nuno Lourenço, Penousal Machado, Luis Paquete, and Darrell Whitley (Eds.). Vol. 11102. Springer International Publishing, Cham, 181–193.
- [18] Sébastien Verel, Arnaud Liefoghe, Laetitia Jourdan, and Clarisse Dhaenens. 2013. On the Structure of Multiobjective Combinatorial Search Space: MNK-Landscapes with Correlated Objectives. *Eur. J. Oper. Res.* 227, 2 (2013), 331–342.
- [19] J. L. Walsh. 1923. A Closed Set of Normal Orthogonal Functions. *American Journal of Mathematics* 45, 1 (1923), 5.
- [20] Martin Zaefferer, Jörg Stork, Martina Friese, Andreas Fischbach, Boris Naujoks, and Thomas Bartz-Beielstein. 2014. Efficient Global Optimization for Combinatorial Problems. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. 871–878.
- [21] Saúl Zapotecas Martínez and Carlos A. Coello Coello. 2013. MOEA/D Assisted by Rbf Networks for Expensive Multi-objective Optimization Problems. In *the 15th Annual Conference on Genetic and Evolutionary Computation (GECCO '13)*. 1405–1412.
- [22] Q. Zhang, W. Liu, E. Tsang, and B. Virginas. 2010. Expensive Multiobjective Optimization by MOEA/D With Gaussian Process Model. *IEEE Transactions on Evolutionary Computation* 14, 3 (2010), 456–474.
- [23] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca. 2003. Performance Assessment of Multiobjective Optimizers: An Analysis and

Review. *IEEE Trans. Evol. Comput.* 7, 2 (2003), 117–132.