



Metaheuristics for (Variable-Size) Mixed Optimization Problems: A Unified Taxonomy and Survey

Prof. El-Ghazali Talbi

► To cite this version:

Prof. El-Ghazali Talbi. Metaheuristics for (Variable-Size) Mixed Optimization Problems: A Unified Taxonomy and Survey. Université de Lille; INRIA Lille Nord Europe; CRISTAL; BONUS. 2024, pp.55. hal-04381240

HAL Id: hal-04381240

<https://inria.hal.science/hal-04381240>

Submitted on 9 Jan 2024

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.



Distributed under a Creative Commons Attribution 4.0 International License

Metaheuristics for (Variable-Size) Mixed Optimization Problems: A Unified Taxonomy and Survey

Prof. El-Ghazali Talbi

Abstract—Many real world optimization problems are formulated as mixed-variable optimization problems (MVOPs) which involve both continuous and discrete variables. MVOPs including dimensional variables are characterized by a variable-size search space. Depending on the values of dimensional variables, the number and type of the variables of the problem can vary dynamically. MVOPs and variable-size MVOPs (VMVOPs) are difficult to solve and raise a number of scientific challenges in the design of metaheuristics.

Standard metaheuristics have been first designed to address continuous or discrete optimization problems, and are not able to tackle (V)MVOPs in an efficient way. The development of metaheuristics for solving such problems has attracted the attention of many researchers and is increasingly popular. However, to our knowledge there is no well established taxonomy and comprehensive survey for handling this important family of optimization problems.

This paper presents an unified taxonomy for metaheuristic solutions for solving (V)MVOPs in an attempt to provide a common terminology and classification mechanisms. It provides a general mathematical formulation and concepts of (V)MVOPs, and identifies the various solving methodologies than can be applied in metaheuristics. The advantages, the weaknesses and the limitations of the presented methodologies are discussed. The proposed taxonomy also allows to identify some open research issues which needs further in-depth investigations.

Index Terms—Metaheuristics, Mixed-variable optimization problem, Variable-size mixed-variable optimization problem, Mixed optimization, Mixed variable programming, Variable-space design, Decomposition-based optimization.

I. INTRODUCTION

Decision variables can be categorized into continuous and discrete variables. Continuous variables refer to real numbers defined within a given interval. Discrete variables can be divided into two categories: ordinal (i.e. integer, quantitative) and categorical (i.e. qualitative, nominal). According to the involved type of variables, optimization problems can be classified into continuous optimization problems (CnOPs) and discrete optimization problems (DOPs).

However, many modern real world optimization problems (e.g. engineering design, process synthesis, logistics and transportation, bio-medical, machine learning [1] [2]) are formulated as mixed-variable optimization problems (MVOPs)¹. MVOPs contain both continuous and discrete variables. The most popular families of MVOPs are:

- **Mixed-Integer Programming (MIP)**: it represents a popular family of problems in mathematical programming with various domains of application such as logistics, transportation, manufacturing, energy, finance [3]. According to the linearity of objectives and constraints, one can classify this family into *Mixed-Integer Linear Programming (MILP)* or *Mixed-Integer Non-Linear Programming (MINLP)* problems. MIP problems are generally non convex and NP-hard [4]. The challenging difficulty of MINLP problems is their non-linearity and non-differentiability due to the mixed decision space.
- **Black-box MVOPs**: they represent MVOPs problems in which the objective function and/or constraints cannot be formulated analytically. Moreover, they are generally characterized by expensive objective functions (e.g. intensive simulations). For instance, many engineering design (e.g. aerospace [5], automotive, nanotechnology, mechanics) and machine learning problems (e.g. neural architecture search) are black-box MVOPs [6].

MVOPs including dimensional variables are characterized by a variable-size decision space. This family of variable-size MVOPs (VMVOPs) is referred in the literature as mixed-variable optimization problems [7] or variable-size design space problem [8] [9]. Depending on the values of dimensional variables, the number and type of variables of the problem can vary dynamically. Dimensional variables can also influence the number and type of constraints a given solution is subject to. For instance, in engineering design problems, dimensional variables can represent the choice between several possible sub-system components. Each component is usually characterized by a partially different set of design variables, and therefore, depending on the considered choice, different continuous and discrete design variables must be optimized. In the automated design of deep neural networks, the architectures are composed of hundreds of mixed variables: continuous (i.e. learning rate), discrete (e.g. number of layers) and dimensional (e.g. type of layer: convolution, pooling, fully-connected layer) [1].

Designing efficient metaheuristics to solve complex (V)MVOPs² raises several challenges. Indeed, this family of problems are difficult to solve by standard metaheuristics according to the following distinguishing characteristics:

University of Lille and INRIA, France, e-mail: el-ghazali.talbi@univ-lille.fr.

¹Also known in the literature as mixed variable programming (MVP).

²Family of optimization problems including VMVOPs and MVOPs

- **Mixed continuous and discrete variables:** most metaheuristics have been initially designed for CnOPs or DOPs. For example, standard differential evolution (DE), evolution strategies (ES), cuckoo search (CS), firefly algorithm (FA) have been initially developed for fixed-size CnOPs, while particle swarm optimization (PSO), ant colony optimization (ACO), simulated annealing (SA), and tabu search (TS) have been developed first for handling fixed-size DOPs. Hence, their application to (V)MVOPs need new methodologies for metaheuristics design.
- **Categorical variables:** they represent elements (e.g. colors, material type) which are constrained to take values from a finite set of non-numerical and unordered values. They can be encoded as a set of discrete numbers. However, their relaxation results in non-computable objective functions and constraints, and they cannot be assigned intermediate values.
- **Dimensional variables:** they can be represented by a vector of integer values and induce VMVOPs. Indeed, each value of a dimensional variable has an impact on the number and type of the involved variables, the number of constraints and the structure of the objective function. Moreover, the discreteness of dimensional variables cannot be relaxed using continuous variables.
- **Constraint handling:** in general, the integer restriction in (V)MVOPs divides the feasible region into discontinuous feasible sub-regions of different sizes. Thus, constraint handling in (V)MVOPs represents a difficult issue to handle during the search.

The development of metaheuristics for solving (V)MVOPs has attracted the attention of many researchers and is increasingly popular. However, to our knowledge there is no well established taxonomy and comprehensive survey for handling this important family of optimization problems. Many papers concerned by MVOPs focus on a specific methodology for a given metaheuristic such as DE [10] [11] [12], ES [13] [14] [15], PSO [16] [17], ACO [18] [19] [20], estimation distribution algorithms (EDA) [21] [22], genetic algorithms (GA) [23] [24] [25], TS [26] [27], SA [28] [29] [30], artificial bee colony (ABC) [31] and so on. Some papers deal with specific families of MVOPs such as those where continuous variables are linearly constrained [7]. Moreover, very few papers deal with VMVOPs. Developing efficient metaheuristics for VMVOPs is in its infancy.

This paper presents a unified taxonomy for general metaheuristic methodologies for solving (V)MVOPs in an attempt to provide a common terminology and classification mechanisms. It provides a general mathematical formulation and concepts of (V)MVOPs, and identifies the various solving methodologies for metaheuristics that can be applied. The advantages, the weaknesses and the limitations of the presented methodologies are discussed. The proposed taxonomy also allows to identify some open research issues which need further in-depth investigations.

The paper is structured as follows. In section II, general formulations and concepts of (V)MVOPs are provided. In

section III, the main search components of metaheuristics (e.g. initialization, encodings of solutions, neighborhoods, variation operators, model construction) that have to be adapted for solving (V)MVOPs are highlighted in a unified and most general way. The following sections IV and V present the taxonomy, classify, detail and analyze the various metaheuristic methodologies for solving respectively MVOPs and VMVOPs (e.g. global versus decomposition-based, encoding adaptation versus variation operators adaptation, relaxation versus discretization, sequential versus nested versus co-evolutionary). Finally, the last section VI presents the main conclusions and opens some important research perspectives.

II. MIXED-VARIABLE OPTIMIZATION PROBLEMS

In this section, a general mathematical formulation of MVOPs is provided, and is followed by the formulation of VMVOPs.

A. Fixed-size mixed-variable optimization problems

Fixed-size MVOPs are concerned by three different types of variables: continuous, discrete ordinal, and discrete nominal [32]. Continuous and discrete ordinal variables are related to measurable values and by consequence, a relation of order between the possible values can be defined. Discrete nominal variables (i.e. categorical) are non-relaxable variables defined within a finite set of choices, in which no relation of order can be defined between the possible values.

Let us introduce a fixed-size vector of mixed decision variables (x, y) , where $x \in \mathbb{R}^{n_x}$ is a vector of n_x continuous variables, $y \in \mathbb{Z}^{n_y}$ is a vector of n_y discrete variables. The discrete vector $y = (y_o, y_c)$ contains both ordinal and categorical variables. A MVOP can be formulated as follows (Eq.1):

$$\begin{aligned} \min f(x, y), f : \mathcal{F}_x \times \mathcal{F}_y &\longrightarrow \mathbb{R} \\ \text{subject to } x \in \mathcal{F}_x &\subseteq \mathbb{R}^{n_x} \\ y \in \mathcal{F}_y &\subseteq \mathbb{Z}^{n_y} \end{aligned} \quad (1)$$

where $f(\cdot)$ is the objective function, \mathcal{F}_x (resp. \mathcal{F}_y) are the feasible region for continuous (resp. discrete) variables. The constraints are generally represented by bounding constraints $x = \prod_{i=1}^{n_x} [x_i^{\min}, x_i^{\max}]$, $y = \prod_{i=1}^{n_y} [y_i^{\min}, y_i^{\max}]$, and inequality constraints $g(x, y) \leq 0$, $g_i : \mathcal{F}_{x_i} \times \mathcal{F}_{y_i} \longrightarrow \mathcal{F}_{g_i} \subseteq \mathbb{R}$ for $i = 1, \dots, n_g$, where n_g is the number of constraints.

The feasible decision space of a MVOP can be decomposed into separated subsets according to the discrete vector y . For any fixed discrete vector $y^* = (y_1^*, \dots, y_{n_y}^*)$, the set $A_{y^*} = \{(x, y) : x \in \mathcal{F}_x, y = y^*\}$ defines an n_x -dimensional affine sub-space. These affine sub-spaces for different values of y^* are mutually disjoint. The distance δ between A_{y_1} and A_{y_2} , $\delta = \min\{d((x, y_1), (x', y_2)), \forall (x, y_1) \in A_{y_1}, \forall (x', y_2) \in A_{y_2}\}$ is greater or equal than 1 [33]. Hence, the minimization of f on $\cup_{y \in \mathcal{F}_y} A_y$ can be reduced as:

$$\min_{y \in \mathcal{F}_y} [\min_{(x, y) \in A_y} f(x, y)]$$

An exhaustive solving of the sub-problems for all $y \in \mathcal{F}_y$ is impractical, since the size of the set \mathcal{F}_y is generally large.

B. Variable-size mixed-variable optimization problems

VMVOPs include dimensional variables which are non-relaxable variables defined within a finite set of choices. The size of the decision space and the definition of the constraint functions vary dynamically during the optimization process as a function of the values of dimensional variables [7] [34]. Let us introduce z as a vector of n_z dimensional variables, a general VMVOP can be formulated as follows (Eq.2):

$$\begin{aligned} \min f(x, y, z), f : \mathcal{F}_x \times \mathcal{F}_y \times \mathcal{F}_z &\longrightarrow \mathbb{R} \\ \text{subject to } x \in \mathcal{F}_x(z) &\subseteq \mathbb{R}^{n_x(z)} \\ y \in \mathcal{F}_y(z) &\subseteq \mathbb{Z}^{n_y(z)} \\ z \in \mathcal{F}_z &\subseteq \mathbb{Z}^{n_z} \end{aligned} \quad (2)$$

where x and y are variable-size vectors of size respectively equal to $n_x(z)$ and $n_y(z)$. The number of constraints $n_g(z)$ the problem is subject to is also variable: $g(x, y, z) \leq 0, g_i : \mathcal{F}_{x_i}(z) \times \mathcal{F}_{y_i}(z) \times \mathcal{F}_z \longrightarrow \mathcal{F}_{g_i} \subseteq \mathbb{R}, \text{ for } i = 1, \dots, n_g(z)$.

The concept of global optimality is straightforward to generalize for VMVOPs.

Definition 1: A solution $(\hat{x}, \hat{y}, \hat{z})$ is a global optimum if $f(\hat{x}, \hat{y}, \hat{z}) \leq f(x, y, z), \forall z \in \mathcal{F}_z, \forall y \in \mathcal{F}_y(z), \forall x \in \mathcal{F}_x(z)$.

Local optimality for VMVOPs is less obvious to define. One has to take into account simultaneously both discrete and continuous variables [32]. Modifying the discrete variables make sense only if the continuous variables change as well. Indeed, variations of the discrete variables will involve changes in the continuous neighborhoods. A neighborhood $\mathcal{N}_c(x)$ for continuous variables x is represented by a ball $\mathcal{B}(\epsilon, x)$ of radius ϵ around x . Local optimality can be defined as: x^* is a local minimum if $f(x^*) \leq f(x)$ for all $x \in \mathcal{B}(\epsilon, x^*)$. The same definition holds for discrete variables using combinatorial finite neighborhoods which will depend on the nature of the variables of the VMVOPs to solve (e.g. binary, bounded ordinal, categorical, permutation). The discrete feasible finite neighborhood $\mathcal{N}_d(x, y, z)$ take into account the fact that changes of the discrete variables can also have an impact on the values of the continuous variables. Local optimality in VMVOPs is then related to the balls centered at the solutions belonging to the discrete neighborhood. A solution (x^*, y^*, z^*) is a local optimum if there are no better solutions in the continuous neighborhood of all the solutions belonging to its discrete neighborhood.

Definition 2: Given a discrete neighborhood $\mathcal{N}_d(x, y, z)$, a solution (x^*, y^*, z^*) is a local optimum if for all $(\bar{x}, \bar{y}, \bar{z}) \in \mathcal{N}_d(x^*, y^*, z^*), f(x^*, y^*, z^*) \leq f(\bar{x}, \bar{y}, \bar{z}), \forall (\bar{x}, \bar{y}, \bar{z}) \in \mathcal{B}(\epsilon, \bar{x}) \cap \mathcal{F}_x(\bar{z}), \epsilon > 0$.

III. MAIN METAHEURISTIC COMPONENTS IN SOLVING (V)MVOPs

Metaheuristics represent a class of general-purpose approximate algorithms that can be applied to difficult optimization problems [35]. One can classify metaheuristics into *local-search* based and *population-based metaheuristics*. Local search-based metaheuristics improve a single solution. They could be seen as search trajectories through neighborhoods. They start from a single solution. They iteratively perform

the generation and replacement procedures from the current solution s . In the generation phase, a set of candidate solutions are generated from the neighborhood $\mathcal{N}(s)$ of the current solution. In the replacement phase³, a selection is performed from the candidate solution set to replace the current solution (i.e. a solution $s' \in \mathcal{N}(s)$) is selected to be the new solution. Popular examples of such metaheuristics are local search (LS) (i.e. hill-climbing, gradient), simulated annealing and tabu search.

Population-based metaheuristics could be viewed as an iterative improvement of a population of solutions. They start from an initial population of solutions. Then, they iteratively apply the generation of a new population and the replacement of the current population. Using various bio-inspired metaphors, they may be classified into two main categories [35] [36]:

- **Evolutionary algorithms (EAs):** they are guided by the selection and reproduction concepts of survival of the fittest. The solutions composing a population of solutions (i.e. individuals) are selected and reproduced using variation operators (e.g. mutation, crossover⁴). New offsprings (i.e. solutions) are constructed from the different features of solutions belonging to the current population. The most popular EAs are genetic algorithms, differential evolution [37], evolution strategy, and estimation distribution algorithms.
- **Swarm intelligence (SI):** they are inspired by the emergence of collective intelligence from populations of agents with simple behavioral patterns for cooperation [38]. Particle swarm optimization, ant colony optimization, firefly algorithm, and cuckoo search represent examples of this class of metaheuristics.

Population-based metaheuristics can also be categorized into *regular population* and *model-based population* algorithms [39]. They are distinct in the way the offsprings are generated during the search. In regular population algorithms, the solutions are generated using population-based variation operators (e.g. mutation and crossover for GA, DE, ES, and particle update for PSO), whereas in model-based algorithms, the solutions are generated by using and adapting a model which stores information during the search (e.g. pheromone in ACO, statistical model in EDA).

The common search components for all metaheuristics which have to be adapted to solve (V)MVOPs are the *encoding* of solutions and their *initialization*. Indeed, the representation of mixed and variable-size solutions is a crucial issue and has an important consequence on the efficiency of metaheuristics. Indeed, it defines the decision space in which any metaheuristic will carry out the search process. In general, the initial solutions are randomly and uniformly generated according to the type of variables and the constraints associated to the (V)MVOP. Continuous (resp. ordinal) variables are generally defined by bounded intervals, while categorical and dimen-

³Also named transition rule, pivoting rule and selection strategy.

⁴Also called recombination and merge.

sional variables are defined by a finite domain of possible values.

In addition to encodings, another important search component to be adapted for local search-based metaheuristics is the definition of the neighborhood and its exploration. Finding a local optimal solution for (V)MVOPs (e.g. expensive black-box) according to the neighborhood definition given in Eq.1 can be prohibitive. Concerning population-based metaheuristics, the most important search components involved for solving (V)MVOPs are the variation operators for regular population-based (e.g. GA, DE, PSO) and the model construction and update for model-based algorithms (e.g. ACO, EDA).

IV. A TAXONOMY OF METAHEURISTICS FOR MVOPs

The proposed taxonomy is based on a unifying view of metaheuristics in solving MVOPs according to the main concerned search components. Using this taxonomy, a survey describes and discusses the various general metaheuristic methodologies to solve MVOPs and gives some relevant illustrative algorithms. Solving methodologies that require problem-specific knowledge of MVOPs are not considered in this paper [8]. The first criteria of our taxonomy is related to the definition of the decision space in which metaheuristics will operate. First, the algorithms can be categorized in two types (Fig.1):

- **Global approaches:** the optimization process of metaheuristics is carried out in the whole mixed decision space.
- **Decomposition-based approaches:** some partitioning of the fixed-size mixed decision space is performed. According to a given decomposition, a set of subproblems are generated and solved.

A. Global approaches for MVOPs

Global approaches consist in applying a global search on the whole decision space of mixed variables. The discrete and the continuous variables are treated simultaneously. This approach maintains the relationships between the decision variables, and involves a straightforward design and implementation. The goal here is the reuse of metaheuristics designed for CnOPs or DOPs to be easily deployed to solve MVOPs. However, the features of standard metaheuristics, and the consistency and compatibility of encodings and variation operators can be destroyed. The global approach contains two main categories: *encodings adaptation* and *variation operators adaptation*.

1) *Encodings adaptation:* In a MVOP, a solution is generally represented by a fixed-size vector of mixed variables (x, y) where x (resp. y) represents the continuous (resp. discrete variables). The most popular approach consists in mapping completely the encoding of a MVOP into a continuous or discrete encoding. The two main mapping methods are *relaxation* and *discretization*.

Relaxation: this approach consists in relaxing the discrete variables as continuous ones. Then, a MVOP is transformed into a CnOP, and any continuous metaheuristic can be applied

[40] [41]. Variation operators in those metaheuristics (e.g. neighborhoods, mutation, crossover, particle update) will generate real numbers. The conversion of real values into integer values is necessary just to evaluate the objective function and constraints, and then continuous representations are kept in the search process.

Conversion of discrete ordinal variables: the most popular and simple conversion function for discrete ordinal variables is the *truncation* function $y_i = INT(y_i)$, where $INT(.)$ is a function modifying a real value into an integer one by rounding to the nearest available value. Metaheuristics based on truncation have been widely deployed in the literature to solve MVOPs:

- **Local search-based algorithms:** it concerns mainly the continuous neighborhood definition, and has been applied to SA [28] [29] [30], TS [27], and spiral dynamics optimization algorithm [42].
- **Evolutionary algorithms:** truncation concerns the mutation and some crossover operations in DE [10] [11] [12] [43] [44] [45] [46] [47] [48] [49] [50] [51], and GA (e.g. arithmetic crossover, non-uniform mutation) [52] [53] [54] [23] [24] [25] [55] [56]. Truncation is unnecessary for some crossover operators (e.g. uniform crossover) [57] [58].
- **Swarm Intelligence:** truncation is used in the particle movement operation of PSO [59] [60] [61] [62] [63] [16] [17], CS [64] [65], harmony search (HS) [66], Grey-Wolf [67], teaching-learning-based optimization (TLBO) [68] [69], ABC [70], and FA [71]. For ACO, various statistical models are applied to deal with different type of variables [72] [73] [20].
- **Hybrid metaheuristics:** truncation can be extended to any hybrid continuous metaheuristic. Some combinations have been investigated, such as combining DE with ABC [74], DE with HS [75], ACO with chaotic optimization [76], FA with chaotic optimization [77], DE with TS [78], DE with direct search [79], TLBO with HS [69], and GA with PSO [80].

Another popular conversion method is *rounding* (Fig.1). A better discrete solution is not always selected by rounding [81]. Another conversion approach consists in checking the upper and lower discrete values of the continuous variables and selecting the best one [60]. This approach obtains better results (i.e. accuracy, convergence) especially when the range of discrete variables is important. However, it involves additional evaluations of the objective function, and then higher computational cost [82]. Other more complex value-based relaxations can be used [83] such as the nearest vertex approach (NVA) [61] and the shortest normal approach (SNA) [84].

Relaxation has two main advantages. On one hand, it does not oblige the design of new effective variation operators for discrete variables. On the other hand, the same variation operators are applied to the whole mixed vector. However, discrete variables generated by relaxation strategies do not always satisfy the constraints [85]. Relaxation can also have an influence on entering the feasible regions of a MVOP

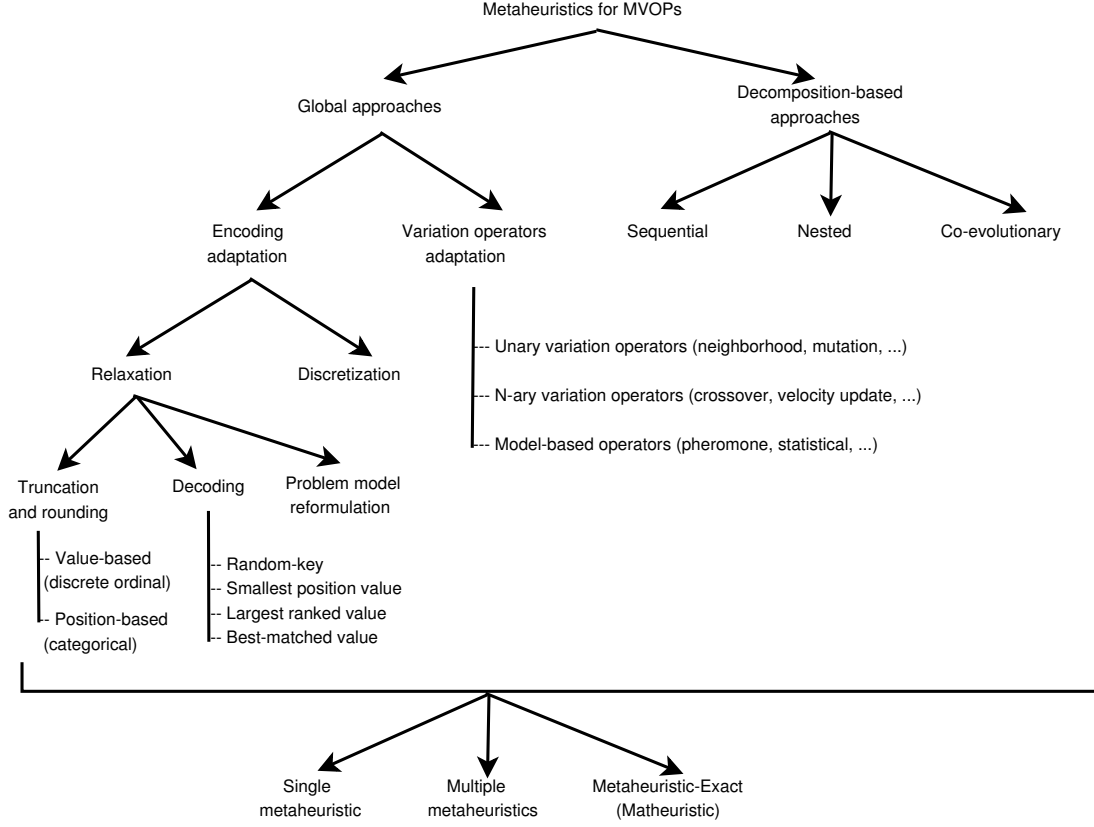


Fig. 1. A taxonomy of metaheuristic methodologies for solving MVOPs.

which are characterized by discontinuous feasible regions. The search process tends to enter the largest feasible regions. Once entering a non-optimal large-size feasible region, it is difficult to escape towards small-size feasible regions. In some problems, the global optimal solutions may be localized in small-size feasible regions, which are neglected.

The encoding adaptation based on relaxation will lead to a biased search procedure and potentially generates a harder optimization problem. The extended continuous search space contains a large number of infeasible solutions. Hence, the search process can generate solutions which are far from the feasible region, and needs a higher computational cost. Moreover, the transformed MVOPs are characterized by a huge number of local optima, which makes them more complex to solve by continuous metaheuristics [86]. Another drawback of relaxation is that it introduces flat areas in the fitness landscape of MVOPs, for which metaheuristic search is less efficient.

Conversion of categorical variables: relaxation strategies have also been applied to treat categorical variables. However, it does not work effectively for categorical variables for which there is no ordering relation between their values, and are not numerically related. There is no sense on carrying out arithmetic operations on categorical values. Hence, the relaxation of categorical variables using value-based conversion is less natural and may decrease the performance of metaheuristics as the number of categorical variables increases [20]. For some MVOPs, the categorical values are associated to a set

of continuous values which can be used during the search. *Position-based encoding* is more adapted to encode categorical variables [87]. Let $Z = \{z_1, \dots, z_i, z_n\}$ be a set values for a given categorical variable. One can use the position (i.e. index) of the discrete variable instead of the discrete value [88]. Then, a continuous variable associated to a categorical variable can be divided into equal intervals which are associated to the index values of the categorical variable. Uniform-spacing integer positions from 1 to n are used and replaced by continuous position variables. Then, the truncation and rounding strategies can be applied for such position-based encodings. For instance, This approach has been investigated in DE [10] and threshold accepting (TA) algorithms [89]. This strategy is based on the assumption that neighboring categorical values contribute similarly to the objective function, which is not always the case. In many categorical variables, the values define qualitative distinctions.

This approach has also been adapted for binary variables [43] [90]. For instance, one can use sigmoid-based conversion by following a logistic probability distribution: $z = 1$ if $r \leq P(x)$, 0 otherwise where $P(x) = \frac{1}{1+e^{-x}}$, r is a random term following a uniform distribution in the interval $[0, 1]$, z is binary variable, and x is the continuous variable to transform. This approach has been integrated in DE [91], PSO [92] and FA [93]. Other functions such as trigonometric function can also be applied [94].

Conversion of order-based encodings: truncation and rounding cannot be applied to some discrete encodings such as

order-based encodings (e.g. permutation). An increasingly popular relaxation method is based on *decoding*. The most popular decoding approach is the *random-key representation* [95]. It transforms a position in a continuous space into a position in a discrete space. It has been used first for permutation-based DOPs (e.g. scheduling [95] [96], routing [97], assignment [98]). Random-key encodings have been used in most of the metaheuristics such as PSO [95] [97] and DE [96]. Other decoding methods have been used in the literature such as the smallest position value (SPV) rule into PSO [99], HS [100], FA [101], CS [102], the largest ranked value (LRV) rule [103], and the best-matched value (BMV) rule [44].

Handling discrete variables in relaxation approaches can also be handled through the *problem model reformulation*. It is generally carried out by penalty constraints through which the correct values of discrete variables cannot be reached. A classical relaxation method for categorical variables encodes them by continuous ones and includes some constraints in the model. This is a popular reformulation methodology in mathematical programming [87] [104] [105] [106]. For instance, the addition of the following non-convex nonlinear function $z(1 - z) = 0, 0 \leq z \leq 1$ to model a binary variable will force z to take either 0 or 1 [107]. In [108] [82], using PSO, the authors incorporate a dynamic penalty approach into the objective function to handle discrete variables as continuous variables.

Discretization approaches: this mapping function consists in discretizing the search range of continuous variables so that they can be optimized as discrete variables. It facilitates the use of standard metaheuristics and variation operators originally developed for DOPs:

- **Genetic algorithms:** it has been applied in simple GAs which encode all continuous variables as binary strings [109] [110] [111] [112] [113] [114]. Thus, the standard binary crossover and mutation variation operators can be applied on continuous variables.
- **Estimations of distribution algorithms:** it is possible to transform each continuous parameter into a discrete one and then use classical statistical models such as Bayesian networks. However, with the increasing precision of continuous variables, the complexity of the statistical models grows exponentially.
- **Local search-based algorithms:** the discretization of the variables facilitates the definition of a discrete neighborhood. Then, any discrete local search-based metaheuristic such as SA [115] can be applied to solve the MVOP as a fully discrete one.

This approach is limited in terms of accuracy. For some MVOPs, continuous variables require high accuracy, and the precision of continuous variables is restricted and suffers from limited solution accuracy. For instance, using the binary discretization, the performance of the metaheuristics may be decreased by the high dimensionality of the binary decision space especially when high accuracy is required.

Although the methodology based on encoding adaptation is easy and simple to design and implement, its efficiency

can suffer from losing important knowledge by applying successive mappings of a continuous (resp. discrete) space into a discrete (resp. continuous) space.

2) *Variation operators adaptation:* This approach does not apply any transformation of the mixed-variable encoding scheme. Instead of modifying the encoding, new variation operators are introduced to handle the type(s) of decision variables that cannot be handled originally by a given metaheuristic [116] [117]. It is based on the adaptation of the variation operators of standard metaheuristics to handle both continuous and discrete variables separately (see Fig.2 and Fig.3). In this combined variation method, the continuous and the discrete variation operators are carried out simultaneously to generate new solutions. Different variation operators will handle different type of variables, and then there is no more restriction to deal with mixed-variable encodings.

Three main design questions arise in the adaptation of variations operators:

- The definition of the variation operators to handle the different types of variables.
- The selection of the variation operators to apply at each iteration of the algorithm.
- The selection of the variables on which to carry out the variation operator.

According to the various families of variation operators, the following methodologies have been investigated:

- **Unary operators (e.g. neighborhood, mutation):** the main issue in local search-based algorithms is the handling of the large mixed neighborhood and the exploration of the various neighborhoods. One can use a complete generation of the mixed neighborhood and applying a first improvement strategy such as proposed in [118] for variable neighborhood search (VNS). However, it is impracticable to consider all neighbors of all neighborhoods. Deterministic or random selection of neighborhoods may be applied. In [119], a local search (LS) algorithm is proposed, in which combinatorial and continuous neighborhoods are treated separately using a random strategy. A random generation of neighbors is carried out for both neighborhoods. In [120], at each iteration a random selection of neighborhood is performed. Then, a SA algorithm is applied for the discrete neighborhood and a non-linear simplex of Nelder and Mead [121] is applied for the continuous neighborhood. In [26], a TS algorithm is proposed, in which the mixed neighborhood is dynamically partitioned into equal-size regions, and a set of random neighbor solutions are generated.

For the mutation operator in EAs, various mutations for continuous, discrete ordinal and categorical variables can be defined and simultaneously applied in a probabilistic way [122] [123] [124] [125] (Fig.2). In [126] [127], a combination of continuous DE with discrete DE is proposed. The continuous variables are initialized and evolved through the original DE, whilst the discrete variables are initialized and evolved through a discrete DE. In [128], the authors combines mutation operators of

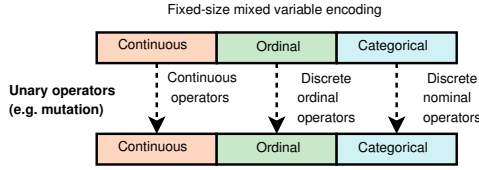


Fig. 2. Adaptation of mutation operators in global approaches for handling MVOPs.

ES in the continuous domain [129], integer domain [122], and binary domain [130]. In opposition to mutation operators in standard ES, the mutation operators for mixed-integer spaces works with a heterogeneous distribution, combining the l_2 symmetric normal distribution for continuous variables, a l_1 symmetric geometric distribution for discrete ordinal variables, and a uniform distribution for the categorical values. The same methodology has been applied to CMA-ES [15], EP [131] and GA [132].

- **N-ary operators (e.g. crossover, particle update):** the different standard N-ary variation operators for continuous and discrete variables are separately and independently applied to generate a part of the encoding. Then, the various parts of encodings are combined to construct a complete solution. For the crossover operators in EAs, most of the standard operators can be used without any change (Fig.3). In [124], the authors proposed an orthogonal crossover combined with the DE crossover. The same principle has been applied in ES [13], in which the standard dominant recombination (resp. intermediate) crossover operators is performed to discrete (resp. continuous) vectors [14].

In the particle update operator of PSO, various operators, respectively for continuous and discrete variables, are simultaneously applied to the mixed vector [133] [134] [135]. Specific operators have been designed for discrete variables [136] [137], binary [138] [133], and categorical variables [139] [111]. A random selection of particle update operators in PSO is performed (i.e. continuous or discrete) in [140]. Different priorities can be assigned for variables. In [117], a higher priority is associated to categorical variables. Among categorical variables, a random selection is performed. Subsequently, the discrete variable are prioritized as a function of both their performance and their feasibility with the constraints.

Other swarm intelligence metaheuristics have been investigated for MVOPs. In [141], the authors adapt the HLO metaheuristic (Human Learning Optimization) in which real-coded parameters are optimized by the continuous linear learning operators of continuous HLO while the rest variables of problems are handled by the binary learning operators of HLO.

Model-based operators (e.g. pheromone, statistical model): in model-based metaheuristics such as ACO and EDA, the concerned operation is the construction and the update of the model. In ACO, one has to extend the initialization and update of the pheromone model for

CnOPs. In DOPs, the ants construct a solution by making a probabilistic decision according to some discrete probability distribution (i.e. pheromone). In case of CnOPs, the domain changes from discrete to continuous. Instead of a discrete probability distribution, the Probability Density Function (PDF) is used [18]. The ants generate a random number according to a certain PDF $P_i(x_i)$. The construction for continuous variables is followed by discrete variables [19]. In [20], the continuous and ordinal variables are treated in a similar way as in the original ACO, while the categorical variables values are fixed through a random sampling which depends on the number and the performance of the solutions of the archive that use the given value. Concerning the EDA metaheuristic, the statistical models have been adapted to deal with continuous variables (e.g. Gaussian network [142], Gaussian kernel [143], decision trees [144]). In [21], EDA has been extended by effective histogram models to handle mixed variables. In [22], the authors propose a variable-width histogram to deal with continuous variables. The continuous space is decomposed into discrete bins, in which continuous variable are sampled.

The variation operators are not necessarily inherited from the same metaheuristic. Using heterogeneous variation operators from different metaheuristics has been investigated in the literature. In [145], the authors propose an efficient hybrid approach based on GA and PSO combining integer and continuous variation operations of GA and PSO (mutation, crossover, particle update). In [146], DE and GA variation operators are handling respectively the continuous and discrete parts. In [147], a GA is sequentially combined with pattern search. A set of elite solutions found by the GA are improved using pattern search.

There are two main advantages using the variation operators adaptation methodology [148] [21]. First, no transformation of the mixed encoding is necessary to convert the variables, which generates higher computational cost. Second, different variation operators designed for various types of variables can be combined. Those variation operators are inherited from the same metaheuristic or not. Hence, this methodology does not need any major design change of metaheuristics. Many well-studied variation operators for CnOPs and DOP can be simply combined to solve MVOPs. Different probabilities can be assigned to the various variation operators [149]. Although this methodology has no limitations on the mixed type of variables, there is no guarantee that the variation operators for different types of variables are compatible and consistent. Therefore, this class of algorithms can be inefficient in guiding the search in mixed spaces.

B. Decomposition-based approaches for MVOPs

Decomposition approaches consist in partitioning the mixed decision space according to the type of variables. The discrete and the continuous variables are not treated simultaneously. This approach alternates iterations for which different types of variables are optimized. Decomposition strategies which

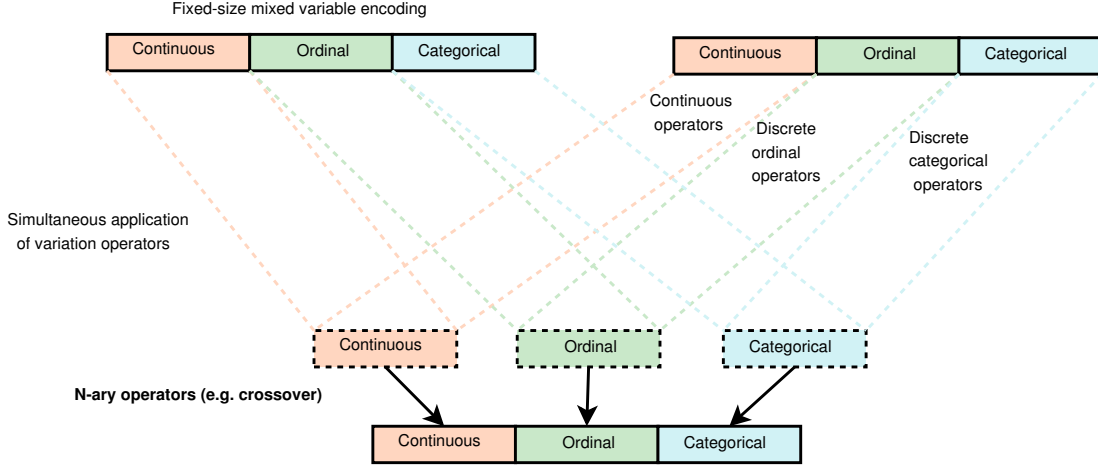


Fig. 3. Adaptation of crossover operators in global EA approaches for handling MVOPs.

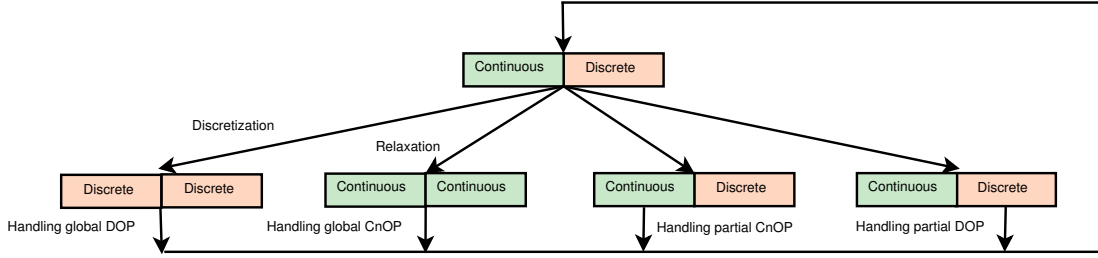


Fig. 4. Building blocks for decomposition-based sequential approaches in handling MVOPs.

use problem-specific knowledge of the MVOP are not considered in this section [62]. There exists three main general decomposition-based approaches: *sequential*, *nested*, and *co-evolutionary*.

1) *Sequential approaches for MVOPs*: In the sequential approach, the generated subproblems are solved in a sequential way. The subproblems are generated according to a given decomposition of the mixed decision space (Fig.4). Two design questions arise in this methodology:

- **Sequence of optimization**: this question concerns the order in which the different types of decisions variables are optimized.
- **Complete or partial optimization**: in each sequential phase, the optimization process may concern a given partition of the decision space or a relaxation/discretization of the whole decision space.

According to those design questions, many sequential approaches may be designed to solve MVOPs. A popular approach consists in performing first an optimization of the continuous variables, followed by the optimization of the discrete ones:

$$\begin{aligned}
 \min f(x, y^*) &\xrightarrow{x^*} \min f(x^*, y) \\
 \text{w.r.t } x \in F_x &\xleftarrow{y^*} \text{w.r.t. } y \in F_y \\
 \text{s.t. } g(x, y^*) \leq 0 &\quad \text{s.t. } g(x^*, y) \leq 0
 \end{aligned} \quad (3)$$

In [150] [151] [152] [7], the common methodology, called *generalized pattern search*⁵, consists in alternating between

a partial continuous and partial discrete search according to various neighborhoods. The size of the neighborhood is an important issue in this family of algorithms. In [153] [150], the neighborhood size varies during the search as a function of the quality of solutions obtained during the previous iterations. The continuous phase can also handle the whole decision variables by using relaxation. In [154], the sequential approach handles a continuous optimization of the whole decision space by relaxing the subset of discrete variables. Next, the discrete variables are rounded-off and fixed, and a continuous optimization is performed over the subset of continuous variables.

Another straightforward sequential approach consists in performing successively a discrete and continuous optimization process [155]. In [156], a two-phase sequential approach is proposed: Sequential(ACO(Discrete, Partial), DE(Continuous, Partial)). First, ACO performs a search in the discrete decision space F_y . A set of elite solutions are archived. Once the discrete values are fixed, the second phase applies a DE to optimize the continuous variables in the subspace F_x . Another complementary sequential methodology, Sequential(Meta(Discrete, Complete), Meta(Continuous, Partial)), has been investigated in [157] [52]. It consists first in discretizing the continuous variables with some specified resolution, and the problem is globally solved as a DOP, using for instance SA [115] and GA [52]. Subsequently, the discrete variables are frozen, and a gradient is applied over the continuous variables to solve the CnOP subproblem [115], or a global relaxation of the MVOP [52].

⁵Also called mesh-based local search.

Memetic approaches have also been considered. In [158], a global approach based on continuous relaxation is first applied. Then, a local search algorithm, based on Nelder-Mead algorithm, is applied on the subset of continuous variables, by fixing the values of discrete variables. Matheuristics combining metaheuristics and mathematical programming can also be considered. In [159], branch-and-bound (B&B) is sequentially combined with PSO to solve MIPs. First, B&B is applied to define a feasible initial solution for PSO. Whenever, an improvement of the global best solution is found by PSO, this solution is passed to B&B as a starting solution.

Decomposition-based sequential strategies allow the reuse of standard metaheuristics in solving the subproblems. However, they ignore the correlation between the different types of variables by evolving separately the continuous and the discrete variables. Hence, the characteristics of the applied metaheuristics, the consistency and compatibility of their search operators (e.g. variation operators) can be destroyed. Moreover, the convergence of such methodology can be prohibitively slow and needs many sequential iterations. Instead of using only the type of variables as a criterion of decomposition, some strategies use some knowledge related to the structure properties of a MVOP to carry out the partitioning of the decision variables [160].

2) *Nested approaches for MVOPs*: Nested approaches are generally based on a two-level decomposition strategy in which the entire domain of decision variables is partitioned into two levels, one involving the continuous variables and the other involving the discrete variables. The subproblems are solved in a hierarchical way, in which a subproblem associated to the continuous (resp. discrete) variables is optimized within the inner loop and the remaining discrete (resp. continuous) one at the outer loop level. Variables in one level are optimized for fixed values of the variable from the other level. Solutions at the outer subproblem can be evaluated once the inner problem is solved (Fig.5). Hence, decision variables of a given partition are optimized independently for fixed values of the decision variables of the other partition [161] [7].

Decomposition-based nested approach represents a major approach in mathematical programming: *bi-level programming* [162] [163] [164], *Danzig-Wolfe decomposition* for MILP (Mixed Integer Linear Programming) [165], the *generalized Benders' decomposition* [166] and the *outer approximation method* [167] for MINLP (Mixed Integer Non-Linear Programming) [168]. In general, nested mathematical programming methodologies have to solve relaxed problems in which the integer constraints are dropped, or they solve a serie of non-linear optimization problems in which some integer values are fixed. These methodologies are complex and they require the objective functions and constraints to be differentiable which restricts their application to many real-life problems [33].

The nested approach on the discrete variables y at the outer

level can be formulated as follows:

$$\begin{aligned}
 & \min f(x^*, y) \\
 & \text{w.r.t } y \in F_y \\
 & \text{s.t. } g(x^*, y) \leq 0
 \end{aligned}
 \tag{4}$$

$$\begin{aligned}
 & \text{with } x^* = \operatorname{argmin} f(x, y) \\
 & \text{w.r.t } x \in F_x \\
 & \text{s.t. } g(x, y^*) \leq 0
 \end{aligned}$$

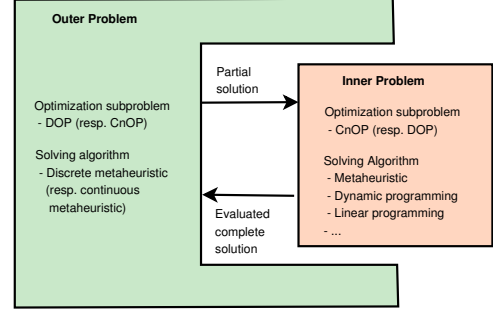


Fig. 5. Nested decomposition-based approaches for handling MVOPs.

Various algorithms in solving CnOPs and DOPs can be carried out to properly handle both optimization levels:

- **Combining metaheuristics**: a straightforward approach consists in applying metaheuristics (e.g. single or multiple) at both levels. Hence, a discrete (resp. continuous) metaheuristic deals with the discrete (resp. continuous) outer problem and another inner-level metaheuristic focuses on the continuous (resp. discrete) one. The notation *Nested(Outer-Algo(Outer-Problem-Type, Inner-Algo(Inner-Problem-Type))* specifies the type and optimization algorithms used at the two levels. Some nested approaches have been proposed in the literature such as *Nested(PSO(Discrete,GA(Continuous)))* [169], *Nested(EA(Discrete,Gradient(Continuous)))* [170], and *Nested(ACO(Discrete,Gradient-based(Continuous)))* [171]. In [58], the same metaheuristic, an EA, is used at both levels. In [170], the algorithm first separates the discrete and continuous variables of the MVOP. An outer subproblem is solved using EGO (Evolutionary Global optimization), a surrogate-based evolutionary approach to solve expensive optimization problems [172]. Considering the discrete variables fixed, a gradient-based optimizer is applied for inner CnOPs. It may be possible that for some initial discrete solutions, the continuous optimization may not find a feasible solution.
- **Combining metaheuristics and mathematical programming**: a popular and obvious approach for solving MINLP (Mixed Integer Non-Linear Programming) problems defines the outer problem (i.e. master problem) by the discrete variables. Discrete metaheuristic approaches are then used to solve the master problem (e.g. GA [173] [174], DE [175], Hyper-heuristic [176], PSO [177]). The inner problem includes only continuous variables. If the inner problem can be reduced to a linear programming (LP) problem, an exact linear

solver can be applied [176] [178] [174] [177]. The same exact methodology has been used within an inner discrete subproblem solved by dynamic programming (DP), and a master continuous problem solved by DE (i.e. Nested(DE(Continuous,DP(Discrete)))) [175].

The hierarchical decomposition of the variables could be based on other criteria than the type of variables. In [179], the problem is hierarchically decomposed into many low-dimension subproblems by using some problem-specific knowledge, and PSO is used to solve each sub-problem.

Nested decomposition-based optimization allows the reuse of efficient metaheuristics for solving CnOPs and DOPs subproblems, but ignores the correlation between the continuous and discrete variables and interrupts the co-evolution of variables. Indeed, as the interdependence between the variables is not explicitly carried out, this approach is predisposed to converge towards sub-optimal and non feasible solutions. Moreover, this approach often leads to an important number of function evaluations. Hence, an efficient inner metaheuristic must be used. In [33], while the outer discrete problem is solved by a GA, an efficient modified grid search is applied to the continuous inner problem to speedup the search. An interesting idea to reduce the complexity of solving inner problems is to avoid making further iterations if the quality of the current solution remain far to the best found solution.

3) *Co-evolutionary approaches for MVOPs*: Co-evolutionary algorithms have been successfully applied for various optimization problems such as CnOPs, DOPs [180], and bi-level problems [181]. Recently, some algorithms have been investigated for MVOPs. Co-evolutionary algorithms decompose a MVOP into multiple subproblems, and each subproblem is solved in a parallel way using a separate search process (e.g. generally an EA) to generate a partial solution [182]. Those multiple search processes cooperate to construct a complete solution for the problem by assembling the partial solutions found.

The main design questions for co-evolutionary approaches are (Fig.6):

- **Decomposition strategies**: an important issue in co-evolutionary approaches is the decomposition of the mixed vector of decision variables. A straightforward decomposition strategy of MVOPS consists in partitioning the decision space according to the type of variables (i.e. continuous and discrete subspaces). Constraint handling can also play a role into the decomposition strategy. In [183], a constrained MVOP is transformed into an unconstrained problem using augmented Lagrangian and then formulating the MVOP within a Min-Max formulation using additional decision variables (i.e. Lagrangian multipliers).

To minimize the relationships between the subproblems, the decomposition should also be based on the variables interaction [184]. Indeed, if the interacting decision variables are not in the same subproblem, the search tends to converge towards local optima [185]. Many partitioning strategies based on variable interaction may be designed:

- **Static versus dynamic decomposition**: most of the proposed decomposition strategies for MVOPs are static. In static strategies, a priori decomposition of the variables is carried out and is fixed during the search, while in dynamic strategies the decomposition of the mixed decision space may be updated during the search. Dynamic decomposition allows to define the number of subproblems and their associated variables in an adaptive way, by learning the extracted knowledge related to variable interactions [186].
- **Disjoint versus overlapped decomposition**: in general, each decision variable belongs to a single subproblem. Overlapping the variables into subproblems can be an interesting strategy if multiple interactions between the variables occur [187]. A complete overlapping of variables has been used in [188], where various metaheuristics (i.e. DE, PSO, GA, and SA) are cooperating in parallel to solve the same VMOP.
- **Knowledge-based decomposition**: some proposed approaches are application-specific and include some knowledge on the MVOP to decompose the MVOP [189]. In this case, the application expert uses some knowledge on the variable interaction. For instance, in complex engineering design problems, the well known correlation between the design variables are used to generate manually the subproblems [190]. Some other MVOPs involve the joint optimization of various optimization problems. In [191], the MVOP involves two problems namely group scheduling and job scheduling within each group.
- **Optimization algorithm**: a metaheuristic is associated to solve a given sub-problem (i.e. subset of decision variables). Each metaheuristic applies variation operators to each sub-problem to generate new solutions. The most popular metaheuristic algorithms used are based on EAs (e.g. GA [190] [191]). Some proposed co-evolutionary approaches use heterogeneous metaheuristics. In [183], two different metaheuristics, namely PSO and artificial immune systems (AIS), are used to solve the subproblems.
- **Cooperation strategies**: in addition to the decomposition strategy, the cooperation between the subproblems is a key issue [192]. The evaluation of a solution of each subproblem requires the cooperation between the subproblems. The selection of solution(s) to be communicated can be either solution-centric or population-centric [193]. Most of the proposed strategies proposed in the literature select a single solution. Many strategies can be used such as the best solution [194], worst solution [195], random solution [196], roulette/tournament selection [197], and elite solutions [198].
- **Evaluation of solutions**: the evaluation of a subproblem solution is approximated based on how well the solution cooperates with other subproblems to generate a complete solution. In general, a solution in a given subproblem is evaluated by using a single solution from each subprob-

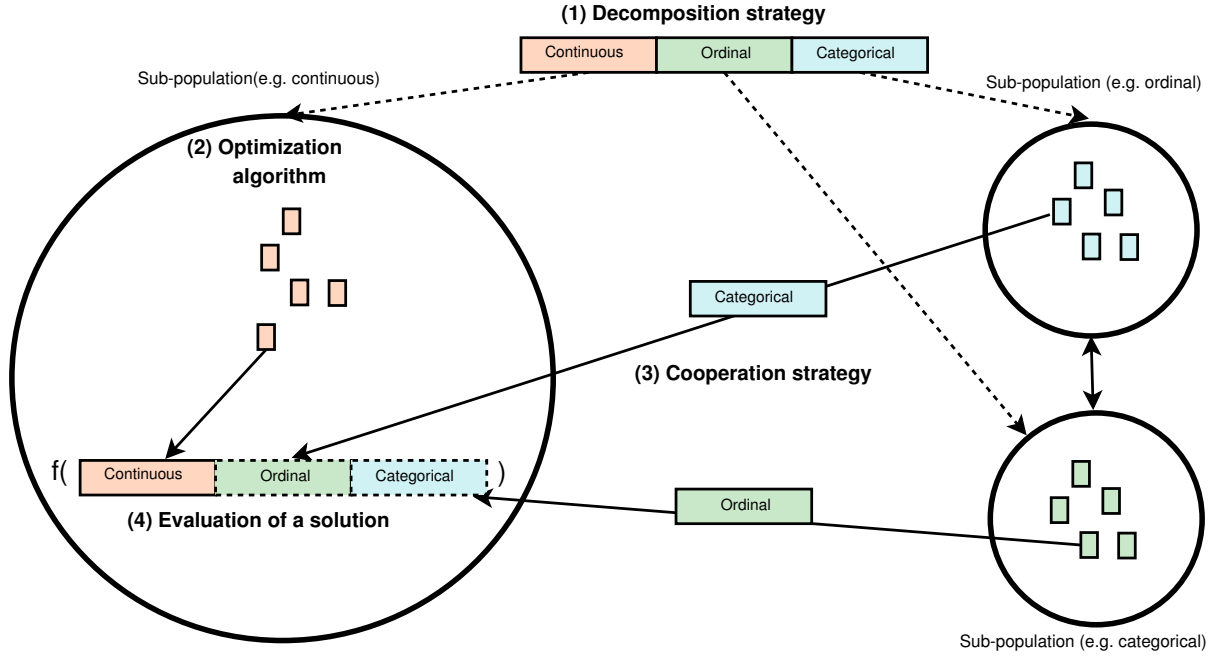


Fig. 6. Decomposition-based co-evolutionary approaches for handling MVOPs.

lem. For instance, in [183], each metaheuristic communicates the *best solution* found according to its subproblem. Then, the j th solution $U_{i,j}$ of the i th subpopulation is evaluated as following: $f(V_1, \dots, V_{i-1}, U_{i,j}, V_{i+1}, \dots, V_n)$ where V_i is the best solution from the i th subpopulation.

Instead of constructing a single complete solution to evaluate, some evaluation strategies construct multiple complete solutions to assess the evaluation of solutions into a given subproblem. The most representative strategies are the *complete cooperative selection* and the *archive-based cooperative selection*. In the complete cooperative selection strategy, each solution in a subproblem is evaluated by assembling it with all candidate combinations from other subproblems [199], while in archive-based cooperative selection, solutions are selected from a given archive (e.g. non-dominated solutions [200]). Then, those multiple evaluations are used into a Pareto dominance framework (e.g. quality, diversity) [200] or aggregated to estimate the quality of the solution (e.g. best, worse, average) [195].

Co-evolutionary algorithms have many advantages. Parallel solving of sub-problems allows to speedup the search and solve large scale MVOPs. It encourages to maintain the diversity of solutions. It allows also to improve the robustness and adaptability in solving dynamic MVOPs [201]. However, co-evolutionary algorithms can be trapped by local optima. Indeed, the speed of convergence and state of each subproblem may be inconsistent [202]. The subproblems may be different in size and complexity. The allocation of computational resources to the subproblems is also a crucial issue in the design of efficient co-evolutionary algorithms. There is a possibility that subproblems with small search space size or complexity converge first or trapped by local optima, while

subproblems with large search space or higher complexity require more iterations to converge towards good quality solutions. Some machine learning strategies (e.g. Q-learning) can be applied to adapt the size or the budget (e.g. population size) of each subproblem [202] [203].

According to the proposed taxonomy, combining multiple approaches such as global and decomposition-based approaches can be performed. Some methodologies based on multiple approaches have been investigated in the literature to solve MVOPs. In [51], a hybrid approach combining a global approach (i.e. relaxation with DE) and a decomposition-based approach (i.e. sequential) is proposed. The combination between global and decomposition-based nested approaches has been investigated in [204] [62] [115], by using various metaheuristics such as ES, PSO and gradient algorithms.

V. A TAXONOMY OF METAHEURISTICS FOR VMVOPS

Many real-world applications are formulated as variable-size MVOPs (VMVOP) such as multidisciplinary design optimization (MDO) [205], classification [206] [207], grouping problems [208]. MDO involves mutually dependent disciplines that must be synchronized. For instance, in the design of an aircraft wing, the following variables are involved: the number of spars in the wing (integer), their location (continuous), and the type of material used (dimensional). Moreover, many new problems in automated deep machine learning (AutoDeepML) such as neural architecture search (NAS) and hyperparameter optimization represent an important family of VMVOPs [1] [209]. Variable-length optimization problems (VOPs) in which there is a single type of variables is a sub-family of VMVOPs.

If the size of the dimensional space is reasonable, its *complete exploration* can be performed (Fig.7). Hence, the

VMVOP will be transformed into a set of independent MVOPs. Indeed, for every solution of the dimensional space, one can generate a fixed-size MVOP:

$$\begin{aligned} &\text{for all } z_i \in \mathcal{F}_z = \{z_1, z_2, \dots, z_{n_z}\} \\ &\min f(x, y, z_i), f : \mathcal{F}_x(z_i) \times \mathcal{F}_y(z_i) \longrightarrow \mathbb{R} \quad (5) \\ &\text{s.t. } x \in \mathcal{F}_x(z_i) \subseteq \mathbb{R}^{n_x}, y \in \mathcal{F}_y(z_i) \subseteq Z^{n_y} \end{aligned}$$

In that case, any metaheuristic approach for MVOPs can be applied for solving this set of MVOPs. Even though this complete methodology is massively parallel, it still prohibitive in terms of computational time if the size of the dimensional space is not small. Indeed, this approach is unfeasible for VMVOPs characterized by a large dimensional space and/or expensive VMVOPs where a complete enumeration of the dimensional space is computationally intractable.

This approach has been used in engineering design problems such as launcher design [210]. Once all MVOPs are solved by a GA, all the results are aggregated to generate the best found solutions. Non feasibility of some dimensional variables values (e.g. incompatibility between dimensional choices) allows to reduce the number of MVOPs to solve. Human expertise and application specific considerations can also be used to reduce the dimensional space. The computational budget (e.g. number of iterations of the metaheuristic, number of objective function evaluations) assigned to solve each MVOP is an important issue. For instance, it is interesting to assign more computational budgets to more promising MVOPs. In the following, we will extend the taxonomy for VMVOPs in the case where a *partial exploration* of the dimensional space is applied (Fig.7).

A. Global approaches for VMVOPs

In global approaches for VMVOPs, a number of additional challenges arise because of the presence of dimensional variables and the dynamically varying decision space.

1) *Encodings adaptation*: In VMVOPs, the natural encoding of solutions is defined as a variable-length vector of mixed variables. To reuse standard variation operators, some fixed-size representations have been proposed such as *hidden genes* and *indirect encodings*.

Hidden genes⁶: this fixed-size encoding has been introduced in [211] [212]. A solution is represented by two structures: a global mixed vector which encodes all mixed decision variables of a VMVOP, and a binary tag vector which specifies which variables are expressed (Fig.8a). The global mixed vector contains the maximum number of variables that characterizes the VMVOP at hand. Not all decision variables are taken into account in the evaluation of a given solution. To each variable is associated a binary tag. If the tag is not activated, the corresponding decision variable will not participate to the evaluation of the solution [213]. For a given VMVOP, there exists various ways to define dimensional variables in order to describe the global and the tag vectors [214]. This approach has been mainly used in EAs (e.g. GA [203], DE [215] [216]) but can be naturally extended to other families of metaheuristics such as PSO [217].

⁶Also called non coding segments.

As the representation of solutions has a fixed length, the standard variation operators of metaheuristics can still be performed. All the elements of the two vectors, even the hidden ones, take part in the variation operators. Unary variation operators (e.g. neighborhoods, mutation) will alter both the value of the mixed decision variables and the tag binary values. Crossover operators act in both vectors: global and tag vectors (Fig.8b). It can also be applied separately [218]. Statistical models in model-based metaheuristics (e.g. EDA, ACO) can also be easily constructed [216].

This approach is simple and easy to implement. However, the variation operators can generate inconsistent and non feasible solutions, which result in additional computational time and ineffective numerical operations. Moreover, for VMVOPs containing a large number of dimensional variables, the global mixed vector might become substantially large.

Indirect encodings: indirect fixed-size encodings (e.g. permutations, matrices) represent partial implicit representations of variable-size mixed solutions [219]. A problem-specific decoding procedure must be developed to map the partial implicit fixed-size representation to a complete explicit variable-size representation. Indirect encodings may define a many-to-one mapping, in the sense that many indirect encodings may represent the same solution. Once the fixed-size indirect encodings defined, standard variation operators can be applied. For instance, this approach has been integrated in ACO [220], EA [221] [222], DE [223], and memetic algorithms [224] to solve various VMVOPs.

B. Variation operators adaptation

Variable-length encodings may be represented by *linear* or *non-linear representations*. In a linear representation, all variables are at the same level. It is well adapted for VOPs characterized by a single type of variables (e.g. all continuous, all binary, all discrete ordinal). Many papers in the evolutionary computation (resp. swarm intelligence) literature address this family of encodings which is called variable-length chromosomes (VLC) [225] [226] (resp. variable number of dimensions (VND) [227] [228]). Standard variation operators can be extended easily for such encodings, in case the position independent encoding of variables is ensured in the linear representation.

Mutation including operators such as adding and deleting variables has been introduced in [229]. Adapted crossover operators have been proposed in [216] [211] [230]. For instance, various crossover operators preserving high-quality building blocks in variable-length linear representations have been proposed in [231] [232] [233] [234] [235] [236]. In [232], the authors introduce a synapsing variable-length crossover (SVLC). In the SVLC crossover, the common components of the parents are preserved in the offsprings, and the different components are exchanged or removed, independent of the length of such differences. Based on the common sequence similarity between the parents, the operators select non random crossover points. Movement update operations (e.g. PSO) have also been adapted to variable-length chromosomes [237] [238]

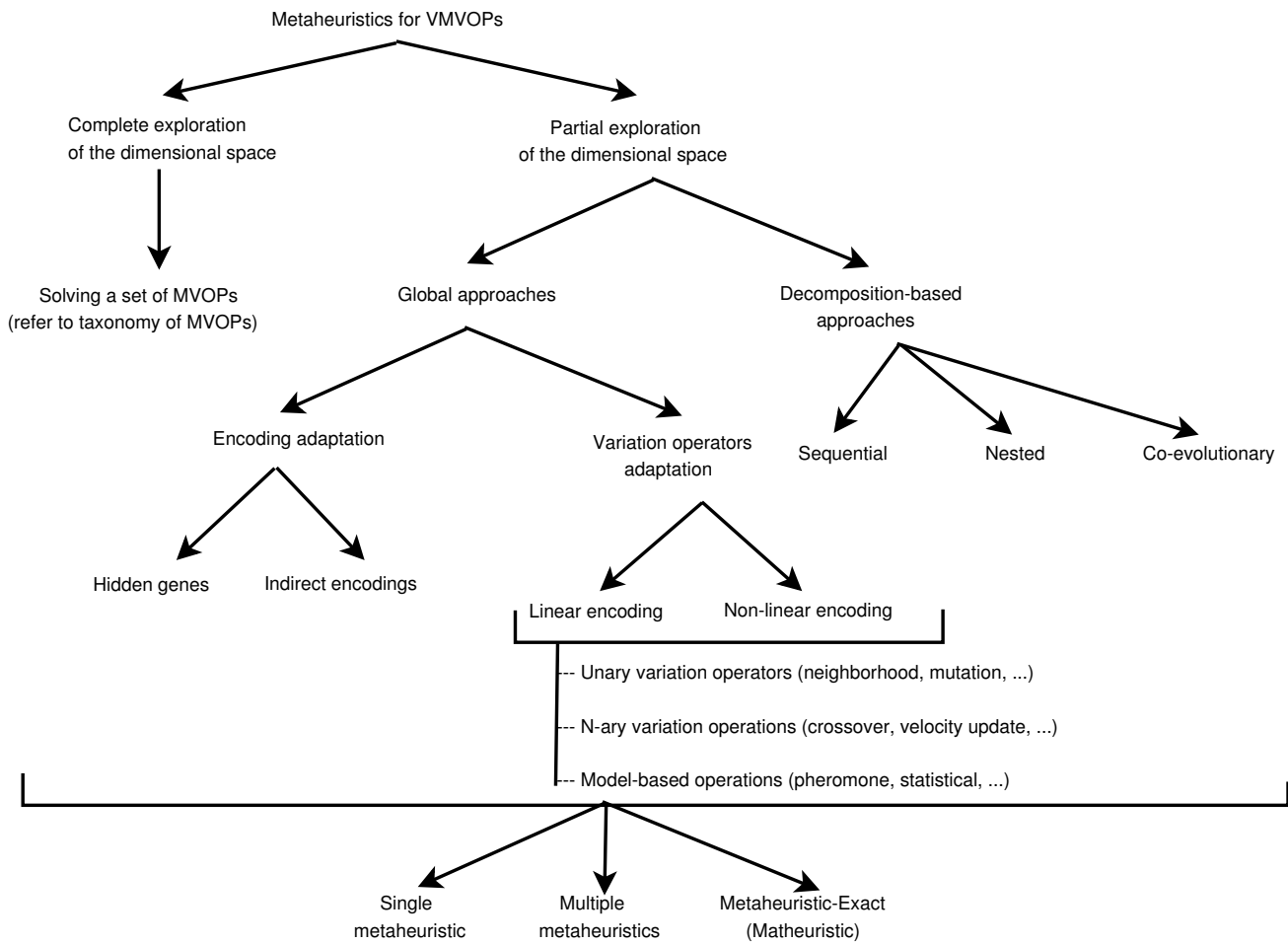


Fig. 7. A taxonomy of metaheuristic methodologies to solve VMVOPs.

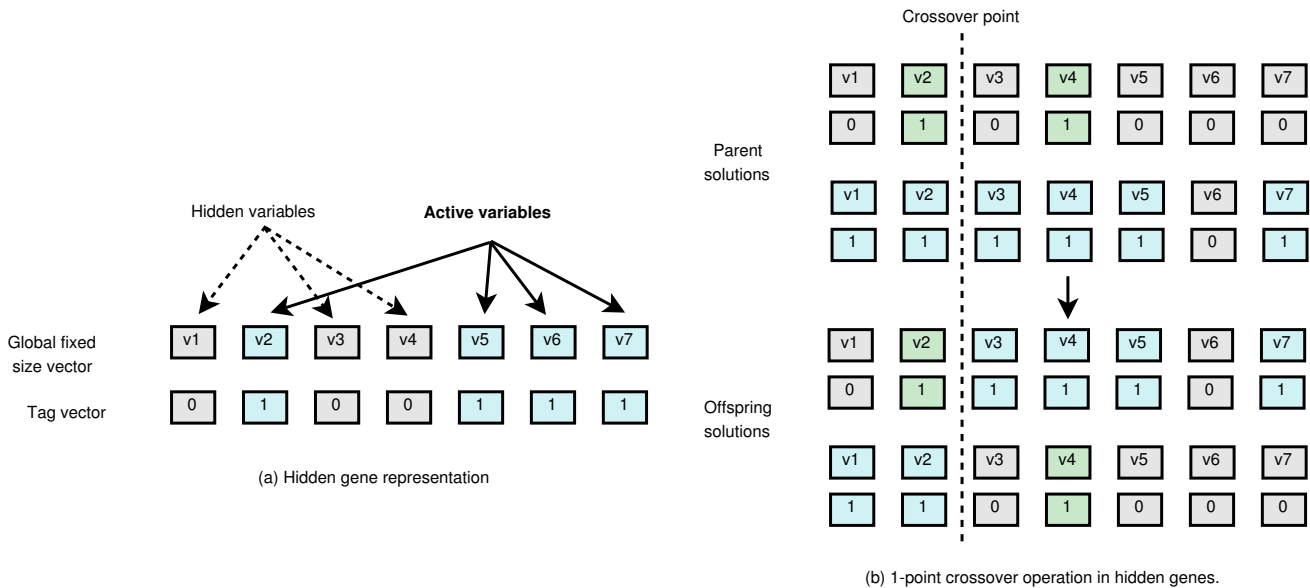


Fig. 8. Hidden genes representation and variation operators for VMVOPs.

[239] [227]. In [240], the authors extend the PSO movement updating mechanism proposed in the comprehensive learning

PSO (CLPSO) [241], which may change the length of a particle.

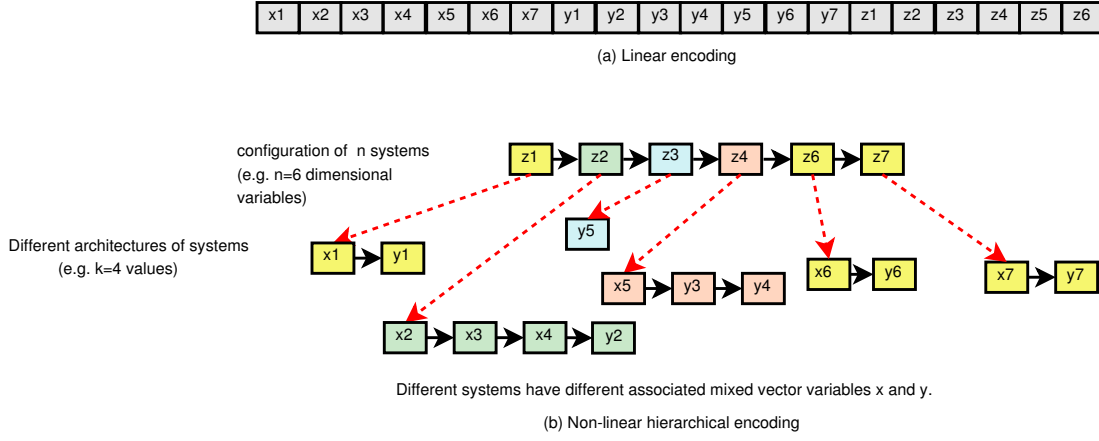


Fig. 9. Linear and non-linear representations for a simple VMVOPs.

A non-linear encoding based on hierarchical linked list of variables is more adapted to VMVOPs [242] [8] [243] [244]. The variables are classified as dependent or independent variables and are linked by both vicinity and hierarchical relations. Let us illustrate those encodings with a simple design problem, in which we have to configure a set of n systems. Each system is characterized by a dimensional variable of order k . Each given configuration of the system is defined by a variable set of mixed variables. In the linear encoding, the variables are stacked in a linear vector of variable-length (Fig.9). The data structure does not link explicitly the dependent variables. In the non-linear encoding, the variables are hierarchically structured in different levels. Solid arrows connect next neighbor variables, and dotted arrows connect child variables (Fig.9). For instance, the value of the dimensional variable z_1 will define the number and type of variables in the second level (e.g. x_1 and y_1). Variables that are pointed are dependent variables. Their existence depends on the value of the ancestor variables. Hence, the nodes z_1 , x_1 and y_1 are dependent nodes.

Standard variation operators typically operate on fixed-size linear encodings. New variations operators have to be designed for hierarchical representations:

- **Unary operators (e.g. neighborhood, mutation):** for a given solution, many neighborhoods and mutation operators may be defined. The following design questions arise:
 - **Definition of neighborhoods and mutations operators:** the number of neighborhoods and mutation operators is related to the number and type of variables associated to the VMVOP. Those operators perform at all levels of the hierarchy. However, the semantic correctness according to the new value of the variable must be ensured. Unary operators for non dimensional variables will not affect the structure of the graph modeling the hierarchical encoding. Indeed, the new value must be consistent with the type of variable (e.g. continuous, discrete categorical, discrete ordinal) [243]. The operators associated to dimensional variables (i.e. variables with dependent childs) have an impact on the number

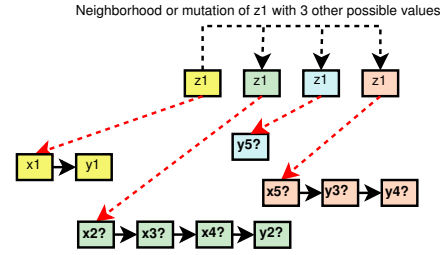


Fig. 10. Unary operator (e.g. neighborhood, mutation) of a dimensional variable in hierarchical encodings.

and type of the other variables. The whole subtree (i.e. size, type of variables) of dimensional variables is transformed according to its new value (Fig.10). Moreover, changing a value of a given dimensional variable has to fix the values of the new dependent child variables (Fig.10). One can randomly initialize those values, or search the sub-optimal (resp. optimal) values using some heuristics (resp. exact algorithms).

- **Order of application of the various operators:** once the unary operators defined, the order of application of those operators is a crucial issue. Operations at the higher hierarchies of the tree generate greater disruption of the solution (i.e. lower degree of locality, higher diversification) than those on the low levels of the hierarchy (i.e. higher degree of locality, higher intensification). A higher degree of locality means that small perturbations to the representation (i.e. genotype) result in small perturbations to the solution (i.e. phenotype).
- **N-ary operators (e.g. crossover, particle update):** as for unary operators, the N-ary operators can be applied at different levels of the hierarchy. For instance, standard crossover operators (e.g. 1-point, n-point, uniform) can be applied at the first layer of the hierarchical encoding [242] (Fig.11). However, all the subtrees (i.e. dependent variables) must be inherited from the parents. Inheriting only a part of the subtree may result in non feasible

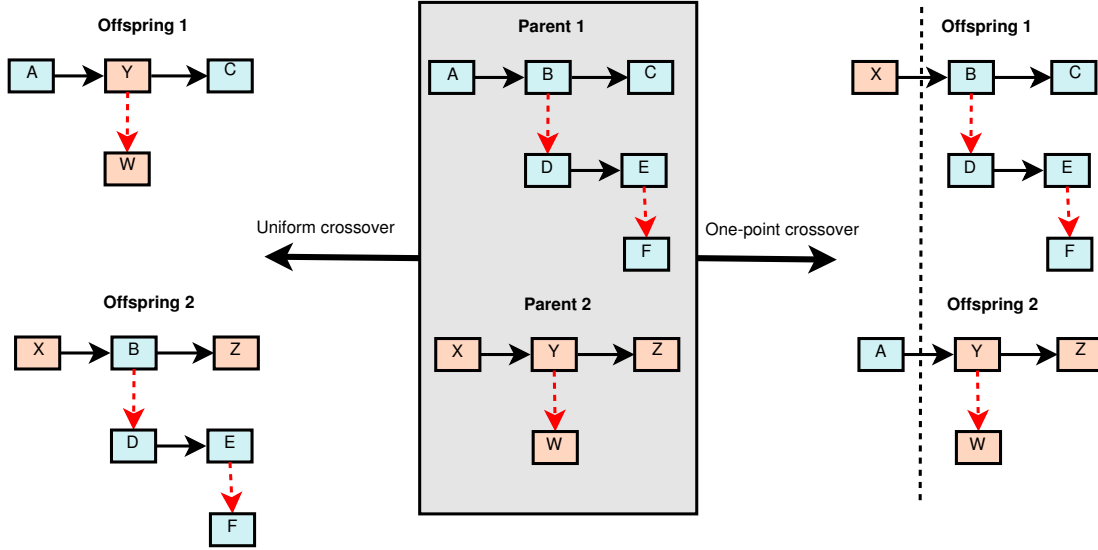


Fig. 11. Crossover operations in hierarchical encodings.

or inconsistent solutions. More complex problem-specific crossover operators can be designed [8]. The adaptation of the movement update operators (e.g. PSO) hierarchical representations is a more complex operation to define, and has not been widely studied in the literature.

Model-based operators (e.g. pheromone in ACO, statistical model in EDA): there has been growing interest in adapting EDAs to construct statistical models on tree representations. In [245], the authors extend EDA to genetic programming (GP) tree representation. GP evolves tree structures which encode computer programs (i.e. mathematical function) [246]. The tree has a limited hierarchical structure in which the terminal set could be $\{x, y, z\}$ and the non-terminal set could be operations such as $\{+, -, \times, /, \sin, \cos, \log, \exp\}$. Very few works in the literature address those important scientific challenges.

Compared to the linear encodings, the hierarchical encoding is more adapted to VMVOPs. It allows more flexibility in the design of efficient variation operators, which generate valid solutions. Compared to the hidden genes approach, hierarchical encoding has the advantage of not performing computationally wasteful variation operations. However, the encoding is much more complex and often requires problem-specific knowledge.

C. Decomposition-based approaches for VMVOPs

The main issue in decomposition-based approaches to solve VMVOPs is the handling of dimensional variables which induce variable-size vectors of mixed variables.

1) *Sequential approaches for VMVOPs:* The sequential approach presented in section IV-B1 may be easily extended to VMVOPs. The design questions to be adapted are:

- **Sequence of optimization:** one can alternate iterations in which the dimensional variables are first considered fixed, with iterations in which the continuous and discrete

variables (i.e. MVOP) are optimized (Fig.12). Indeed, once the dimensional variables are fixed, a VMOP has to be solved [247]. This sequential approach can be defined as follows:

$$\begin{aligned}
 \min f(x, y, z^*) &\longrightarrow^{(x^*, y^*)} \min f(x^*, y^*, z) \\
 \text{w.r.t } x \in F_x(z^*), y \in F_y(z^*) &\qquad \qquad \qquad \text{w.r.t } z \in F_z \\
 \text{s.t. } g(x, y, z^*) \leq 0 &\longleftarrow^{z^*} \text{s.t. } g(x^*, y^*, z) \leq 0
 \end{aligned} \tag{6}$$

In [32], a sequential approach applying continuous search (e.g. pattern local search algorithm) followed by discrete search is proposed.

- **Complete versus partial optimization:** partial optimization allows to reduce the size of decision space, which is suitable for large scale VMVOPs (Fig.12). In [248], a multi-stage partial GA strategy is applied in which the design space is gradually refined during the search. Maintaining the diversity in the population in such an approach is a very important issue. The partial optimization can be based on the type of variables or on some problem-specific knowledge. In [62], a problem-specific knowledge is used to carry out a partial sequential optimization. At each phase, a VMOP mixing continuous and discrete optimization has to be solved.

2) *Nested approach for VMVOPs:* A straightforward nested approach consists in optimizing at the outer-level (resp. inner-level) the dimensional variables (resp. continuous and discrete variables). A metaheuristic search over the dimensional variables can be applied (Fig.13). When fixing the dimensional variables at the outer-level, one has to solve a MVOP at the inner-level. Indeed, a MVOP subproblem is associated for each possible combination of dimensional variables. For each iteration of the outer loop, which specifies the values of the dimensional variables, a complete optimization of the problem with respect to the relevant continuous and discrete variables is performed in the inner loop. Then, any metaheuristic presented

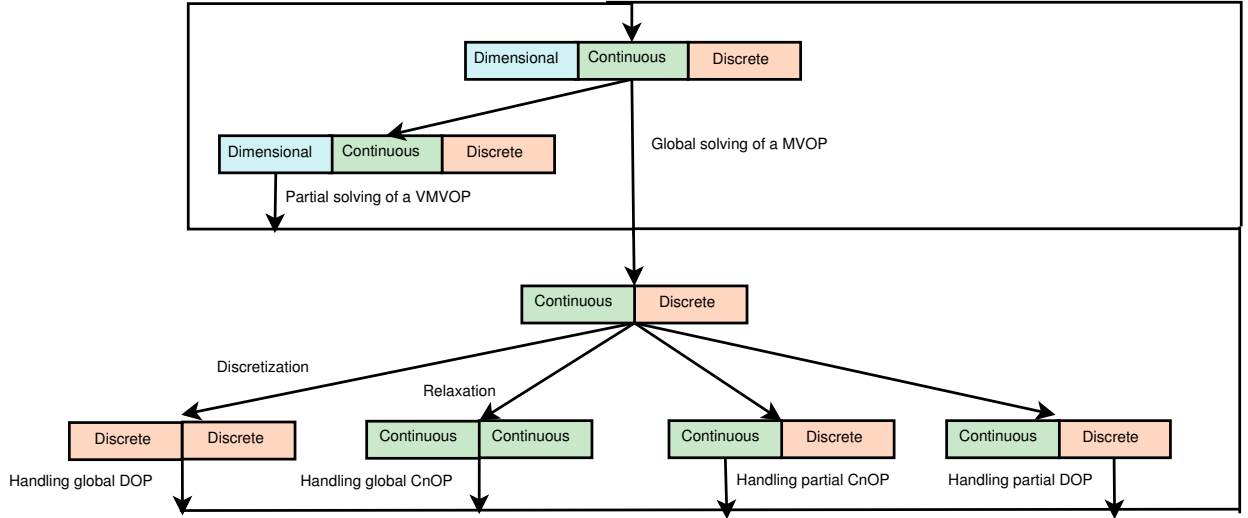


Fig. 12. Decomposition-based sequential strategies for solving VMVOPs.

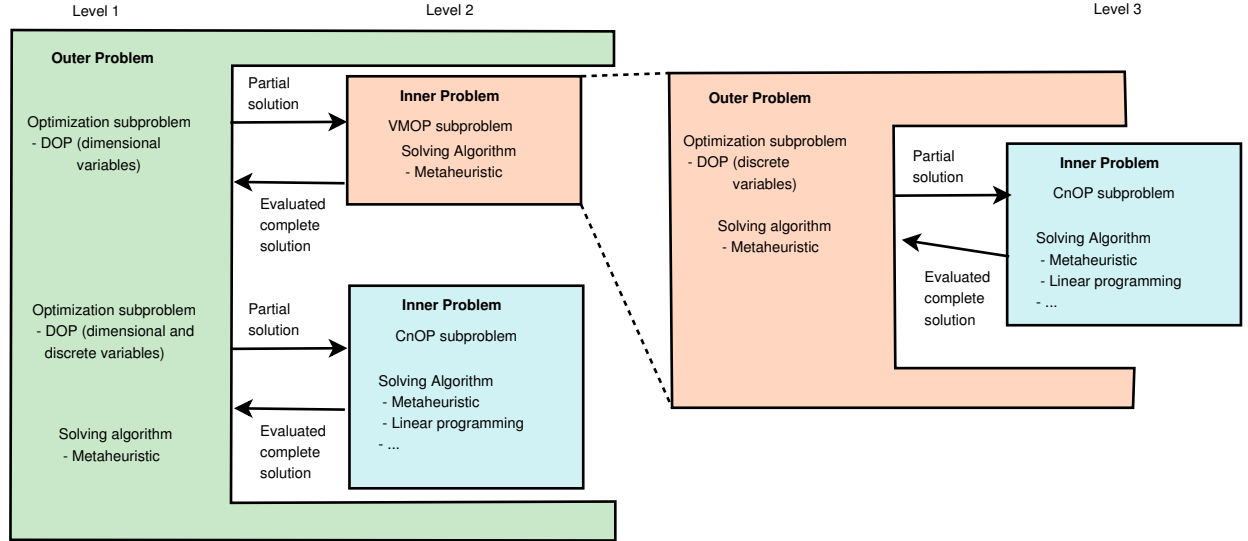


Fig. 13. Nested approaches for VMVOPs.

in section IV can be used to solve the VMOP. The nested approach on the dimensional can be formulated as follows:

$$\begin{aligned}
 &\min f(x^*, y^*, z) \\
 &\text{w.r.t } z \in F_z \\
 &\text{with } (x^*, y^*) = \underset{\text{w.r.t } x \in F_x(z), y \in F_y(z)}{\operatorname{argmin}} f(x, y, z) \quad (7) \\
 &\text{s.t } g(x^*, y^*, z) \leq 0
 \end{aligned}$$

This approach has been used in [249] [249] [250]. In [249], a binary GA algorithm (resp. hybrid PSO and DE) is applied at the outer-level (resp. inner-level). This approach is not efficient in case the inner-level MVOP is complex to solve, and there is a large proportion of inner-level MVOPs with infeasible solutions. Another two-level nested approach maps the dimensional and the discrete variables at the outer-level and the continuous variables at the inner-level. Finally, one can

also use a 3-level nested approach for handling respectively the dimensional, discrete and continuous variables (Fig.13).

3) *Co-evolutionary-based approaches for VMVOPs*: To our knowledge, no co-evolutionary approaches have been applied to VMVOPs. Most of the proposed strategies have been employed to VOPs with single type of variables [202]. One can extend the design question of co-evolutionary algorithms as following:

- **Decomposition strategies**: promising decomposition strategies for VMVOPs can be applied in the dimensional space. Decomposition should also be based on the dimensional variables interaction. The decomposition criteria shown in section IV can also be used here: static versus dynamic, disjoint versus overlapped, and knowledge-based decomposition. For instance, a static and disjoint partitioning of the dimensional space can be performed using a sequential or random division of the dimensional space [202].

- **Optimization algorithm:** the selected metaheuristic depends on the type of the generated sub-problems (i.e. VMVOP or MVOP). If the generated subproblems are VMOPs, any metaheuristic shown in section IV can be used. In [203], the subpopulations have fixed-size design spaces and are randomly initialized. All the solutions in each sub-population have the same encoding length, and then global relaxation GAs are applied.
- **Cooperation strategies and evaluation of solutions:** the same cooperative strategies as for MVOPs can be used. However, the evaluation of solutions represents a difficult challenge. Indeed, assembling the various partial solutions to construct a global consistent and feasible solution is complexified by the presence of dimensional variables.

VI. CONCLUSIONS AND PERSPECTIVES

This paper presents a unified taxonomy for metaheuristics in solving (V)MVOPs in an attempt to provide a common terminology and classification mechanisms. It provides a general mathematical formulation and concepts of (V)MVOPs, and identifies the various solving methodologies that can be applied. The advantages, the weaknesses and the limitations of the presented methodologies have been discussed.

Nevertheless, developing efficient and effective metaheuristics for handling (V)MVOPs is still challenging. One has to extend the proposed taxonomy to include surrogate-based metaheuristics for (V)MVOPs. Very few works try to solve (V)MVOPs that involve both expensive objective functions and high number of mixed variables. Indeed, most of the work in the literature dealing with Bayesian optimization applies to medium-size continuous optimization problems [251] [5] [252].

Extending the proposed taxonomy and critical analysis for solving multi-objective (V)MVOPs [253] [254], bi-level (V)MVOPs [255] [256], dynamic and uncertain (V)MVOPs [257] [258] is a challenging task. Indeed, many real-life (V)MVOPs in science and industry involve multiple objectives, Stackelberg sequential games and/or uncertainty. Even though many papers dealing with such problems have been referenced and analyzed in this work, one has to deepen the analysis of the proposed algorithmic methodologies to solve such complex (V)MVOPs.

There is no substantial work dealing with the design and implementation of massively parallel metaheuristics for (V)MVOPs. Nowadays, supercomputers composed of an important number of heterogeneous devices such as multi-core processors and GPUs are becoming more and more popular. Parallel co-evolutionary metaheuristics represent a promising perspective in the development of efficient and effective algorithms to solve large scale and expensive (V)MVOPs.

REFERENCES

[1] E.-G. Talbi, "Automated design of deep neural networks: A survey and unified taxonomy," *ACM Computing Surveys*, vol. 54, no. 2, 2021.

[2] Z. Meng, G. Li, X. Wang, S. Sait, and A. Yıldız, "A comparative study of metaheuristic algorithms for reliability-based design optimization problems," *Archives of Computational Methods in Engineering*, vol. 28, no. 3, 2021.

[3] A. Lodi, "Mixed integer programming computation," in *50 years of integer programming 1958-2008*. Springer, 2010.

[4] M. Cooper, "A survey of methods for pure nonlinear integer programming," *Management Science*, vol. 27, no. 3, 1981.

[5] J. Pelamatti, L. Brevault, M. Balesdent, E.-G. Talbi, and Y. Guerin, "Mixed variable gaussian process-based surrogate modeling techniques," *Journal of Aerospace Information Systems*, vol. 18, no. 11, 2021.

[6] J. Martins and A. Ning, *Engineering design optimization*. Cambridge University Press, 2021.

[7] S. Lucidi, V. Piccialli, and M. Sciandrone, "An algorithm model for mixed variable programming," *SIAM Journal on Optimization*, vol. 15, no. 4, 2005.

[8] H. M. Nyew, O. Abdelkhalik, and N. Onder, "Structured-chromosome evolutionary algorithms for variable-size autonomous interplanetary trajectory planning optimization," *J. Aerosp. Inf. Syst.*, vol. 12, no. 3, 2015.

[9] O. Abdelkhalik, *Algorithms for Variable-Size Optimization: Applications in Space Systems and Renewable Energy*. CRC Press, 2021.

[10] J. Lampinen and I. Zelinka, "Mixed integer-discrete-continuous optimization by differential evolution," in *5th Int. Conf. on Soft Computing*, 1999.

[11] V. Ho-Huu, T. Nguyen-Thoi, M. Nguyen-Thoi, and L. Le-Anh, "An improved constrained differential evolution using discrete variables for layout optimization of truss structures," *Expert Systems with Applications*, vol. 42, no. 20, 2015.

[12] F. Zhao, L. Zhao, L. Wang, and H. Song, "An ensemble discrete differential evolution for the distributed blocking flowshop scheduling with minimizing makespan criterion," *Expert Systems with Applications*, vol. 160, 2020.

[13] M. Emmerich, M. Grötzner, B. Groß, and M. Schütz, "Mixed-integer evolution strategy for chemical plant optimization with simulators," in *Evolutionary Design and Manufacture*. Springer, 2000.

[14] T. Bäck, F. Hoffmeister, and H.-P. Schwefel, "A survey of evolution strategies," in *ICGA Fourth Int. Conf. on Genetic Algorithms*, 1991.

[15] A. Miyagi, Y. Akimoto, and H. Yamamoto, "Well placement optimization for carbon dioxide capture and storage via CMA-ES with mixed integer support," in *Genetic and Evolutionary Computation Conference Companion*, 2018.

[16] L. dos Santos Coelho, "Gaussian quantum-behaved PSO approaches for constrained engineering design problems," *Expert Systems with Applications*, vol. 37, no. 2, 2010.

[17] T.-H. Kim, M. Cho, and S. Shin, "Constrained mixed-variable design optimization based on PSO with a diversity classifier for cyclically neighboring subpopulations," *Mathematics*, vol. 8, no. 11, 2020.

[18] K. Socha, "Aco for continuous and mixed-variable optimization," in *Int. Workshop on Ant Colony Optimization and Swarm Intelligence*. Springer, 2004.

[19] A. Rivas and L. Pareja, "Coordination of directional overcurrent relays that uses an ant colony optimization algorithm for mixed-variable optimization problems," in *IEEE Int. Conf. on Environment and Electrical Engineering*.

[20] T. Liao, K. Socha, M. A. M. de Oca, T. Stützle, and M. Dorigo, "ACO for mixed-variable optimization problems," *IEEE Trans. Evolutionary Computation*, vol. 18, no. 4, 2014.

[21] F. Wang, Y. Li, A. Zhou, and K. Tang, "An estimation of distribution algorithm for mixed-variable newsvendor problems," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 3, 2019.

[22] A. Zhou, J. Sun, and Q. Zhang, "An estimation of distribution algorithm with cheap and expensive local search methods," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, 2015.

[23] H. Jalota and M. Thakur, "GA designed for solving linear or nonlinear mixed-integer constrained optimization problems," in *Int. Conf. on advances in Soft Computing, Intelligent Systems and Applications*, 2018.

[24] D. Liu, C. Liu, C. Zhang, C. Xu, Z. Du, and Z. Wan, "Efficient hybrid algorithms to solve mixed discrete-continuous optimization problems: A comparative study," *Engineering Computations*, 2018.

[25] A. Maiti, A. Bhunia, and M. Maiti, "An application of real-coded GA for mixed integer non-linear programming," *Applied Mathematics and computation*, vol. 183, no. 2, 2006.

- [26] O. Exler, L. Antelo, J. Egea, A. Alonso, and J. Banga, "A tabu search-based algorithm for mixed-integer nonlinear problems and its application to integrated process and control system design," *Computers & Chemical Engineering*, vol. 32, no. 8, 2008.
- [27] H. Mashinchi, M. Orgun, and W. Pedrycz, "Hybrid optimization with improved tabu search," *Applied Soft Computing*, vol. 11, no. 2, 2011.
- [28] C. Zhang and H.-P. Wang, "Mixed-discrete nonlinear optimization with simulated annealing," *Engineering Optimization*, vol. 21, no. 4, 1993.
- [29] P. Koken, H. Seok, and S. Yoon, "A simulated annealing algorithm with neighbourhood list for capacitated dynamic lot-sizing problem with returns and hybrid products," *International Journal of Computer Integrated Manufacturing*, vol. 31, no. 8, 2018.
- [30] C. Mohan and H. Nguyen, "A controlled random search technique incorporating the simulated annealing concept for solving integer and mixed integer global optimization problems," *Computational Optimization and Applications*, vol. 14, no. 1, 1999.
- [31] B. Akay, D. Karaboga, B. Gorkemli, and E. Kaya, "A survey on the artificial bee colony algorithm variants for binary, integer and mixed integer programming problems," *Applied Soft Computing*, vol. 106, 2021.
- [32] C. Audet and J. E. Dennis, "Pattern search algorithms for mixed variable programming," *SIAM Journal on Optimization*, vol. 11, no. 3, 2001.
- [33] B. Cheung, K. Langevin, and H. Delmaire, "Coupling genetic algorithm with a grid search method to solve mixed integer nonlinear programming problems," *Computers & Mathematics with Applications*, vol. 34, no. 12, 1997.
- [34] J. Pelamatti, L. Brevault, M. Balesdent, E.-G. Talbi, and Y. Guerin, "How to deal with mixed-variable optimization problems," in *World Congress of Structural and Multidisciplinary Optimisation*, 2017.
- [35] E.-G. Talbi, *Metaheuristics: from design to implementation*. Wiley, 2009.
- [36] v. y. n. p. J. Del Ser et al., journal=Swarm and Evolutionary Computation, "Bio-inspired computation: Where we stand and what's next."
- [37] R. Storn and K. V. Price, "Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, 1997.
- [38] R. Eberhart, Y. Shi, and J. Kennedy, *Swarm intelligence*. Elsevier, 2001.
- [39] J. Stork, A. E. Eiben, and T. Bartz-Beielstein, "A new taxonomy of global optimization algorithms," *Natural Computing*, 2020.
- [40] A. Yildiz, H. Abderazek, and S. Mirjalili, "A comparative study of recent non-traditional methods for mechanical design optimization," *Archives of Computational Methods in Engineering*, vol. 27, no. 4, 2020.
- [41] S. Gupta, H. Abderazek, B. Yildiz, A. Yildiz, S. Mirjalili, and S. Sait, "Comparison of metaheuristic optimization algorithms for solving constrained mechanical design optimization problems," *Expert Systems with Applications*, vol. 183, 2021.
- [42] A. Kania and K. Sidarto, "Solving mixed integer nonlinear programming problems using spiral dynamics optimization algorithm," in *AIP Conference*, vol. 1716, no. 1, 2016.
- [43] R. Angira and B. Babu, "Optimization of process synthesis and design problems: A modified differential evolution approach," *Chemical Engineering Science*, vol. 61, no. 14, 2006.
- [44] I. Ali, D. Essam, and K. Kasmarik, "A novel design of differential evolution for solving discrete traveling salesman problems," *Swarm and Evolutionary Computation*, vol. 52, 2020.
- [45] S. Zheng, B. Xiang, X. Zhang, and J. Zhang, "Differential evolution optimization algorithm for antenna designs with mixed discrete-continuous variables," in *Int. Conf. on Microwave and Millimeter Wave Technology (ICMMT)*, 2019.
- [46] A. W. Mohamed, "An efficient modified differential evolution algorithm for solving constrained non-linear integer and mixed-integer global optimization problems," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 3, 2017.
- [47] A. Ponsich and C. Coello, "Differential evolution performances for the solution of mixed-integer constrained process engineering problems," *Applied Soft Computing*, vol. 11, no. 1, 2011.
- [48] M. Varadarajan and K. Swarup, "Differential evolution approach for optimal reactive power dispatch," *Applied soft computing*, vol. 8, no. 4, 2008.
- [49] A. Mohamed, A. Mohamed, E. Elfeky, and M. Saleh, "Solving constrained non-linear integer and mixed-integer global optimization problems using enhanced directed differential evolution algorithm," in *Machine Learning Paradigms: Theory and Application*. Springer, 2019.
- [50] Y.-C. Lin, K.-S. Hwang, and F.-S. Wang, "A mixed-coding scheme of evolutionary algorithms to solve mixed-integer nonlinear programming problems," *Computers & Mathematics with Applications*, vol. 47, no. 8, 2004.
- [51] H. Peng, Y. Han, C. Deng, J. Wang, and Z. Wu, "Multi-strategy co-evolutionary differential evolution for mixed-variable optimization," *Knowledge-Based Systems*, vol. 229, 2021.
- [52] S. Rao and Y. Xiong, "A hybrid GA for mixed-discrete design optimization," *Journal of Mechanical Design*, vol. 127, no. 6, 2005.
- [53] K. Deep, K. Singh, M. Kansal, and C. Mohan, "A real coded genetic algorithm for solving integer and mixed integer optimization problems," *Applied Mathematics and Computation*, vol. 212, no. 2, 2009.
- [54] S.-J. Wu and P.-T. Chow, "GAs for nonlinear mixed discrete-integer optimization problems via meta-genetic parameter optimization," *Engineering Optimization*, vol. 24, no. 2, 1995.
- [55] T. Yokota, M. Gen, and Y.-X. Li, "Genetic algorithm for non-linear mixed integer programming problems and its applications," *Computers & Industrial Engineering*, vol. 30, no. 4, 1996.
- [56] L. Yan, K. Shen, and S. Hu, "Solving mixed integer nonlinear programming problems with line-up competition algorithm," *Computers & Chemical Engineering*, vol. 28, no. 12, 2004.
- [57] W. Jun, G. Yuelin, and Y. Lina, "An improved differential evolution algorithm for mixed integer programming problems," in *IEEE Int. Conf. on Computational Intelligence and Security*, 2013.
- [58] Y.-C. Lin, K.-S. Hwang, and F.-S. Wang, "Co-evolutionary hybrid differential evolution for mixed-integer optimization problems," *Engineering Optimization*, vol. 33, no. 6, 2001.
- [59] C.-X. Guo, J.-S. Hu, B. Ye, and Y.-J. Cao, "Swarm intelligence for mixed-variable design optimization," *Journal of Zhejiang University-SCIENCE A*, vol. 5, no. 7, 2004.
- [60] H. Nahvi and I. Mohagheghian, "A particle swarm optimization algorithm for mixed variable nonlinear problems," 2011.
- [61] S. Chowdhury, W. Tong, A. Messac, and J. Zhang, "A mixed-discrete particle swarm optimization algorithm with explicit diversity-preservation," *Structural and Multidisciplinary Optimization*, vol. 47, no. 3, 2013.
- [62] G. Venter and J. Sobieszcanski-Sobieski, "Multidisciplinary optimization of a transport aircraft wing using PSO," *Structural and Multidisciplinary optimization*, vol. 26, no. 1-2, 2004.
- [63] K. Parsopoulos and M. Vrahatis, "Recent approaches to global optimization problems through PSO," *Natural Computing*, vol. 1, no. 2, 2002.
- [64] G. Kanagaraj, S. Ponnambalam, N. Jawahar, and M. Nilakantan, "An effective hybrid cuckoo search and genetic algorithm for constrained engineering design optimization," *Engineering Optimization*, vol. 46, no. 10, 2014.
- [65] A. Gandomi, X.-S. Yang, and A. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Engineering with Computers*, vol. 29, no. 1, 2013.
- [66] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Applied mathematics and computation*, vol. 188, no. 2, 2007.
- [67] S. Gupta and K. Deep, "An efficient grey wolf optimizer with opposition-based learning and chaotic local search for integer and mixed-integer optimization problems," *Journal for Science & Engineering*, vol. 44, no. 8, 2019.
- [68] V. Rao, V. Savsani, and D. Vakharia, "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-aided design*, vol. 43, no. 3, 2011.
- [69] S. Talatahari and V. Goodarzi, "A discrete hybrid teaching-learning-based optimization algorithm for optimization of space trusses," *Journal of Structural Engineering and Geo-Techniques*, vol. 10, no. 1, 2020.
- [70] S. Das, A. Verma, and P. Bijwe, "Transmission network expansion planning using a modified artificial bee colony algorithm," *International Transactions on Electrical Energy Systems*, vol. 27, no. 9, 2017.
- [71] A. Gandomi, X.-S. Yang, and A. Alavi, "Mixed variable structural optimization using firefly algorithm," *Computers & Structures*, vol. 89, no. 23-24, 2011.
- [72] M. Schlüter, J. Egea, and J. Banga, "Extended ant colony optimization for non-convex mixed integer nonlinear programming," *Comput. Oper. Res.*, vol. 36, no. 7, 2009.
- [73] T. Liao, R. Kuo, and J. Hu, "Hybrid ACO algorithms for mixed discrete-continuous optimization problems," *Applied Mathematics and Computation*, vol. 219, no. 6, 2012.

- [74] Y. Miao, Q. Su, Z. Hu, and X. Xia, "Modified differential evolution algorithm with onlooker bee operator for mixed discrete-continuous optimization," *SpringerPlus*, vol. 5, no. 1, 2016.
- [75] W. Liao, "Two hybrid differential evolution algorithms for engineering design optimization," *Applied Soft Computing*, vol. 10, no. 4, 2010.
- [76] L. C. dos Santos and V. Mariani, "Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization," *Expert Systems with Applications*, vol. 34, no. 3, 2008.
- [77] C. L. dos Santos, B. de Andrade, and V. Mariani, "A chaotic firefly algorithm applied to reliability-redundancy optimization," in *IEEE Congress of Evolutionary Computation (CEC)*, 2011.
- [78] M. Srinivas and G. Rangaiah, "Differential evolution with tabu list for solving nonlinear and mixed-integer nonlinear programming problems," *Industrial & engineering chemistry research*, vol. 46, no. 22, 2007.
- [79] H. Yi, Q. Duan, and W. Liao, "Three improved hybrid metaheuristic algorithms for engineering design optimization," *Applied Soft Computing*, vol. 13, no. 5, 2013.
- [80] H. Garg, "A hybrid PSO-GA algorithm for constrained optimization problems," *Applied Mathematics and Computation*, vol. 274, 2016.
- [81] S. Rao, *Engineering optimization: theory and practice*. John Wiley & Sons, 2019.
- [82] S. Kitayama and K. Yasuda, "A method for mixed integer programming problems by PSO," *Electrical Engineering in Japan*, vol. 157, no. 2, 2006.
- [83] A. Lotfipour and H. Afrakhte, "A discrete teaching-learning-based optimization algorithm to solve distribution system reconfiguration in presence of distributed generation," *International Journal of Electrical Power & Energy systems*, vol. 82, 2016.
- [84] S. Chowdhury, A. Messac, and R. Khire, "Developing a non-gradient based mixed-discrete optimization approach for comprehensive product platform planning (cp3)," in *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, 2010.
- [85] J.-F. Fu, R. Fenton, and W. Cleghorn, "A mixed integer-discrete-continuous programming method and its application to engineering design optimization," *Engineering Optimization*, vol. 17, no. 4, 1991.
- [86] W. Murray and K.-M. Ng, "An algorithm for nonlinear optimization problems with binary variables," *Computational optimization and applications*, vol. 47, no. 2, 2010.
- [87] K. Abhishek, S. Leyffer, and J. Linderoth, "Modeling without categorical variables," *Optimization and Engineering*, vol. 11, no. 2, 2010.
- [88] S. He, E. Prempain, and Q. Wu, "An improved PSO for mechanical design optimization problems," *Engineering optimization*, vol. 36, no. 5, 2004.
- [89] H. Schmidt and G. Thierauf, "A combined heuristic optimization technique," *Advances in Engineering Software*, vol. 36, no. 1, 2005.
- [90] B. Crawford, R. Soto, G. Astorga, J. García, C. Castro, and F. Paredes, "Putting continuous metaheuristics to work in binary search spaces," *Complexity*, vol. 2017, 2017.
- [91] L. Yu, L. Hu, and L. Tang, "Stock selection with a novel sigmoid-based mixed discrete-continuous differential evolution algorithm," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, 2016.
- [92] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm algorithm," in *IEEE Int. Conf. on Systems, Man, and Cybernetics*, vol. 5, 1997.
- [93] S. Palit, S. Sinha, M. Molla, A. Khanra, and M. Kule, "A cryptanalytic attack on the knapsack cryptosystem using binary firefly algorithm," in *IEEE Int. Conf. on Computer and Communication Technology (ICCCCT)*, 2011.
- [94] G. Pampara, A. Engelbrecht, and N. Franken, "Binary differential evolution," in *IEEE Int. Conf. on Evolutionary Computation*, 2006.
- [95] B. Liu, L. Wang, and Y. Jin, "An effective pso-based memetic algorithm for flow shop scheduling," *IEEE Trans. Syst. Man Cybern. Part B*, vol. 37, no. 1, 2007.
- [96] I. Ali, S. Elsayed, T. Ray, and R. Sarker, "A differential evolution algorithm for solving resource constrained project scheduling problems," in *ACALCI'2016 Conference on Artificial Life and Computational Intelligence*, 2016.
- [97] A. Ouassab, B. Ahiod, and X.-S. Yang, "Random-key cuckoo search for the travelling salesman problem," *Soft Computing*, vol. 19, no. 4, 2015.
- [98] F. Hafiz and A. Abdenour, "PSO algorithm variants for the quadratic assignment problems - a probabilistic learning approach," *Expert Systems with Applications*, vol. 44, 2016.
- [99] F. Tasgetiren, Y.-C. Liang, M. Sevkli, and G. Gencyilmaz, "A PSO algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem," *European Journal of Operational Research*, vol. 177, no. 3, 2007.
- [100] E. Ülker, "Adaptation of harmony search algorithm for dna fragment assembly problem," in *IEEE Computing Conference (SAI)*, 2016.
- [101] A. Yousif, S. Nor, A. Abdullah, and M. Bashir, "A discrete firefly algorithm for scheduling jobs on computational grid," in *Cuckoo Search and Firefly Algorithm*.
- [102] A. Kumar and S. Chakraverty, "Design optimization for reliable embedded system using cuckoo search," in *IEEE Int. Conf. on Electronics Computer Technology*, vol. 1, 2011.
- [103] X. Li and M. Yin, "A hybrid cuckoo search via lévy flights for the permutation flow shop scheduling problem," *International Journal of Production Research*, vol. 51, no. 16, 2013.
- [104] J. Arora, M. Huang, and C. Hsieh, "Methods for optimization of nonlinear problems with discrete variables: a review," *Structural optimization*, vol. 8, no. 2, 1994.
- [105] E. Davydov and I. Sigal, "Application of penalty function method in integer programming problems," *Engineering Cybernetics*, vol. 10, no. 1, 1972.
- [106] D. Shin, Z. Gürdal, and J. Griffin, "A penalty approach for nonlinear optimization with discrete design variables," *Engineering Optimization*, vol. 16, no. 1, 1990.
- [107] H.-L. Li, "An approximate method for local optima for nonlinear mixed integer programming problems," *Comput. Oper. Res.*, vol. 19, no. 5, 1992.
- [108] S. Kitayama, M. Arakawa, and K. Yamazaki, "Penalty function approach for the mixed discrete nonlinear problems by PSO," *Structural and Multidisciplinary Optimization*, vol. 32, no. 3, 2006.
- [109] D. Goldberg, K. Deb, and J. H. Clark, "Genetic algorithms, noise, and the sizing of populations," *Complex Systems*, vol. 6, no. 4, 1992.
- [110] Z. Michalewicz and Z. Michalewicz, *Genetic algorithms+ data structures= evolution programs*. Springer Science & Business Media, 1996.
- [111] G. Dimopoulos, "Mixed-variable engineering optimization based on evolutionary and social metaphors," *Computer methods in applied mechanics and engineering*, vol. 196, no. 4-6, 2007.
- [112] L. Costa and P. Oliveira, "Evolutionary algorithms approach to the solution of mixed integer non-linear programming problems," *Computers & Chemical Engineering*, vol. 25, no. 2-3, 2001.
- [113] N. Turkkan, "Discrete optimization of structures using a floating-point genetic algorithm," in *Annual Conf. of the Canadian Society for Civil Engineering*, 2003.
- [114] C.-Y. Lin and P. Hajela, "GAs in optimization problems with discrete and integer design variables," *Engineering optimization*, vol. 19, no. 4, 1992.
- [115] M. Stelmack, N. Nakashima, and S. Batill, "Genetic algorithms for mixed discrete/continuous optimization in multidisciplinary design," in *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 1998.
- [116] L. Coelho, "Gaussian quantum-behaved particle swarm optimization approaches for constrained engineering design problems," *Expert Systems with Applications*, vol. 37, no. 2, 2010.
- [117] J. Wang and Z. Yin, "A ranking selection-based particle swarm optimizer for engineering design optimization problems," *Structural and multidisciplinary optimization*, vol. 37, no. 2, 2008.
- [118] A. De, J. Wang, and M. Tiwari, "Hybridizing basic variable neighborhood search with PSO for solving sustainable ship routing and bunker management problem," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, 2019.
- [119] F. Gardi and K. Nouioua, "Local search for mixed-integer nonlinear optimization: a methodology and an application," in *European Conference on Evolutionary Computation in Combinatorial Optimization*, 2011.
- [120] M. F. Cardoso, R. L. Salcedo, F. de Azevedo, and D. Barbosa, "A simulated annealing approach to the solution of MINLP problems," *Computers & Chemical Engineering*, vol. 21, no. 12, 1997.
- [121] D. Olsson and L. Nelson, "The nelder-mead simplex procedure for function minimization," *Technometrics*, vol. 17, no. 1, 1975.
- [122] R. Günter, "An evolutionary algorithm for integer programming," in *Int. Conf. on Parallel Problem Solving from Nature*. Springer, 1994.
- [123] K. V. der Blom, S. Boonstra, H. Hofmeyer, and M. Emmerich, "Multicriteria building spatial design with mixed integer evolutionary algorithms," in *Int. Conf. on Parallel Problem Solving from Nature*. Springer, 2016.
- [124] H. Li and L. Zhang, "A discrete hybrid differential evolution algorithm for solving integer programming problems," *Engineering Optimization*, vol. 46, no. 9, 2014.

- [125] R. L. et al., "Mixed integer evolution strategies for parameter optimization," *Evolutionary Computation*, vol. 21, no. 1, 2013.
- [126] Y. Lin, Y. Liu, W.-N. Chen, and J. Zhang, "A hybrid differential evolution algorithm for mixed-variable optimization problems," *Information Sciences*, vol. 466, 2018.
- [127] D. Datta and J. Figueira, "A real integer discrete-coded differential evolution," *Applied Soft Computing*, vol. 13, no. 9, 2013.
- [128] R. L. et al., "Metamodel-assisted mixed integer evolution strategies and their application to intravascular ultrasound image analysis," in *IEEE Congress on Evolutionary Computation CEC*, 2008.
- [129] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies - a comprehensive introduction," *Natural computing*, vol. 1, no. 1, 2002.
- [130] T. Back, *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press, 1996.
- [131] Y. J. Cao, L. Jiang, and Q. H. Wu, "An evolutionary programming approach to mixed-variable optimization problems," *Applied Mathematical Modelling*, vol. 24, no. 12, 2000.
- [132] R. Kincaid, M. Griffith, R. Sykes, and J. Sobieszczanski-Sobieski, "Bell-curve GA for mixed continuous and discrete optimization problems," *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, 2004.
- [133] L. Gao and A. Hailu, "Comprehensive learning particle swarm optimizer for constrained mixed-variable optimization problems," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, 2010.
- [134] F. Wang, H. Zhang, and A. Zhou, "A particle swarm optimization algorithm for mixed-variable optimization problems," *Swarm and Evolutionary Computation*, vol. 60, 2021.
- [135] V. Mokarram and M. Banan, "A new pso-based algorithm for multi-objective optimization with continuous and discrete design variables," *Structural and Multidisciplinary Optimization*, vol. 57, no. 2, 2018.
- [136] V. Hinojosa and R. Araya, "Modeling a mixed-integer-binary small-population evolutionary PSO for solving the optimal power flow problem in electric power systems," *Applied Soft Computing*, vol. 13, no. 9, 2013.
- [137] C. Sun, J. Zeng, and J.-S. Pan, "A modified PSO with feasibility-based rules for mixed-variable optimization problems," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 6, 2011.
- [138] J. Rezaee, "A mixed binary-continuous PSO algorithm for unit commitment in microgrids considering uncertainties and emissions," *International Transactions on Electrical Energy Systems*, vol. 30, no. 11, 2020.
- [139] C. Liao, C. Tseng, and P. Luarn, "A discrete version of particle swarm optimization for flowshop scheduling problems," *Comput. Oper. Res.*, vol. 34, no. 10, 2007.
- [140] L. Yiqing, L. Xigang, and L. Yongjian, "An improved PSO algorithm for solving non-convex NLP/MINLP problems with equality constraints," *Computers & Chemical Engineering*, vol. 31, no. 3, 2007.
- [141] L. Wang, J. Pei, M. Menhas, J. Pi, M. Fei, and P. Pardalos, "A hybrid-coded human learning optimization for mixed-variable optimization problems," *Knowledge-Based Systems*, vol. 127, 2017.
- [142] P. Larranag, R. Etxeberria, J. Lozano, and J. Pena, "Optimization by learning and simulation of bayesian and gaussian networks. university of the basque country technical report ehu-kzaa-ik-4/99," 1999.
- [143] P. Bosman and D. Thierens, "Mixed ideas," *Utrecht University Technical Report UU-CS-2000-45*. Utrecht University, Utrecht, Netherlands, 2000.
- [144] J. Ocenasek and J. Schwarz, "Estimation of distribution algorithm for mixed continuous-discrete optimization problems," in *2nd Euro-Int. Symp. on Computational Intelligence*. IOS Press Kosice, 2002.
- [145] L. Sahoo, A. Banerjee, A. Bhunia, and S. Chattopadhyay, "An efficient GA-PSO approach for solving mixed-integer nonlinear programming problem in reliability optimization," *Swarm and Evolutionary Computation*, vol. 19, 2014.
- [146] Y. Gao, Y. Sun, and J. Wu, "Difference-genetic co-evolutionary algorithm for nonlinear mixed integer programming problems," *Journal of Nonlinear Science and Its Applications*, vol. 9, no. 3, 2016.
- [147] A.-R. Hedar and A. Fahim, "Filter-based GA for mixed variable programming," *Numerical Algebra, Control & Optimization*, vol. 1, no. 1, 2011.
- [148] W. Shi, W.-N. Chen, Y. Lin, T. Gu, S. Kwong, and J. Zhang, "An adaptive estimation of distribution algorithm for multipolicy insurance investment planning," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 1, 2017.
- [149] W. Li, G. Zhang, X. Yang, Z. Tao, and H. Xu, "Sizing a hybrid renewable energy system by a coevolutionary multiobjective optimization algorithm," *Complexity*, vol. 2021, 2021.
- [150] M. A. Abramson, C. Audet, J. W. Chrissis, and J. G. Walston, "Mesh adaptive direct search algorithms for mixed variable optimization," *Optimization Letters*, vol. 3, no. 1, 2009.
- [151] M. A. Abramson, C. Audet, and J. E. Dennis, "Filter pattern search algorithms for mixed variable constrained optimization problems," Rice University, Tech. Rep., 2004.
- [152] M. A. Abramson, *Pattern search algorithms for mixed variable general constrained optimization problems*. Rice University, 2003.
- [153] C. Audet and J. E. Dennis, "Mesh adaptive direct search algorithms for constrained optimization," *SIAM Journal on Optimization*, vol. 17, no. 1, 2006.
- [154] F. Capitanescu and L. Wehenkel, "Sensitivity-based approaches for handling discrete variables in optimal power flow computations," *IEEE Transactions on Power Systems*, vol. 25, no. 4, 2010.
- [155] W. Yan, F. Liu, C. Chung, and K. Wong, "A hybrid GA-interior point method for optimal reactive power flow," *IEEE Transactions on Power Systems*, vol. 21, no. 3, 2006.
- [156] Y. Gao, X. Chang, and J. Liu, "Hybrid coding collaborative DE-ACO algorithm for solving mixed-integer programming problems," *Journal of Computing*, vol. 9, no. 1, 2014.
- [157] M. Stelmack, S. Batill, M. Stelmack, and S. Batill, "Concurrent subspace optimization of mixed continuous/discrete systems," in *38th Structures, Structural Dynamics, and Materials Conf.*, 1997.
- [158] Y. Lin, "Mixed-integer constrained optimization based on memetic algorithm," *Journal of Applied Research and Technology*, vol. 11, no. 2, 2013.
- [159] S. Nema, J. Goulernas, G. Sparrow, and P. Cook, "A hybrid particle swarm branch-and-bound optimizer for mixed discrete nonlinear programming," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 38, no. 6, 2008.
- [160] Z. Hua and F. Huang, "An effective genetic algorithm approach to large scale mixed integer programming problems," *Applied Mathematics and Computation*, vol. 174, no. 2, 2006.
- [161] S. Praharaj and S. Azarm, "Two-level nonlinear mixed discrete-continuous optimization-based design," *Adv. Design Autom.*, vol. 1, no. 44, 1992.
- [162] E.-G. Talbi, "A taxonomy of metaheuristics for bi-level optimization," in *Metaheuristics for bi-level optimization*. Springer, 2013.
- [163] —, *Metaheuristics for bilevel optimization*. Springer, 2013.
- [164] A. Sinha, P. Malo, and K. Deb, "A review on bilevel optimization: from classical to evolutionary approaches and applications," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 2, 2017.
- [165] F. Vanderbeck and M. Savelsbergh, "A generic view of dantzig-wolfe decomposition in mixed integer programming," *Operations Research Letters*, vol. 34, no. 3, 2006.
- [166] A. Geoffrion, "Generalized benders decomposition," *Journal of Optimization Theory and Applications*, vol. 10, no. 4, 1972.
- [167] M. Duran and I. Grossmann, "An outer-approximation algorithm for a class of mixed-integer nonlinear programs," *Mathematical Programming*, vol. 36, no. 3, 1986.
- [168] C. Floudas, *Nonlinear and mixed-integer optimization: fundamentals and applications*. Oxford University Press, 1995.
- [169] M. Chanthasuwannasin, B. Kottitum, and T. Srinophakun, "A mixed coding scheme of a particle swarm optimization," *Chemical Engineering Communications*, vol. 204, no. 8, 2017.
- [170] S. Roy, W. Crossley, B. Stanford, K. Moore, and J. S. Gray, "A mixed integer efficient global optimization algorithm with multiple infill strategy-applied to a wing topology optimization problem," in *AIAA Scitech 2019 Forum*, 2019.
- [171] W. Wang, S. Guo, and W. Yang, "Simultaneous partial topology and size optimization of a wing structure using ant colony and gradient based methods," *Engineering Optimization*, vol. 43, no. 4, 2011.
- [172] D. Jones, M. Schonlau, and W. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, no. 4, 1998.
- [173] Z. Garroussi, R. Ellaia, E.-G. Talbi, and J.-Y. L. Lucas, "A matheuristic for a bi-objective demand-side optimization for cooperative smart homes," *Electrical Engineering*, vol. 102, no. 4, 2020.
- [174] Z. Chiam, A. Easwaran, D. Mouquet, S. Fazlollahi, and J. Millás, "A hierarchical framework for holistic optimization of the operations of district cooling systems," *Applied Energy*, vol. 239, 2019.
- [175] R. Balamurugan and S. Subramanian, "Hybrid integer coded differential evolution," *Energy Conversion and Management*, vol. 49, no. 4, 2008.

- [176] M. Gonzalez, J. López-Espín, J. Aparicio, and E.-G. Talbi, "A hyper-matheuristic approach for solving mixed integer linear optimization models in the context of data envelopment analysis," *PeerJ Computer Science*, vol. 8, 2022.
- [177] Y. Ye, H. Jinhou, C. Chen, X. Jiarong, L. Zuwei, L. Xinggao, C. Jinshui, and L. Jiangang, "A PSO-LP cooperative algorithm for mixed integer nonlinear programming," in *12th Asian Control Conference (ASCC)*, 2019.
- [178] Z. Garroussi, R. Ellaia, E.-G. Talbi, and J.-Y. Lucas, "Hybrid evolutionary algorithm for residential demand side management with a photovoltaic panel and a battery," in *Int. Conf. on Control, Artificial Intelligence, Robotics & Optimization*, 2017.
- [179] Y.-J. Zheng and S.-Y. Chen, "Cooperative particle swarm optimization for multiobjective transportation planning," *Applied Intelligence*, vol. 39, no. 1, 2013.
- [180] X. M. et al., "A survey on cooperative co-evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 3, 2018.
- [181] F. Legillon, A. Liefvooghe, and E.-G. Talbi, "Cobra: A cooperative coevolutionary algorithm for bi-level optimization," in *IEEE Congress on Evolutionary Computation (CEC)*, 2012.
- [182] M. Potter and K. K. Jong, "A cooperative coevolutionary approach to function optimization," in *Int. Conf. on Parallel Problem Solving from Nature (PPSN)*, 1994.
- [183] N. Hiremath, S. Sahu, and M. Tiwari, "Designing a multi echelon flexible logistics network using co-evolutionary immune PSO with penetrated hyper-mutation," in *Applied Mechanics and Materials*, vol. 110, 2012.
- [184] M. Omidvar, X. Li, Y. Mei, and X. Yao, "Cooperative co-evolution with differential grouping for large scale optimization," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, 2013.
- [185] F. V. den Bergh and A. Engelbrecht, "A cooperative approach to PSO," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, 2004.
- [186] Y. Mei, M. Omidvar, X. Li, and X. Yao, "A competitive divide-and-conquer algorithm for unconstrained large-scale black-box optimization," *ACM Transactions on Mathematical Software*, vol. 42, no. 2, 2016.
- [187] S. Strasser, J. Sheppard, N. Fortier, and R. Goodman, "Factored evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 2, 2016.
- [188] T. Vinkó and D. Izzo, "Global optimisation heuristics and test problems for preliminary spacecraft trajectory design," *Advanced Concepts Team, ESATR ACT-TNT-MAD-GOHTPPSTD*, 2008.
- [189] Q.-K. Pan, L. Gao, and L. Wang, "An effective cooperative co-evolutionary algorithm for distributed flowshop group scheduling problems," *IEEE Transactions on Cybernetics*, 2020.
- [190] X. Zhang, X. Guan, I. Hwang, and K. Cai, "A hybrid distributed-centralized conflict resolution approach for multi-aircraft based on cooperative co-evolutionary," *Science China Information Sciences*, vol. 56, no. 12, 2013.
- [191] S. Yuan, T. Li, and B. Wang, "A co-evolutionary genetic algorithm for the two-machine flow shop group scheduling problem with job-related blocking and transportation times," *Expert Systems with Applications*, vol. 152, 2020.
- [192] M. Shi and S. Gao, "Reference sharing: a new collaboration model for cooperative coevolution," *Journal of Heuristics*, vol. 23, no. 1.
- [193] E. Popovici, A. Bucci, R. Wiegand, and E. D. de Jong, "Coevolutionary principles," in *Handbook of Natural Computing*, G. Rozenberg, T. Bäck, and J. N. Kok, Eds. Springer, 2012.
- [194] Z. Yang, K. Tang, and X. Yao, "Large scale evolutionary optimization using cooperative coevolution," *Information sciences*, vol. 178, no. 15, 2008.
- [195] P. Wiegand, W. Liles, and K. D. Jong, "An empirical analysis of collaboration methods in cooperative coevolutionary algorithms," in *Genetic and Evolutionary Computation Conf. (GECCO)*, vol. 2611, 2001.
- [196] F. de Oliveira, R. Enayatifar, H. Sadaei, F. Guimarães, and J.-Y. Potvin, "A cooperative coevolutionary algorithm for the multi-depot vehicle routing problem," *Expert Systems with Applications*, vol. 43, 2016.
- [197] Y. Son and R. Baldick, "Hybrid coevolutionary programming for nash equilibrium search in games with local optima," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 4, 2004.
- [198] E. Glorieux, B. Svensson, F. Danielsson, and B. Lennartson, "Improved constructive cooperative coevolutionary differential evolution for large-scale optimisation," in *IEEE Symposium Series on Computational Intelligence*, 2015.
- [199] L. Panait and S. Luke, "Selecting informative actions improves cooperative multiagent learning," in *5th Int. Joint Conf. on Autonomous Agents and Multiagent Systems*, 2006.
- [200] A. Bucci and J. Pollack, "On identifying global optima in cooperative coevolution," in *7th Annual Conference on Genetic and Evolutionary Computation*, 2005.
- [201] M. Nguyen, H. Abbass, and R. McKay, "Analysis of ccme: Coevolutionary dynamics, automatic problem decomposition, and regularization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 1, 2007.
- [202] X.-F. Song, Y. Zhang, Y.-N. Guo, X.-Y. Sun, and Y.-L. Wang, "Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 5, 2020.
- [203] O. Abdelkhalik and A. Gad, "Dynamic-size multiple populations genetic algorithm for multigravity-assist trajectory optimization," *Journal of Guidance, Control and Dynamics*, vol. 35, no. 2, 2012.
- [204] L. Hansen and P. Horst, "Multilevel optimization in aircraft structural design evaluation," *Computers & structures*, vol. 86, no. 1-2, 2008.
- [205] J. Sobieszcanski-Sobieski and R. Haftka, "Multidisciplinary aerospace design optimization: survey of recent developments," *Structural optimization*, vol. 14, no. 1, 1997.
- [206] S. Bandyopadhyay and S. Pal, "Pixel classification using variable string genetic algorithms with chromosome differentiation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 2, 2001.
- [207] U. Maulik and I. Saha, "Modified differential evolution based fuzzy clustering for pixel classification in remote sensing imagery," *Pattern Recognition*, vol. 42, no. 9, 2009.
- [208] E. Falkenauer, "A new representation and operators for GA applied to grouping problems," *Evolutionary computation*, vol. 2, no. 2, 1994.
- [209] V. Costa and C. Rodrigues, "Hierarchical ant colony for simultaneous selection and hyperparameter optimization," in *IEEE Congress on Evolutionary Computation (CEC)*, 2018.
- [210] C. Frank, R. Marlier, O. Pinon-Fischer, and D. Mavris, "An evolutionary multi-architecture multi-objective optimization algorithm for design space exploration," in *57th Structures, Structural Dynamics, and Materials Conf.*, 2016.
- [211] R. Zebulum, M. Vellasco, and M. Pacheco, "Variable length representation in evolutionary electronics," *Evolutionary Computation*, vol. 8, no. 1, 2000.
- [212] A. Gad and O. Abdelkhalik, "Hidden genes genetic algorithm for multi-gravity-assist trajectories optimization," *Journal of Spacecraft and Rockets*, vol. 48, no. 4, 2011.
- [213] O. Abdelkhalik, "Hidden genes genetic optimization for variable-size design space problems," *Journal of Optimization Theory and Applications*, vol. 156, no. 2, 2013.
- [214] J. Gamot, M. Balesdent, A. Tremolet, R. Wuilbercq, N. Melab, and E.-G. Talbi, "Hidden-variables GA for variable-size design space optimal layout problems with application to aerospace vehicles," *Engineering Applications of Artificial Intelligence*, vol. 121, 2023.
- [215] O. Abdelkhalik, "Autonomous planning of multigravity-assist trajectories with deep space maneuvers using a differential evolution approach," *International Journal of Aerospace Engineering*, vol. 2013, 2013.
- [216] Y. C. et al., "Reconfiguration of satellite orbit for cooperative observation using variable-size multi-objective differential evolution," *European Journal of Operational Research*, vol. 242, no. 1, 2015.
- [217] A. Mukhopadhyay and M. Mandal, "Identifying non-redundant gene markers from microarray data," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 11, no. 6, 2014.
- [218] S. Darani and O. Abdelkhalik, "Space trajectory optimization using hidden genes genetic algorithms," *Journal of Spacecraft and Rockets*, vol. 55, no. 3, 2018.
- [219] F. Rothlauf, "Representations for genetic and evolutionary algorithms," in *Representations for Genetic and Evolutionary Algorithms*. Springer, 2006.
- [220] G. Gao, Y. Mei, Y.-H. Jia, W. Browne, and B. Xin, "Adaptive coordination ant colony optimization for multipoint dynamic aggregation," *IEEE Transactions on Cybernetics*, 2021.
- [221] G. Gao, B. Xin, Y. Mei, S. Ding, and J. Li, "A hybrid decomposition-based multi-objective evolutionary algorithm for the multi-point dynamic aggregation problem," *arXiv preprint arXiv:2105.04934*, 2021.
- [222] C. Reuter, M. Schwiengershausen, and P. Pirsch, "Heterogeneous multiprocessor scheduling and allocation using evolutionary algorithms," in *IEEE Int. Conf. on Application-Specific Systems, Architectures and Processors*, 1997.

- [223] B. Wang, Y. Sun, B. Xue, and M. Zhang, "A hybrid differential evolution approach to designing deep convolutional neural networks for image classification," in *Australasian Joint Conference on Artificial Intelligence*, 2018.
- [224] G. Gao, Y. Mei, B. Xin, Y.-H. Jia, and W. Browne, "A memetic algorithm for the task allocation problem on multi-robot multi-point dynamic aggregation missions," in *IEEE Congress on Evolutionary Computation (CEC)*, 2020.
- [225] A. Merlevede, H. Åhl, and C. Troein, "Homology and linkage in crossover for linear genomes of variable length," *Plos one*, vol. 14, no. 1, 2019.
- [226] M. Ryerkerk, R. Averill, K. Deb, and E. Goodman, "A survey of evolutionary algorithms using metameric representations," *Genetic Programming and Evolvable Machines*, vol. 20, no. 4, 2019.
- [227] M. Marek and P. Kadlec, "Another evolution of generalized differential evolution: variable number of dimensions," *Engineering Optimization*, vol. 54, no. 1, 2022.
- [228] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj, "Fractional PSO in multidimensional search space," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 2.
- [229] G. Neumann and D. Cairns, "Targeted eda adapted for a routing problem with variable length chromosomes," in *IEEE Congress on Evolutionary Computation*, 2012.
- [230] P. Dwivedi, V. Kant, and K. Bharadwaj, "Learning path recommendation based on modified variable length genetic algorithm," *Education and information technologies*, vol. 23, no. 2, 2018.
- [231] M. Ryerkerk, R. Averill, K. Deb, and E. Goodman, "Optimization for variable-size problems using genetic algorithms," in *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conf.*
- [232] B. Hutt and K. Warwick, "Synapsing variable-length crossover: Meaningful crossover for variable-length genomes," *IEEE transactions on Evolutionary Computation*, vol. 11, no. 1, 2007.
- [233] D. Goldberg, B. Korb, and K. Deb, "Messy genetic algorithms: Motivation, analysis, and first results," *Complex systems*, vol. 3, no. 5, 1989.
- [234] I. Harvey, R. Manner, and B. Manderick, "The saga cross: the mechanics of crossover for variable-length genetic algorithms," *Parallel Problem Solving from Nature*.
- [235] D. Burke, K. D. Jong, J. Grefenstette, C. Ramsey, and A. Wu, "Putting more genetics into genetic algorithms," *Evolutionary Computation*, vol. 6, no. 4, 1998.
- [236] D. Dasgupta and D. McGregor, "Nonstationary function optimization using the structured genetic algorithm," in *Parallel Problem Solving from Nature Conference (PPSN)*, vol. 2, 1992.
- [237] Y. Yan and L. Osadciw, "Density estimation using a new dimension adaptive PSO algorithm," *Swarm Intelligence*, vol. 3, no. 4, 2009.
- [238] Z. Y. et al., "PSO for base station placement in mobile communication," in *IEEE Int. Conf. on Networking, Sensing and Control*, vol. 1, 2004.
- [239] P. Kadlec and V. Šeděnka, "PSO for problems with variable number of dimensions," *Engineering Optimization*, vol. 50, no. 3, 2018.
- [240] B. Tran, B. Xue, and M. Zhang, "Variable-length PSO for feature selection on high-dimensional classification," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 3, 2018.
- [241] J. Liang, K. Qin, P. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE transactions on Evolutionary Computation*, vol. 10, no. 3, 2006.
- [242] E.-G. Talbi and H. Meunier, "Hierarchical parallel approach for gsm mobile network design," *Journal of Parallel and Distributed Computing*, vol. 66, no. 2, 2006.
- [243] W.-L. Liu, Y.-J. Gong, W.-N. Chen, Z. Liu, H. Wang, and J. Zhang, "Coordinated charging scheduling of electric vehicles: A mixed-variable differential evolution approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 12, 2019.
- [244] L. Gentile, C. Greco, E. Minisci, T. Bartz-Beielstein, and M. Vasile, "Structured-chromosome ga optimisation for satellite tracking," in *Genetic and Evolutionary Computation Conference (GECCO)*, 2019.
- [245] Y. Shan, R. McKay, D. Essam, and H. Abbass, "A survey of probabilistic model building genetic programming," in *Scalable optimization via probabilistic modeling*. Springer, 2006.
- [246] J. Koza, "On the programming of computers by means of natural selection," *Genetic programming*, 1992.
- [247] O. J. Isebor, "Constrained production optimization with an emphasis on derivative-free methods," Ph.D. dissertation, Stanford University Stanford, CA, 2009.
- [248] Y. Kim and O. D. Weck, "Variable chromosome length GA for progressive refinement in topology optimization," *Structural and Multidisciplinary Optimization*, vol. 29, no. 6, 2005.
- [249] J. Englander, B. Conway, and T. Williams, "Automated mission planning via evolutionary algorithms," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 6, 2012.
- [250] C. Chilan and B. Conway, "Automated design of multiphase space missions using hybrid optimal control," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 5, 2013.
- [251] M. Morar, *Bayesian optimisation over mixed parameter spaces*. University of Manchester (United Kingdom), 2021.
- [252] J. Manson, T. Chamberlain, and R. Bourne, "Mvmoo: Mixed variable multi-objective optimisation," *Journal of Global Optimization*, vol. 80, no. 4, 2021.
- [253] R. Tanabe and H. Ishibuchi, "An easy-to-use real-world multi-objective optimization problem suite," *Applied Soft Computing*, vol. 89, 2020.
- [254] H. Li, K. Deb, and Q. Zhang, "Variable-length pareto optimization via decomposition-based evolutionary multiobjective algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 6, 2019.
- [255] Y. Tang, J.-P. Richard, and C. Smith, "A class of algorithms for mixed-integer bilevel min-max optimization," *Journal of Global Optimization*, vol. 66, no. 2, 2016.
- [256] T. Kleinert, M. Labbé, I. Ljubić, and M. Schmidt, "A survey on mixed-integer programming techniques in bilevel optimization," *EURO Journal on Computational Optimization*, vol. 9, 2021.
- [257] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numerica*, vol. 22, 2013.
- [258] W. Yao, X. Chen, W. Luo, T. Van, and J. Guo, "Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles," *Progress in Aerospace Sciences*, vol. 47, no. 6, 2011.

VII. BIOGRAPHY SECTION



El-Ghazali TALBI Prof. El-ghazali Talbi received the Master and Ph.D degrees in Computer Science, both from the Institut National Polytechnique de Grenoble in France. Since 2001, he is a full Professor at the University of Lille. In 2000, he founded and was the head of the optimization research team at the Computer Science laboratory (CRISTAL) and INRIA Lille Nord Europe research institute (France). His research interests are in the field of optimization, metaheuristics, high-performance computing, optimization and machine learning, and application to energy, logistics/transportation, bio-medical and networks.

Professor E-G. Talbi has to his credit more than 250 publications in journals, chapters in books, and conferences. He is the co-editor of 15 books. His book on metaheuristics (Wiley, 2009) is a reference in the area. He was a guest editor of more than 27 special issues in different journals (e.g. Journal of Heuristics, Journal of Parallel and Distributed Computing, European Journal of Operational Research, Theoretical Computer Science, Journal of Global Optimization). He has many collaborative national, European and international projects. He served as a conference chair of more than 20 international conferences.