SMS-EDA-MEC: Extending Copula-based EDAs to Multi-Objective Optimization

Luis Martí*[†], Harold D. de Mello Jr.[‡], Nayat Sanchez-Pi[‡] and Marley Vellasco[§]

*Universidade Federal Flumnense, Niterói (RJ) Brazil
[†]TAO team, CNRS/INRIA/LRI, Université Paris-Saclay, Paris, France.
[‡]Universidade do Estado do Rio de Janeiro, Rio de Janeiro (RJ) Brazil
[§]Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro (RJ) Brazil

Abstract—It can be argued that in order to produce a substantial improvement in multi-objective estimation of distribution algorithms it is necessary to focus on a particular group of issues, in particular, on the weaknesses derived from multi-objective fitness assignment and selection methods, the incorrect treatment of relevant but isolated (precursor) individuals; the loss of population diversity, and the use of 'general purpose' modeling algorithms without taking note of the particular requirements of the task. In this work we introduce the $\mathcal{S}$-Metric Selection Estimation of Distribution Algorithm based on Multivariate Extension of Copulas (SMS-EDA-MEC). SMS-EDA-MEC was devised with the intention of dealing with those issues in mind. It builds the population model relying on the comprehensive Clayton's copula and incorporates methods for automatic population restarting and for priming precursor individuals. The experimental studies presented show that SMS-EDA-MEC yields better results than current and 'traditional' approaches.

I. INTRODUCTION

Many real-world optimization problems involve more than one objective function to be optimized simultaneously. These type of problem is known as multi-objective optimization problems (MOPs). In the general case, finding the solution of a MOP is an *NP*–complete problem [1]. Because of that, metaheuristic approaches are necessary. MOP-related research has seen a great deal of development as a result of the involvement of the evolutionary computation community. This fact has lead to the creation of *multi-objective evolutionary algorithms* (MOEAs) (cf. [2]).

The scalability issues of MOEAs with regard to the number of objective functions problems have triggered a sizable amount of research that aims to provide alternative approaches that can properly handle what has been denominated as the many-objective problems.

The inclusion of learning as part of the search process has been pointed out as a relevant alternative to "traditional" MOEAs [3]. Multi-objective estimation of distribution algorithms (MOEDAs) [4] are one of those alternatives, as they are capable of learning the problem structure. Estimation of distribution algorithms (EDAs) replace the application of evolutionary operators with the creation of a statistical model of the fittest elements of the population in a process known as model building. This model is then sampled to produce new elements.

However, MOEDAs have not yielded the anticipated results. An analysis of this issue [5] lead us to distinguish a number of possible shortcomings, in particular: i) weaknesses derived from multi-objective fitness assignment and selection methods; ii) the incorrect treatment of precursor individuals —that is, relatively isolated and relevant individuals—; iii) the loss of population diversity; and iv) the use of 'general purpose' modeling algorithms for model building without taking into account that MOPs have particular requirements that should be addressed.

In recent years different EDA models based on copula theory [6], [7] have been proposed. However, in spite of its *a priori* advantages, there has been paid relatively little attention to the application of copula-based approaches to multi-objective optimization. For instance, Gaussian and t copulas have already been used in optimization, mainly in the variable space to estimate distribution in evolutionary algorithms [8], [9]. Similarly, [10] focused on objective space by proposing a first estimation of the Pareto front relying on Archimedean copulas. [11], [12] also focused on Archimedean copulas for their ability to model dependence in high dimensions with only one parameter, which has the positive effect of speeding up multi-objective optimization computation time.

In this paper we propose a novel MOEDA algorithm: the $\mathcal{S}$-Metric Selection Estimation of Distribution Algorithm based on Multivariate Extension of Copulas (SMS-EDA-MEC). SMS-EDA-MEC deals with the above issues by combining the successful $\mathcal{S}$-metric selection mechanism [13] with an augmented version of the (single-objective) Estimation of Distribution Algorithm based on Multivariate Extension of Copulas (EDA-MEC) [14] which relies on Clayton's copula [15] and is complemented with heuristics meant to avoid loss of population diversity and to promote search by priming precursor individuals.

As part of the proposal we present a discussion on current MOEDA issues and drawbacks and how to overcome them. We formally described SMS-EDA-MEC and its components and a series of experiments on community-accepted benchmarks. These experiments are devised to study the influence of each of the improvements that are put forward as part of the proposal and to compare our approach with comparable state-of-the-art and 'classical' algorithms.

The rest of the paper is organized as follows. It first introduces the theoretical support matters that are required for correctly understanding the proposal. Subsequently, in Section

III, we describe the essential elements of EDA-MEC, as they are central to our proposal. Next, SMS-EDA-MEC is then presented in Section IV. Following that, in Section V, we present the experimental results that have been carried out. Finally, in Section VI, some final remarks are put forward.

II. FOUNDATIONS

In this section we briefly describe the theoretical pillars that support our proposal.

A. Multi-Objective Optimization

A MOP can be expressed as the problem in which a set of *objective functions* $f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x})$ should be jointly optimized;

$$\min \ \boldsymbol{F}(\boldsymbol{x}) = \langle f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x}) \rangle \, ; \ \boldsymbol{x} \in \mathcal{S} \, ; \qquad (1)$$

where $\mathcal{S} \subseteq \mathbb{R}^n$ is known as the *feasible set* and could be expressed as a set of restrictions over the decision set, in our case, $\mathbb{R}^n$. The image set of $\mathcal{S}$ produced by function vector $\boldsymbol{F}(\cdot)$, $\mathcal{O} \subseteq \mathbb{R}^M$, is called *feasible objective set* or *criterion set*.

The solution to these type of problems is a set of trade-off points. The optimality of a solution can be expressed in terms of the Pareto dominance relation.

Definition 1 (Pareto dominance relation): For the optimization problem specified in (1) and having $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{S}$, $\boldsymbol{x}$ is said to dominate $\boldsymbol{y}$ (expressed as $\boldsymbol{x} \prec \boldsymbol{y}$) iff $\forall f_j$, $f_j(\boldsymbol{x}) \leq f_j(\boldsymbol{y})$ and $\exists f_i$ such that $f_i(\boldsymbol{x}) < f_i(\boldsymbol{y})$.

Definition 2 (non-dominated subset): In problem (1) and having the set $\mathcal{A} \subseteq \mathcal{S}$. $\hat{\mathcal{A}}$, the *non-dominated subset* of $\mathcal{A}$, is defined as

$$\widehat{\mathcal{A}} = \{ \boldsymbol{x} \in \mathcal{A} \,|\, \nexists \boldsymbol{x}' \in \mathcal{A} : \boldsymbol{x}' \prec \boldsymbol{x} \} \, .$$

The solution of (1) is $\hat{\mathcal{S}}$, the non-dominated subset of $\mathcal{S}$. $\hat{\mathcal{S}}$ is known as the *efficient set* or *Pareto-optimal set* [16]. The *Pareto-optimal front*, $\hat{\mathcal{O}}$, is the image of $\hat{\mathcal{S}}$ in the feasible objective set.

If problem (1) has certain characteristics, e. g., linearity or convexity of the objective functions or convexity of $\mathcal{S}$, the efficient set can be determined by mathematical programming approaches [16]. However, in the general case, finding the solution of (1) is an *NP–complete problem* [1]. In this case, heuristic or metaheuristic methods, like evolutionary algorithms, can be applied in order to have solutions of practical value at an admissible computational cost.

B. $\mathcal{S}$-Metric Selection Evolutionary Multiobjective Optimization Algorithm

The $\mathcal{S}$-metric selection evolutionary multiobjective optimization algorithm (SMS-EMOA) [13] is an steady-state algorithm. Which means that, in every iteration, only one individual is created and only one has to be deleted from the population in each generation.

The key element of SMS-EMOA is the method for determining which element of the population will be substituted by the offspring. This is done, by applying a non-domination ranking. From the individuals that are dominated by the rest of the population, one individual is selected such that it has the minimum contribution to the hypervolume [17] of the set. This individual is to be removed from the population and substituted by a new individual generated by the usual variation operators. It may happen, that there is only one non-dominated front (all individuals are non-dominated). In this case, the individual with least hypervolume contribution is selected.

C. MOEAs Stagnation Detection and Stopping Criteria

The formal determination of convergence or optimality criteria in MOPs (and MOEAs, for that matter) is often impossible when gradient information is not available. This is a common situation in real-world application. Because of that, sophisticated heuristic stopping criteria have become a subject of intensive research [18]. The on-line convergence detection criterion (OCD) [18] is a robust method for convergence detection. OCD computes a set of performance indicators applying them on a given number of consecutive populations. Relying on the values of the performance indicators, OCD determines if they have remained stable in a non-progress state applying an statistical hypothesis test. In this work we use OCD to determine if the algorithm has become stagnated and, therefore, a part of the population should be restarted.

D. Multi-Objective Estimation of Distribution Algorithms

As already mentioned in the previous section, estimation of distribution algorithms (EDAs) are a class of optimization algorithms that combine evolutionary computation, statistical modeling, and machine learning. The most important step — and bottleneck— of EDAs is estimating the joint probability distribution associated with the variables from the most promising solutions determined by the evaluation function. One main advantage of EDAs is that the search distribution may encode dependences between the domain problem parameters, thus performing a more effective search.

Multi-objective optimization EDAs (MOEDAs) are the extensions of EDAs to the multi-objective domain. Most MOEDAs are a modification of existing EDAs whose fitness assignment strategy has been replaced by a previously existing method used by MOEAs.

E. Copulas

The concept of copulas was introduced by Sklar [6], but remained largely dormant until it was applied to finance only many years later [7]. Currently, copulas constitute a very useful tool for allowing the construction of multivariate models that adequately characterize linear and nonlinear dependencies between the variables of a problem.

Formally, a function $C(u_1, \ldots, u_n) : [0,1]^n \rightarrow [0,1]$ is an n-dimensional copula if it satisfies axiomatic conditions related to boundary conditions and increasing property [19]. Thus, copula C is a distribution function in $[0,1]^n$ that has marginal uniform functions $u_k \in (0,1)$, with $k = 1, \ldots, n$.

According to Sklar's theorem [6], a joint cumulative distribution function (JCDF), F, of random variables $X_1, X_2, \ldots, X_n$, with continuous marginal distributions

$F_1, F_2, \ldots, F_n$, respectively, can be characterized by a single n-dimensional dependence function or copula C, such that for all vectors $\boldsymbol{x} \in \mathbb{R}^n$:

$$F(x_1, x_2, \ldots, x_n) = C(F_1(x_1), F_2(x_2), \ldots, F_n(x_n)) \quad (2)$$

As an alternative representation, (2) can be written in the inverted form. For any vector $\boldsymbol{u} \in [0,1]^n$:

$$C(u_1, u_2, \ldots, u_n) = \boldsymbol{F}\left[F_1^{-1}(u_1), F_2^{-1}(u_2), \ldots, F_n^{-1}(u_n)\right], \quad (3)$$

where C_F denotes the copula associated with F and $F^{-1}(\boldsymbol{u}) = inf\{x \in \mathbb{R} \mid F_i(x_i) \geq \boldsymbol{u}\}$ with $i = 1, \ldots, n$, constitutes the generalized inverse function of F. Then $x_1 = F_1^{-1}(u_1) \sim F_1, \ldots, x_n = F_n^{-1}(u_n) \sim F_n$. It follows that an n-copula is a multivariate distribution with all n univariate margins being uniformly distributed as $U(0,1)$.

A particular group of copulas that has proved useful in empirical modeling is the Archimedean class. In addition to the fact that many parametric copulas are Archimedean, their most common subclasses can i) be expressed in an explicit closed form and ii) capture wide ranges of dependence. Nevertheless, this type of copula is not directly derived from Sklar's theorem. For the bivariate case, this copula holds the following representation:

$$C(u_1, u_2) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2)), \quad (4)$$

where $u_1, u_2 \in (0,1)$, $\varphi(\cdot)$ is known as a generator function of the copula, and φ^{-1} is its inverse.

Different generator functions yield different Archimedean copulas when plugged into (4). Several Archimedean copula families are described in [19]; most of them depend on a single parameter θ that controls the dependence structure and is embedded in the functional form of the generator. Quantifying dependence is relatively straightforward for Archimedean copulas because Kendall's τ simplifies to a function of the generator function.

When $\varphi(t) = t^{-\theta} - 1$, a family of the Clayton's bivariate copula [15] is generated. It is one of the most frequently used and best known Archimedean copulas in empirical applications and takes the form

$$C(u_1, u_2) = \left(u_1^{-\theta} + u_2^{-\theta} - 1\right)^{-1/\theta} \quad (5)$$

The Clayton family has lower-tail dependence $2^{-1/\theta}$ but does not have upper-tail dependence. In this copula, the perfect dependence among the random variables occurs when $\theta \to \infty$, while $\theta = 1$ indicates total independence among the variables. The Clayton copula's parameter is often estimated via Kendall's τ [19]:

$$\tau = \frac{\theta}{\theta + 2} \quad (6)$$

An additional advantage of Archimedean copulas is that they can be easily used to generate multivariate distributions from extensions of Archimedean 2-copulas. Although, the simplest method used in this work is named an exchangeable multivariate Archimedean copula (EAC), it poses a limitation:

the dependence structure between any pair of variables is described by the same parameter θ, regardless of dimension. In this respect, copulas created by the nesting of Archimedean copula generators and *pair-copulas* models [20] are more flexible.

F. Copula-based Estimation of Distribution Algorithms

In recent years, a new approach for developing EDAs to solve real-valued optimization problems has been developed based on copula theory, the Copula-based Estimation of Distribution Algorithms (CEDAS). The first attempt to associate copulas with EDAs was a technical report [21]. Since then, a considerable number of CEDAs have been proposed in the literature. See [22] to a review. Nonetheless, most EDAs are applicable to single-objective problems and only a few CEDAs have been proposed to deal with multi-objective problems.

In CEDAs, the step for estimating the probabilistic model is divided into two parts: i) estimating the marginal distributions, F_i, for $i = 1, \ldots, n$, and ii) estimating the dependence structure. Typically, a specific distribution is assumed to be the correct distribution for each marginal distribution, and its parameters are estimated based on maximum likelihood. In other cases, kernel density estimators (KDE), marginal empirical distributions or estimators based on Kendall's inversion were used.

After the marginal distributions are estimated, the selected population is transformed into uniform variables in the interval $(0, 1)$ by evaluating each cumulative marginal distribution. This transformed population is then used to estimate a copula-based model C that describes the dependence among the variables.

III. THE EDA BASED ON MULTIVARIATE EXTENSION OF COPULAS

The Estimation of Distribution Algorithm based on Multivariate Extension of Copulas (EDA-MEC) [14] is a Copula-based EDA that implements two approaches to also address the previously described issues of restarting and precursor individuals handling. In this work we extend EDA-MEC to the multi-objective problem domain. Therefore, it is convenient to describe the single-objective version first.

The use of copulas in EDAs can simplify their operation, once the construction of the probability models is performed in less time and with greater accuracy than other estimation techniques. Since a copula completely determines the dependence structure of variables and the multivariate distribution F is obtained by (2), the issue of estimating the probabilistic model of a copula-based EDA is split into two parts:

1) selecting or building the copula C, and
2) estimating each univariate marginal F_i for $i = 1, \ldots, n$.

After estimating the probabilistic model, the next step is sampling it to generate the new solutions. In order to do this, the algorithm generates a random vector $\{u_1, u_2, \ldots, u_n\} \in [0, 1]^n$ that follows C (basic property of the copula theory). Next, the x_i values are calculated by the inverse function of their corresponding marginal, i.e., $x_i = F^{-1}(u_i)$, thereby

generating a new individual $\{x_1, x_2, \ldots, x_n\}$ that is a sample that follows the multivariate distribution F.

Thus, the most significant differences between copula-based EDAs reside in the steps of learning, construction and sampling of the probabilistic model. [14] generated a random variable $\{u_1, u_2, \ldots, u_n\}$ from the Clayton's copula using the Marshall and Olkin's method for constructing multivariate copulas [7], [23]. The dependence parameter (θ) was estimated dynamically with an alternative and less computationally intense method compared to other copula-based EDAs. In this method, the correlation matrix for all variables is calculated using the current population as the data inputs. Subsequently, it was possible to calculate Kendall's τ and to obtain the parameter θ for Clayton's copula using (6).

Additionally, EDA-MEC incorporates a restarting method proposed in [14], [24] to maintain the population diversity and avoid the premature convergence. Whenever the perfect dependence among the random variables occurs ($\theta \to \infty$), the best individual is retained, a population's percentage is restarted uniformly in the feasible set, and the remaining population is restarted uniformly within the limits of the best individual variables at that generation. Nevertheless, the population size is not changed after a restarting that differ from the mechanisms cited previously. Subsequently, a new rank matrix is calculated.

Subsequently, x_i new individuals are generated using the calculated dependence parameter and the samples **u** from the Clayton's copula. This offspring population is merged with the parents. After that, the worst individuals are selected from the merged population and removed from it.

After a transformation in the samples generated in the next step, precursors $r_n = F_n^{-1}(1-u_n)$ are generated. According to [14] the precursor individuals correspond to samples of a new probability distribution is obtained as a variation of the learned one so as to generate individuals with a lower probability of appearing in the evolutionary process. This concept has been used to determine the inverse probability for discrete variables [25] and explore other regions of the search space which helps to avoid premature convergence. It consists of using the information which is contained in the probabilistic model to generate individuals which introduces diversity. It must be highlighted that the precursor individuals do not necessarily have to be obtained from the complement of samples; they can be generated from any process able to explore other regions of the search space.

Finally, the r_n individuals replace the worst individuals from the resulting population. This entire process is repeated until a stop condition is attained, and the best individual is found.

IV. $\mathcal{S}$-METRIC SELECTION EDA-MEC

The $\mathcal{S}$-Metric Selection EDA-MEC (SMS-EDA-MEC) extrapolates the concepts developed for EDA-MEC to the multi-objective optimization problem domain by combining it with the multi-objective $\mathcal{S}$-metric selection mechanism. Although the ideas put forward by EDA-MEC are valid in the multi-objective case some modifications are necessary to deal with the particularities of the problem.

A. Restarting in the Multi-Objective Case

In EDA-MEC, the need for restarting is established by either the difference between the values of the objective function of the best and worst individuals being too small, the loss of population diversity, or after a given number of iterations had passed. The first condition can not be directly extrapolated to the multi-objective domain. Instead, what is necessary is to determine when progress towards the Pareto-optimal front is taking place too slow pace. That situation can be detected with the OCD stopping criteria described in Section II-C. OCD can be configured in such way as to not detect convergence in the most strict way but to detect (momentarily) stagnation situations that might be good candidates for restarting. Similarly, SMS-EDA-MEC extends the idea of population diversity by measuring the approximated volume of the convex hull [26] defined by the individuals in feasible set. In this point, it should be mentioned that other diversity indicators could be used. They will be explored in future works.

After the conditions for restart are met, care should be taken not to make an excessively aggressive restart. In EDA-MEC the whole population with the exception of the best individual was restarted. In the multi-objective case, there is more than one optimal solution and hence, equally good (non-dominated) individuals represent different trade-off of the objective functions. Therefore restarting, although needed should substitute a relatively small amount of individuals.

Taking these arguments into consideration, in SMS-EDA-MEC restarting takes place by generating two sets of restarted individuals, $\mathcal{P}_{\text{local}}^{\text{restarted}}$ that does a 'local' restart and its 'global' counterpart, $\mathcal{P}_{\text{global}}^{\text{restarted}}$, such that

$$\left|\mathcal{P}_{\text{local}}^{\text{restarted}}\right| = \left|\mathcal{P}_{\text{local}}^{\text{restarted}}\right| = \frac{\lceil \nu n_{\text{pop}} \rceil}{2}, \qquad (7)$$

where ν is the restarting percentile that control how much of the population is to be substituted by restarted individuals.

$\mathcal{P}_{\text{local}}^{\text{restarted}}$ is created by sampling a uniform distribution on the interval of the feature set currently occupied by the population $\mathcal{P}$ as,

$$\mathcal{P}_{\text{local}}^{\text{restarted}} = \left\{ x \sim U(\boldsymbol{\ell}^{\text{lower}}, \boldsymbol{\ell}^{\text{upper}}) \right\}, \qquad (8)$$

where $\ell_i^{\text{lower}} = \min_{\boldsymbol{x} \in \mathcal{P}}(x_i)$ and $\ell_i^{\text{lower}} = \max_{\boldsymbol{x} \in \mathcal{P}}(x_i)$ for $i = 1, \ldots, n$.

Having $\mathcal{P}_{\text{local}}^{\text{restarted}}$ and $\mathcal{P}_{\text{global}}^{\text{restarted}}$ they are merged with the original population $\mathcal{P}$ by substituting randomly selected individuals from it.

It should be noted that in practical applications it is convenient to disable restarting in the last part of the optimization. For example, in our experiments we included a condition to prevent restarting in the last 10% of the execution of the algorithm.

B. The SMS-EDA-MEC Algorithm

The SMS-EDA-MEC algorithm goes by creating an initial population, $\mathcal{P}_0$, of n_{pop} random individuals.

Then, the optimization loop starts by determining if the current population, $\mathcal{P}_t$, calls for a restart. This decision is made relying on the $\texttt{OCD}(\cdot)$ and $\texttt{convex\_hull\_volume}(\cdot)$ functions as described in Section IV-A and forcing a pause of $\Delta t_{\text{restart}}$ iterations between restarts to avoid continuous restarting. If a restart is required the restarted population is computed by the $\texttt{restart}(\cdot)$ function.

After that the Clayton's copula C is estimated from the population by function $\texttt{estimate\_copula}(\cdot)$. This copula is used to synthesize n_{off} individuals that are united with the current population to form $\mathcal{P}'$.

Subsequently, the $\mathcal{S}$-metric selection mechanism is applied to $\mathcal{P}'$. That is, individuals are removed from $\mathcal{P}'$ until its size reaches $n_{\text{pop}} - \lceil \varepsilon n_{\text{pop}} \rceil$ Here ε regulates how many precursor individuals will be kept across iterations. Individuals to be removed are selected by their hypervolume contribution by function $\texttt{worst\_individual}(\cdot)$.

At this point, the subset $\Upsilon \subseteq \mathcal{P}_t$ containing $\lceil \varepsilon n_{\text{pop}} \rceil$ precursor individuals are generated as described in Section III using as reference $\mathcal{P}_t$ by function $\texttt{precursors}(\cdot)$. The selected precursors, Υ and the selected offspring $\mathcal{P}'$ are combined to form to population of the next generation,

$$\mathcal{P}_{t+1} = \mathcal{P} + \Upsilon. \qquad (9)$$

Figure 1 details the aforementioned process as pseudocode.

V. Experiments

This section reports the results of the experiments involving different variants of SMS-EDA-MEC and some of the current state-of-the-art MOEDAs and MOEAs in a set of current community-accepted problems.

As stated, the scope of the experiments is to not only compare our proposal with similar approaches. SMS-EDA-MEC implements different features, that is, model building using copulas, the restarting procedure and the selection of precursors. Therefore, it is also important to identify which of those features have a more relevant contribution in a positive outcome. That is why the test includes the complete SMS-EDA-MEC, identified as SMS-EDA-MEC+RP, as it includes restarting and precursors, SMS-EDA-MEC with restart (SMS-EDA-MEC+R), SMS-EDA-MEC with precursors (SMS-EDA-MEC+P) and SMS-EDA-MEC without restarting and precursors (SMS-EDA-MEC-RP).

Besides SMS-EDA-MEC, we also tested MONEDA [5], [27], MAMaLGaM-X+ [28], naïve MIDEA [29] MO-CMA-ES [30] and MrBOA [31] MOEDAs and the SMS-EMOA [13], NSGA-II [2] and NSGA-III [32] MOEAs.

MONEDA and MAMaLGaM-X+ have been included as they put forward approaches to improve MOEDAs by correctly handling precursor and restarting, although. Naïve MIDEA and MrBOA, on the other hand, are well-known and established MOEDAs. It is interesting to compare our results against the ones yielded by SMS-EMOA as it allows to assess the impact of the EDA component as optimization search engine. Similarly, NSGA-II is a popular and well-known MOEA, while NSGA-III builds upon it and uses an approach

---

Parameters:
- $\triangleright$ $n_{\text{pop}} \in \mathbb{N}$, population size.
- $\triangleright$ $n_{\text{off}} \in \mathbb{N}$, number of offspring to generate.
- $\triangleright$ $\varepsilon \in [0, 1]$, precursor selection percentile.
- $\triangleright$ $\nu \in [0, 1]$, restarting percentile.
- $\triangleright$ $\sigma_{\min} \in \mathbb{R}^+$, minimum OCD variance for restarting.
- $\triangleright$ $\Delta t_{\text{OCD}} \in \mathbb{N}$, number of iterations to apply OCD.
- $\triangleright$ $\gamma \in \mathbb{R}^+$, population convex hull volume threshold.
- $\triangleright$ $\Delta t_{\text{r}} \in \mathbb{N}$, min. number of iterations before restarts.

Algorithm:
$t \leftarrow 0$, $t_{\text{r}} \leftarrow 0$, $\mathfrak{P} \leftarrow \varnothing$.
$\mathcal{P}_0 \leftarrow \texttt{random\_population}(n_{\text{pop}})$.
while not $\texttt{end\_condition}(\mathcal{P}_t)$ **do**
  $\mathfrak{P} = \mathfrak{P} \cup \{\mathcal{P}_t\}$.
  if $t > \Delta t_{\text{OCD}}$ **and**
    $t - t_{\text{r}} > \Delta t_{\text{r}}$ **and**
    $[\texttt{OCD}(\mathfrak{P}, \sigma_{\min}, \Delta t_{\text{OCD}}) = \textbf{true}$ **or**
    $\texttt{convex\_hull\_volume}(\mathcal{P}_t) < \gamma]$ **then**
    $t_{\text{r}} \leftarrow t$. // *doing restart*
    $\mathcal{P}_t \leftarrow \texttt{restart}(\mathcal{P}_t, \lceil \nu n_{\text{pop}} \rceil)$.
  end if
  $C \leftarrow \texttt{estimate\_copula}(\mathcal{P}_t)$.
  $\mathcal{P}' \leftarrow \mathcal{P}_t \cup \texttt{sample\_copula}(C, n_{\text{off}})$.
  while $|\mathcal{P}'| > n_{\text{pop}} - \lceil \varepsilon n_{\text{pop}} \rceil$ **do**
    $\mathcal{P}' \leftarrow \mathcal{P}' \setminus \{\texttt{worst\_individual}(\mathcal{P}')\}$.
  end while
  $\Upsilon \leftarrow \texttt{precursors}(\mathcal{P}, \lceil \varepsilon n_{\text{pop}} \rceil)$.
  $\mathcal{P}_{t+1} \leftarrow \mathcal{O} \cup \Upsilon$.
  $t \leftarrow t + 1$.
end while
return $\hat{\mathcal{P}}$, the non-dominated subset of $\mathcal{P}_t$.

Figure 1. Algorithmic description of SMS-EDA-MEC. See Table **??** for a description of the functions used.

---

based on reference points to yield a better performance in higher objective dimensions.

We selected the Walking Fish Group multi-objective problem toolkit [33] as benchmark tests. It describes nine complex problems (from WFG1 to WFG9) that test whether the optimization algorithms are capable of handling different challenges, like separability, multi-modality, deceptive local optima, etc.

Each problem was addressed with 3, 5, 7 and 9 objectives. The number of objective function evaluation was used as stopping criterion, stopping the execution of the algorithm when the number of evaluations reached $10^{3+\frac{M}{3}}$, thus allowing longer runs for more complex problems. Population sizes for the algorithms were to automatically increase with problem dimension as $n_{\text{pop}} = 50 \times 10^{\frac{M}{3}}$.

In these experiments, SMS-EDA-MEC was run with $n_{\text{off}} = n_{\text{pop}}$, $\varepsilon = 0.1$, $\nu = 0.1$, $\sigma_{\min} = 10^{-5}$, $\Delta t_{\text{OCD}} = 15$, $\gamma = 10^{-5}$ and $\Delta t_{\text{restart}} = 10$. The parameters of the other algorithms were set as in the respective referenced papers. For each algorithm/problem/number-of-objectives combination 30 runs of the algorithm were executed in order to have statistically

valid results.

The stochastic nature of the algorithms prompts the use of statistical tools in order to reach a valid judgement of the quality of the solutions and how different algorithms compare with each other. Box plots [34] are one of such representations and have been repeatedly applied in our context. Although box plots allows a visual comparison of the results and, in principle, some conclusions could be deduced out of them.

Figure 2 shows the results of the experiments in the form of box plots. Examining that figure it becomes evident that SMS-EDA-MEC is able to consistently yield good results across all problems and number of objectives.

Although illustrative, box plots can not be used to reach a definitive conclusion. That is why statistical hypothesis tests are called for. In our case, for each problem/dimension combination, we performed a Kruskal-Wallis test [35] with the indicator values yielded by each algorithm's run. In this context, the null hypothesis of this test is that all algorithms are equally capable of solving the problem. If the null hypothesis was rejected, which was actually the case in all instances of the experiment, the Conover-Inman procedure [36] was applied in a pairwise manner to determine whether one algorithm had significantly better results than another. A significance level, α, of 0.05, corrected using the Dunn–Šidák correction was used for all tests.

The separate results of the tests in tabular form could not be included in the paper because of the imposed length restrictions[1]. It is evident, that, in any case, comprehensive analysis of the results is rather difficult as it implies cross-examining and comparing the results presented separately. That is why we decided to adopt a more integrative representation such as the one proposed in [37]. That is, for a given set of algorithms $A_1, \ldots, A_K$, a set of P test problem instances $\Phi_{1,m}, \ldots, \Phi_{P,m}$, configured with m objectives, the function $\delta(\cdot)$ is defined as

$$\delta\left(A_i, A_j, \Phi_{p,m}\right) = \begin{cases} 1 & \text{if } A_i \gg A_j \text{ solving } \Phi_{p,m} \\ 0 & \text{in other case} \end{cases}, \quad (10)$$

where the relation $A_i \gg A_j$ defines if A_i is significantly better than A_j when solving the problem instance $\Phi_{p,m}$, as computed by the statistical tests previously described.

Relying on $\delta(\cdot)$, the performance index $P_{p,m}(A_i)$ of a given algorithm A_i when solving $\Phi_{p,m}$ is then computed as

$$P_{p,m}\left(A_i\right) = \sum_{j=1; j \neq i}^{K} \delta\left(A_i, A_j, \Phi_{p,m}\right). \quad (11)$$

This index intends to summarize the performance of each algorithm with regard to its peers.

Figures 3a and 3b exhibit the results computing the performance indexes grouped by problems and dimensions.

Figure 3a represents the mean performance indexes yielded by each algorithm when solving each problem in all of its configured objective dimensions,

$$\bar{P}_p\left(A_i\right) = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} P_{p,m}\left(A_i\right). \quad (12)$$

Similarly, Figure 3b presents the mean values of the index grouped by for each feasible objective set dimension,

$$\bar{P}_m\left(A_i\right) = \frac{1}{P} \sum_{p=1}^{P} P_{p,m}\left(A_i\right), \quad (13)$$

with $m = \{3, 5, 7, 9\}$.

These figures lead to solid conclusions. They allow us to assert that SMS-EDA-MEC with restarting and precursors has consistently outperformed the other approaches. It is also noticeable the importance of restarting and precursors. Both SMS-EDA-MEC+R and SMS-EDA-MEC+P yielded better results than the 'plain' (-RP) version. The only case where SMS-EDA-MEC was outperformed by other algorithm was by MONEDA, an algorithm that also has as working principle the correct handling of precursors.

VI. FINAL REMARKS

The present paper introduced the $\mathcal{S}$-Metric Selection EDA-MEC (SMS-EDA-MEC), which is the extension to the multi-objective domain of the Estimation of Distribution Algorithm based on Multivariate Extension of Copulas (EDA-MEC). SMS-EDA-MEC constructs its model of the population relying on the Clayton's copula and includes a process for automatic population restarting to overcome early stagnation and a novel mechanism to prioritize precursor areas of the feasible set. These features have rendered SMS-EDA-MEC capable of outperforming the other approaches included in the experiments.

In spite of the promising results obtained so far further studies are necessary. More experiments are necessary in order to gain a better comprehension of the proposed algorithm. In one hand, different test problems must be addressed to realize if the results obtained here can be generalized. On the other hand, it is also of interest to further scale the problems to more objective functions. The analysis of the behavior of the algorithms in those situations might lead to their adaptation to the problem.

From a theoretical perspective some points still need to be explored. For example, a computational cost and complexity study is necessary in order to grasp the resource consumption of SMS-EDA-MEC when advancing into higher dimensions. Because of its reliance on computationally demanding components, like the hypervolume indicator, it is necessary to assess lower-cost variants, like the approximation of the hypervolume via Monte Carlo sampling. On the other hand, more model building approaches based on copulas should be studied. Such models could be found by doing a more exhaustive survey or by proposing completely new ones. If such approaches show themselves as efficient as the one described here, it would help to corroborate the working hypothesis of this contribution.

---

[1]The tables with the results of the statistical tests are available online at http://lmarti.com/smsedamec-cec2016.
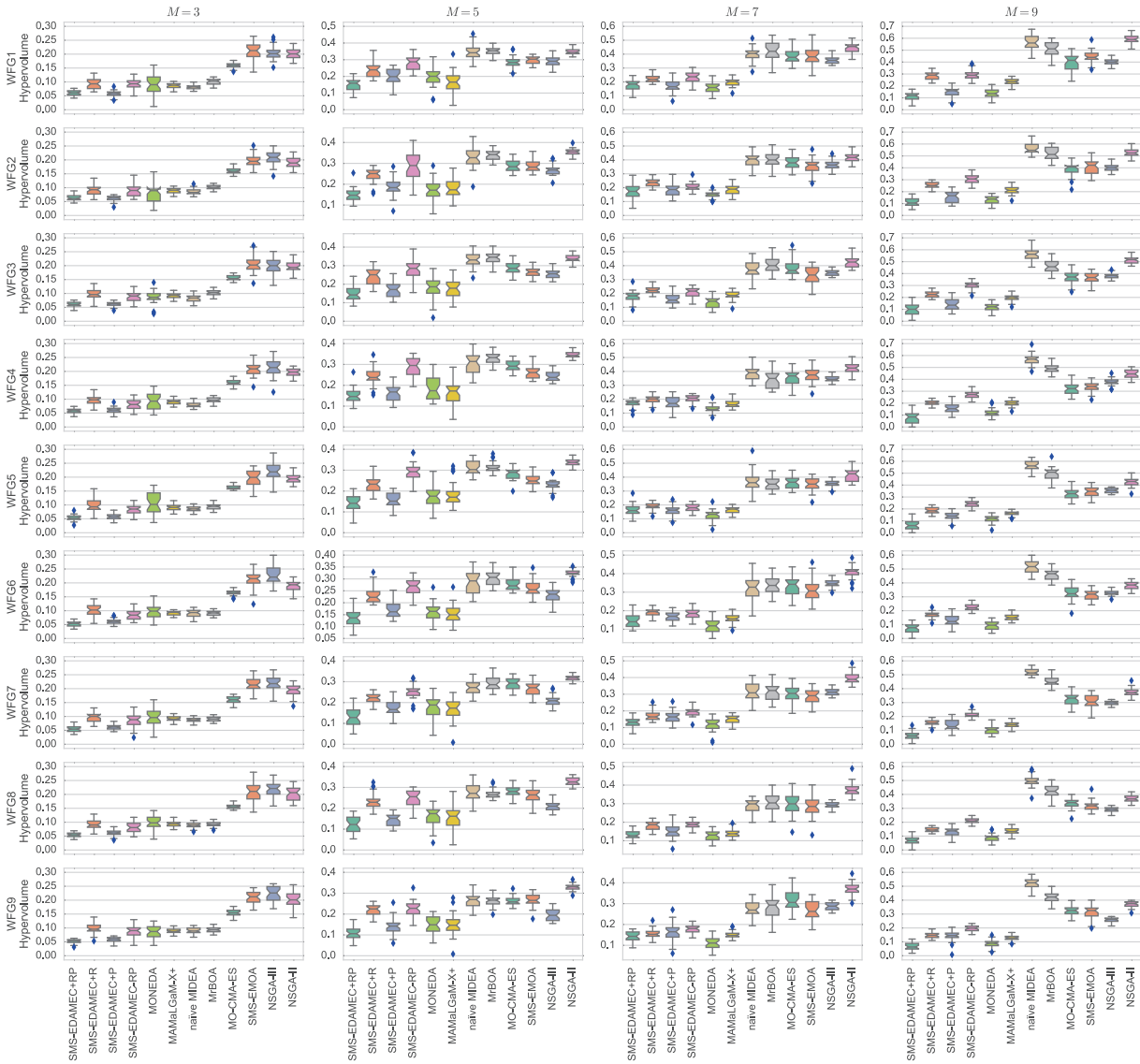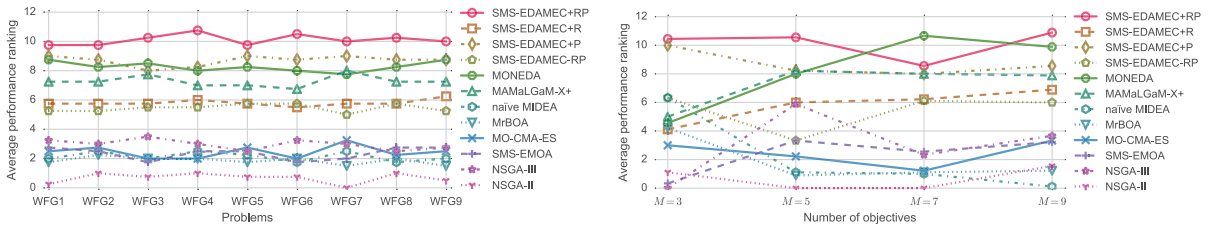
Figure 2. Box plots produced by the experiments organized by number of objectives ($M = \{3, 5, 7, 9\}$) and problem.

(a) Mean values of the performance index grouped by problem.

(b) Mean values of the performance index grouped by number of objectives.

Figure 3. Summaries of the statistical hypothesis tests grouped by problem and by number of objectives.

encouraging comments. The source code open and available at http://lmarti.com/smsedamec-cec2016.

REFERENCES

[1] T. Bäck, *Evolutionary Algorithms in Theory and Practice*. New York — Oxford: Oxford University Press, 1996.

[2] C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed., ser. Genetic and Evolutionary Computation. New York: Springer, 2007. [Online]. Available: http://www.springer.com/west/home/computer/foundations?SGWID=4-156-22-173660344-0

[3] D. W. Corne, "Single objective = past, multiobjective = present, ??? = future," in *2008 IEEE Conference on Evolutionary Computation (CEC), part of 2008 IEEE World Congress on Computational Intelligence (WCCI 2008)*, Z. Michalewicz, Ed. Piscataway, New Jersey: IEEE Press, 2008.

[4] M. Pelikan, K. Sastry, and D. E. Goldberg, "Multiobjective estimation of distribution algorithms," in *Scalable Optimization via Probabilistic Modeling: From Algorithms to Applications*, ser. Studies in Computational Intelligence, M. Pelikan, K. Sastry, and E. Cantú-Paz, Eds. Springer–Verlag, 2006, pp. 223–248.

[5] L. Martí, "Scalable multi-objective optimization," Ph.D. dissertation, Departmento de Informática, Universidad Carlos III de Madrid, Colmenarejo, Spain, 2011. [Online]. Available: http://hdl.handle.net/10016/11896

[6] A. Sklar, "Fonctions de répartition à n dimensions et leurs marges," *Publications de l'Institut de Statistique de L'Université de Paris*, vol. 8, pp. 229–231, 1959.

[7] U. Cherubini, E. Luciano, and W. Vecchiato, *Copula Methods in Finance*, 1st ed. Chichester: John Wiley & Sons Ltd, 2004.

[8] Y. Gao, L. Peng, F. Li, M. Liu, and X. Hu, "Pareto-based multi-objective estimation of distribution algorithm with gaussian copulas and application in RFID network planning," in *2012 IEEE Fifth International Conference on Advanced Computational Intelligence (ICACI)*, Oct 2012, pp. 370–373.

[9] ——, "Estimation of distribution algorithm with multivariate t-copulas for multi-objective optimization," *Intelligent Control and Automation*, vol. 4, no. 1, pp. 63–69, 2013.

[10] M. Binois, D. Rullière, and O. Roustant, "On the estimation of Pareto fronts from the point of view of copula theory," *Information Sciences*, vol. 324, pp. 270–285, Dec. 2015. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01097403

[11] A. Cheriet and F. Cherif, "A posteriori pareto front diversification using a copula-based estimation of distribution algorithm," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 6, no. 12, pp. 23–35, 2015.

[12] Y. Gao, L. Peng, F. Li, M. Liu, and X. Hu, "Multiobjective estimation of distribution algorithms using multivariate archimedean copulas and average ranking," in *Foundations of Intelligent Systems*, ser. Advances in Intelligent Systems and Computing, Z. Wen and T. Li, Eds. Springer Berlin Heidelberg, 2014, vol. 277, pp. 591–601.

[13] N. Beume, B. Naujoks, and M. Emmerich, "SMS–EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, September 2007. [Online]. Available: http://ideas.repec.org/a/eee/ejores/v181y2007i3p1653-1669.html

[14] H. D. de Mello Jr., A. V. Abs da Cruz, and M. M. B. R. Vellasco, "Nonconvex functions optimization using an estimation of distribution algorithm based on a multivariate extension of the clayton copula," in *Intelligent Data Engineering and Automated Learning (IDEAL 2014)*, ser. Lecture Notes in Computer Science, E. Corchado, J. A. Lozano, H. Quintián, and H. Yin, Eds. Springer International Publishing, 2014, vol. 8669, pp. 318–326.

[15] D. G. Clayton, "A model for association in bivariate life tables and its application in epidemiological studies of familial tendency in chronic disease incidence," *Biometrika*, vol. 65, no. 1, pp. 141–151, 1978. [Online]. Available: http://biomet.oxfordjournals.org/content/65/1/141.abstract

[16] J. Branke, K. Miettinen, K. Deb, and R. Słowiński, Eds., *Multiobjective Optimization*, ser. Lecture Notes in Computer Science. Berlin/Heidelberg: Springer–Verlag, 2008, vol. 5252.

[17] R. Lacour, K. Klamroth, and C. M. Fonseca, "A box decomposition algorithm to compute the hypervolume indicator," *CoRR*, vol. abs/1510.01963, 2015. [Online]. Available: http://arxiv.org/abs/1510.01963

[18] T. Wagner, H. Trautmann, and L. Martí, "A taxonomy of online stopping criteria for multi-objective evolutionary algorithms," in *6th International Conference on Evolutionary Multi-Criterion Optimization (EMO 2011)*, R. H. C. Takahashi, K. Deb, E. F. Wanner, and S. Greco, Eds., vol. 6576. Berlin/Heidelberg: Springer, 2011, pp. 16–30, 10.1007/978-3-642-19893-9_2.

[19] R. B. Nelsen, *An Introduction to Copulas*, 2nd ed. New York: Springer, 2006.

[20] K. Aas, C. Czado, A. Frigessi, and H. Bakken, "Pair-copula constructions of multiple dependence," *Insurance, Mathematics and Economics*, vol. 44, no. 2, pp. 182–198, 2009.

[21] M. Soto, A. Ochoa, and R. J. Arderí, "Gaussian copula estimation of distribution algorithm," Institute of Cybernetics, Mathematics and Physics, Cuba, Tech. Rep. ICIMAF 2007-406, 2007.

[22] Y. González Fernández and M. Soto, "copulaedas: An r package for estimation of distribution algorithms based on copulas," *Journal of Statistical Software*, vol. 58, no. 9, 2014. [Online]. Available: http://www.jstatsoft.org/v58/i09/

[23] J.-F. Mai and M. Scherer, *Simulating Copulas: Stochastic Models, Sampling Algorithms and Applications*, ser. Series in Quantitative Finance - Vol. 4. London: Imperial College Press, 2012.

[24] H. D. De Mello Jr., A. Abs da Cruz, and M. Vellasco, "Estimation of distribution algorithm based on a multivariate extension of the archimedean copula," in *2013 Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence (BRICS-CCI CBIC 2013)*, Sept 2013, pp. 75–80.

[25] L. de la Ossa, J. A. Gamez, J. L. Mateo, and J. M. Puerta, "Avoiding premature convergence in estimation of distribution algorithms," in *IEEE Congress on Evolutionary Computation 2009 (CEC 2009)*, May 2009, pp. 455–462.

[26] M. Dyer, A. Frieze, and R. Kannan, "A random polynomial-time algorithm for approximating the volume of convex bodies," *Journal of the ACM (JACM)*, vol. 38, no. 1, pp. 1–17, 1991.

[27] L. Martí, J. García, A. Berlanga, and J. M. Molina, "MONEDA: scalable multi-objective optimization with a neural network-based estimation of distribution algorithm," *Journal of Global Optimization*, pp. 1–40, 2016.

[28] P. A. N. Bosman, "The anticipated mean shift and cluster registration in mixture-based EDAs for multi-objective optimization," in *Proceedings of the 12th annual conference on Genetic and evolutionary computation - GECCO '10*. New York, New York, USA: ACM Press, 2010, p. 351. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1830483.1830549

[29] P. A. N. Bosman and D. Thierens, "The Naive MIDEA: A Baseline Multi–objective EA," in *Evolutionary Multi-Criterion Optimization*, 2005, pp. 428–442. [Online]. Available: http://www.springerlink.com/content/gh7kkhhwbc5lw47a

[30] C. Igel, N. Hansen, and S. Roth, "Covariance matrix adaptation for multi-objective optimization," *Evolutionary Computation*, vol. 15, no. 1, pp. 1–28, 2007.

[31] C. Ahn, *Advances in Evolutionary Algorithms*, ser. Studies in Computational Intelligence. Berlin/Heidelberg: Springer-Verlag, 2006, vol. 18. [Online]. Available: http://link.springer.com/10.1007/3-540-31759-7

[32] K. Deb and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, aug 2014. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6600851

[33] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.

[34] J. Chambers, W. Cleveland, B. Kleiner, and P. Tukey, *Graphical Methods for Data Analysis*. Belmont: Wadsworth, 1983.

[35] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion analysis of variance," *Journal of the American Statistical Association*, vol. 47, pp. 583–621, 1952.

[36] W. J. Conover, *Practical Nonparametric Statistics*, 3rd ed. New York: John Wiley & Sons, 1999.

[37] J. Bader, "Hypervolume-Based Search for Multiobjective Optimization: Theory and Methods," Ph.D. dissertation, ETH Zurich, Switzerland, 2010.