The objective that freed me: a multi-objective local search approach for continuous single-objective optimization

Pelin Aspar[1,3] · Vera Steinhoff[1] · Lennart Schäpermeier[2] · Pascal Kerschke[2] · Heike Trautmann[1,4] · Christian Grimme[1]

Abstract

Single-objective continuous optimization can be challenging, especially when dealing with multimodal problems. This work sheds light on the effects that multi-objective optimization may have in the single-objective space. For this purpose, we examine the inner mechanisms of the recently developed sophisticated local search procedure SOMOGSA. This method solves multimodal single-objective continuous optimization problems based on first expanding the problem with an additional objective (e.g., a sphere function) to the bi-objective domain and subsequently exploiting local structures of the resulting landscapes. Our study particularly focuses on the sensitivity of this multiobjectivization approach w.r.t. (1) the parametrization of the artificial second objective, as well as (2) the position of the initial starting points in the search space. As SOMOGSA is a modular framework for encapsulating local search, we integrate Nelder–Mead local search as optimizer in the respective module and compare the performance of the resulting hybrid local search to its original single-objective counterpart. We show that the SOMOGSA framework can significantly boost local search by multiobjectivization. Hence, combined with more sophisticated local search and metaheuristics, this may help solve highly multimodal optimization problems in the future.

Keywords Multiobjectivization · Multimodal optimization · Continuous optimization · Local search

✉ Pelin Aspar
asparp@uni-muenster.de

✉ Christian Grimme
christian.grimme@uni-muenster.de

Vera Steinhoff
vera.steinhoff@uni-muenster.de

Lennart Schäpermeier
lennart.schaepermeier@tu-dresden.de

Pascal Kerschke
pascal.kerschke@tu-dresden.de

Heike Trautmann
trautmann@uni-muenster.de

[1] Statistics and Optimization Group, University of Münster, Münster, Germany

[2] Big Data Analytics in Transportation, TU Dresden, Dresden, Germany

[3] LIACS, University of Leiden, Leiden, The Netherlands

[4] Data Management & Biometrics Group, University of Twente, Enschede, The Netherlands

1 Introduction

The basic idea of multiobjectivization for single-objective (SO) optimization problems is simple (Knowles et al. 2001; Segura et al. 2016): instead of optimizing the desired objective alone, introduce a second objective and solve the resulting multi-objective (MO)—more precisely bi-objective—optimization problem using MO solvers, e.g., evolutionary multi-objective algorithms (EMOAs). There are multiple reasons for doing this: First and foremost, local optima are obstacles in search space, which may lead to premature convergence of single-objective (local) search methods. And even for explicitly designed global methods, multimodality can be a curse (Preuss 2015). A first promising report on the benefits of transforming SO problems into the MO domain was provided by Knowles et al. (2001). The authors empirically showed that multiobjectivization can reduce the amount of local optima in search space. Follow-up studies showed that helper-objectives can be beneficial (Jensen 2004), while others

theoretically demonstrate that an additional objective can improve the search behavior of evolutionary algorithms (Neumann and Wegener 2008). In the global optimization domain, Dunlavy and O'Leary (2005) proposed a so-called homotopy optimization approach, which integrates a second (helper) objective into a homotopy function and minimizes different linear aggregations of both objectives.

A second reason for the multiobjectivization of single-objective problems is to exploit the power of EMOAs, which are known to be capable of solving many difficult problems (Tran et al. 2013). However, the result of multiobjectivization may be ambivalent and thus have positive and negative effects on an algorithm's performance (Handl et al. 2008; Brockhoff et al. 2007). Nevertheless, a central and beneficial property of multiobjectivization remains: if we add another objective, there is often more information available that may help in guiding the search. Some authors even report on plateau networks that may level out local optima and make it easier to avoid them (Garza-Fabre et al. 2015). Using new visual representations of MO landscapes (Kerschke and Grimme 2017; Grimme et al. 2019b), we were recently able to show that MO landscapes in fact comprise structures that can be exploited for escaping local optima (Steinhoff et al. 2020). In that work, we not only described the observations in the MO landscapes, but also integrated this behavior in a local search method and demonstrated the general working principle.

Given a single-objective problem $f_1(x)$ with $x \in \mathbb{R}^n$ that is to be optimized, we additionally introduce a sphere function $f_2(x) = (x_1 - y_1^*)^2 + \cdots + (x_n - y_n^*)^2$ as second objective, with $y^* \in \mathbb{R}^n$ denoting the position of the sphere's center and thus the only optimum of f_2. Then, the resulting bi-objective problem $F(x) = (f_1(x), f_2(x))^T$ is solved by a multi-objective local search approach called *multi-objective gradient sliding algorithm* (MOGSA) (Grimme et al. 2019a), which essentially moves from a starting point in decision space towards the nearest local efficient set (in the MO sense, see 2.1 for terminology), explores that local efficient set, and moves on to the next (and dominating) locally efficient set of $F(x)$. Thereby, it exploits the special structure of MO landscapes (for details, see Sect. 2). Tracing the search path from multiple starting points, we were able to show in Steinhoff et al. (2020), that we pass better local optima for f_1 in that process than standard local search mechanisms (e.g., Nelder and Mead 1965) could possibly reach for many starting points. Depending on the structure of f_1, those trajectories oftentimes even crossed the global optimum.

This paper extends previous work (Aspar et al. 2021), in which we presented work-in-progress results for the aforementioned MO search approach based on MOGSA towards an effective SO local search mechanism. By combining multiobjectivization and MOGSA, our modularized heuristic enables the boosting of classical single-objective local search mechanisms like Nelder–Mead (NM). Specifically, we extend our experimental setup compared to Aspar et al. (2021) and focus on the single-objective MOGSA search heuristic (SOMOGSA) that integrates Nelder–Mead local search.

To assess the benefits of our approach compared with the capabilities of single-objective local search and to highlight the specific properties and challenges for this approach, we evaluate the method for the 24 BBOB problems proposed by Hansen et al. (2009) in two-dimensional search spaces. For those low-dimensional search spaces we are able to investigate the positive effect of the proposed hybridization visually as well as using new measures for quality gain. To complement the findings, we apply this approach to highly multimodal problem instances in five dimensions.

Our results show that the SO variant of MOGSA (SOMOGSA) significantly outperforms the considered Nelder–Mead local search. Our systematic study of starting points in search space, as well as the examination of the approach's sensitivity to different positions of the helper objective f_2, suggests that further, possibly more advanced, SO local search mechanisms can easily be incorporated into the SOMOGSA framework. Alternatively, our algorithm could be used within global search procedures to support global convergence.

The remainder of this work is structured as follows: Sect. 2 starts with providing necessary terminology and notation before we explain the idea of SOMOGSA. Section 3 describes the experimental setup, the conducted experiments and the evaluation strategy. Section 4 details the results, before Sect. 5 summarizes our findings and highlights perspectives for future work.

2 Single-objective optimization via multiobjectivization

Within this section, we first describe the fundamental notion of multi-objective optimization problems as well as some aspects of locality to lay a foundation for further discussion. Then, we address multiobjectivization in general, and detail our respective specific approach. Afterwards, the core components of our hybrid and modularized local search algorithm SOMOGSA (and its hybridization with Nelder–Mead local search) will be outlined.

2.1 Multi-objective optimization and locality

We earlier described the general scheme of converting a single-objective optimization problem into a multi-

objective one by adding an additional objective. Here, we formally introduce some of the necessary terminology summarized and extended by Grimme et al. (2021):

Definition 1 (Multi-objective optimization problem) A *multi-objective optimization problem* (*MOP*) is commonly denoted as a vector-valued function

$$\mathbf{f} : \mathcal{X} \rightarrow \mathbb{R}^m, \quad \mathbf{x} \mapsto \big(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})\big)^\top,$$

with m real-valued single-objective functions $f_i : \mathcal{X} \rightarrow \mathbb{R}$, $i = 1, \ldots, m$, which are to be optimized simultaneously.

As we focus on *continuous* multi-objective optimization, the MOP's search space $\mathcal{X}$ is real-valued, i.e., $\mathcal{X} \subseteq \mathbb{R}^n$. It is clear that the domination of a solution over another solution is not as simple as in SO optimization. In contrast to a totally ordered set $(\mathbb{R}, \leq)$ in SO optimization, with its natural total order $\leq$ on $\mathbb{R}$, the set of solution candidates of an MOP follows the *weak Pareto order* $\preceq$ on $\mathbb{R}^m$, which is defined as follows:

Definition 2 (Pareto order or Pareto dominance) Let $\mathbf{a} = (a_1, \ldots, a_m) \in \mathbb{R}^m$ and $\mathbf{b} = (b_1, \ldots, b_m) \in \mathbb{R}^m$. We say $\mathbf{a}$ weakly dominates $\mathbf{b}$ (written as $\mathbf{a} \preceq \mathbf{b}$), if and only if $a_i \leq b_i$ for all $i = 1, \ldots, m$. Then the *Pareto order* $\prec$ on $\mathbb{R}^m$ is defined as follows: $\mathbf{a}$ dominates $\mathbf{b}$ ($\mathbf{a} \prec \mathbf{b}$), if and only if $\mathbf{a} \preceq \mathbf{b}$ and $\mathbf{a} \neq \mathbf{b}$. The weak Pareto order can also be extended to sets of points: for $A, B \subseteq \mathbb{R}^m$, A weakly dominates B, if and only if $\forall \mathbf{b} \in B \, \exists \mathbf{a} \in A$ such that $\mathbf{a} \preceq \mathbf{b}$. The order $\prec$ can be generalized similarly: A dominates B, if and only if $\forall \mathbf{b} \in B \, \exists \mathbf{a} \in A$ such that $\mathbf{a} \prec \mathbf{b}$.

Thus, the solution set of an MOP, the Pareto set, consists of optimal trade-off solutions, i.e., nondominated solutions in decision space. The image of the Pareto set is called Pareto front:

Definition 3 (Pareto set and Pareto front) A point $\mathbf{x} \in \mathcal{X}$ is denoted as *globally efficient point* of $\mathcal{X}$ (or of $\mathbf{f}$) if there is no point $\tilde{\mathbf{x}} \in \mathcal{X}$ s.t. $\mathbf{f}(\tilde{\mathbf{x}}) \prec \mathbf{f}(\mathbf{x})$. The set $\mathcal{X}_E$ of all globally efficient points of $\mathcal{X}$ is called *Pareto set* (or globally efficient set) of $\mathbf{f}$. The image of $\mathcal{X}_E$ under $\mathbf{f}$ is called the *Pareto front* of $\mathbf{f}$ and denoted by $\mathbf{f}(\mathcal{X}_E)$.

Traditional multi-objective solvers specifically aim for approximating (or even exactly determining) the Pareto sets and Pareto fronts. However, as shown in Grimme et al. (2021), *local* structures in decision space and objective space thereby not necessarily hinder algorithm success but can rather be efficiently exploited by sophisticated approaches. Thus, with the methodological concept of multiobjectivization in mind, we need to define locality and locally efficient sets following the aforementioned work in terminology and notation (see Fig. 1 for a schematic visualization).
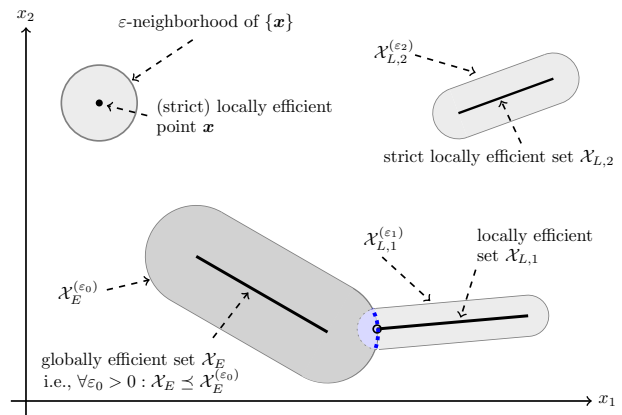

Fig. 1 Schematic illustration of the introduced concepts for a decision space $\mathcal{X} = \mathbb{R}^2$, adopted from Grimme et al. (2021)

Definition 4 (Locally efficient point) A point $\mathbf{x} \in \mathcal{X}$ is called a *locally efficient point* of $\mathcal{X}$ (or of $\mathbf{f}$) if there is an open set $U \subseteq \mathcal{X}$ with $\mathbf{x} \in U$ and there is no point $\tilde{\mathbf{x}} \in U$ such that $\mathbf{f}(\tilde{\mathbf{x}}) \prec \mathbf{f}(\mathbf{x})$. The set of all locally efficient points of $\mathcal{X}$ is denoted by $\mathcal{X}_{\text{LE}}$.

Yet, as the consideration of locally efficient sets goes beyond the definition of locally efficient points, this additionally requires a notion of connectedness.

Definition 5 (Connectedness and connected component) The subset $A \subseteq \mathcal{X}$ is called *connected*, if and only if there do not exist two open subsets U_1 and U_2 of $\mathcal{X}$ such that $A \subseteq (U_1 \cup U_2)$, $(U_1 \cap A) \neq \emptyset$, $(U_2 \cap A) \neq \emptyset$, and $(U_1 \cap U_2 \cap A) = \emptyset$; or equivalently, there do not exist two non-empty subsets A_1 and A_2 of A which are open in the relative topology of A such that $(A_1 \cup A_2) = A$ and $(A_1 \cap A_2) = \emptyset$. Let B be a non-empty subset of $\mathcal{X}$. A subset C of B is a *connected component* of B, if and only if C is non-empty, connected, and there exists no strict superset of C that is connected.

Now it is possible to define the locally efficient set as a structure in search space and with its corresponding image in objective space.

Definition 6 (Locally efficient set and locally efficient front) A subset $A \subseteq \mathcal{X}$ is a *locally efficient set* of $\mathbf{f}$, if A is a connected component of $\mathcal{X}_{\text{LE}}$. The image $f(A)$ under $\mathbf{f}$ is called a *locally efficient front* of $\mathbf{f}$.

Clearly, solutions in locally efficient sets, which are not dominated by any other solution, are contained in the Pareto set. This implies $\mathcal{X}_E \subseteq \mathcal{X}_{LE}$. Following the defined distinction of globally and locally efficient points, we can also speak of a globally efficient set of solutions and a globally efficient front of solutions instead of a Pareto set

or a Pareto front, respectively. Examples for both globally and locally efficient sets are shown in Fig. 1.

To even further sharpen the notion of localness, we adopt the definition of the ε-neighborhood of a set in analogy to the ε-ball, which surrounds a single solution (refer to Definition 4).

Definition 7 (ε-neighborhood of a set) Let $A \subseteq \mathcal{X}$ and $\varepsilon > 0$. The set $A^{(\varepsilon)} := \{x \in \mathcal{X} \mid \exists a \in A \text{ with } \|x - a\| < \varepsilon\}$ is the ε-*neighborhood* of A, where $\| \cdot \|$ is the Euclidean norm.

Then, a strict locally efficient set is completely surrounded by an ε-neighborhood for some ε (see also Fig. 1, set $\mathcal{X}_{L,2}$).

Definition 8 (Strict, locally efficient set) Let $C \subseteq \mathcal{X}_{\mathrm{LE}}$ be a locally efficient set. Then C is a *strict, locally efficient set*, if and only if $\exists \varepsilon > 0$ such that $C \preceq C^{(\varepsilon)}$.

For a schematic example on how a locally efficient set fails being a strict locally efficient set see Fig. 1, again. Therein, the set $\mathcal{X}_{L,1}$ is a half-open segment of a straight line, whose leftmost end point is excluded (indicated by an open circle). Moreover, it is an accumulation point of $\mathcal{X}_E^{(\varepsilon_0)}$, i.e., for a fixed $\varepsilon_0 > 0$ there exists $\varepsilon_1 > 0$ such that *none* of the points of $\mathcal{X}_E^{(\varepsilon_0)} \cap \mathcal{X}_{L,1}^{(\varepsilon_1)}$ (the blue intersecting area) is dominated by $\mathcal{X}_{L,1}$.

The non-strict locally efficient sets are of special interest here. Their ε-neighborhood is—as demonstrated visually—obviously superposed by the ε-neighborhood of a dominating efficient set. We now consider the ε-neighborhood of a locally efficient set as subset of the basin of attraction of a local set. Therefore, we define a *basin of attraction* as the set of points for which a multi-objective gradient-based descent (w.l.o.g. for minimization problems) strategy leads to the same locally efficient set. This multi-objective gradient descent can be considered as movement from one point in decision space to a (neighboring) dominating point in decision space following the common direction of the normalized gradients of all objectives.

The superposition which makes a locally efficient set non-strict thus opens up the way along the effcient set into the superposing (i.e., dominating) basin of attraction. This can be algorithmically exploited and is the motivation for the method presented and investigated in this paper.

2.2 Concept of multiobjectivization

For this work, we consider the optimization of box-constrained single-objective continuous optimization problems as our main goal. These problems are formally given by

$\min_{x \in [l,u]} f(x)$ with $f : \mathbb{R}^n \to \mathbb{R}$, and $l, u \in \mathbb{R}^n$ being the problem's (lower and upper) box constraints.

However, instead of optimizing the single-objective problem directly – w.l.o.g. we denote this objective f_1—we transform the single-objective problem into a bi-objective problem $F = (f_1, f_2)$ by introducing a second objective f_2, which w.l.o.g. shall be minimized as well. This simple concept is known as *multiobjectivization* (Knowles et al. 2001; Jensen 2004; Neumann and Wegener 2008; Steinhoff et al. 2020) and its motivation is that the resulting bi-objective problem landscape possesses structural properties which could potentially be exploited by a sophisticated multi-objective optimizer.

Among these characteristics are the previously defined *locally efficient sets* (see Def. 6, i.e., the multi-objective pendants of local optima). Per definition, each of these sets is a connected set of locally efficient points.

Interestingly, in multi-objective optimization, the existence of locally efficient sets can be very beneficial for local solvers. In contrast to their SO counterparts, these optima oftentimes are no traps for local solvers, but instead guide the way to more promising regions of the search space (Grimme et al. 2019b; Kerschke and Grimme 2021; Schäpermeier et al. 2022), sometimes even towards the globally efficient set.

This is rooted in superpositions of basins of attraction as described in Sect. 2.1 and exemplarily visualized in Fig. 2. This figure presents the sequential construction of a multi-objective problem from two single-objective functions. The left-hand image shows a heatmap-based landscape plot of the highly multimodal Rastrigin problem (Hoffmeister and Bäck 1991), which we consider as f_1. The middle plot shows a respective plot of the sphere function, which we consider as f_2. The right-hand plot shows a heatmap-based gradient field plot (Kerschke and Grimme 2017) of the multi-objective problem $F = (f_1, f_2)^T$, where the color denotes proximity to the next locally efficient set from red (far) to blue (close). In a nutshell, the gradient heatmap-based plot of a multi-objective problem depicts by color for each point in a discretized decision space the distance (number of necessary steps via neighboring points) towards a local efficient point, if we follow the normalized multi-objective gradient direction. More detailed information on this visualization technique, related approaches, and visualization tools can be found in Schäpermeier et al. (2020, 2021). Obviously, the multi-objective landscape of the combined problems exposes many locally efficient sets (visible as green-yellow lines) surrounded by red colored basins of attraction. However, in contrast to the
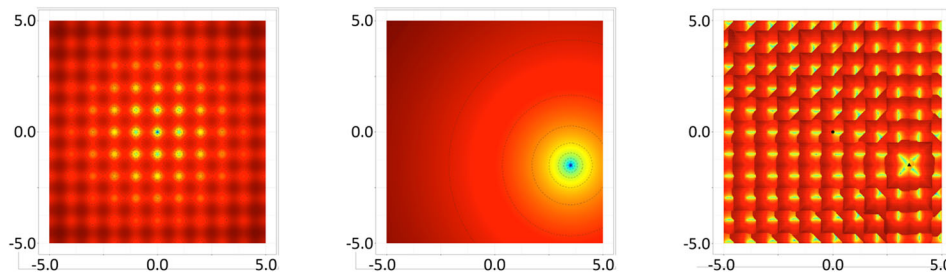
Fig. 2 Exemplary visualization of the multiobjectivization approach using the Rastrigin function (left) and a sphere function (middle) to create a multi-objective landscape (right). The latter exposes some interesting structural properties, which could be exploited by a local search mechanism

classical view on the Rastrigin problem with its many local traps (and local basins of attraction) the multi-objective basins superpose each other: basins of one set directly cut locally efficient sets of superposed basins. This innovative structural view allows for new ideas how to roam the search space in a directed way.

What we have observed formally means: if we are not trapped in a strict locally efficient set (see Def. 8), then dominating basins of attraction overlap dominated basins. As shown schematically in Fig. 1 and exemplarily in Fig. 2, it is then possible to move from the dominated basin towards the dominating one along the efficient set. Thereby, we definitely reach a "better" multi-objective locally efficient set—and possibly also a better area of the single-objective search space.

By making use of the Fritz John necessary condition (John 1948), potentially local efficient points can be identified easily during multi-objective downhill movement in the search space: let $x \in \mathbb{R}^n$ be a locally efficient point and all objective functions of F, i.e., f_1 and f_2, continuously differentiable in $\mathbb{R}^n$. Then, there is a weight vector $v \in [0,1]^m$ with $\sum_{i=1}^m v_i = 1$, such that

$$\sum_{i=1}^m v_i \nabla f_i(x) = 0.$$

In the bi-objective case, when we are looking at a locally efficient point, the (single-objective) gradients of f_1 and f_2 cancel each other out given a suitable weight vector $v \in \mathbb{R}^2$. This property provides useful fundamentals for understanding and optimizing multi-objective optimization problems and (as will be shown within this work) for single-objective problems as well. For instance, it can be used for visualizing MO landscapes (Kerschke and Grimme 2017; Schäpermeier et al. 2020), and is essential for the lately proposed *multi-objective gradient sliding algorithm (MOGSA)* (Grimme et al. 2019a, b). As recently shown by Steinhoff et al. (2020), multiobjec-

tivization enables the aforementioned multi-objective algorithm to even optimize single-objective problems. In the following subsection, we will describe this single-objective variant of MOGSA—dubbed SOMOGSA—in more detail.

2.3 The modularized search heuristic SOMOGSA

Contrary to previous research in which global methods were applied to multiobjectivized problems—for an extensive review, we refer to Segura et al. (2013, 2016)—we will use a deterministic MO *local search* algorithm. It efficiently exploits properties of MO landscapes, which we previously identified using a recently developed visualization method based on *gradient field heatmaps* (Grimme et al. (2019b), see Fig. 2 for an example). In our setting the multi-objective problem (MOP) degenerates to a bi-objective problem $F(x) = (f_1(x), f_2(x))^T \in \mathbb{R}^2$. To ensure simplicity of our approach and maximal comprehensibility, we used a simple sphere function $f_2(x) = \sum_{i=1}^n (x_i - y_i^*)^2$ as second objective with $x, y^* \in \mathbb{R}^n$. It comes with two benefits: (1) the resulting bi-objective problem is simple enough to avoid unwanted distractions, but still capable of guiding an MO algorithm, and (2) it allows for direct, analytical determination of the corresponding gradient (which is utilized by SOMOGSA). As such, this choice seems to be cost-effective and logical for this study. Note that in principle a different and more complex choice of f_2 is possible. A clever choice of f_2 could help in creating specific MO descent structures to guide SOMOGSA directly towards a desired path. It is still unclear, however, what effort has to be put into finding such a structure of f_2 and how the interaction of f_1 and f_2 can be predicted for generating a beneficial MO structure, specifically if f_1 is a gray or even a black box problem. In this study we rely on Occam's principle of choosing the simplest model for evaluation.

Algorithm 1 SOMOGSA

Require: a) start point $x_s \in \mathbb{R}^n$, **b)** function f_1 to be optimized **c)** termination
    angle $t_\angle \in [0, 180]$ for switching to local search w.r.t. f_1, **d)** step size $\sigma_{MO} \in \mathbb{R}$
    for MO gradient descent, **e)** step size $\sigma_{SO} \in \mathbb{R}$ for SO gradient descent w.r.t. f_2,
    f) $y^* \in \mathbb{R}^n$ optimum of f_2
1: $f_2(x) = (x_1 - y_1^*)^2 + \cdots + (x_n - y_n^*)^2$ ▷ *use sphere function for multiobjectivization*
2: $x = x_s$
3: **while** optimum of f_2 not yet reached **do**
4:     $x' = x$
5:     **while** $|\nabla f_1(x)| > 0$ and $\angle(\nabla f_1(x), \nabla f_2(x)) \leq t_\angle$ **do**
6:         $x = x - \sigma_{MO} \cdot \left(\frac{\nabla f_1(x)}{|\nabla f_1(x)|} + \frac{\nabla f_2(x)}{|\nabla f_2(x)|} \right)$ ▷ *MO gradient descent*
7:     **end while**
8:     $x^{t-1} = x = LocalSearch(x, f_1)$ ▷ *local search w.r.t.* f_1
9:     **if** $f_1(x) < f_1(x')$ **then**   ▷ *check for decreasing f_1 value*
10:        **if** $|\nabla f_2(x)| > 0$ **then**
11:           **while** $\angle(\nabla f_1(x), \nabla f_2(x)) \geq 90°$ and $\angle(\nabla f_2(x^{t-1}), \nabla f_2(x)) \leq 90°$ **do**
12:             $x^{t-1} = x$
13:             $x = x - \sigma_{SO} \cdot \frac{\nabla f_2(x)}{|\nabla f_2(x)|}$ ▷ *gradient descent towards f_2*
14:           **end while**
15:        **else**
16:           $x' = x$
17:        **end if**
18:     **else**
19:        leave outer while-loop   ▷ *f_1 value deteriorated: break*
20:     **end if**
21: **end while**
22: **return** x', $f_1(x')$

The general idea of our proposed hybrid and modularized search heuristic SOMOGSA is given schematically in Fig. 3. While SOMOGSA enables the exploitation of the multi-objective landscape structures, it can encapsulate an arbitrary local search for refinement of f_1 results. In Sect. 3, we will exemplarily show this for the Nelder–Mead local search algorithm. The SOMOGSA framework is described in more detail in Algorithm 1. Apart from adding a sphere

as second objective (line 1), SOMOGSA essentially (repeatedly) performs the following steps:

1. Perform an MO gradient descent (i.e., 'slide down') into the vicinity of the attracting locally efficient set (lines 5 and 6).
2. Once an (almost) locally efficient point has been reached, SOMOGSA will perform a single-objective local search towards the corresponding (single-objective) local optimum of f_1 (line 7). Here, the user can choose his/her favorite local search strategy.
3. As the local optimum defines one end of the locally efficient set, SOMOGSA now can traverse that set towards the neighboring (better) attraction basin (lines 10-12) by performing a single-objective gradient descent in the direction of the second objective (f_2).

Note that the criterion for stopping multi-objective downhill movement is based on the Fritz-John condition, which is only a necessary criterion and may lead to premature termination of step 1. This case is not explicitly handled by our approach. Instead, we heuristically assume that the following single-objective local search w.r.t. f_1 will lead the search near a locally efficient solution again, where step 3 will continue as planned.

The above mentioned steps are repeated until either (a) SOMOGSA has reached the sphere's optimum (line 3), or
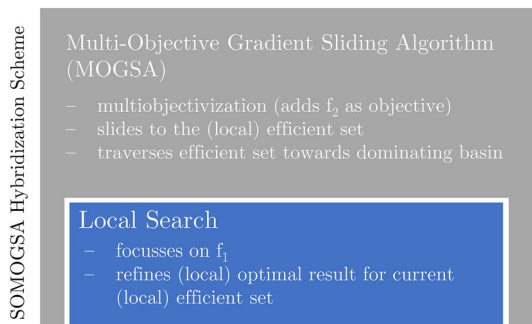

Fig. 3 Hybridization concept of SOMOGSA: while the *multi-objective* local search algorithm MOGSA realizes a descent towards dominating MO basins by approaching and traversing locally efficient sets, the encapsulated *single-objective* local search focuses on the refinement of the f_1 values in the visited MO basins. Taken from Aspar et al. (2021)

(b) the objective value of f_1 got worse between two subsequently found local optima of f_1 (line 8). In the latter case, we interrupt the optimization, as SOMOGSA wouldn't improve any longer w.r.t. f_1 but instead waste the remaining budget to eventually reach the optimum of f_2. Note that the gradient of the original objective function f_1 usually needs to be computed numerically, e.g., by two-sided finite differences, while the gradient of the helper objective f_2 may be known analytically.

As described above, SOMOGSA basically provides a modularized framework, which allows to make use of the strengths of MOGSA. However, the performance of SOMOGSA obviously depends on its configuration (e.g., which problem is used as objective f_2, where is the optimum of f_2 located, which local search strategy is used, which starting point is chosen etc.). Therefore, in the following, we will investigate the sensitivity of our method w.r.t. different parameters and module choices. In addition, we experimentally demonstrate that SOMOGSA can significantly boost the potential of simple single-objective local search algorithms.

3 Experiments

3.1 Setup

In this work, we expand our test environment used in Aspar et al. (2021) and consider the 24 noiseless single-objective BBOB problems from the COCO platform (Hansen et al. 2009) in the two-dimensional decision space. For SOMOGSA, these functions are referred to as objective f_1 in the multiobjectivized problem. As second objective, we consider $f_2(x) = \sum_{i=1}^{2}(x_i - y_i^*)^2$ with $x, y^* \in \mathbb{R}^2$, as explained above. Besides investigating the principal benefits of applying SOMOGSA together with a classical SO local search, we specifically focus on the sensitivity of the approach w.r.t the starting position of the search, as well as the location of the optimum of f_2. We thus discretize the decision space using a regular grid X for generating $N = d \times d$ starting points. We choose $d = 50$, as (empirically) larger values of d do not provide additional insights and lower values do not reveal all desired structures. The methods to be assessed are executed for all points in the grid centers of X. Ten positions y^* of the f_2 optimum were generated[1] by using Latin Hypercube Sampling (McKay et al. 1979) and subsequently mapping those points to their nearest neighbors in X[2]. The (default) parameters mentioned in Algorithm 1 were set to $t_\angle = 170$ as angle for denoting sufficient proximity to a potentially local efficient

set, $\sigma_{MO} = 0.05$ as step size for multi-objective descent, and $\sigma_{SO} = 0.1$ as step size for fast traversal by following the approximated gradient of f_2 for reaching the next basin of attraction. In Aspar et al. (2021) Gradient Search (GS) and Nelder–Mead (NM) (Nelder and Mead 1965) were used as local search mechanisms inside the SOMOGSA framework, resulting in two variants of SOMOGSA: SOMOGSA+GS and SOMOGSA+NM. As a baseline, we also evaluated GS and NM as stand-alone methods on each considered SO problem. In order to prevent possible infinite cycling of local search (specifically NM can run into cyclic behavior), the number of local search steps was restricted to a maximum of 400. Since the method SOMOGSA+NM consistently outperformed SOMOGSA+GS, in this work, we solely focus on SOMOGSA+NM and Nelder–Mead as the corresponding standalone method. Furthermore, our previous research revealed that both algorithms barely require more than 2000 function evaluations for $n = 2$ until one of the internal termination criteria has been reached. Therefore, we set the optimizer budget to $n \times 1000 = 2000$ function evaluations. Thereby, an optimizer run is considered successful, if it comes sufficiently close to the optimum, i.e., if it stagnates in an ε-neighborhood of the optimum of f_1 (using a precision of $p = \varepsilon = 0.01$). This is a common experimental approach within the BBOB benchmark setting (Hansen et al. 2009). Moreover, we report the amount of function evaluations the optimizers actually need. For Nelder–Mead, we assume an average number of 1.5 function evaluations per iteration due to varying simplex generation procedures depending on the local landscape structure[3].

In order to ensure to stay within the boundaries of our box-constrained problems, we integrate the following approach: As soon as Nelder–Mead spans the next simplex beyond any boundary, this simplex is clipped at the crossed boundary, respectively, and the best reached solution by the time is selected. This kind of constraint handling is advantageous, but may lead to early stoppings in some cases, especially for the stand-alone method of Nelder–Mead. For starting points near a boundary, the algorithm often passes the closest boundary rapidly and thus stops.

As a proof-of-concept study, we investigate how our findings generalize to higher decision space dimensions and expand our experiments to the two highly multimodal BBOB functions 21 and 22 (called Gallagher's 101 and 21 Peak functions) in the 5-dimensional decision space. Likewise, these problems are evaluated for ten different

---

[1] LHS implementation of the pyDOE package: https://pythonhosted.org/pyDOE/.

[2] The ten considered sphere positions are: $(3.5, -1.5)$, $(-1.5, 0.5)$, $(-0.5, 2.5)$, $(2.5, -2.5)$, $(-4.5, -0.5)$, $(-2.5, -3.5)$, $(1.5, 3.5)$, $(4.5, -4.5)$, $(-3.5, 4.5)$, $(0.5, 1.5)$

[3] The NM Python procedure only reports number of iterations used in total.

sphere positions and 2500 starting points. Both the center of the sphere function and the starting points are generated via the Latin Hypercube Sampling method (McKay et al. 1979). The precision value remains at $p = 0.01$. We have assigned the optimizers a fixed budget of $n \times 1000$ function evaluations as well (with $n = 5$).

3.2 Evaluation concept

We investigate the sensitivity of SOMOGSA regarding (a) the parametrization of the second objective, (b) the position of the starting point, and (c) the local search strategy. We define specific metrics for capturing the performances of each variant in all considered settings (see Fig. 4).

As first metric, we define the *gain* (a value we want to maximize) as

$$g_{LS}(x_s) = \frac{|f_1(x_b) - f_1(x_s)|}{|f_1(x^*) - f_1(x_s)|} \in [0, 1]$$

where x_s is a starting point, x_b the best local search result and x^* the known global optimum w.r.t. f_1. This measure quantifies the achieved performance gain by the applied local search. Therefore, we compute the ratio of the gap between $f(x_s)$ and $f(x^*)$, which is closed by applying the considered search heuristic. As a (global) performance measure, we also compare the *gap* (a value we want to minimize) that cannot be closed by the local search, relating to *all* starting points. This is quantified by

$$G_{LS}(x_s) = \frac{|f_1(x^*) - f_1(x_b)|}{|f_1(x^*) - \max_{x \in X} f_1(x)|} \in [0, 1]$$

This measure expresses the gap between $f(x_b)$ and $f(x^*)$ w.r.t the global maximum gap of all starting points function values from X to the optimum value. For a schematic representation of both measures refer to Fig. 4.
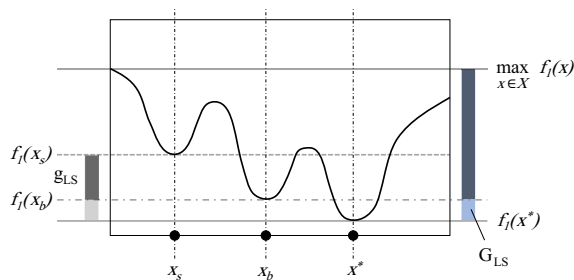
Fig. 4 Illustration of g_{LS} and G_{LS} in a one-dimensional setting. Taken from Aspar et al. (2021)

Both measures are evaluated for all combinations of N starting points and considered benchmark problem. The overall aggregated performance of each method can be statistically analyzed and visualized in the decision space, see, e.g., Fig. 5.

Furthermore, we assess the algorithm's capability to reach the optimal solution (of f_1) from any of the starting points of the (discretized) search space X by determining the relative frequency of success (to be maximized by the algorithm), denoted as (success) *ratio*

$$r_N(LS) = \frac{H_N(LS)}{N} \in [0, 1]$$

where $H_N(LS)$ is the absolute frequency of the N algorithm runs that successfully converged to the optimum of f_1 w.r.t. the precision $p = 0.01$.

4 Results

In the following, the results of our experiments are analyzed on the basis of the aforementioned performance indicators. Our investigations have shown that the results of the G_{LS} measure qualitatively confirm those of g_{LS}. Thus, in this work, we focus on presenting only results for g_{LS} and the (success) ratio r_N.

Figure 5 shows heatmaps of the g_{LS} measures for the considered search space of the Gallagher's Gaussian 101-me Peaks function (BBOB function 21), Gallagher's 21-hi Peak function (BBOB function 22), and the Lunacek bi-Rastrigin function (BBOB function 24). The idea behind this visualization is to assign the $g_{LS}(x_s)$ value of the SOMOGSA variant or of the stand-alone LS run, respectively, to each starting point x_s (i.e., pixel of the corresponding heatmap) of the two-dimensional search space. This technique leads to a heatmap, in which the blue color denotes zero or little gain while the red color denotes maximum gain. Note that the top left heatmap for each function merely shows the single-objective problem landscape itself, while the second left heatmap depicts the g_{LS} values for any starting point when applying the stand-alone version of NM. The remaining heatmaps show the results for each of the considered ten positions of the optimum of f_2.

Considering the heatmaps, which illustrate the BBOB functions F21 and F22 (i.e., the first four rows), we observe that the heatmaps of the original (NM) local search show that local optima are often traps for these approaches. In our hybridized method SOMOGSA+NM, this turns out differently. Trapping areas (denoted as blue or green-yellowish structures for NM) vanish from the heatmaps and become red. Such changes indicate that for those starting
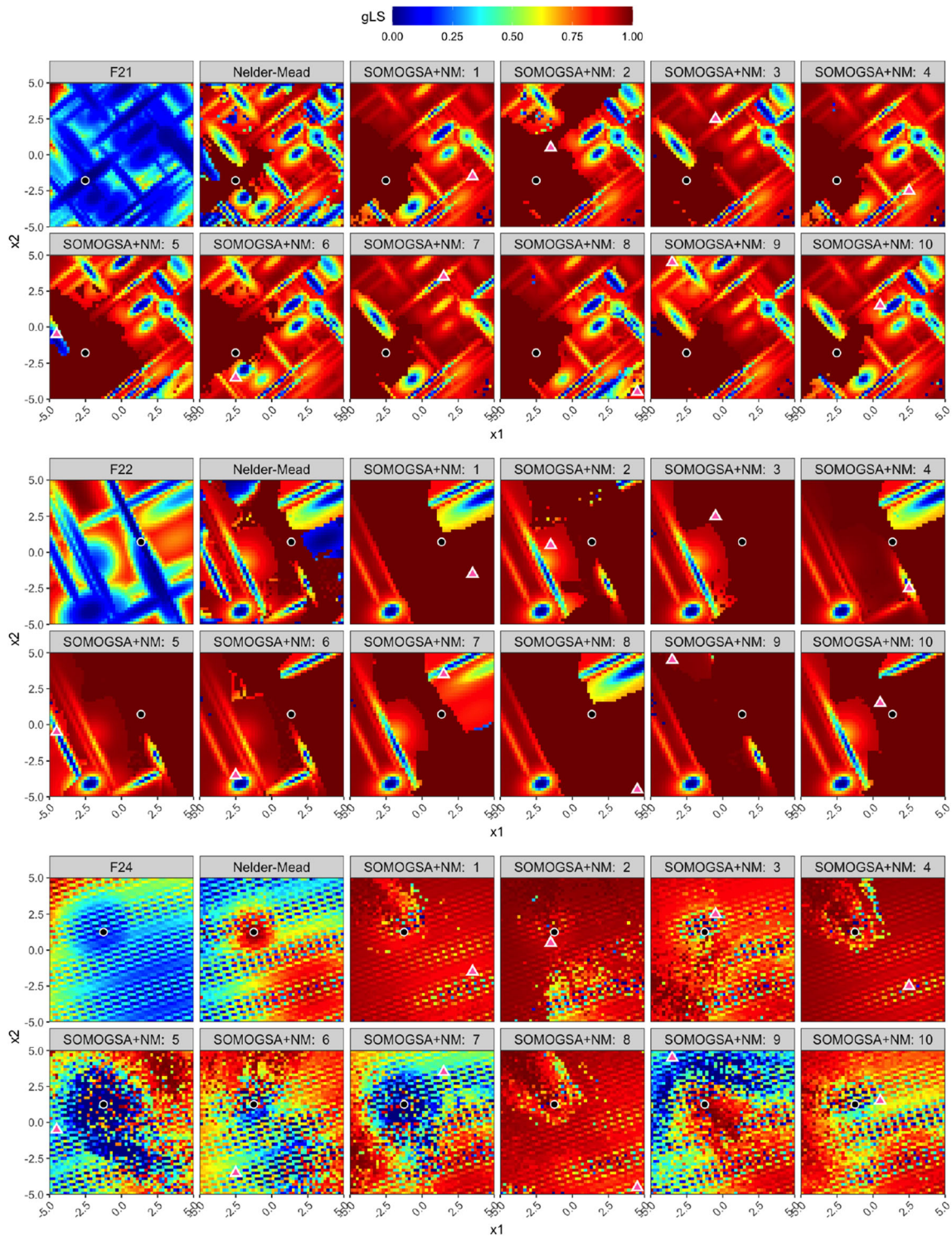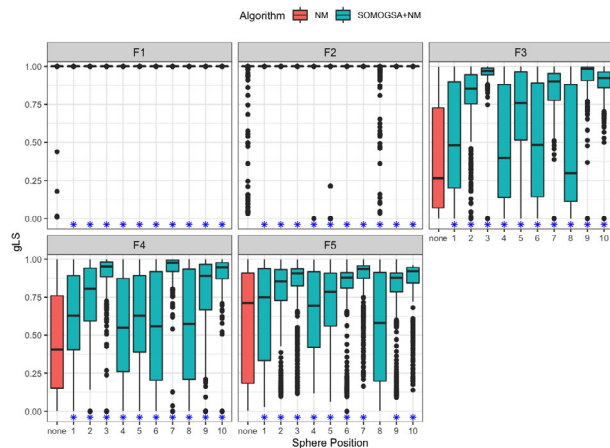
Fig. 5 Heatmaps of the g_{LS} measures for the BBOB functions 21 (top), 22 (middle) and 24 (bottom). For each of the BBOB functions, the respective top left heatmap shows the landscape of the function itself (the black dot indicates the function's optimum), followed by a heatmap with the g_{LS} performance of the original version of NM on f_1. The following ten charts display the g_{LS} values of SOMOGSA+NM applied to all ten bi-objective problems with varying sphere position. The pink triangles indicate the position of the respective optimum of f_2

points, the SOMOGSA framework realizes a significant gain and is clearly superior. As we observe the entire decision space, here, we can state that multiobjectivization and specifically SOMOGSA has the effect of removing (or opening up) traps for the single-objective optimizer. Although we observe different structures for varying locations of f_2, the general behaviour coincides for these multiobjectivized problems.

The heatmaps of the Lunacek bi-Rastrigin function (BBOB function 24), see Fig. 5, reveal a slightly different situation. Differences in the performance w.r.t. the sphere position (i.e., the position of the additional objective) become apparent. Placing the global optimum in one of the locations 1, 2, 3, 4, 8, or 10 helps the optimizer to avoid local points or even areas as traps. In contrast, placing the sphere optimum in location 5, 7, or 9 results in an opposite effect. Despite a shifting of the values, we can assume a related performance to the original method for position 6. Thus, a correlation between the sphere optimum and the

results is directly visible for this specific problem. This leads to the assumption that the choice of the position of the spherical optimum is irrelevant for some function classes. Still, for other problems, one should choose the position of the spherical optimum appropriately.
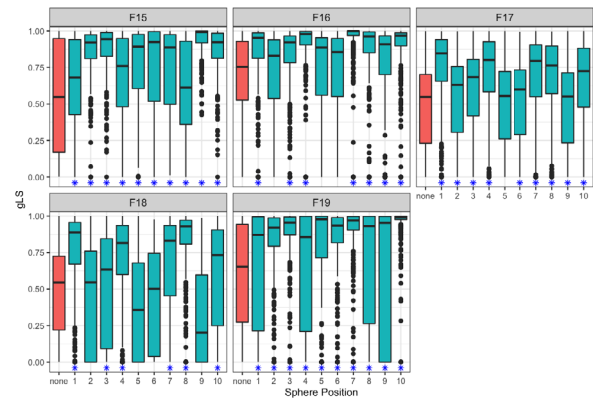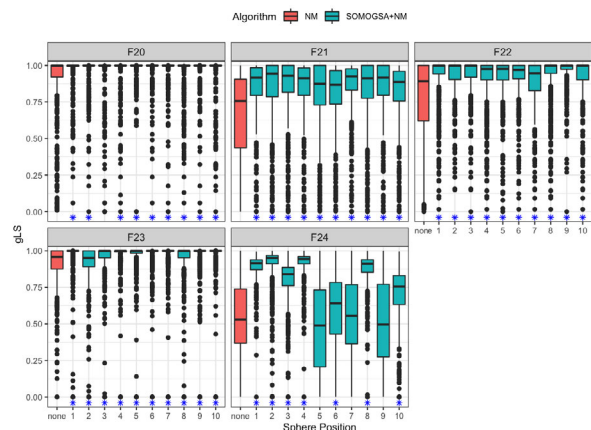

(c) Functions with high conditioning and unimodal: F10 − F14.

(a) Separable functions: F1 − F5.

(d) Multi-modal functions with adequate global structure: F15 − F19.
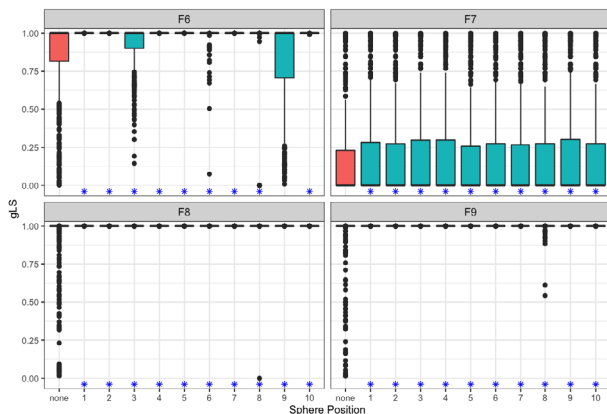
(b) Functions with low or moderate conditioning: F6 − F9.

(e) Multi-modal functions with weak global structure: F20 − F24.

Fig. 6 Boxplots for the g_{LS} measure for each function divided into the BBOB groups. The numbers 1 to 10 indicate the ten different sphere positions. *None* indicates the stand-alone method Nelder–Mead

Fig. 6 continued

This is also underlined by Fig. 6, which displays the g_{LS} values as boxplots. Therein, we compare the original NM local search (indicated by red boxes) to the hybridized SOMOGSA+NM method (indicated by the blue boxes) for each of the 24 BBOB functions. The results are splitted according to the five function groups provided by Hansen et al. (2009). Examining the individual groups in more detail, functions 3 and 4 stand out within the group of *separable functions* (F1–F5, see Fig. 6a). An adequately chosen sphere optimum can significantly boost the performance concerning these problems. Moreover, function 7 stands out (see Fig. 6b). Contrary to all other problems, here, even our hybridized method is hindered in making progress, attributed to the specific problem structure. Given numerous plateaus, this function consists of many areas in which the gradient is zero. Since SOMOGSA+NM is a gradient-based method, premature convergence occurs. However, an improvement can be seen for the group of functions with *high conditioning and unimodal* shape (F10–F14, see Fig. 6a). It appears that these functions have structures that are not significantly affected by the position of the sphere center. In addition, according to the sphere optimum, the performance fluctuates for the *multi-modal functions with adequate global structure* (F15–F19, see

Fig. 6b) as well and is particularly visible for problems 16, 17, and 18. Again, we see an improvement in the performance by adopting multiobjectivization for the BBOB functions 21 and 22, and the varying values for the BBOB function 24, see Fig. 6. In order to further consolidate our observations, we also tested this utilizing a (pairwise) paired Wilcoxon rank-sum test between the stand-alone NM and each bi-objective problem variant to which we apply SOMOGSA+NM (significance level $\alpha = 5\%$). We indicated the significantly better-performing variant by a blue *. In general, the stand-alone method does not outperform our hybridized approach for any bi-objective problem regarding the ten global sphere optima.

Figure 7 also confirms the aforementioned results. Note that this figure displays the results of our 2-dimensional test environment and the 5-dimensional study (third row second column). The red boxplots represent the ratio values achieved by the hybridized SOMOGSA+NM method, resulting in 10 ratio values per function (one for each sphere position). The blue line indicates the success ratio of the stand-alone NM method applied to the single-objective problem f_1. For nearly all considered positions of f_2, the multiobjectivization approach reaches the global optimum more often than the pure local search. Furthermore,
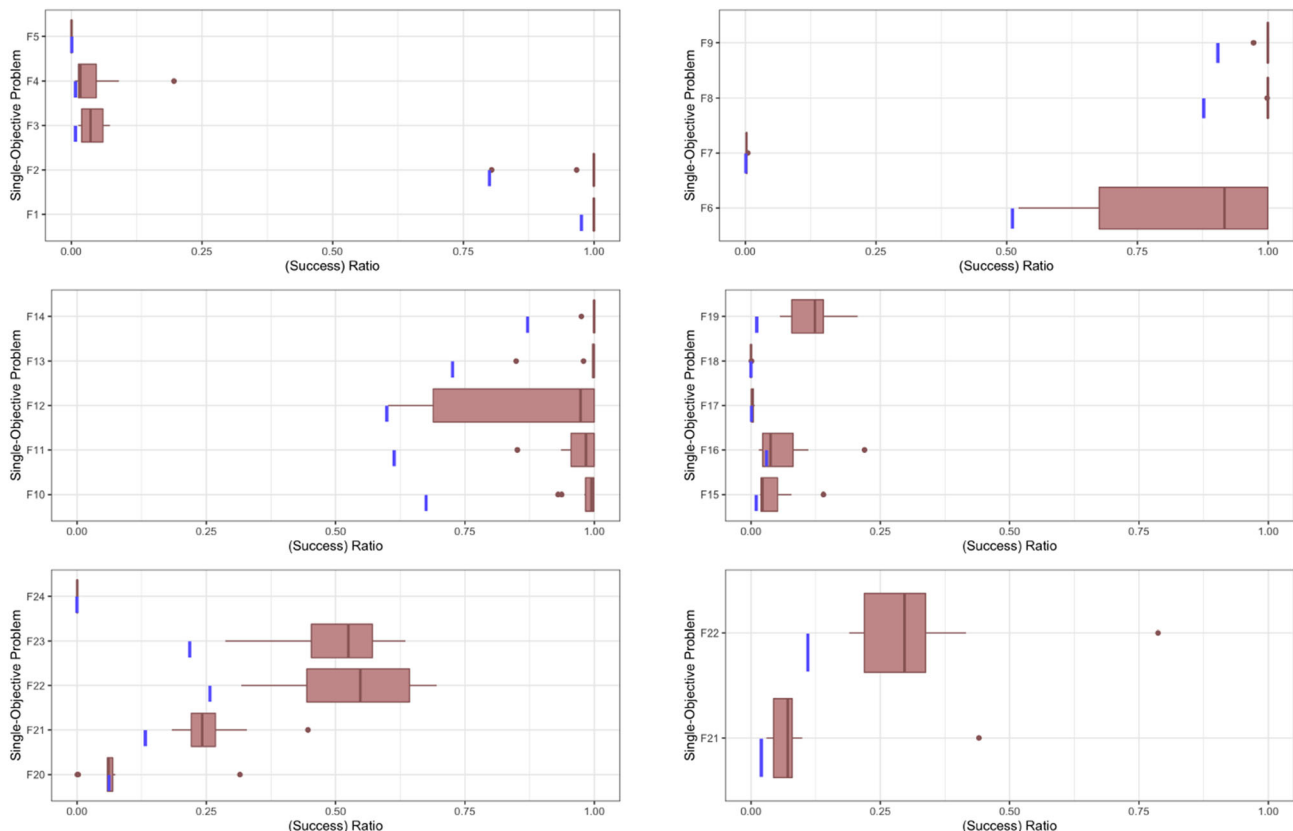

Fig. 7 Boxplots of the (success) ratio displayed per group and function. The blue line indicates the (success) ratio of the stand-alone method Nelder–Mead. Bottom right: This image shows the ratio values for the BBOB functions 21 and 22 in the 5-dimensional decision space
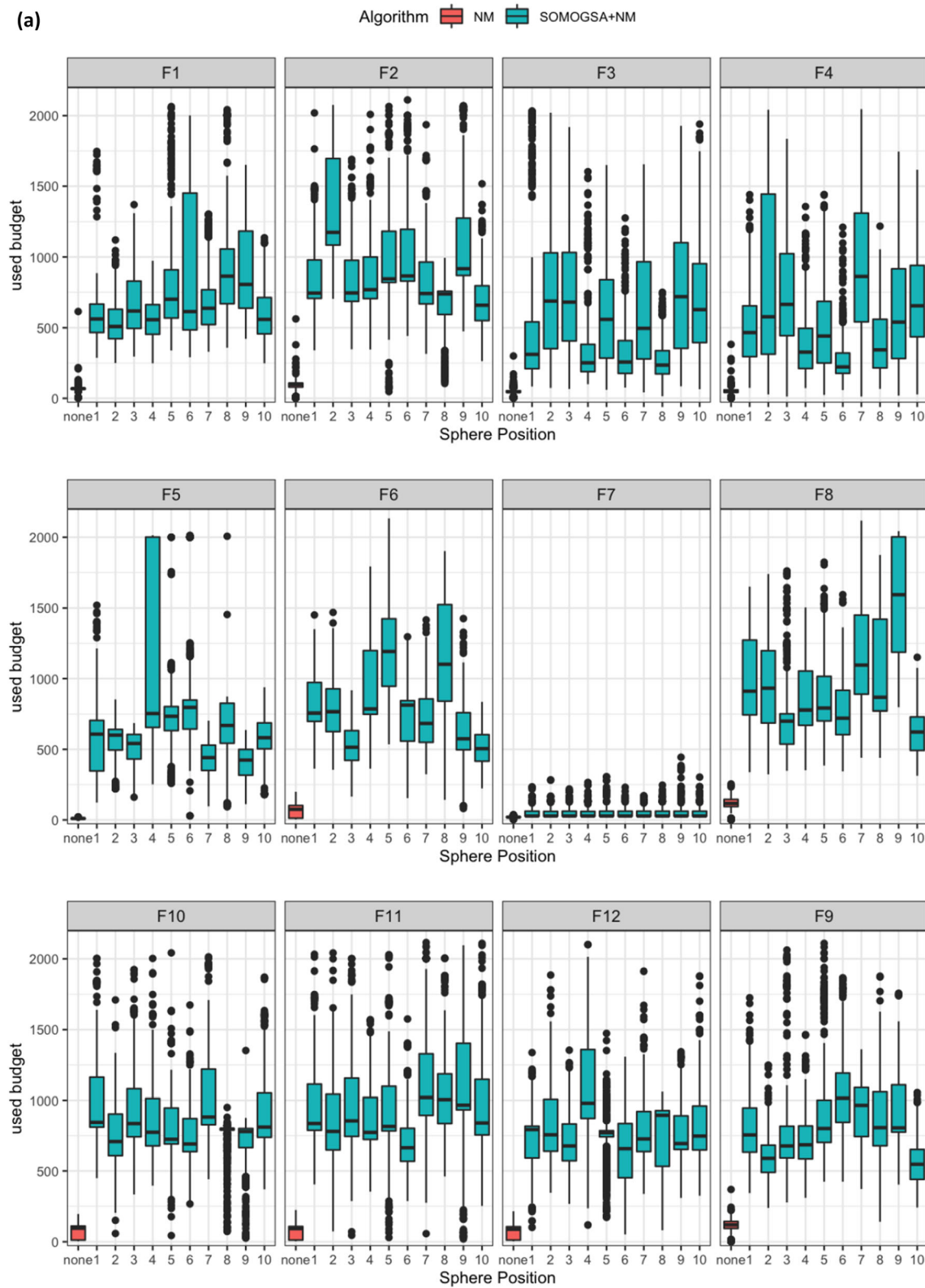
(a)

Fig. 8 Boxplots illustrating the used budget for the functions F13 to F24. The numbers 1 to 10 indicate the sphere optimum's position. *None* indicates the stand-alone method Nelder–Mead
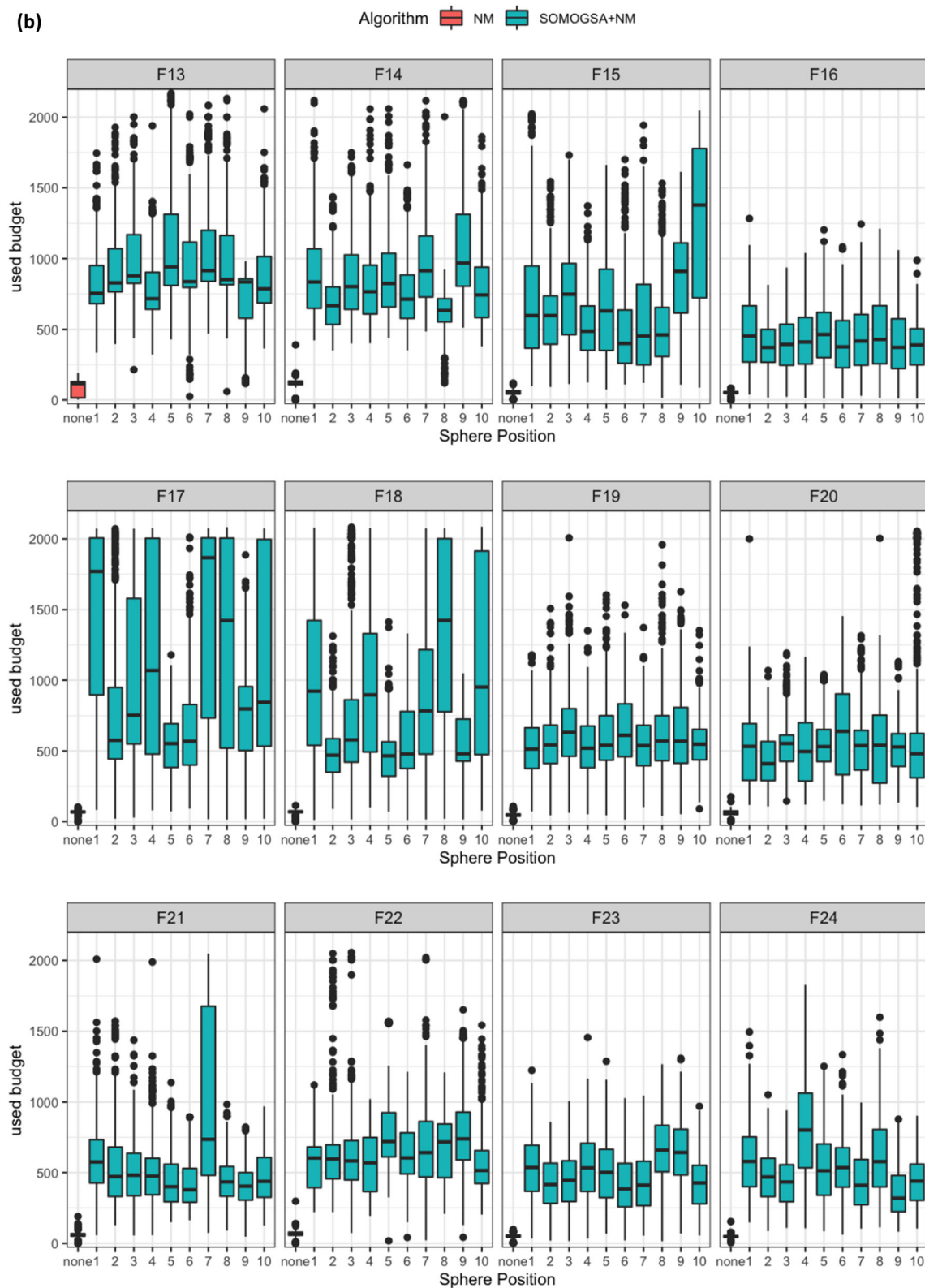
Fig. 8 continued

considering the results in the 5-dimensional decision space, it becomes apparent that there is potential for a benefit due to the multiobjectivization as well. For this dimension, SOMOGSA+NM consistently was more successful than NM for both considered functions (i.e., F21 and F22).

In addition to the allocation of a fixed budget, we recorded the number of function evaluations the algorithms actually used, respectively, and displayed them in Fig. 8 using boxplots. Clearly, SOMOGSA+NM as a more sophisticated hybrid local search technique requires a higher number of function evaluations than its original counterpart Nelder–Mead. However, this is also rooted in the exploratory steps of SOMOGSA along multi-objective efficient sets for escaping basins, which would have been traps for standard local search. As such, SOMOGSA proceeds further and gains better results than standard local search. Additionally, keeping the large gain in performance in mind, a sophisticated hybridization, which uses a global optimizer combined with a landscape-based determination of the most appropriate combination of sphere optimum and starting position, constitutes a very promising line of future research.

5 Conclusion

This paper shows the usefulness of transforming multimodal single-objective problems into multi-objective counterparts by artificially adding a second objective. Specifically, we investigate the performance of our algorithm SOMOGSA+NM, which hybridizes the sophisticated multi-objective MOGSA solver with Nelder–Mead local search, on the noiseless BBOB benchmark set.

The single-objective BBOB functions are complemented by a sphere function with varying center position. Specifically developed performance indicators reveal that SOMOGSA+NM consistently outperforms the stand-alone version of Nelder–Mead in the two-dimensional decision space. However, it also requires a considerably larger number of function evaluations. Sensitivity regarding the sphere position is quite low for unimodal functions, but increases for multimodal functions. Therefore, an individual yet automated configuration of this parameter based on landscape characteristics—i.e., numerical features derived by means of Exploratory Landscape Analysis (see, e.g., Kerschke and Trautmann 2019)—of the original singleobjective problem is a very promising perspective to reach optimal SOMOGSA+NM performance.

Of course, SOMOGSA itself is still a local search mechanism also after hybridization with Nelder–Mead (or other local search approaches). The proximity to the optimum of individual runs depends on the starting position within the search space. Next steps will thus include the integration of SOMOGSA+NM into a meta-heuristic which will be able to determine suitable starting points in a sophisticated manner including systematic restarts. Similar to the aforementioned automated determination of the sphere's optimum, Exploratory Landscape Analysis techniques can be highly supportive for the identification of promising starting positions. Moreover, based on first very promising studies, we will extend our analysis to higherdimensional search spaces in order to construct high-performing hybrid metaheuristics in the multiobjectivized problem space.

References

Aspar P, Kerschke P, Steinhoff V, et al (2021) Multi³: optimizing multimodal single-objective continuous problems in the multiobjective space by means of multiobjectivization. In: Proceedings of the 11th international conference on evolutionary multicriterion optimization (EMO). Springer, pp 311–322. https://doi.org/10.1007/978-3-030-72062-9_25

Brockhoff D, Friedrich T, Hebbinghaus N, et al (2007) Do additional objectives make a problem harder? In: Proceedings of the 9th annual conference on genetic and evolutionary computation (GECCO), pp 765–772

Dunlavy DM, O'Leary DP (2005) Homotopy optimization methods for global optimization. Technical report, Sandia National Laboratories (SNL), Albuquerque, NM, and Livermore, CA

Garza-Fabre M, Toscano-Pulido G, Rodriguez-Tello E (2015) Multiobjectivization, fitness landscape transformation and search performance: a case of study on the HP model for protein structure prediction. Eur J Oper Res (EJOR) 243(2):405–422

Grimme C, Kerschke P, Emmerich MTM, et al (2019a) Sliding to the global optimum: how to benefit from non-global optima in multimodal multi-objective optimization. In: AIP conference proceedings. AIP Publishing, pp 020,052–1–020,052–4

Grimme C, Kerschke P, Trautmann H (2019b) Multimodality in multi-objective optimization—more boon than bane? In: Proceedings of the 10th international conference on evolutionary multi-criterion optimization (EMO). Springer, pp 126–138

Grimme C, Kerschke P, Aspar P et al (2021) Peeking beyond peaks: challenges and research potentials of continuous multimodal

multi-objective optimization. Comput Oper Res 136(105):489. https://doi.org/10.1016/j.cor.2021.105489

Handl J, Lovell SC, Knowles J (2008) Multiobjectivization by decomposition of scalar cost functions. In: Proceedings of the 10th international conference on parallel problem solving from nature (PPSN X). Springer, pp 31–40

Hansen N, Finck S, Ros R, et al (2009) Real-parameter black-box optimization benchmarking 2009: noiseless functions definitions. Research Report RR-6829, INRIA. https://hal.inria.fr/inria-00362633

Hoffmeister F, Bäck T (1991) Genetic algorithms and evolution strategies: similarities and differences. In: Parallel problem solving from nature (PPSN). Springer, pp 455–469

Jensen MT (2004) Helper-objectives: using multi-objective evolutionary algorithms for single-objective optimisation. J Math Model Algorithms 3(4):323–347

John F (1948) Extremum problems with inequalities as subsidiary conditions, studies and essays presented to R. Courant on his 60th Birthday, January 8, 1948

Kerschke P, Grimme C (2017) An expedition to multimodal multi-objective optimization landscapes. In: Proceedings of the 9th international conference on evolutionary multi-criterion optimization (EMO). Springer, pp 329–343

Kerschke P, Grimme C (2021) Lifting the multimodality-fog in continuous multi-objective optimization. In: Metaheuristics for finding multiple solutions. Springer, pp 89–111

Kerschke P, Trautmann H (2019) Comprehensive feature-based landscape analysis of continuous and constrained optimization problems using the R-package flacco. In: Applications in statistical computing. Springer, pp 93–123. https://doi.org/10.1007/978-3-030-25147-5_7

Knowles JD, Watson RA, Corne DW (2001) Reducing local optima in single-objective problems by multi-objectivization. In: Proceedings of the international conference on evolutionary multi-criterion optimization (EMO). Springer, pp 269–283

McKay MD, Beckman RJ, Conover WJ (1979) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21(2):239–245

Nelder JA, Mead R (1965) A simplex method for function minimization. Comput J 7(4):308–313

Neumann F, Wegener I (2008) Can single-objective optimization profit from multiobjective optimization? In: Multiobjective problem solving from nature. Springer, pp 115–130

Preuss M (2015) Multimodal optimization by means of evolutionary algorithms. Springer, Berlin

Schäpermeier L, Grimme C, Kerschke P (2020) One PLOT to show them all: visualization of efficient sets in multi-objective landscapes. In: International conference on parallel problem solving from nature (PPSN). Springer, pp 154–167

Schäpermeier L, Grimme C, Kerschke P (2021) To boldly show what no one has seen before: a dashboard for visualizing multi-objective landscapes. In: Proceedings of the 11th international conference on evolutionary multi-criterion optimization (EMO). Springer, pp 632–644. https://doi.org/10.1007/978-3-030-72062-9_50

Schäpermeier L, Grimme C, Kerschke P (2022) MOLE: digging tunnels through multimodal multi-objective landscapes. In: Proceedings of the 24th annual conference on genetic and evolutionary computation (GECCO). ACM

Segura C, Coello Coello CA, Miranda G et al (2013) Using multi-objective evolutionary algorithms for single-objective optimization. 4OR 11(3):201–228

Segura C, Coello Coello CA, Miranda G et al (2016) Using multi-objective evolutionary algorithms for single-objective constrained and unconstrained optimization. Ann Oper Res 240(1):217–250

Steinhoff V, Kerschke P, Aspar P, et al (2020) Multiobjectivization of local search: single-objective optimization benefits from multi-objective gradient descent. In: Proceedings of the IEEE symposium series on computational intelligence (SSCI), pp 2445–2452. https://doi.org/10.1109/SSCI47803.2020.9308259

Tran TD, Brockhoff D, Derbel B (2013) Multiobjectivization with NSGA-II on the noiseless BBOB testbed. In: Proceedings of the 15th annual conference on genetic and evolutionary computation (GECCO) companion. ACM, pp 1217–1224