Stochastic and deterministic algorithms for continuous black-box optimization

Wang, H.

Author: Wang, H.
Title: Stochastic and deterministic algorithms for continuous black-box optimization
Issue Date: 2018-11-01

CHAPTER 5

Numerical Multi-objective Optimization

Many multi-objective optimization (MOO) algorithms have been proposed and exploited in real-world problems over the years, e.g., NSGA-II (Deb et al., 2000), SPEA2 (Zitzler et al., 2001) and SMS-EMOA (Beume et al., 2007). These evolutionary multi-criteria optimization (EMO) algorithms employ heuristic operators (e.g., random variation and selection operators), instead of using the gradient information of the objective functions. For a large subclass of such problems, that is the **continuous** multi-objective optimization problem, gradient-based algorithms are of interest due to the fact that they are generally fast, precise and stable with respect to local convergence. Various gradient-based approaches have been proposed for the multi-objective optimization task (Fliege and Svaiter, 2000; López et al., 2012; Hillermeier, 2001; Schütze et al., 2011). A relatively new idea is proposed by (Emmerich et al., 2007; Emmerich and Deutz, 2014), in which the gradient of the hypervolume indicator with respect to a set of decision vectors is computed. In this chapter, we adopt the definition and the computation of the *hypervolume indicator gradient* to steer the search points within the decision space. By using the hypervolume indicator gradient (Emmerich and Deutz, 2014), the search points are moved into the direction of steepest ascent w.r.t. the hypervolume indicator. Therefore, the proposed numerical multi-objective optimization algorithm is termed *hypervolume indicator gradient ascent multi-objective optimization* (HIGA-MO). The major benefits of exploiting hypervolume gradients are 1) the points in the objective space will be well distributed on the Pareto front, 2) it is almost free of control parameters, and 3) the algorithm has a high precision of convergence to the Pareto front.

However, the first implementation of this idea showed numerical problems. As a remedy, ideas that were developed in the field of evolutionary multi-criterion

optimization are adopted in this thesis. Firstly, the hypervolume indicator may have zero gradient components at some decision vectors, e.g., the dominated points. The well-known non-dominated sorting technique is adopted and combined with the hypervolume indicator gradient computation, in order to equip each decision vector with a multi-layered gradient. Secondly, the normalization of the hypervolume indicator *sub*-gradient is used to overcome the "creepiness" phenomenon observed in earlier versions of hypervolume gradient ascent, and caused by an imbalance in the length of sub-gradients which leads to a slow convergence speed (Sosa Hernández et al., 2014). Thirdly, the usage of constant step-sizes is no longer appropriate if the precise convergence to the Pareto front is aimed for. Instead, a cumulative step-size control inspired by the optimal gradient ascent is proposed to dynamically adapt the step-size. Such a cumulative step-size control resembles the step-size adaptation mechanism in the well-known CMA-ES (Hansen and Ostermeier, 2001), an evolutionary algorithm for single objective continuous optimization. The resulting algorithm is tested on problems named ZDT1-4 and ZDT6 from (Zitzler et al., 2000). Its performance is compared to three evolutionary algorithms: NSGA-II (Deb et al., 2000), SPEA2 (Zitzler et al., 2001) and SMS-EMOA (Beume et al., 2007), as well as the other methods for steering the dominated points.

In addition, the hypervolume-based numerical MOO is extended by differentiating the hypervolume gradient again, yielding the *hypervolume indicator Hessian matrix*. We furthermore investigate the condition on which the Hessian matrix stays *non-singular*, showing that it is "safe" to apply the Hessian in general applications. Based on the Hessian matrix, the hypervolume indicator Newton method is proposed and validated.

In the following, the general settings/notations on *set-oriented numerics* are given first. In multi-objective optimization problems (MOPs), a collection of functions, represented as the m-tuple below, are optimized simultaneously:

$$(f_1 \colon \mathrm{S}_1 \to \mathbb{R}, f_2 \colon \mathrm{S}_2 \to \mathbb{R}, \ldots, f_m \colon \mathrm{S}_m \to \mathbb{R}), \quad \mathrm{S}_1, \mathrm{S}_2, \ldots, \mathrm{S}_m \subseteq \mathbb{R}^d.$$

where d denotes the dimension of the domain of each function and m denotes the number of objective functions. Without loss of generality, we assume all the functions above are to be minimized (maximization problems can be transformed into minimization problems). In this thesis, it is assumed that each objective function f_i is continuously differentiable almost everywhere in S_i. Thus, the MOP

can be formulated as follows:

$$\min_{\mathbf{x}\in \mathrm{S}} \mathbf{f}(\mathbf{x}), \quad \mathrm{S} = \bigcap_{i=1}^{m} \mathrm{S}_i \subseteq \mathbb{R}^d,$$

where $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}))^\top$ is a vector-valued function composed of m objective functions: $\mathbf{f}\colon \mathrm{S} \to \mathbb{R}^m$. Note that the minimization of the vector-valued function $\mathbf{f}$ is understood with respect to the *Pareto order* $\prec$ as defined in Section 1.2. Let $\mathbf{a}, \mathbf{b}$ be two distinct points in $\mathbb{R}^m$. We say $\mathbf{a} \prec \mathbf{b}$ iff $a_i \leq b_i$, $i = 1, \ldots, m$, where $\leq$ is the natural total order on the real numbers. Because of the continuous differentiability assumption on each objective function, $\mathbf{f}$ is again continuously differentiable almost everywhere in S. The gradient information is expressed as transpose of the Jacobian matrix as follows:

$$\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} = [\nabla f_1(\mathbf{x}), \nabla f_2(\mathbf{x}), \ldots, \nabla f_m(\mathbf{x})], \quad \nabla f_i(\mathbf{x})\colon \mathrm{S} \to \mathbb{R}^d, \quad i = 1, 2, \ldots, m.$$

In addition, it is assumed that each gradient vector above (column vector) can be computed either analytically or numerically. In MOPs, a set of decision vectors are moved in *decision space* S to approximate the Pareto efficient set, which is the so-called Pareto efficient set approximation:

$$X = \left\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(\mu)}\right\}, \ \mathbf{x}^{(i)} \in \mathrm{S}, \ i = 1, 2, \ldots, \mu.$$

with corresponding Pareto front approximation set (objective vectors) in the *objective space*:

$$Y = \left\{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \ldots, \mathbf{y}^{(\mu)}\right\}, \ \mathbf{y}^{(i)} = \mathbf{f}(\mathbf{x}^{(i)}) \in \mathbb{R}^m, \ i = 1, 2, \ldots, \mu.$$

In order to measure and compare the quality among Pareto front approximation sets Y, one approach is to quantify the quality by constructing a proper indicator. The most common one is the hypervolume indicator H (Zitzler and Thiele, 1998; Zitzler et al., 2003). Given a reference point $\mathbf{r} \in \mathbb{R}^m$, the **hypervolume indicator** of the Pareto front approximation set Y can be expressed as:

$$H(Y;\ \mathbf{r}) = \lambda^m \left(\bigcup_{\mathbf{y}\in Y} [\mathbf{r}, \mathbf{y}] \right),$$

where λ^m denotes the Lebesgue measure on $\mathbb{R}^m$, which is the size of the hypervolume dominated by the approximation set Y with respect to the reference space. Note that the reference point $\mathbf{r}$ will be assumed to be a given constant and thus omitted

in the following notations for brevity. The hypervolume indicator gradient is defined as the gradient of the hypervolume indicator with respect to the approximation of the Pareto efficient set, which is proposed in Emmerich and Deutz (2014); Emmerich et al. (2007). In this thesis, the derivation of the hypervolume indicator gradient is reformulated and the notation is simplified. In the following, we shall use matrix calculus notations with denominator layout, meaning that the derivative of a vector/matrix is laid out according to the denominator.

5.1 Mixed-Peak Test Problem

Prior to the discussion of the numerical MOO algorithm, a bi-objective problem class, called *Mixed-Peak* problems are introduced for investigating the behavior of the proposed algorithms. Such a problem class is chosen over other standard benchmark problems, e.g., the so-called ZDT problems (Zitzler et al., 2000), because 1) it allows for controlling the problem difficulty of its instance, by varying the number of peaks in each objective function. 2) it is smooth and differentiable almost everywhere in its domain, which makes it a perfect test problem for the gradient and Hessian methods. As no analytical property is available on this problem, the detailed analysis is conducted on this problem and as a result, the expressions of the Pareto front and efficient set are derived.

5.1.1 Mixed-Peak Functions

In this this, a sophisticated problem generator, called *Multiple Peaks Model 2* (MPM2, Wessing (2015)), is adopted to illustrate the proposed topological definitions and further analyze the behavior of explorative algorithms. Such a function class is a mixture of similar unimodal functions, i.e., the peaks, that have *convex* local level sets, which is typically combined with the well-known Karush-Kuhn-Tucker theorem to identify local efficient points. In addition, the complexity of the problem can be easily controlled by the number of peaks. The mixed-peak function is defined as an unconstrained function $f : \mathbb{R}^d \to \mathbb{R}$ that is subject to

minimization:

$$f(\mathbf{x}) = 1 - \max_{1 \le i \le N} \{g_i(\mathbf{x})\}, \quad \mathbf{x} \in \mathbb{R}^d. \tag{5.1}$$

$$g_i(\mathbf{x}) = h_i \left(1 + \frac{\left(\sqrt{(\mathbf{x} - \mathbf{c}_i)^\top \mathbf{\Sigma}_i (\mathbf{x} - \mathbf{c}_i)} \right)^{s_i}}{r_i} \right)^{-1}, \quad i = 1, \ldots, N. \tag{5.2}$$

The function g above defines a parameterized *quasi-concave* unimodal peak, whose negative leads to *quasi-convex* valleys on function f. According to the `optproblems` package (Wessing, 2016), it has the following parameters: (1) number of peaks $N \in \mathbb{Z}_{>0}$, (2) center $\mathbf{c}_i \in \mathbb{R}^d$, height $h_i \in [0, 1]$ and radius $r_i \in [0.25\sqrt{d}, 0.5\sqrt{d}]$ per peak, with decision space dimension d, (3) "shape" $s_i \in [1.5, 2.5]$ per peak, controlling the landscape's steepness, (4) rotation of the elliptical level sets based on a *positive definite* matrix $\mathbf{\Sigma}_i$. In the following, we will use the norm notation $\|\mathbf{x} - \mathbf{c}_i\|_{\mathbf{\Sigma}_i} := \sqrt{(\mathbf{x} - \mathbf{c}_i)^\top \mathbf{\Sigma}_i (\mathbf{x} - \mathbf{c}_i)}$ as it can be considered as the Mahalanobis distance w.r.t. $\mathbf{\Sigma}_i$.

Ridges: As a result from the definition of f (Eq. (5.1)), the landscape can contain ridges. The set of all ridges of f can be represented by:

$$\mathcal{R} = \left\{ \mathbf{x} \in \mathbb{R}^d \mid \exists i \neq j \in \{1, 2, \ldots, N\}, g_i(\mathbf{x}) = g_j(\mathbf{x}) \text{ and } g_i(\mathbf{x}) = \max_{1 \le k \le N} \{g_k(\mathbf{x})\} \right\},$$

i.e., the set of all points on which the value of f is simultaneously attained by at least two peak functions. In the simple case, when the $\mathbf{\Sigma}_i$'s are identical and the peaks differ only in centers, the ridges actually form a Voronoi diagram in the decision space. According to Eq. (5.1), for any point that is not on the ridge, $\mathbf{x} \in \mathbb{R}^d \setminus \mathcal{R}$, there is only one peak function that is effective or *active*. From now on, the active peak function at $\mathbf{x}$ is denoted as g_τ w.r.t. $\tau = \arg\max_{1 \le i \le N} \{g_i(\mathbf{x})\}$. In fact, ridges separate the decision space into many *active regions*, on each of which only a single peak function g is active:

$$\mathcal{A}_i = \left\{ \mathbf{x} \in \mathbb{R}^d \mid \forall k \in \{1, 2, \ldots, N\} \setminus \{i\},\ g_i(\mathbf{x}) > g_k(\mathbf{x}) \right\}, \quad i = 1, 2, \ldots, N.$$

Note that the active regions $\mathcal{A}_i$'s are open and mutually disjoint and the union of all such active regions $\mathcal{A} = \cup_{1 \le i \le N} \mathcal{A}_i$ is equal to the set of non-ridge points.

Convex Local Level Sets: Given the quasi-concavity of each peak g_i, $1-g_i$ has local convex level sets in $\mathbb{R}^d$. If the function $1-g_i$ is restricted to an ε-Euclidean ball $B_\varepsilon(\mathbf{x}^*) = \left\{\mathbf{x} \in \mathbb{R}^d \mid \|\mathbf{x}-\mathbf{x}^*\| < \varepsilon\right\}$ for every $\mathbf{x}^* \in \mathbb{R}^d$ and every $\varepsilon > 0$, the resulting function $1-g_i\big|_{B_\varepsilon(\mathbf{x})} : B_\varepsilon(\mathbf{x}) \to \mathbb{R}$ also has local convex level sets. Also, due the fact that the active regions $\mathcal{A}_i$'s are disjoint and open, for every non-ridge point $\mathbf{x}^*$, it is possible to find a $\delta > 0$ (depending on $\mathbf{x}^*$) such that $B_\delta(\mathbf{x}^*) \subset \mathcal{A}_\tau$ and $B_\delta(\mathbf{x}^*) \cap \mathcal{A}_i = \emptyset$, $\forall i \neq \tau$ (τ is the unique index of the active peak function at $\mathbf{x}^*$). Then the restricted f to $B_\delta(\mathbf{x}^*)$, $f\big|_{B_\delta(\mathbf{x}^*)}$ equals $1-g_\tau\big|_{B_\delta(\mathbf{x}^*)}$ and thus it has local convex level sets. Therefore, we have the following conclusion:

$$\forall \mathbf{x}^* \in \left(\mathbb{R}^d \setminus \mathcal{R}\right)\ \exists \delta > 0,\ f\big|_{B_\delta(\mathbf{x}^*)} \text{ has local convex level sets.} \tag{5.3}$$

For the points on the ridge, $\mathbf{x}^* \in \mathcal{R}$, the conclusion above does not hold because it is not possible to find a δ such that $B_\delta(\mathbf{x}^*)$ has no intersection with all $\mathcal{A}_i$'s except $\mathcal{A}_\tau$.

As the gradient of the mixed-peak function is required to derive the Pareto front, we given it as follows:

$$\nabla f(\mathbf{x}) = \frac{h_\tau s_\tau}{r_\tau}\left(1 + \frac{\|\mathbf{x}-\mathbf{c}_\tau\|_{\mathbf{\Sigma}_\tau}^{s_\tau}}{r_\tau}\right)^{-2} \|\mathbf{x}-\mathbf{c}_\tau\|_{\mathbf{\Sigma}_\tau}^{s_\tau-2}\, \mathbf{\Sigma}_\tau(\mathbf{x}-\mathbf{c}_\tau). \tag{5.4}$$

5.1.2 Mixed-Peak Bi-objective Problem

By generating two different configurations for the parameters in Eq. (5.1), two different multimodal functions are constructed, naturally defining a bi-objective optimization problem:

$$f_1(\mathbf{x}) = 1 - \max_{1\le i\le N} g_i(\mathbf{x}) \to \min, \qquad f_2(\mathbf{x}) = 1 - \max_{1\le i\le N'} g'_i(\mathbf{x}) \to \min\,.$$

Note that the peak function g and g' (and its parameters N and N') are distinguished by the superscript. Next, the efficient set and Pareto front are derived analytically. In the following, the analytical efficient set and Pareto front are derived.

One Peak Scenario We first consider a simple case in which each objective function consists of one peak without any ridges in the domain. In this case, the

objective functions degenerate to:

$$f_1(\mathbf{x}) = 1 - h\left(1 + \frac{\|\mathbf{x}-\mathbf{c}\|_{\mathbf{\Sigma}}^{s}}{r}\right)^{-1}, \quad f_2(\mathbf{x}) = 1 - h'\left(1 + \frac{\|\mathbf{x}-\mathbf{c}'\|_{\mathbf{\Sigma}'}^{s'}}{r'}\right)^{-1}.$$

According to the Karush-Kuhn-Tucker (KKT) condition (Ehrgott, 2006) for multi-objective optimization problems, a *necessary condition* for $\mathbf{x}^* \in \mathbb{R}^d$ being efficient is:

$$\exists \lambda_1 > 0, \lambda_2 > 0,\ \lambda_1 \nabla f_1(\mathbf{x}^*) + \lambda_2 \nabla f_2(\mathbf{x}^*) = \mathbf{0}.$$

Substituting the condition above by the gradient expression (Eq. (5.4)) leads to:

$$\begin{aligned} &\lambda_1 C(\mathbf{x}^*)\mathbf{\Sigma}(\mathbf{x}^* - \mathbf{c}) + \lambda_2 C'(\mathbf{x}^*)\mathbf{\Sigma}'(\mathbf{x}^* - \mathbf{c}') = \mathbf{0}, \\ &\text{with } \ C(\mathbf{x}^*) := \frac{hs}{r}\left(1 + \frac{\|\mathbf{x}^*-\mathbf{c}\|_{\mathbf{\Sigma}}^{s}}{r}\right)^{-2} \|\mathbf{x}^* - \mathbf{c}\|_{\mathbf{\Sigma}}^{s-2}. \end{aligned}$$

And C' is defined similarly to C by adding prime superscripts to all parameters. As a result, the condition above can further be simplified to:

$$\exists \lambda_1 > 0, \lambda_2 > 0,\ \ \mathbf{\Sigma}(\mathbf{x}^* - \mathbf{c}) = -\frac{\lambda_2 C'(\mathbf{x}^*)}{\lambda_1 C(\mathbf{x}^*)}\mathbf{\Sigma}'(\mathbf{x}^* - \mathbf{c}'). \tag{5.5}$$

Let us denote $k := \lambda_2 C'(\mathbf{x}^*)/\lambda_1 C(\mathbf{x}^*)$. Thus, $\lambda_1, \lambda_2 > 0$ and $C, C' \geq 0$ result in $k \geq 0$. In addition, $C \to 0$ leads to $k \to \infty$, i.e., $\mathbf{x}^* \to \mathbf{c}$. Due to the fact that C and C' are continuous functions w.r.t. $\mathbf{x}^*$, k is also *continuous* in $\mathbb{R}^d$. Therefore, it must take any value between its minimum and maximum, resulting in $0 \leq k < \infty$. Taking the range of k into account, every point that satisfies Eq. (5.5) can be written as:

$$\forall k > 0,\ \ \mathbf{x}^* = \mathbf{c} - \left(\frac{\mathbf{\Sigma}}{k} + \mathbf{\Sigma}'\right)^{-1} \mathbf{\Sigma}'(\mathbf{c} - \mathbf{c}'). \tag{5.6}$$

Note that the points above are not necessarily local efficient points (as defined in Section 1.2). The sufficiency can be shown as follows: for any point $\mathbf{x}^* \in \mathbb{R}^d$ satisfying Eq. (5.6) – remember, there is no ridge in this scenario – there exists an $\varepsilon > 0$ such that the restricted objective function $f_1\big|_{B_\varepsilon(\mathbf{x}^*)}$ has local convex level sets according to Eq. (5.3). Similarly, there exists an $\varepsilon' > 0$ such that $f_2\big|_{B_{\varepsilon'}(\mathbf{x}^*)}$ has local convex level sets. It is then possible to construct a Euclidean ball with radius $\varepsilon^* := \min\{\varepsilon, \varepsilon'\}$ such that: $f_1\big|_{B_{\varepsilon^*}(\mathbf{x}^*)}$ and $f_2\big|_{B_{\varepsilon^*}(\mathbf{x}^*)}$ both have local **convex** level sets. This implies that it is always possible to find a neighborhood around a point where the local level sets of both objective functions are convex. Thus, it is sufficient

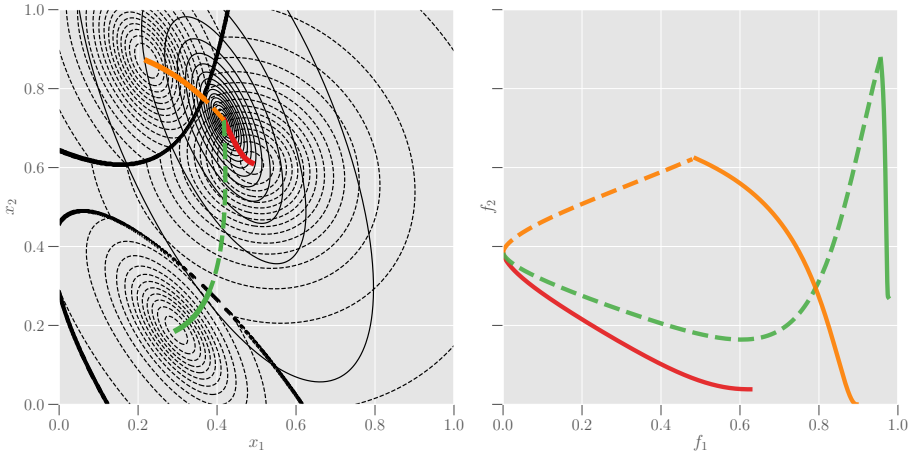

Figure 5.1: Example of analytical Pareto fronts and efficient sets: the contour lines of f_1 (solid curves, 1 peak) and f_2 (dashed curves, 3 peaks) are drawn in the decision space (left) with ridges shown as thick solid curves. Three local efficient sets are drawn in different colors while the dashed extensions of them represent the pseudo-efficient sets. The corresponding Pareto fronts are shown on the right.

to conclude that points satisfying Eq. (5.6) are *locally Pareto efficient* and the efficient set of the problem is expressed as:

$$\mathcal{X}_{LE} = \left\{ \mathbf{c} - \left(\frac{\mathbf{\Sigma}}{k} + \mathbf{\Sigma}' \right)^{-1} \mathbf{\Sigma}'(\mathbf{c} - \mathbf{c}') \mid 0 \leq k < \infty \right\}. \tag{5.7}$$

Consequently, the Pareto front can implicitly be obtained by applying the objective functions to the efficient set from above. When the contour lines are spherical for both objective functions, the arguments here can be largely simplified. We omit such a special case, since it has already been discussed in detail in Kerschke et al. (2016).

Multiple Peaks If each of the objective functions consists of multiple peak functions, namely $N > 1$, the efficient set derived in Eq. (5.7) can be adapted in the following manner: suppose function f_1 and f_2 contain N and N' peaks, respectively. For each pair of peaks between two objective functions (e.g., g_i and g'_j), a *pseudo-efficient set* can be calculated according to Eq. (5.7) as if the rest of

the peaks in both objective functions were not existing:

$$\mathcal{P}_{ij} = \left\{ \mathbf{c}_i - \left(\frac{\mathbf{\Sigma}_i}{k} + \mathbf{\Sigma}'_j \right)^{-1} \mathbf{\Sigma}'_j (\mathbf{c}_i - \mathbf{c}'_j) \mid 0 \leq k < \infty \right\},$$

where $\mathbf{c}_i$ and $\mathbf{c}'_j$ are the centers of the i-th and j-th peak of function f_1 and f_2, respectively. Note that Eq. (5.7) requires that no ridge is present in the function domain and thus for the set defined above, it is not necessarily a local efficient set. Let us denote the active region of peak g_i and g'_j as $\mathcal{A}_i$ and $\mathcal{A}'_j$, respectively. Then the region on which g_i and g'_j are both active is $\mathcal{A}_i \cap \mathcal{A}'_j$. Consider the intersections of $\mathcal{P}_{ij}$ and the ridges $\mathcal{R}$ of f_1 for instance: at such points, any infinitesimal movement towards a different active region other than $\mathcal{A}_i \cap \mathcal{A}'_j$ will revert the direction of ∇f_1 and therefore this movement will improve both f_1 and f_2 values of the intersection points. This implies that the points in $\mathcal{P}_{ij}$ intersecting or crossing the ridges are not efficient for g_i and g'_j. In other words, the efficient set $\mathcal{X}^*_{ij} = \mathcal{P}_{ij} \cap \mathcal{A}_i \cap \mathcal{A}'_j$ associated with peak g_i and g'_j is the intersection of $\mathcal{P}_{ij}$ with the active regions of both peak functions. In addition, all local efficient sets can be enumerated by calculating the local efficient set associated with each pair of peaks between two objective functions: $\mathcal{X}^* = \bigcup_{i=1}^{N} \bigcup_{j=1}^{N'} \mathcal{X}^*_{ij}$. An example of this is illustrated in Fig. 5.1. Here, three pseudo-efficient sets are depicted in different colors (red, orange and green) and the orange and green sets are truncated by the ridges (thick black lines), where the valid local efficient sets are depicted as solid curves.

5.2 Hypervolume Indicator Gradient

Intuitively, the hypervolume indicator can be expressed as a function of the Pareto efficient set approximation X, which allows for the differentiation of hypervolume indicator with respect to decision vectors. More specifically, by concatenation of all the vectors in this set, we obtain a so-called $\mu \cdot d$-vector:

$$\mathbf{X} = \left[\mathbf{x}^{(1)^\top}, \mathbf{x}^{(2)^\top}, \ldots, \mathbf{x}^{(\mu)^\top} \right]^\top \in \mathrm{S}^\mu \subseteq \mathbb{R}^{\mu \cdot d}.$$

and its corresponding Pareto front approximation vector can be written as a $\mu \cdot m$-vector:

$$\mathbf{Y} = \left[\mathbf{y}^{(1)^\top}, \mathbf{y}^{(2)^\top}, \ldots, \mathbf{y}^{(\mu)^\top} \right]^\top \in \mathbb{R}^{\mu \cdot m}.$$

In order to establish a connection between $\mu \cdot d$-vectors and $\mu \cdot m$-vectors, we define a mapping $\mathbf{F}\colon \mathrm{S}^{\mu} \to \mathbb{R}^{\mu \cdot m}$,

$$\mathbf{F}(\mathbf{X}) := \left[\mathbf{f}(\mathbf{x}^{(1)})^{\top}, \mathbf{f}(\mathbf{x}^{(1)})^{\top}, \ldots, \mathbf{f}(\mathbf{x}^{(\mu)})^{\top}\right]^{\top}.$$

Now consider that the hypervolume indicator, that is normally defined in the objective space, can be re-written as a function of $\mu \cdot d$-vectors by composition:

$$\mathcal{H}_{\mathbf{F}}(\mathbf{X}) := H(\mathbf{F}(\mathbf{X})),$$

which is a continuous mapping from S^{μ} to $\mathbb{R}$, for which under certain regularity conditions the gradient is defined (in case of differentiable objective functions only for a zero measure subset of $\mathbb{R}^{\mu \cdot d}$ the gradient is undefined, in which case one-sided derivatives still exist). Given $\mathcal{H}_{\mathbf{F}}$, its derivatives (hypervolume indicator gradient) are defined (given they exist) by:

$$\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{X}} = \left[\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(1)}}^{\top}, \ldots, \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(\mu)}}^{\top}\right]^{\top}, \tag{5.8}$$

where each of the term in the RHS of the equation above is called *sub-gradient*, which is the local hypervolume change rate by moving each decision vector infinitesimally. It has been shown in Emmerich and Deutz (2014) that the hypervolume indicator gradient is the concatenation of the hypervolume contribution gradients. Moreover, the sub-gradients can be calculated by applying the chain rule:

$$\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(i)}} = \frac{\partial \mathbf{y}^{(i)}}{\partial \mathbf{x}^{(i)}} \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{y}^{(i)}} \tag{5.9}$$

$$= \sum_{k=1}^{m} \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k(\mathbf{x}^{(i)})} \nabla f_k(\mathbf{x}^{(i)}). \tag{5.10}$$

The first partial derivative in Eq. (5.9) is the gradient of $\mathcal{H}_{\mathbf{F}}$ in the objective space while the second one is the transpose of the Jacobian matrix of the mapping $\mathbf{F}$. Eq. (5.10) is the detailed form. From it, it is clear that the hypervolume indicator gradient is a *linear combination of gradient vectors of objective functions*, where the weight for an objective function is the partial derivative of the hypervolume indicator at this objective value. We omit the calculation for gradients of $\mathcal{H}_{\mathbf{F}}$ in the objective space for simplicity, noting that in the bi-objective case they correspond to the length of the steps of the attainment curve. For the high dimensional case and efficient computation, see Emmerich and Deutz (2014).

Note that in practice the length of the sub-gradients usually differs by orders of magnitude, leading to the "creepiness" behavior (Sosa Hernández et al., 2014) that some decision vectors move much faster than the rest, Such a behavior results in a very slow convergence speed and points might get dominated by others. As a remedy, it is suggested to normalize all the sub-gradients.

5.2.1 Steering Dominated Points

The difficulty increases when applying the hypervolume indicator gradient direction for steering the decision vectors: the hypervolume indicator can either be zero or only one-sided at decision vectors. For example, at every strictly dominated search point, the hypervolume indicator sub-gradient is zero, because the Pareto front and thus the hypervolume indicator remain unchanged if it is moved locally in an infinitesimally small neighborhood. For every weakly dominated point, the hypervolume indicator sub-gradient at this point, even does not exist due to the fact that only one-sided partial derivatives exist. Consequently, such decision vectors will become stationary in the gradient ascent method. One obvious solution to such a problem is to apply evolutionary operators (mutation and crossover) on those search points (decision vectors) until they become non-dominated. However, as we are aiming for a fully deterministic multi-objective optimization algorithm, randomized operators are not adopted in this thesis.

Some methods have been proposed to steer dominated points (Ren et al., 2015; Wang et al., 2017; López et al., 2012). The most prominent one, proposed in López et al. (2012), computes the gradient at dominated points as follows (for bi-objective problems):

$$-\left(\frac{\nabla f_1(\mathbf{x}^{(i)})}{\left\|\nabla f_1(\mathbf{x}^{(i)})\right\|}+\frac{\nabla f_2(\mathbf{x}^{(i)})}{\left\|\nabla f_2(\mathbf{x}^{(i)})\right\|}\right), \quad \mathbf{x}^{(i)} \text{ is dominated.}$$

which is a sum of normalized gradients of each objective function (the minus symbol is for the minimization problem). It guarantees that dominated decision vectors move into the *dominance cone* (Wang et al., 2017). However, such a method only considers the movement of single points, instead of a set of search points and it does not generalize to more than two dimensions. We shall call this method **Lara's direction** in the following experiments, where it is compared with the method proposed in this thesis. Another method for steering the dominated points is

proposed by the authors in Wang et al. (2017). It steers dominated points towards the nearest gap on the non-dominated set. The search direction is determined as the gradient of the distance of the dominated objective vector to the center of its nearest gap. Again, this method steers dominated points independently and is termed **gap-filling** in this thesis. In the above methods, dominated points are steered widely independent of each other, which might result in a diversity loss.

In this thesis, we propose to use the *non-dominated sorting* technique that is developed in the NSGA-II algorithm (Srinivas and Deb, 1994), in order to compute the hypervolume indicator gradients of multiple layers of non-dominated sets. In detail, the decision and objective vectors are partitioned into q subsets, or *layers* according to their dominance rank in the objective space:

$$\mathbf{X} \to \left\{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^q\right\},$$
$$\mathbf{X}^i = \left[\mathbf{x}^{(i_1)^\top}, \mathbf{x}^{(i_2)^\top}, \ldots, \mathbf{x}^{(i_\mu)^\top}\right]^\top,$$

where X^i indicates a layer of order i and i_μ indexes decision vectors in the ith rank layer. The layers can be recursively defined as (given ND as the procedure

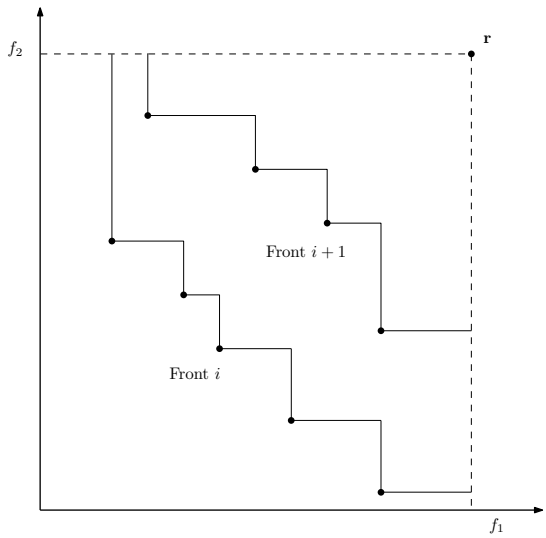

Figure 5.2: Schematic graph showing the partition of the objective vectors using non-dominated sorting. For each partition (layer), a hypervolume indicator is defined and thus its gradient can be computed.

that selects the non-dominated subset from an approximation set):

$$X^1 = \text{ND}(X), \quad X^{i+1} = \text{ND}\left(X - \bigcup_{j=1}^{i} X^j\right),$$

where q is the highest index i such that $X^i \neq \emptyset$. Note that the $\mu \cdot m$-vector is also partitioned as above. In principle, it is possible to compute the hypervolume indicator gradient for any layer by ignoring all the layers that dominate it (have a lower rank) temporarily. This partition is illustrated in Fig. 5.2. The hypervolume volume indicator gradient on the approximation set $\mathbf{X}$ can be (re-)written as the concatenation of hypervolume indicator gradients on each layer:

$$\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{X}} := \left[\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X}^1)}{\partial \mathbf{X}^1}^\top, \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X}^2)}{\partial \mathbf{X}^2}^\top, \ldots, \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X}^q)}{\partial \mathbf{X}^q}^\top\right]^\top. \tag{5.11}$$

Note that again q is the number of layers obtained from non-dominated sorting techniques. The gradient computation given in Eq. (5.10) can be used to compute each gradient term above. Thus, each decision vector is associated with a steepest ascent direction that maximizes its hypervolume contribution on each layer.

There are two advantages of using the non-dominated sorting procedure. Firstly, maximizing the hypervolume will not only steer the points towards the Pareto front, but also spread out the points across the intermediate Pareto front approximation. By applying the hypervolume indicator gradient direction on each layer, the decision vectors on each layer will be well distributed before a dominated layer merges into the global Pareto front and thus the additional cost to spread out points after merging is small. Moreover, when the Pareto efficient set is disconnected in the decision space, the proposed approach will increase the convergence speed due to the fact that each connected efficient set is treated as one layer and the decision vectors on it are spread quickly over the efficient sets. This effect can be shown by visualizing the trajectories of the approximation set on a simple objective landscape. In Fig. 5.3, trajectories of the approximation set are illustrated in both decision and objective space, on MPM2 functions (from the R `smoof` package[1]). In the decision space, it is clear that our layering approach (Fig. 5.3) manages to approximate five disconnected efficient sets with a good distribution of points. Secondly, on the real landscape, it is possible that local Pareto fronts exist (e.g., consider the well-known ZDT4 problem (Zitzler et al., 2000)). Using the non-dominated sorting, it is more

[1] https://github.com/jakobbossek/smoof

likely to identify those local Pareto fronts, which could be helpful to balance global and local search. This advantage of the proposed approach is exploited by the authors in multi-objective multi-modal landscape analysis (Kerschke et al., 2016).

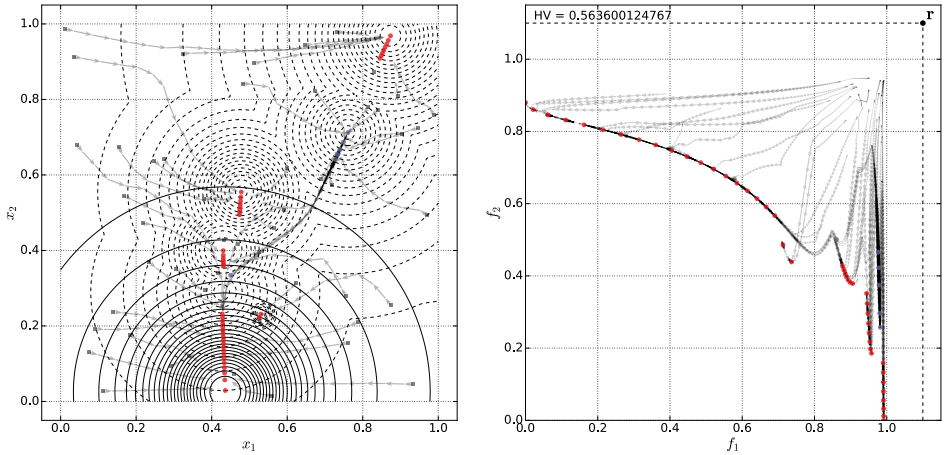

Figure 5.3: Trajectories of 50 points under hypervolume indicator gradient direction to approximate the Pareto front using 10^3 function evaluations. The experiment is conducted on a bi-objective problem MPM2 (from the R `smoof` package) in the 2-D decision space. All five disconnected components of the Pareto front are obtained with well distributed points. **Left**: the decision space. **Right**: the objective space.

5.2.2 Step-size adaptation

The constant step-size setting that is common in gradient descent (ascent) for the single objective optimization task, is no longer appropriate. Usually, the length of the gradient vector (in the gradient field) gradually goes to zero when approaching the local optimum. In this case, a properly set constant step-size will lead to the local optimum in a stable manner. However, in our case, due to the normalization, the length of the search steps is always 1 when decision vectors are approaching the Pareto efficient set. If a constant step-size is applied here, the decision vector will *overshoot* its optimal position and begin to oscillate (even diverge). In order to tackle this issue, the step-size of the decision vectors needs to 1) gradually decrease when approaching the Pareto efficient set and 2) increase quickly when

the decision vectors are far away from the efficient set. In addition, it is reasonable to use individual step-sizes that are controlled independently for each decision vector because their optimal step-size differs largely.

A cumulative step-size adaptation mechanism is proposed to approximate the optimal step-size in the optimization process. It is inspired by the following observation: in single objective gradient optimization, if the step-size is set optimally, then consecutive search directions are perpendicular to each other. In order to approximate the optimal step-size setting, the inner product of consecutive normalized hypervolume indicator gradients is calculated. If such an inner product is positive, it indicates the current step-size is smaller than the optimal one and vice versa:

$$I_t^{(i)} = \left\langle \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(i)}}^{(t-1)}, \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(i)}}^{(t)} \right\rangle, \quad i = 1, \ldots, \mu, \quad t = 1, 2, \ldots.$$

Note that superscripts $(t), (t-1)$ are iteration indices. In addition, such an inner product computed in each iteration fluctuates hugely and direct use of it leads to unstable adaptation behavior. Therefore, the inner product is cumulated using exponentially decreasing weights through the iterations to get a more stable indicator for the step-size adaptation. The cumulative rule for the inner product is written as follows:

$$p_t^{(i)} \leftarrow (1-c)p_{t-1}^{(i)} + cI_t^{(i)}, \quad i = 1, \ldots, \mu, \quad t = 1, 2, \ldots. \tag{5.12}$$

Note that $p_t^{(i)}$ denotes the cumulated inner product for search point i at iteration t and c $(0 < c < 1)$ is the accumulation coefficient. Such an inner product accumulation rule is similar to the cumulative step-size adaptation mechanism in the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) (Hansen and Ostermeier, 2001), where consecutive mutation steps are accumulated for step-size adaptation. Based on the cumulated inner product, a simple control rule is designed to adapt the step-size online:

$$\sigma_{t+1}^{(i)} = \begin{cases} \alpha\sigma_t^{(i)} & \text{if } p_t^{(i)} < 0, \\ \sigma_t^{(i)} & \text{if } p_t^{(i)} = 0, \\ \sigma_t^{(i)}/\alpha & \text{if } p_t^{(i)} > 0. \end{cases} \quad 0 < \alpha < 1. \tag{5.13}$$

where $\sigma_t^{(i)}$ is the individual step-size for search point i at iteration t. In this thesis, the settings of $c = 0.7, \alpha = 0.8$ are suggested by tuning the algorithmic performance on MPM2 functions.

The backtracking line search (Nocedal and Wright, 2000), which is a common technique to approximate the optimal step-size in single objective gradient ascent, is not suitable for the proposed algorithm. It requires additional function evaluations for each search point to estimate the optimal step-size setting. Such additional costs are no longer acceptable for the set-based algorithm. In contrast, the proposed cumulative step-size adaptation mechanism does not bring any additional overheads.

5.2.3 Hypervolume Indicator Gradient Ascent Algorithm

In this section, the algorithmic components developed in the previous sections are combined into the *Hypervolume Indicator Gradient Ascent Multi-objective Optimization* (HIGA-MO) algorithm.

In practice, the continuous objective function can be non-differentiable at some points, even if the function is almost everywhere differentiable (e.g., on the constraint boundary of the ZDT1 problem). To overcome this issue, it is suggested to *mutate* those points in the decision space. Given a point $\mathbf{x} \in \mathbb{R}^d$, it is mutated in the decision space S when the gradient of objective functions at $\mathbf{x}$ contains invalid values (e.g., the derivative becomes infinite when approaching the origin, on function $f = 1/x$). The mutation of $\mathbf{x}$ should be local but large enough to escape from the non-differentiable regions. For this purpose, the mutation operator in Differential Evolution (Storn and Price, 1997) is adopted here because it is adaptive and only contains a single parameter. Suppose $\mathbf{x}$ is in the ith ranked layer ($\mathbf{x} \in \mathbf{X}^i$), then the following mutation operation is applied on $\mathbf{x}$:

$$\mathbf{x} \leftarrow \mathbf{x} + F(\mathbf{x}^{(a)} - \mathbf{x}^{(b)}), \tag{5.14}$$

where $\mathbf{x}^{(a)}, \mathbf{x}^{(b)}$ are randomly picked from $\mathbf{X}^i$. Furthermore, $F \in [0, 2]$ is the differential weight that is set according to the literature. It is necessary to compute the differential vector within the same layer of $\mathbf{x}$ because the Pareto efficient set is possibly disconnected in the decision space and differential vectors computed across layers possibly create non-local mutations.

The resulting algorithm is presented in Alg. 10. In line 4, the non-dominated sorting procedure is called to partition the approximation set. In line 7 the hypervolume indicator gradient is computed for every decision vector on each layer. If a decision vector has either zero gradient or is not differentiable, it is

mutated in line 9 according to Eq. (5.14). In line 11, the hypervolume indicator sub-gradient is normalized before decision vectors are moved in the steepest ascent manner (line 12). The cumulative step-size adaptation (Eq. (5.12) and (5.13)) is then applied in line 13. In addition to the common usage of the function evaluation budget for the termination criterion, it is suggested here to check stationarity of search points: a decision vector is considered stationary if the norm of its

Algorithm 10 Hypervolume Indicator Gradient Ascent Multi-Objective Optimization

1: **procedure** HIGA-MO$(\mu, \mathrm{S}, \mathbf{f}, \nabla\mathbf{f})$ ▷ $\nabla\mathbf{f}$: Jacobian of the objective function
2: $c \leftarrow 0.7,\ \alpha \leftarrow 0.8,\ F \in [0, 2]$. ▷ endogenous parameters
3: Initialize μ search points $X = \left\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(\mu)}\right\} \subset \mathrm{S}$ uniformly.
4: **while** the termination criteria are not satisfied **do**
5: $Y \leftarrow \left\{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \ldots, \mathbf{y}^{(\mu)}\right\} \leftarrow \left\{\mathbf{f}(\mathbf{x}^{(1)}), \mathbf{f}(\mathbf{x}^{(2)}), \ldots, \mathbf{f}(\mathbf{x}^{(\mu)})\right\}$
6: $\left\{X^1, X^2, \ldots, X^q\right\} \leftarrow$ NON-DOMINATED-SORTING(X, Y)
7: **for** $k = 1$ to q **do**
8: **for** every $\mathbf{x}^{(i)}$ in X^k **do**
9: Compute the sub-gradient (Eq. (5.9)):

$$\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(i)}} \leftarrow \nabla\mathbf{f}(\mathbf{x}^{(i)}) \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{y}^{(i)}}$$

10: **if** $\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(i)}}$ is undefined **then**
11: Randomly pick $\mathbf{x}^{(a)} \neq \mathbf{x}^{(b)}$ from X^k
12: $\mathbf{x}^{(i)} \leftarrow \mathbf{x}^{(i)} + F(\mathbf{x}^{(a)} - \mathbf{x}^{(b)})$
13: **else**
14: $\mathbf{g}^{(i)} \leftarrow \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(i)}} / \left\| \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(i)}} \right\|$ ▷ sub-gradient normalization
15: $\mathbf{x}^{(i)} \leftarrow \mathbf{x}^{(i)} + \sigma^{(i)}\mathbf{g}^{(i)}$ ▷ gradient ascending
16: $p^{(i)} \leftarrow (1 - c)p^{(i)} + c\langle \mathbf{g}^{(i)}, \mathbf{g}^{(i)}_{\mathrm{old}} \rangle$ ▷ cumulation
17: $\sigma^{(i)}_{t+1} = \begin{cases} \alpha\sigma^{(i)}_t & \text{if } p^{(i)}_t < 0, \\ \sigma^{(i)}_t & \text{if } p^{(i)}_t = 0, \\ \sigma^{(i)}_t/\alpha & \text{if } p^{(i)}_t > 0. \end{cases}$ ▷ step-size control
18: $\mathbf{g}^{(i)}_{\mathrm{old}} \leftarrow \mathbf{g}^{(i)}$
19: **end if**
20: **end for**
21: **end for**
22: **end while**
23: **return** X, Y
24: **end procedure**

sub-gradient multiplied by the step-size is close to zero ($\leq 10^{-8}$).

5.2.4 Experiments

Experiment settings To test the performance of HIGA-MO, the well-known ZDT problems (Deb et al., 2000) are selected as benchmark problem set. The proposed algorithm is compared to three well-established evolutionary multi-objective optimization algorithm: NSGA-II, SPEA2 and SMS-EMOA. The parameters in those two algorithms are set according to the literature (Deb et al., 2000; Beume et al., 2007; Zitzler et al., 2001). In addition, other methods for steering the dominated point (Section 4), Lara's direction and Gap-filling, are tested against HIGA-MO. For these two methods, the non-dominated points are moved using the hypervolume indicator gradient.

The hypervolume indicator and convergence measure used in Beume et al. (2007), are adopted here as the performance metrics. The convergence measure is calculated numerically by discretizing the Pareto front into 1000 points. For the hypervolume indicator computation, the reference point $[11, 11]^\top$ is used for the test problems ZDT1 $-$ 4 and ZDT6. Two experiments are conducted: one with a relatively small population setting $\mu = 40$ while the other uses a large population, $\mu = 100$. A relatively small function evaluation budget, 100μ, is chosen here due to the reason that in long runs, all deterministic methods stagnate to local optima. All the algorithms terminate if the maximal function evaluation budget is reached. For each algorithm, 15 independent runs are conducted to obtain average performance measures. The initial step-size of the proposed HIGA-MO algorithm is set to 0.05 multiplied by the maximum range of the decision space. The internal reference point to compute the hypervolume indicator gradient is set to $[11, 11]^\top$ to ensure every objective vector is within the reference space.

Results The test results are shown in Tab. 5.1 for $\mu = 40$ and Tab. 5.2 for $\mu = 100$. The hypervolume of the non-dominated set after termination is used to compute the performance measures. For the small population setting, HIGA-MO outperforms the evolutionary algorithms (NSGA-II, SPEA2 and SMS-EMOA) on ZDT1-3 and ZDT6 problems, both in terms of hypervolume indicator and convergence measure. By checking the standard deviation, it is obvious that HIGA-MO generates more stable results compared to evolutionary algorithms and

such deviations are only affected by the initialization of the approximation set and the technique to handle the non-differentiable points (Eq. (5.14)). Comparing it to the other two methods, namely, Lara's direction and Gap-filling, that steer the dominated points independently, HIGA-MO gives a higher hypervolume indicator value on ZDT1-3 while Lara's method performs better on ZDT6.

Table 5.1: $\mu = 40$: performance measures on ZDT1-4 and ZDT6 problems.

Test-function	Algorithm	Convergence measure			Hypervolume indicator		
		Average	Std. dev.	Rank	Average	Std. dev.	Rank
ZDT1	HIGA-MO	**0.00500490**	1.3075e-02	1	**120.62948062**	4.0750e-03	1
	Lara's direction	0.07747718	6.4031e-02	3	120.33761711	1.2309e-01	2
	Gap-filling	0.06061863	1.2352e-01	2	120.22307239	4.6840e-01	3
	NSGA-II	0.10960371	3.2542e-02	5	119.33541376	3.7345e-01	4
	SMS-EMOA	0.09376444	3.5934e-02	4	119.20965862	4.8101e-01	5
	SPEA2	0.32006024	5.9788e-02	6	116.27370195	1.6826e+00	6
ZDT2	HIGA-MO	0.00036082	3.6233e-05	3	**120.31634691**	9.8307e-04	1
	Lara's direction	**0.00011253**	5.0289e-05	1	118.92812930	3.5019e+00	3
	Gap-filling	0.00015973	2.0645e-04	2	119.45871166	2.5324e+00	2
	NSGA-II	0.16511979	7.7092e-02	4	114.03423180	3.7806e+00	4
	SMS-EMOA	0.24929199	8.4178e-02	5	109.17629732	3.2584e+00	5
	SPEA2	0.67688451	1.5708e-01	6	104.54506810	3.3537e+00	6
ZDT3	HIGA-MO	0.00031903	5.0492e-05	2	128.55259300	7.9970e-01	2
	Lara's direction	**0.00028076**	5.0842e-05	1	125.78304061	3.5114e+00	6
	Gap-filling	0.00034568	5.4557e-05	3	**128.75911576**	9.2658e-03	1
	NSGA-II	0.00228282	5.9689e-03	4	126.56081625	2.8857e+00	3
	SMS-EMOA	0.00405046	5.7238e-03	5	125.88966563	2.9289e+00	5
	SPEA2	0.00635668	1.0852e-02	6	126.55026001	2.5895e+00	4
ZDT4	HIGA-MO	38.13060527	7.6780e+00	4	0.00000000	0.0000e+00	6
	Lara's direction	43.19742796	1.1544e+01	5	0.00000000	0.0000e+00	5
	Gap-filling	52.35972878	1.2465e+01	6	1.16325406	4.3525e+00	4
	NSGA-II	4.07411956	1.6869e+00	2	75.28344930	1.8038e+01	2
	SMS-EMOA	**3.52099683**	1.7386e+00	1	**78.04608227**	1.8555e+01	1
	SPEA2	11.17677922	4.9514e+00	3	19.34577362	2.2000e+01	3
ZDT6	HIGA-MO	3.83694298	1.3668e+00	6	113.28359226	1.3577e+00	2
	Lara's direction	**0.00010409**	4.3909e-05	1	**116.86127498**	1.6820e+00	1
	Gap-filling	3.02249489	2.7090e+00	5	106.81768735	2.0573e+01	3
	NSGA-II	1.28139859	3.0071e-01	2	97.53535725	3.8143e+00	4
	SMS-EMOA	1.36426329	3.1163e-01	3	96.84386232	4.2309e+00	5
	SPEA2	2.22799304	7.2398e-01	4	86.25780584	7.9570e+00	6

In terms of the convergence measure, Lara's direction always outperforms HIGA-MO on ZDT1-3 and 6. Lara's direction moves the dominated points toward the Pareto front without considering their distribution while HIGA-MO is designed to achieve both. Thus, HIGA-MO requires more efforts to approach the Pareto front than Lara's direction, in terms of the convergence measure. On ZDT4, which

has a highly multi-modal landscape, none of the gradient-based methods (HIGA-MO, Lara's direction and Gap-filling) achieves comparable results to evolutionary algorithms. The gradient-based methods easily stagnate in the local Pareto-front and fail to move towards the global one. For such a highly multi-modal optimization problem, a restart heuristic could improve the performance of gradient-based algorithms. For the large population setting, Tab. 5.2 shows roughly the same results for algorithm comparisons as for the small population setting.

Table 5.2: $\mu = 100$: performance measures on ZDT1-4 and ZDT6 problems.

Test-function	Algorithm	Convergence measure			Hypervolume indicator		
		Average	Std. dev.	Rank	Average	Std. dev.	Rank
ZDT1	HIGA-MO	**0.00031201**	4.1269e-05	1	**120.64580412**	1.7718e-03	1
	Lara's direction	0.02103585	4.7314e-02	5	120.48926778	5.2474e-02	2
	Gap-filling	0.02091304	6.1387e-02	4	120.42616648	2.7937e-01	5
	NSGA-II	0.01769266	4.6048e-03	3	120.45030137	4.5135e-02	4
	SMS-EMOA	0.01234011	2.6377e-03	2	120.48071780	3.6130e-02	3
	SPEA2	0.06017346	1.7966e-02	6	119.86686583	2.1615e-01	6
ZDT2	HIGA-MO	0.00028335	3.3303e-05	3	**120.31710222**	2.3560e-03	1
	Lara's direction	**0.00005498**	1.2085e-05	1	120.30338190	2.9998e-03	2
	Gap-filling	0.00007857	8.7094e-05	2	120.14758158	1.5778e-01	3
	NSGA-II	0.02834448	4.4153e-03	5	119.16220851	1.0985e+00	4
	SMS-EMOA	0.02338094	7.0938e-03	4	118.40070248	2.7352e+00	5
	SPEA2	0.08566545	4.8472e-02	6	114.48551919	4.4285e+00	6
ZDT3	HIGA-MO	0.00047505	7.5997e-05	3	128.77154126	8.5828e-03	3
	Lara' direction	0.00046485	5.9553e-05	2	128.77257561	5.2596e-03	2
	Gap-filling	**0.00039660**	4.9392e-05	1	128.77099724	3.3611e-03	4
	NSGA-II	0.00063823	5.1880e-05	5	**128.77436195**	1.1318e-03	1
	SMS-EMOA	0.00055256	3.5594e-05	4	128.34841609	1.0889e+00	6
	SPEA2	0.00243258	6.6391e-03	6	128.55447469	7.9741e-01	5
ZDT4	HIGA-MO	31.34155544	3.9090e+00	4	0.00000000	0.0000e+00	6
	Lara's direction	40.35930710	1.1041e+01	5	0.00000000	0.0000e+00	5
	Gap-filling	43.47103886	1.5933e+01	6	5.23444012	1.5425e+01	4
	NSGA-II	**0.80498648**	5.0038e-01	1	**109.60569075**	5.4368e+00	1
	SMS-EMOA	1.01209147	6.3095e-01	2	107.14186469	7.1460e+00	2
	SPEA2	2.80155378	1.3959e+00	3	83.82023960	1.5461e+01	3
ZDT6	HIGA-MO	3.54689504	1.2985e+00	5	113.79978098	8.8488e-01	2
	Lara's direction	**0.00004369**	1.2553e-05	1	**116.49314419**	1.4990e+00	1
	Gap-filling	4.12388484	2.9230e+00	6	86.58598768	3.4123e+01	6
	NSGA-II	0.43202530	7.1773e-02	3	109.28079070	1.2513e+00	4
	SMS-EMOA	0.40028650	1.1394e-01	2	109.87049482	1.8951e+00	3
	SPEA2	0.49692387	1.2882e-01	4	108.17997611	1.9177e+00	5

As shown in the experimental results on ZDT4, the proposed algorithm fails to approach the global Pareto front and gets stuck in local ones instead. In practice, such an issue can be tackled by using restart heuristics to re-sample the stagnated

points. In addition, it is possible to hybridize HIGA-MO with an evolutionary multi-objective (EMO) algorithm, where the global search ability of an EMO helps the algorithm to escape from a deceptive, local Pareto front and HIGA-MO could achieve fast convergence speed when approaching the global Pareto front. Such an approach has been proposed in López et al. (2012) and the optimal way to combine HIGA-MO with EMOs should be investigated.

The experiments conducted in this thesis are on a small number of problems. In future research, the proposed algorithm should be investigated on more multi-objective problems. When using a large number of search points, the objective vectors on the Pareto front are close to each other, which might result in relatively slow movement. In this case, its performance needs to be further tested. In addition, it is of interest to compare HIGA-MO empirically to other set-based scalarization method (Schütze et al., 2016).

5.3 Hypervolume Indicator Hessian

In this section, we first derive the Hessian matrix of the hypervolume indicator for the general *multi-objective* optimization scenario. The Hessian matrix in *bi-objective* cases is treated in Section 5.3.1. For conciseness, matrix calculus notations are used in the following derivation, which helps to understand the structure of the Hessian matrix. The hypervolume Hessian matrix is the "Jacobian" of the hypervolume gradient defined as follows:

$$\begin{aligned}
\nabla^2 \mathcal{H}_{\mathbf{F}}(\mathbf{X}) &= \frac{\partial}{\partial \mathbf{X}} \left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{X}} \right) \qquad (5.15) \\
&= \left(\underbrace{\frac{\partial}{\partial \mathbf{X}} \left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(1)}} \right)}_{\mu \cdot d \times d}, \ldots, \frac{\partial}{\partial \mathbf{X}} \left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(\mu)}} \right) \right) \\
&= \begin{pmatrix} \frac{\partial}{\partial \mathbf{x}^{(1)}} \left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(1)}} \right) & \cdots & \frac{\partial}{\partial \mathbf{x}^{(1)}} \left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(\mu)}} \right) \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial \mathbf{x}^{(\mu)}} \left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(1)}} \right) & \cdots & \frac{\partial}{\partial \mathbf{x}^{(\mu)}} \left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(\mu)}} \right) \end{pmatrix},
\end{aligned}$$

where each *sub-gradient* is differentiated with respect to $\mathbf{X}$. This results in μ^2 block partitions $(d \times d)$ of the Hessian matrix. The (i, j)-block matrix can be

further expressed as follows:

$$\begin{aligned}
&\frac{\partial}{\partial \mathbf{x}^{(i)}}\left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{x}^{(j)}}\right) = \frac{\partial}{\partial \mathbf{x}^{(i)}}\left(\frac{\partial \mathbf{y}^{(j)}}{\partial \mathbf{x}^{(j)}}\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial \mathbf{y}^{(j)}}\right)\\
&= \sum_{k=1}^{m}\frac{\partial}{\partial \mathbf{x}^{(i)}}\left(\frac{\partial f_k(\mathbf{x}^{(j)})}{\partial \mathbf{x}^{(j)}}\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k(\mathbf{x}^{(j)})}\right)\\
&= \underbrace{\sum_{k=1}^{m}\frac{\partial}{\partial \mathbf{x}^{(i)}}\left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k(\mathbf{x}^{(j)})}\right)\nabla f_k(\mathbf{x}^{(j)})^{\top}}_{\mathbf{A}_{ij}} + \underbrace{\sum_{k=1}^{m}\frac{\partial^2 f_k(\mathbf{x}^{(j)})}{\partial \mathbf{x}^{(i)}\partial \mathbf{x}^{(j)}}\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k(\mathbf{x}^{(j)})}}_{\mathbf{B}_{ij}}.
\end{aligned} \tag{5.16}$$

According to the differentiation above, each (i,j)-block matrix is a combination of two components: $\mathbf{A}_{ij}$ and $\mathbf{B}_{ij}$. Note that matrix $\mathbf{A}_{ij}$, $\frac{\partial}{\partial \mathbf{x}^{(i)}}\left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k(\mathbf{x}^{(j)})}\right)$ is a column vector of size n and stands for the sub-gradient of $\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k(\mathbf{x}^{(j)})}$ at $\mathbf{x}^{(j)}$. In the following, we abbreviate $f_k(\mathbf{x}^{(i)})$ as $f_k^{(i)}$ and its gradient $\nabla f_k(\mathbf{x}^{(i)})$ as $\nabla f_k^{(i)}$.

The first component: $\mathbf{A}_{ij}$ Due the fact that the matrix $\mathbf{a}_{ij}$ is a sum of m outer products, this term has *at most* rank m. It is possible to make $\mathbf{A}_{ij}$ to have full rank only if $m \geq n$. In other cases, $\mathbf{A}_{ij}$ is always rank deficient ($rank(\mathbf{A}_{ij}) \leq m < d$). This indicates that in the "usual" multi-objective optimization case, where the number of objective functions is smaller than the number of decision variables, such a matrix $\mathbf{A}_{ij}$ is *always singular*.

In the following lemma, a detailed expression of $\mathbf{A}_{ij}$ is given for the bi-objective case ($m = 2$). Without loss of generality, we assume that the objective vectors (and corresponding decision vectors) are arranged according to the ascending order of the first objective values.

Lemma 5.1. *Let $m = 2$, $i = 1, \ldots, \mu$ and $j = 1, \ldots, \mu$. Assume that all vectors $\mathbf{x}^{(i)}$ are mutually non-dominated, then the first component $\mathbf{A}_{ij}$ is non-zero only if the block matrix is located on the main diagonal ($i = j$) or the first diagonal above/below the main diagonal ($|i - j| = 1$), and it can be written as:*

$$\mathbf{A}_{ij} = \begin{cases} \nabla f_2^{(j)}\nabla f_1^{(j)\top} + \nabla f_1^{(j)}\nabla f_2^{(j)\top} & \text{if } i = j \\ -\nabla f_1^{(j+1)}\nabla f_2^{(j)\top} & \text{if } i = j+1 \\ -\nabla f_2^{(j-1)}\nabla f_1^{(j)\top} & \text{if } i = j-1 \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{5.17}$$

Proof. Assume a fixed reference point $\mathbf{r} = (r_1, r_2)^{\top}$. To simplify the formulation, we denote $f_1^{(\mu+1)} := r_1$ and $f_2^{(0)} := r_2$. The partial derivative of the hypervolume

indicator w.r.t. the objective value is derived in Emmerich and Deutz (2014), which corresponds to the length of the steps of the attainment curve:

$$\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_1^{(j)}} = f_2^{(j)} - f_2^{(j-1)}, \ \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_2^{(j)}} = f_1^{(j)} - f_1^{(j+1)}. \tag{5.18}$$

It is clear that $\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_1^{(j)}}$ is a function of only $\mathbf{x}^{(j)}$ and $\mathbf{x}^{(j-1)}$ (similar argument holds for $\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_2^{(j)}}$). The gradient of the partial derivatives can be given, for example: $\frac{\partial}{\partial \mathbf{x}^{(j)}}\left(\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k^{(j)}}\right) = \nabla f_2^{(j)}$. Such a gradient is nonzero for at least one objective function, when $i = j$, $i = j+1$ or $i = j-1$. By substituting the required gradients into Eq. (5.16), the expression of $\mathbf{A}_{ij}$ can be obtained. □

The second component: $\mathbf{B}_{ij}$ $\mathbf{B}_{ij}$ is a weighted sum of second order derivatives of the objective functions, where the weights are partial derivatives of the hypervolume indicator at each objective value (cf. Eq. (5.10)). Note that the second order derivative $\frac{\partial^2 f_k^{(j)}}{\partial \mathbf{x}^{(i)} \partial \mathbf{x}^{(j)}}$ is not zero if and only if $i = j$:

$$\mathbf{H}_k^{(j)} := \frac{\partial^2 f_k^{(j)}}{\partial {\mathbf{x}^{(j)}}^2},$$

is the Hessian matrix of objective function f_k at point $\mathbf{x}^{(j)}$. Consequently, matrix $\mathbf{B}_{ij}$ can be written as:

$$\mathbf{B}_{ij} = \begin{cases} \sum_{k=1}^{m} \frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k^{(j)}} \mathbf{H}_k^{(j)} & \text{if } i = j \\ \mathbf{0} & \text{if } i \neq j. \end{cases} \tag{5.19}$$

Note that $\frac{\partial \mathcal{H}_{\mathbf{F}}(\mathbf{X})}{\partial f_k^{(j)}}$ can be obtained from Eq. (5.18). The singularity of matrix $\mathbf{B}_{ij}$ depends on the properties of the Hessian matrices of the objective functions. Under the assumption that all objective functions are convex (the objective-wise Hessian matrices are positive-definite), matrix $\mathbf{B}_{ij}$ is also positive-definite, under the condition that all objective functions are subject to maximization (for minimization, $\mathbf{B}_{ij}$ is negative-definite). In general, if each objective function has non-singular Hessian matrix almost everywhere, it is obvious that the matrix $\mathbf{B}_{ij}$ is non-singular.

5.3.1 The Bi-objective Case

For a bi-objective optimization problem, the hypervolume Hessian matrix has the following structure:

$$\nabla^2 \mathcal{H}_{\mathbf{F}}(\mathbf{X}) = \begin{pmatrix} \mathbf{D}_1 & \tilde{\mathbf{A}}_1 & & & \\ \tilde{\mathbf{A}}_1^\top & \mathbf{D}_2 & \tilde{\mathbf{A}}_2 & & \\ & \tilde{\mathbf{A}}_2^\top & \ddots & \ddots & \\ & & \ddots & \ddots & \tilde{\mathbf{A}}_{\mu-1} \\ & & & \tilde{\mathbf{A}}_{\mu-1}^\top & \mathbf{D}_\mu \end{pmatrix},$$

where $\mathbf{D}_i = \mathbf{A}_{ii} + \mathbf{B}_{ii}$ and $\tilde{\mathbf{A}}_i = \mathbf{A}_{i(i+1)} = -\nabla f_2^{(i)} \nabla f_1^{(i+1)}{}^\top$ according to Eq. (5.17). Note that the Hessian matrix $\nabla^2 \mathcal{H}_{\mathbf{F}}(\mathbf{X})$ is a *tridiagonal block* matrix. It is important to investigate when the diagonal block matrix is singular. Due to the difficulty of the investigation, we start to discuss the invertibility of the Hessian matrix in two special cases: single-point system, where only a single decision vector is moved and two-point system where the interactions between two points need to be considered.

One-point system In this case, the hypervolume Hessian matrix degenerates to the diagonal block matrix $\mathbf{D}_i$, that can be expressed using Eq. (5.19):[1]

$$\mathbf{D}_i = \underbrace{\left(f_2^{(i)} - f_2^{(i-1)}\right) \mathbf{H}_1^{(i)} + \left(f_1^{(i)} - f_1^{(i+1)}\right) \mathbf{H}_2^{(i)}}_{\mathbf{B}_{ii}} + \underbrace{\nabla f_2^{(i)} \nabla f_1^{(i)}{}^\top + \nabla f_1^{(i)} \nabla f_2^{(i)}{}^\top}_{\mathbf{A}_{ii}}.$$

To investigate the invertibility of such a matrix, we assume that each objective function is **convex**, in addition to the differentiability assumption.

Theorem 5.1. *If the decision vector belongs to the efficient set, $\mathbf{x}^{(i)} \in P_{\mathcal{X}}$ and its two neighbors $\mathbf{x}^{(i-1)}$ and $\mathbf{x}^{(i+1)}$ are not weakly dominated simultaneously, then the diagonal block matrix $\mathbf{D}_i$ is non-singular and negative definite.*

Proof. Note the following facts: 1) Due the assumption that both objective functions are convex, the objective-wise Hessian matrices $\mathbf{H}_1^{(i)}$ and $\mathbf{H}_2^{(i)}$ are positive definite. 2) Since the minimization task is assumed and $\mathbf{x}^{(i-1)}, \mathbf{x}^{(i+1)}$ are not weakly dominated simultaneously, the coefficients $\left(f_2^{(i)} - f_2^{(i-1)}\right)$ and $\left(f_1^{(i)} - f_1^{(i+1)}\right)$ are non-positive but do not take zero value at the same time (Emmerich and Deutz,

[1] As only a single decision vector is considered here, the script index i can be removed. We still keep it because the discussion of invertibility here holds for every diagonal block matrix.

2014). 3) If $\mathbf{x}^{(i)} \in P_{\mathcal{X}}$, then the two objective-wise gradients are anti-parallel to each other, namely $\exists \beta > 0,\ \nabla f_1^{(i)} = -\beta \nabla f_2^{(i)}$, due the Karush-Kuhn-Tucker theorem (Ehrgott, 2006). Then, $\forall \mathbf{y} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$, the quadratic form associated with $\mathbf{D}_i$ is:

$$\mathbf{y}^\top \mathbf{D}_i \mathbf{y} = \left(f_2^{(i)} - f_2^{(i-1)}\right) \mathbf{y}^\top \mathbf{H}_1^{(i)} \mathbf{y} + \left(f_1^{(i)} - f_1^{(i+1)}\right) \mathbf{y}^\top \mathbf{H}_2^{(i)} \mathbf{y} - 2\beta \left(\mathbf{y}^\top \nabla f_2^{(i)}\right)^2.$$

Each term on the right-hand-side of the equation above is negative according to the facts listed above and therefore their sum, $\mathbf{y}^\top \mathbf{D}_i \mathbf{y} < 0$. Consequently, $\mathbf{D}_i$ is negative definite and thus non-singular. □

Theorem 5.2. *If the objective functions are convex and the decision vector $\mathbf{x}^{(i)}$ does not belong to the efficient set $\mathcal{X}$, the matrix $\mathbf{D}_i$ is non-singular if and only if the following condition holds:*

$$\left(\nabla f_2^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_1^{(i)} + 1\right)^2 \neq \left(\nabla f_2^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_2^{(i)}\right) \left(\nabla f_1^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_1^{(i)}\right).$$

Proof. We introduce the following notations:

$$\mathbf{P}_i := \left(\nabla f_1^{(i)}, \nabla f_2^{(i)}\right), \quad \mathbf{Q}_i := \begin{pmatrix} \nabla f_2^{(i)\top} \\ \nabla f_1^{(i)\top} \end{pmatrix}.$$

Then $\mathbf{D}_i$ can be re-written as: $\mathbf{D}_i = \mathbf{B}_{ii} + \mathbf{P}_i \mathbf{Q}_i$. Note that $\mathbf{B}_{ii} = \left(f_2^{(i)} - f_2^{(i-1)}\right) \mathbf{H}_1^{(i)} + \left(f_1^{(i)} - f_1^{(i+1)}\right) \mathbf{H}_2^{(i)}$ is a combination of objective-wise Hessian matrices. Since both of the objective function are *convex*, $\mathbf{H}_1^{(i)}$ and $\mathbf{H}_2^{(i)}$ are positive definite. In addition, the coefficients $\left(f_2^{(i)} - f_2^{(i-1)}\right)$ and $\left(f_1^{(i)} - f_1^{(i+1)}\right)$ are negative in case of minimization. Consequently, $\mathbf{B}_{ii}$ is negative definite and thus *non-singular*. According to the matrix inversion lemma (Woodbury matrix identity), $\mathbf{C}_i$ is invertible if and only if $\mathbf{T}_i = \mathbf{I}_{2\times 2} + \mathbf{Q}_i \mathbf{B}_{ii}^{-1} \mathbf{P}_i$ is invertible:

$$\mathbf{T}_i = \begin{pmatrix} \nabla f_2^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_1^{(i)} + 1 & \nabla f_2^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_2^{(i)} \\ \nabla f_1^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_1^{(i)} & \nabla f_1^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_2^{(i)} + 1 \end{pmatrix}.$$

As matrix $\mathbf{T}_i$ is always of size 2×2, its determinant is much easier to compute than that of $\mathbf{C}_i$:

$$\det(\mathbf{T}_i) = \left(\nabla f_2^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_1^{(i)} + 1\right)^2 - \left(\nabla f_2^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_2^{(i)}\right) \left(\nabla f_1^{(i)\top} \mathbf{B}_{ii}^{-1} \nabla f_1^{(i)}\right).$$

The matrix $\mathbf{D}_i$ is non-singular if and only if the determinant above is non-zero. □

Note that the analysis of the convergence of a single point to the maximal hypervolume is structurally similar to the maximization of the hypervolume contribution of a single point in a set. The only difference is that the reference point is provided by coordinates of the neighboring non-dominated points in the objective space. Therefore, Theorem 5.2 also holds for maximizing the hypervolume contribution of a single point, as long as the neighboring points in the objective space are kept fixed.

Two-point system In this case, the hypervolume Hessian matrix looks as follows:

$$\nabla^2 \mathcal{H}_{\mathbf{F}}(\mathbf{X}) = \begin{pmatrix} \mathbf{D}_1 & \tilde{\mathbf{A}}_1 \\ \tilde{\mathbf{A}}_1^\top & \mathbf{D}_2 \end{pmatrix}.$$

Again, we assume that two decision points $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ are mutually non-dominated. For this 2×2 block matrix, its invertibility is given by the following theorem.

Theorem 5.3. *If the diagonal block matrices $\mathbf{D}_1, \mathbf{D}_2$ are non-singular, then the hypervolume indicator Hessian matrix is non-singular if and only if:*

$$\left(\nabla f_1^{(1)}{}^\top \mathbf{D}_1^{-1} \nabla f_1^{(1)}\right)\left(\nabla f_2^{(2)}{}^\top \mathbf{D}_2^{-1} \nabla f_2^{(2)}\right) \neq 1.$$

Proof. See appendix B.1. □

Note that the non-singularity condition above does not hold even when both of the search points are in the efficient set. Such a situation can be depicted using a simple bi-objective optimization problem with $d = 1$ and $m = 2$:

$$f_1 = x^2, \quad f_2 = (1-x)^2, \quad x \in \mathbb{R}.$$

To illustrate the singularity scenario, only two decision points are used, namely $x^{(1)}$ and $x^{(2)}$. For such a problem, the efficient set is the interval $[0, 1]$ and thus the box $[0, 1]^2$ is the region where $x^{(1)}$ and $x^{(2)}$ are both efficient. In Fig. 5.4, the set where the Hessian matrix is singular is depicted by the curved boundary of the shaded area. As shown in this example, in the two-point system, the Hessian matrix is not always invertible, even if all the search points belong to the Pareto efficient set. Moreover, in the shaded area (in $[0, 1]^2$), the hypervolume Hessian matrix is even *indefinite*, which would make it more difficult for Newton method to converge to the optimum.

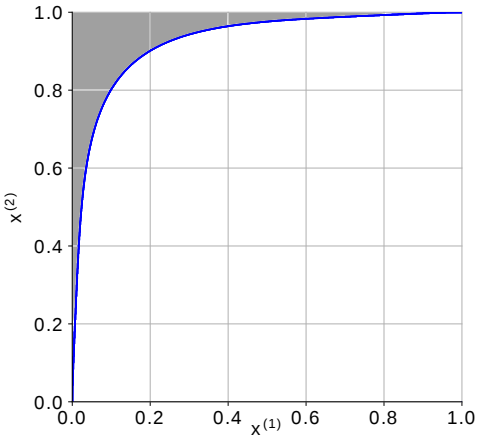

Figure 5.4: An example of two-point case in bi-objective optimization problem: The box $[0,1]^2$ is the region where $x^{(1)}, x^{(2)}$ are both Pareto efficient. The hypervolume Hessian matrix is singular when $(x^{(1)}, x^{(2)})$ is on the blue curve.

Remark. As illustrated in this example, the hypervolume Hessian matrix is only singular on a set of zero measure and therefore the applicability of the Newton method is not affected by the singularity because the probability of entering into such a set is zero. However, in general additional caution is needed if the set where the hypervolume Hessian is singular has nonzero measure, or in case there are regions where the Hessian is indefinite (which happens in our example).

5.3.2 Hypervolume Indicator Newton Method

After having stated the hypervolume gradient and Hessian matrix for a $\mu \cdot n$-vector $\mathbf{X}$ for a given MOP, we are now in the position to address the population based Newton method for hypervolume maximization. For this, we will first consider the unconstrained case and later on discuss first attempts to treat constrained problems. Given an unconstrained MOP and a population of μ points, the Newton step (or Newton function) is defined as follows:

$$\Delta\mathbf{X} := -\sigma \left[\nabla^2 \mathcal{H}_{\mathbf{F}}(\mathbf{X})\right]^{-1} \nabla \mathcal{H}_{\mathbf{F}}(\mathbf{X}). \tag{5.20}$$

In practice, a small step size $\sigma \in (0,1]$ is introduced to ensure the so-called Wolfe conditions (Wright and Nocedal, 1999) are met after each Newton step. As with the treatment in Section 5.2, the Newton step for decision point $\mathbf{x}^{(i)}$ is denoted by $\Delta\mathbf{X}^{(i)} \in \mathbb{R}^d, i = 1, \ldots, \mu$. Since the hypervolume indicator sub-gradient (Eq. (5.9))

for the *strictly dominated point* $\mathbf{x}^{(i)}$ of $\mathbf{X}$ is zero, its corresponding Newton direction is also zero. Consequently, such a point will remain stationary when applying the set-based Newton method. For the sake of simplicity, it is assumed that all the points contained in $\mathbf{X}$ are *mutually non-dominated* by each other. In case any point is dominated, it is always possible to apply the non-dominated-sorting approach proposed in Section 5.2.1. The **Hypervolume Newton Method** (HNM) is thus defined as

$$\mathbf{X}_0 \in \mathbb{R}^{\mu \cdot d}, \quad \mathbf{X}_{i+1} = \mathbf{X}_i + \Delta\mathbf{X}, \quad i = 0, 1, 2, \ldots. \tag{5.21}$$

The pseudo code for HNM is shown in Alg. 11. For the step size control, we suggest to choose the initial step size $\sigma_0 = 1$ and adjust it online using the backtracking line search. If automatic differentiation is used to evaluate the (exact) hypervolume indicator gradient and the Hessian matrix at each iteration, the cost for each Newton step is given by $5\mu + (4 + 6d)\mu$ function evaluations.

Algorithm 11 Hypervolume Newton Method

```
procedure HNM(X, N, ε)          ▷ X: initial approximation set, N: maximal
iteration, ε: tolerance on the length of hypervolume gradient
    for i = 1 → N do
        Compute ∇H_F(X), ∇²H_F(X)
        Compute step size σ by backtracking line search
        X ← X − σ [∇²H_F(X)]^{−1} ∇H_F(X)                ▷ Newton step
        if ‖∇H_F(X)‖ < ε then
            return X
        end if
    end for
    return X
end procedure
```

Example. In order to demonstrate the performance of the HNM we consider the following bi-objective optimization problem (also known as the MOP1 problem):

$$\begin{aligned} f_1 &= (x_1 - 1)^2 + (x_2 - 1)^2 \\ f_2 &= (x_1 + 1)^2 + (x_2 + 1)^2, \end{aligned} \tag{5.22}$$

where we choose as reference point $\mathbf{r} = (20, 20)^\top$.

(a) We choose $\mu = 5$ and the initial approximation set X as

$$\begin{aligned}&\left\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \mathbf{x}^{(3)}, \mathbf{x}^{(4)}, \mathbf{x}^{(5)}\right\}\\ &= \left\{\begin{pmatrix}0\\-2\end{pmatrix}, \begin{pmatrix}0.5\\-1.5\end{pmatrix}, \begin{pmatrix}1\\-1\end{pmatrix}, \begin{pmatrix}1.5\\-0.5\end{pmatrix}, \begin{pmatrix}2\\2\end{pmatrix}\right\}.\end{aligned} \tag{5.23}$$

Fig. 5.5 shows the performance of HNM both in the decision and objective space. As it can be seen, the iterations quickly approach the optimal solution for $\mu = 5$ and a given reference point. This observation is confirmed in Tab. 5.3a, where the hypervolume values, the norm of the gradients, and the error measured in terms of the Hausdorff distance (Schütze et al., 2012) of $\mathbf{X}$ and the optimal solution are displayed for each iteration. The values indicate quadratic convergence.

(b) Next, we consider the same setting but using a different initial approximation set:

$$\begin{aligned}&\left\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \mathbf{x}^{(3)}, \mathbf{x}^{(4)}, \mathbf{x}^{(5)}\right\}\\ &= \left\{\begin{pmatrix}-0.12\\-1.57\end{pmatrix}, \begin{pmatrix}0.48\\-1.24\end{pmatrix}, \begin{pmatrix}1\\-1\end{pmatrix}, \begin{pmatrix}1.32\\-0.26\end{pmatrix}, \begin{pmatrix}1.89\\-0.11\end{pmatrix}\right\}.\end{aligned} \tag{5.24}$$

Fig. 5.6 and Tab. 5.3b show the numerical results of HNM. In step 2, $\mathbf{x}^{(1)}$ gets dominated by $\mathbf{x}^{(3)}$. The iteration thus continues with the remaining 4 point excluding $\mathbf{x}^{(1)}$. HNM converges (again quadratically) toward the optimal hypervolume population, albeit for population size $\mu = 4$.

5.4 Summary

The multi-objective optimization problem is investigated in this chapter. The general goal here is to generalize the first- and second-order optimization method from the single-objective scenario to the multi-objective scenario. In order to achieve this goal, the notion on the gradient is extended to the multi-objective problem: the partial derivatives of the hypervolume indicator is taken w.r.t. to the decision points and the so-called hypervolume indicator gradient is defined as the concatenation of such partial derivatives at all decision points. Based on this extension, a gradient ascent algorithm, called hypervolume indicator gradient ascent multi-objective optimization (HIGA-MO) is proposed to maximize the hypervolume in the steepest manner. Following this treatment, the second-order

Table 5.3: On the MOP1 Problem (Eq. (5.22)), Alg. 11 is executed for seven iterations and the following values are listed iteratively: the hypervolume value, the size of the approximation set μ, the error in the Hausdorff distance to the optimal approximation set and the norm of the hypervolume indicator gradient.

(a) Using Eq. (5.23) for $\mathbf{X}_0$ (cf. Fig. 5.5).

Iter	μ	$\mathcal{H}_{\mathbf{F}}$	Error	$\lVert\nabla\mathcal{H}_{\mathbf{F}}\rVert$
0	5	306.5000	76.5695	48.8262
1	5	369.5622	13.5072	21.0628
2	5	379.0652	4.0042	13.9973
3	5	382.7340	0.3355	2.8800
4	5	383.0680	0.0015	0.2000
5	5	383.0695	0.0000	0.0013
6	5	383.0695	0.0000	0.0000
7	5	383.0695	0.0000	0.0000

(b) Using Eq. (5.24) for $\mathbf{X}_0$ (cf. Fig. 5.6)

Iter	μ	$\mathcal{H}_{\mathbf{F}}$	Error	$\lVert\nabla\mathcal{H}_{\mathbf{F}}\rVert$
0	5	321.5483	61.5212	52.9006
1	5	376.6161	6.4534	14.6855
2	4	373.5446	9.5249	2.0132
3	4	380.6982	2.3713	0.1104
4	4	380.6985	2.3710	0.0002
5	4	380.6985	2.3710	0.0000
6	4	380.6985	2.3710	0.0000
7	4	380.6985	2.3710	0.0000

derivatives of the hypervolume indicator is formulated. In addition, we investigate the condition on which the resulting Hessian matrix is regular (non-singular). This is an essential prerequisite for using the hypervolume indicator Hessian matrix correctly. In addition, to investigate the proposed algorithms, a bi-objective problem class, called Mixed-Peak problems are introduced. This problem class allows for directly controlling the problem difficulty of its instance.

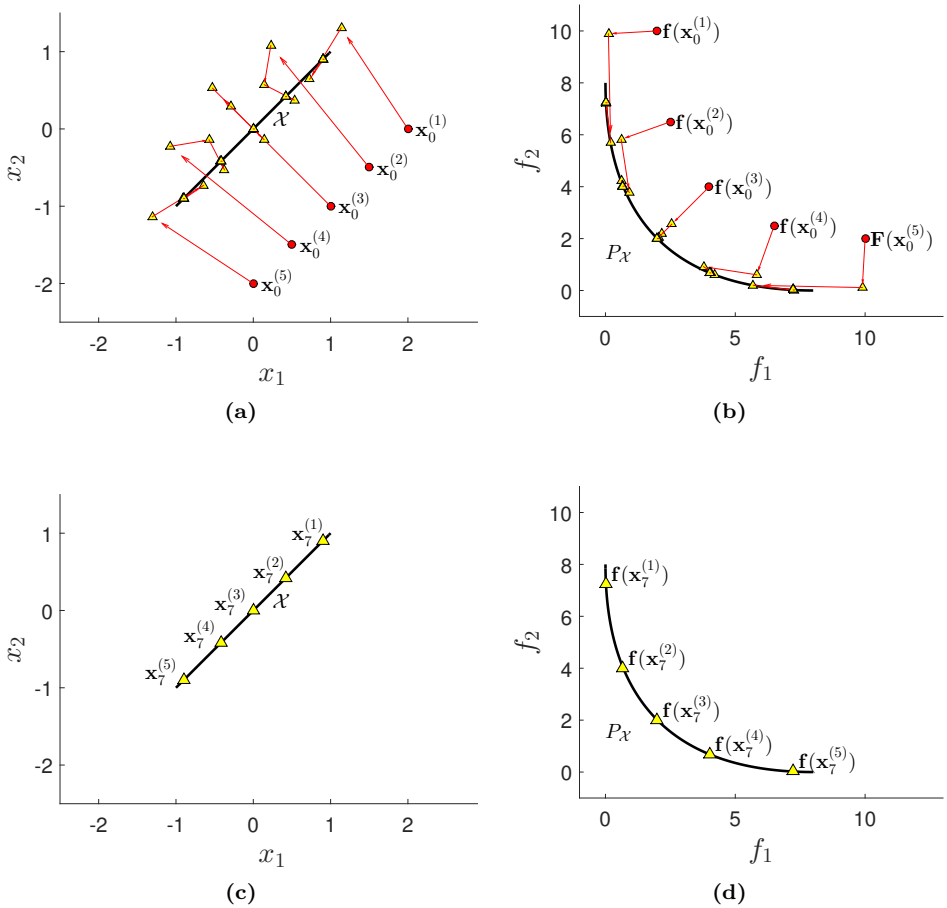

Figure 5.5: Numerical result of HNM on the MOP1 problem using Eq. (5.23) as the initial approximation set. **Top**: the iterations in decision and objective space. **Bottom**: the optimal solution and its image for $\mu = 5$ and $\mathbf{r} = (20, 20)^\top$.

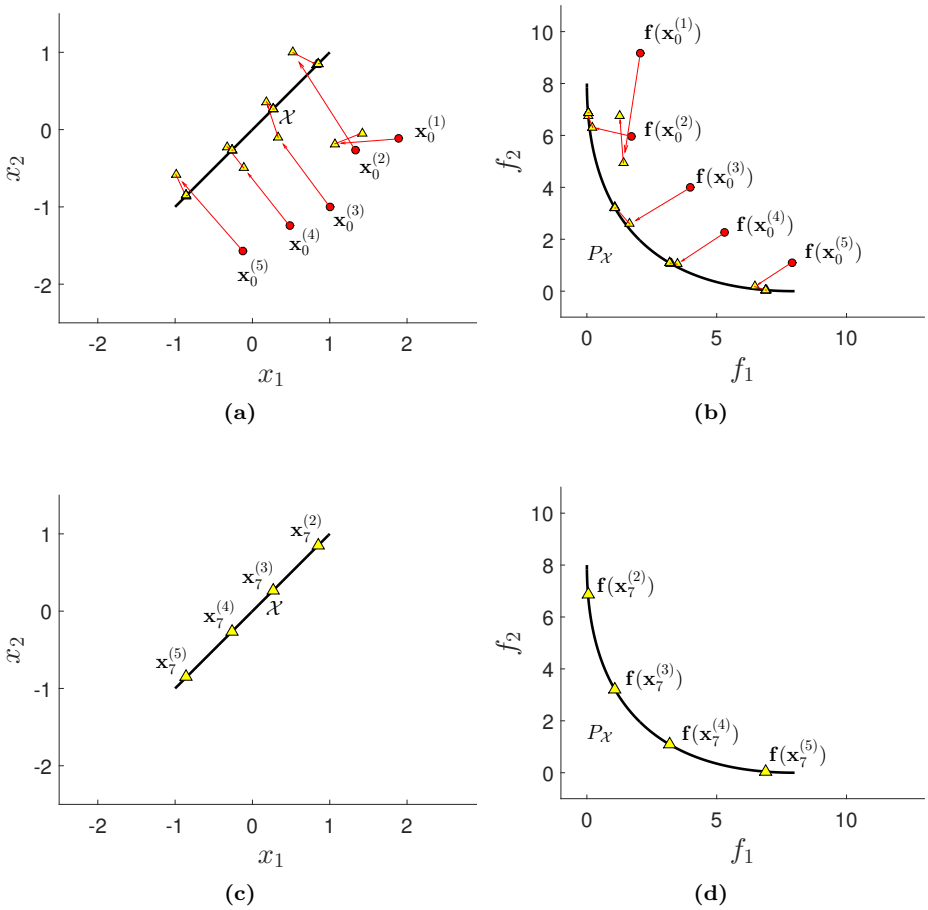

Figure 5.6: Numerical result of HNM on the MOP1 problem using Eq. (5.24) as the initial approximation set. **Top**: the iterations in decision and objective space. **Bottom**: the optimal solution and its image for $\mu = 4$ and $\mathbf{r} = (20, 20)^\top$.