Petri Eskelinen

Objective trade-off rate information in interactive multiobjective optimization methods

A survey of theory and applications

HELSINGIN KAUPPAKORKEAKOULU
HELSINKI SCHOOL OF ECONOMICS

W-445

Petri Eskelinen

Objective trade-off rate information in interactive multiobjective optimization methods

A survey of theory and applications

Quantitative Methods of Economics and
Management Science

February
2008

HELSINGIN KAUPPAKORKEAKOULU
HELSINKI SCHOOL OF ECONOMICS
WORKING PAPERS
W-445

HELSINGIN KAUPPAKORKEAKOULU
HELSINKI SCHOOL OF ECONOMICS
PL 1210
FI-00101 HELSINKI
FINLAND

ISSN 1235-5674
(Electronic working paper)
ISBN 978-952-488-220-0

Helsinki School of Economics -
HSE Print 2008

Objective Trade-off Rate Information in Interactive Multiobjective Optimization Methods: A Survey of Theory and Applications

Petri Eskelinen

Helsinki School of Economics
P.O. Box 1210
FI-00101 Helsinki, Finland
e-mail: petri.eskelinen@hse.fi

Abstract

In this paper we survey interactive multiobjective optimization methods that utilize, in a way or another, so called objective trade-off rate information. Objective trade-off information can be obtained by studying only the properties of the underlying multiobjective optimization problem. In the context of multiobjective methodology trade-off information can be useful because it can be used to indicate relative changes in values of other objectives when the value of some objective is altered from its current value. This kind of information is especially useful when supporting a decision maker. We analyze interactive methods that utilize the trade-off information and discuss some main benefits and drawbacks that can be related to trade-offs in general. As a conclusion we can see that in interactive methods objective trade-off information can be utilized at very different levels ranging from learning to decision making.

1 Introduction

Multiobjective optimization procedures deal with optimization problems having several conflicting objectives defined on a set of feasible decision variables. Typically, when dealing with optimization problems we are interested in determining one such decision variable allocation which in some sense produces an optimal solution for a given problem. However, when we consider multiple conflicting objectives one decision variable vector cannot produce simultaneously an optimal value for every objective at the same time and therefore an additional problem is how to determine a so called best compromise solution. In a general case such a best compromise solution cannot be determined in a satisfying way purely by mathematical means and typically a human decision maker (DM), who can be considered as an expert of an underlying problem, is used to determine subjectively this best compromise. The task of the DM is to find a balance (in a subjective sense) between conflicting objectives, and this can be obtained by finding favorable trade-off between objectives.

When we are dealing with multiobjective problems containing continuous variables and functions the set of potential compromise solutions where to choose from is typically infinite. Especially in the case where we have more than two conflicting objectives approaches such as visualization or pairwise comparison of all potential solutions are not necessarily efficient or possible methods to determine the most preferred solution. This kind of problem make worthwhile to consider interactive solution methods where the DM is acting in collaboration with the method and directs the search procedure to regions that are interesting while trying to avoid uninteresting regions. Since the 1970's several interactive methods for linear and nonlinear multiobjective optimization problems have been developed (see, e.g., [4], [17], [24], [26]).

In a typical interactive method the DM considers at each iteration a solution candidate (or a couple of candidates) and states preferences related to this solution(s). The underlying solution method then utilizes the preference information to generate a new solution candidate at the next iteration. This procedure is continued until the DM is satisfied with some of the obtained solutions. The main differences between interactive methods are related to the ways how and in what form they obtain preference information from the DM and how this information is utilized in the method to produce the next solution(s). In an ideal situation while using an interactive method the DM learns about her/his own preferences and at the same time also learns to know what kind of solutions are attainable. Based on what has been learned it is assumed that the DM is able to direct the method toward the subjectively best compromise solution.

A very important aspect related to interactive methods is how the DM is supported during the interaction. At each iteration some kind of information is shown to the DM who is then supposed to offer preference information related to the solution in such a way that at the next iteration a new more interesting solutions can be generated. Here, for instance, information obtained during the current and the previous iterations can be used to aid the DM to state preferences according to which the next solution is produced. An alternative possibility is to try to approximate what kind of solutions are available if certain kind of preference information is given. In other words, instead of history information we try to forecast what kind of solutions might be available if certain type of preference information is given at the current solution. Trade-off rate information can be seen as this kind of forecasting information because it can be used to indicate relative changes in the values of other objectives when the value of some objective is altered from its current value at the current solution.

From the decision support perspective trade-off rates are used, for instance, to approximate how values of objectives are going to change if certain type of preference information is given (e.g. STOM method [21], and ISTM method [31]). On the other hand, the trade-off information can be used to study whether there locally exists such a solution which is corresponding to the preferences given by the DM (e.g. SPOT method [22], ISWT method [3]), or what will happen for the objective function values if we move from one decision variable vector to another (e.g. ZW method [36]).

Trade-off information can be used in several different ways and our aim in this paper is to survey some central ways how the trade-off information has so far been utilized in the interactive multiobjective optimization methods. We restrict our study to the concept of so called objective trade-off information that can be obtained purely by inspecting the properties of the underlying problem. Another widely used type of trade-off information is so called subjective trade-off which is based more on the preferences of the DM. Due to this restriction, we do not discuss here methods that are directly based on such a trade-off concepts as marginal rate of substitution (e.g. GDF method [6]), trade-off cuts (see, e.g., [25]) or bounded trade-offs (see, e.g., [10] and [12]). In addition, we restrict discussion to the methods that can be in a convenient way implemented in practice. Therefore, we omit here such theoretical approaches as trade-off directions (see, e.g., [9],[15],[18],[29]), even though they are very closely related to the objective trade-off concept.

In what follows, we start in Section 2 by defining a multiobjective optimization problem and what we mean by its solution. We also give a general outline of an interactive method. In Section 3 we give a formal definition for the trade-off concept and discuss how it can be connected to interactive methods. In section 4 we present a couple of scalarization formulations that are later on used in methods to produce solution candidates. Computing trade-off rates using Karush-Kuhn-Tucker -multipliers is discussed in Section 5. In Section 6 we present some central interactive methods that utilize the objective trade-off concept in some unique way. Some discussion and further analysis of the methods is presented in Section 7. Finally, we conclude in Section 8.

2 Multiobjective optimization problem

A *multiobjective optimization problem* can be stated in the following form

$$\begin{aligned} &\text{minimize} \quad && \{f_1(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x})\} \\ &\text{subject to} \quad && \boldsymbol{x} \in S := \{\boldsymbol{x} \in \mathbb{R}^n : g_i(\boldsymbol{x}) \leq 0, i = 1, \ldots, m\} \end{aligned} \tag{1}$$

having k conflicting *objective functions* $f_i : \mathbb{R}^n \to \mathbb{R}$ and a *feasible set* $S \neq \emptyset$ defined by m *constraint functions* $g_i : \mathbb{R}^n \to \mathbb{R}$. In what follows, we say that a *decision vector* $\boldsymbol{x} = (x_1, \ldots, x_n)^T$, containing *decision variables* x_i as its components, belong to a *decision space* $\mathbb{R}^n$ and if vector $\boldsymbol{x} \in S$ it is called a *feasible decision vector*. A mapping $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^k$ can be used to map a decision variable $\boldsymbol{x}$ to an *objective vector* $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \dots, f_k(\boldsymbol{x}))^T \in \mathbb{R}^k$, where $\mathbb{R}^k$ is called an *objective space*. If $\boldsymbol{x} \in S$ then $\boldsymbol{f}(\boldsymbol{x})$ is called a *feasible objective vector*. The image $\boldsymbol{f}(S) \subset \mathbb{R}^k$ of the feasible set is called a *set of feasible objective vectors*.

Because of the notational conveniences we consider problem (1) in the minimization form. However, this does not affect the generality because maximization of some objective function f_i is equal to minimizing $-f_i$, for any $i = 1, \dots, k$. Later on, in this text some single objective subproblems are presented in a maximization form, but only in cases where maximization form supports the underlying idea. For instance, when a value function concept is later on presented it is more intuitive to assume that the value is maximized than minimized, even if from the technical point of view the problem can be treated equivalently in either form.

Furthermore, unless stated otherwise, through out this paper we assume that the objective functions f_i $(i = 1, \dots, k)$ and the constraint functions g_i $(i = 1, \dots, m)$ of problem (1) are nonlinear and continuous. In most cases we also assume that these functions are at least once continuously differentiable but this kind of additional properties are always listed explicitly if they are relevant in a context.

So far, we have only outlined a general multiobjective optimization problem framework and defined some terminology related to it. In order to solve problem (1) we must also characterize what is meant by its solution.

2.1 Solution concepts

In what follows, we simply assume that for the DM less is always more. This kind of an assumption leads to concept of Pareto optimality.

Definition 2.1 (Pareto optimality)**.** A decision vector $\hat{\boldsymbol{x}} \in S$ is *Pareto optimal* (efficient, noninferior, nondominated) if there exists no other decision vector $\boldsymbol{x} \in S$ such that $f_i(\boldsymbol{x}) \leq f_i(\hat{\boldsymbol{x}})$ for all $i = 1, \dots, k$ and at least one of the inequalities is strict. An objective vector $\boldsymbol{f}(\hat{\boldsymbol{x}}) = \hat{\boldsymbol{z}}$ is Pareto optimal if the corresponding decision vector $\hat{\boldsymbol{x}} \in S$ is Pareto optimal.

The concept of Pareto optimality can be used to determine decision vectors that can be considered in a mathematical sense as a potential solutions for the problem (1), therefore we also use term *Pareto optimal solution* to refer a Pareto optimal decision vector. In a Pareto optimal solution $\hat{\boldsymbol{x}} \in S$ the value of the objective f_i $(i = 1, \dots, k)$ cannot be improved without degrading the value at least one other objective f_j $(j = 1, \dots, k, j \neq i)$ at the same time. In the light of this we can say that if the DM is interested in changing one Pareto optimal solution to another one then some kind of trade-off among objectives is necessary, that is, simultaneous improvement of every objective is not possible and something must be always given up in order to gain something. The task of the DM is to determine which of the Pareto optimal solutions is the most preferred one.

In what follows, we use term *Pareto surface* to refer set $\boldsymbol{f}(E)$. Loosely speaking, this is due the topological fact that $\boldsymbol{f}(E)$ belongs to the boundary of $\boldsymbol{f}(S)$, and therefore it can be also seen belonging to a hypersurface of the objective space.

Because in multiobjective optimization methods potential solution candidates are typically considered to belong to the set of Pareto optimal vectors E it is useful to define upper and lower bounds for the set $\boldsymbol{f}(E)$. If $\boldsymbol{x}^{\star,i}$ is a solution of the problem $\min_{\boldsymbol{x}\in S} f_i(\boldsymbol{x})$ and $\boldsymbol{x}^{\dagger,i}$ is a solution of the problem $\max_{\boldsymbol{x}\in E} f_i(\boldsymbol{x})$ then vectors $\boldsymbol{z}^\star = (f_1(\boldsymbol{x}^{\star,1}), \dots, f_k(\boldsymbol{x}^{\star,k}))^T$ and $\boldsymbol{z}^\dagger = (f_1(\boldsymbol{x}^{\dagger,1}), \dots, f_k(\boldsymbol{x}^{\dagger,k}))^T$ are called the *ideal* and *nadir objective vectors*, respectively. In other words, the ideal and the nadir objective vectors give the lower and the upper bounds for objective function values in the Pareto optimal set E, respectively. Because ideal and nadir objective vectors are usually used for scaling purposes it is usually convenient to consider a *utopian objective vector* $\boldsymbol{z}^{\star\star}$ instead of the ideal objective vector. The utopian objective vector can be defined, for example, by setting $z_i^{\star\star} = z_i^\star - \epsilon$ for each $i = 1, \dots, k$, where ϵ is a small strictly positive real number. This allows us to avoid numerical difficulties which might occur, for instance, when we do scaling $f_i(\boldsymbol{x})/(\boldsymbol{z}^\dagger - \boldsymbol{z}^\star)$, for each $i = 1, \dots, k$. In a general case nadir objective vector can not usually be computed exactly and, therefore, it needs to be approximated. A widely used, though generally unreliable, approximation can be obtained by using a *payoff table* (see, e.g., [17]). In some cases also the DM can give an approximation for objective function ranges.

Even if the concept of Pareto optimality is quite an intuitive way to determine a set of potential solutions for problem (1) it is not the only way. When considering optimality of problem (1) we can use the theory of preference relations and end up with different kind of optimality concepts (see, e.g. [35] for more details). However, in this paper we omit more general optimality considerations and use the concept of Pareto optimality. However, due the technical reasons we are in practice also forced to consider the weak and proper forms of Pareto optimality. The explanation for this is given later when we present methods to produce Pareto optimal solutions.

Definition 2.2 (Weak Pareto optimality)**.** A decision vector $\hat{\boldsymbol{x}} \in S$ is *weakly Pareto optimal* if there exists no other decision vector $\boldsymbol{x} \in S$ such that $f_i(\boldsymbol{x}) < f_i(\hat{\boldsymbol{x}})$ for all $i = 1, \ldots, k$. An objective vector $\boldsymbol{f}(\hat{\boldsymbol{x}})$ is weakly Pareto optimal if the corresponding decision vector $\hat{\boldsymbol{x}} \in S$ is weakly Pareto optimal.

Definition 2.3 (Proper (Geoffrion) Pareto optimality)**.** A decision vector $\hat{\boldsymbol{x}} \in S$ is said to be *properly Pareto optimal* if it is Pareto optimal and if there exists a finite $M > 0$ such that for each f_i and each $\boldsymbol{x} \in S$ satisfying $f_i(\boldsymbol{x}) < f_i(\hat{\boldsymbol{x}})$, there exists at least one f_j such that $f_j(\boldsymbol{x}) > f_j(\hat{\boldsymbol{x}})$ and

$$\frac{f_i(\hat{\boldsymbol{x}}) - f_i(\boldsymbol{x})}{f_j(\boldsymbol{x}) - f_j(\hat{\boldsymbol{x}})} \leq M$$

for all $i, j = 1, \ldots, k$ and $i \neq j$. An objective vector $\boldsymbol{f}(\hat{\boldsymbol{x}})$ is called properly Pareto optimal if the corresponding $\hat{\boldsymbol{x}}$ is properly Pareto optimal.

The relationship between Definitions 2.1, 2.2, and 2.3 can be characterized by inclusion $E_P \subseteq E \subseteq E_W \subseteq S$, where sets E_P and E_W are the set of properly and weakly Pareto optimal vectors, respectively. From the DM point of view weakly Pareto optimal solutions are usually uninteresting because if a vector $\hat{\boldsymbol{x}} \in E_W$ is weakly Pareto optimal there may exists one vector $\boldsymbol{x} \in E_W$ at which the value of some objective f_i is improved without degradation in the value of any other objective. On the other hand, restriction to the properly Pareto optimal solutions excludes from consideration such Pareto optimal solutions where a relatively small change in the value of one objective can lead to a very large change in the value of another. These concepts are intuitively very closely connected to the concept of a trade-off which is the main topic of this paper.

2.2 An interactive multiobjective optimization method

As pointed out in the previous section, from the purely mathematical point of view the set of Pareto optimal solutions E can be considered as a solution for problem (1). However, difficulties to evaluate the entire set E arise especially in the case of continuous problems when the set E usually contains infinitely many points. We are usually interested in determining just one vector which is then considered as a final solution for problem (1). We use a term *the best compromise solution* to refer to a vector that is considered in a subjective sense (from the DM's perspective) as the best alternative in the set of Pareto optimal solutions of problem (1). This kind of best compromise cannot be determined by purely mathematical means and therefore some external preference information related to importance of objectives is needed. In multiobjective optimization the external preference information is typically given by the DM. This preference information is then incorporated into a solution method. In a nutshell multiobjective optimization procedures differ from each other mainly in 1) the way they obtain external preference information and 2) the way this preference information is used during the solution process.

One possible way to classify multiobjective optimization methods is to examine when the preference information is obtained from the DM (see, e.g. [11] or [17]). In this way at least four main method categories can be recognized: no-preference, a posteriori, a priori, and interactive methods. In no-preference methods the preference information of the DM is not available and just some Pareto optimal solution is obtained. A posteriori methods generate representative set of Pareto optimal solutions after which the DM selects the most preferred one. In a priori methods the preferences of the DM are determined before the solution process, and then a Pareto optimal solution which is in some sense the closest one is determined. Interactive methods work iteratively in interaction with the DM and solution candidates at each iteration are generated using preference information obtained from the DM.

In this paper the main emphasis is laid on the class of interactive methods. This is due to the fact that in multiobjective optimization the trade-off concept (to be presented) is mostly connected to the interactive methods. Typically, a general interactive method has three steps [17]:

1. Find an initial Pareto optimal solution.
2. Interact with the DM.
3. Generate one or some Pareto optimal solution(s). If the new solution or some of the previous solutions is acceptable to the DM, stop. Otherwise, go to step 2.

In step 1 and 3 Pareto optimal solutions are generated from the set of Pareto optimal solutions E. In step 2 the term 'interact' means that preference information (related to previously shown Pareto optimal solutions) is obtained from the DM. This can be done in several ways and usually the main differences between interactive methods are in the way the preference information is asked from the DM and what kind of information is shown to the DM in this interaction.

The key issue with interactive methods is that the DM is able to learn during the iterative process. Learning is, on the one hand, related to properties of the problem which means that the DM gradually grasps an idea of what kind of Pareto optimal solutions are available. On the other hand, the DM also learns about his/her own preferences. In some interactive methods the DM is allowed to change his/her mind which is intuitively speaking an important aspect when learning something new.

In addition to the learning aspect, interactive methods might be useful also when function evaluations are expensive or when we are dealing with a relatively large number of objectives. This is due the fact that in interactive methods only a relatively small set of Pareto optimal solutions is usually needed because the DM works in collaboration with the method and directs continuously the search to the areas which seems to be most interesting to him/her. In other words, computation of uninteresting Pareto solutions can be avoided. The efficiency of an interactive method is related to the way it supports the DM during the solution process. The key issues are what kind of and which amount of information should be shown to the DM. If too much information is shown to the DM (s)he may become cognitively loaded, and the same goes with too demanding questions related to preferences. One important topic is also that no method works for every DM and every problem, and this is related to the question in what form should the preference information be asked from the DM.

The next section introduces the concept of trade-off information and later on, we will also present some interactive methods which are utilizing trade-off information. Trade-off information offers one way to support the DM during the interactive procedure, but it must be emphasized that it is definitely not the only way. As already pointed out, it depends on the underlying problem and the DM what kind of information is most useful in some context, and also trade-off information has its benefits and drawbacks. A completely other problem is how to determine which kind of information is the most useful in some particular context. All in all the trade-off concept to be presented should be considered as supporting information which is meant to aid the DM during the interactive procedure so that (s)he could get more confidence that the decisions made are corresponding to one's preferences.

3 Trade-off information

If we consider two conflicting objectives, intuitively speaking the term 'trade-off' means that we must sacrifice something in the value of other objective in order to gain something in the value of another one. In other words, we trade some amount of one good to obtain some other good. From a more technical perspective we are considering trade-off information as a value which offers an answer to the question "How much is the relative change in the value of objective f_i when the value of objective f_j changes by one unit?". When considering, for instance, some Pareto optimal solution it is quite clear that in some cases this kind of information can be very useful when incorporated to multiobjective optimization methods. For instance, at some Pareto optimal solution we can use trade-off information to indicate how the objective function values are going to change if we move from the current solution to some nearby solution. The trade-off information can be considered in the methods mainly in two ways: 'what is', and 'what should be'. This means that 'what is' type of *objective trade-off* gives us information about the real interconnections between objective functions whereas 'what should be' type of *subjective trade-off* information is determined by the DM and it is used to reflect the preferences (see e.g. [4, p. 335]). The emphasis in this text is on objective trade-off information, but we will present also methods that combines both of these ways.

The formal definition for the trade-off concept can be given in the following form (see, e.g. [8], [4, pp. 331]):

Definition 3.1 (Trade-off)**.** Let us assume that we have two decision variables $\boldsymbol{x}, \hat{\boldsymbol{x}} \in S$. The *trade-off* (rate of change, rate of transformation, pairwise trade-off, point-to-point trade-off) between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ involving objective functions f_i and f_j is defined as

$$T_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{f_i(\boldsymbol{x}) - f_i(\hat{\boldsymbol{x}})}{f_j(\boldsymbol{x}) - f_j(\hat{\boldsymbol{x}})} = \frac{\Delta f_i(\boldsymbol{x}, \hat{\boldsymbol{x}})}{\Delta f_j(\boldsymbol{x}, \hat{\boldsymbol{x}})} \tag{2}$$

where $f_j(\boldsymbol{x}) \neq f_j(\hat{\boldsymbol{x}})$.

Remark 3.1. Trade-off $T_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}})$ in Definition 3.1 is not defined if $f_j(\boldsymbol{x}) = f_j(\hat{\boldsymbol{x}})$. However, we can consider the following extension:

$$\overline{T}_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}}) := \begin{cases} -\infty & \text{, if } \Delta f_i(\boldsymbol{x}, \hat{\boldsymbol{x}}) < 0 \text{ and } \Delta f_j(\boldsymbol{x}, \hat{\boldsymbol{x}}) = 0 \\ 0 & \text{, if } \Delta f_i(\boldsymbol{x}, \hat{\boldsymbol{x}}) = 0 \text{ and } \Delta f_j(\boldsymbol{x}, \hat{\boldsymbol{x}}) = 0 \\ \infty & \text{, if } \Delta f_i(\boldsymbol{x}, \hat{\boldsymbol{x}}) > 0 \text{ and } \Delta f_j(\boldsymbol{x}, \hat{\boldsymbol{x}}) = 0 \\ T_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}}) & \text{otherwise.} \end{cases} \tag{3}$$

In other words, trade-off can be considered as a mapping $\overline{T}_{ij} : \mathbb{R}^n \times \mathbb{R}^n \to \overline{\mathbb{R}}$, where $\overline{\mathbb{R}} := \{-\infty\} \cup \mathbb{R} \cup \{\infty\}$. The extension above might be useful because infinite trade-off has, at least in theory, intuitive meaning. For instance, in the set of weakly Pareto optimal solutions it may be possible to make changes where one objective value becomes better ($\overline{T}_{ij} = -\infty$) or worse ($\overline{T}_{ij} = \infty$) while another objective maintains its current level. Basically this means that we can gain or loose in the value of one objective with zero cost in the value of another objective.

If problem (1) contains more than two objectives it makes sense to separate the following two cases.

Definition 3.2 (Partial and total trade-off)**.** Trade-off $T_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}})$ is called a *partial trade-off* if $f_l(\boldsymbol{x}) = f_l(\hat{\boldsymbol{x}})$ for all $l = 1, \ldots, k$ and $l \neq j, i$. If $f_l(\boldsymbol{x}) \neq f_l(\hat{\boldsymbol{x}})$ for at least one l then $T_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}})$ is called a *total trade-off.*

Let us point out that when there are only two objective functions then partial trade-off is always equal to total trade-off, but in the case of more than two objectives this equality does not generally hold.

In some cases it maybe convenient to consider Definition 3.1 in a very small neighborhood of some decision variable $\hat{\boldsymbol{x}}$. We can, for instance, study what kind of trade-offs are available if we vary a decision variable $\hat{\boldsymbol{x}}$ to some certain direction $\boldsymbol{d} \in \mathbb{R}^n$.

Definition 3.3 (Trade-off rate)**.** Let us assume that there is a direction $\boldsymbol{d}$ such that $\hat{\boldsymbol{x}} + \alpha \boldsymbol{d} \in S$ for all $\alpha \in [0, \bar{\alpha}]$ and for some $\bar{\alpha} > 0$. If the limit

$$t_{ij}(\hat{\boldsymbol{x}}; \boldsymbol{d}) = \lim_{\alpha \searrow 0} T_{ij}(\hat{\boldsymbol{x}} + \alpha \boldsymbol{d}, \hat{\boldsymbol{x}}) = \lim_{\alpha \searrow 0} \frac{f_i(\hat{\boldsymbol{x}} + \alpha \boldsymbol{d}) - f_i(\hat{\boldsymbol{x}})}{f_j(\hat{\boldsymbol{x}} + \alpha \boldsymbol{d}) - f_j(\hat{\boldsymbol{x}})}$$

exists we say that $t_{ij}(\hat{\boldsymbol{x}}; \boldsymbol{d})$ is a *trade-off rate* (marginal rate of transformation) at $\hat{\boldsymbol{x}}$ to the direction $\boldsymbol{d}$ involving objectives f_i and f_j. Furthermore, if f_i and f_j are both continuously differentiable at $\hat{\boldsymbol{x}}$ then

$$t_{ij}(\hat{\boldsymbol{x}}; \boldsymbol{d}) = \frac{\nabla f_i(\hat{\boldsymbol{x}})^T \boldsymbol{d}}{\nabla f_j(\hat{\boldsymbol{x}})^T \boldsymbol{d}}.$$

Definition 3.4 (Partial and total trade-off rate)**.** Trade-off rate $t_{ij}(\hat{\boldsymbol{x}};\boldsymbol{d})$ is called a *partial trade-off rate* if $f_l(\boldsymbol{x}) = f_l(\hat{\boldsymbol{x}} + \alpha \boldsymbol{d})$ for all $\alpha \in [0, \bar{\alpha}]$, for some $\bar{\alpha} > 0$, and for all $l = 1, \dots, k$ and $l \neq j, i$. Otherwise $t_{ij}(\hat{\boldsymbol{x}};\boldsymbol{d})$ is called a *total trade-off rate.*

In what follows, we use notation $t_{ij}(\hat{\boldsymbol{x}})$ at $\hat{\boldsymbol{x}} \in S$ without any explicit direction to denote partial trade-off.

From Definition 3.3 we see that trade-off rate at some $\hat{\boldsymbol{x}} \in S$ can be computed easily using formula $\nabla f_i(\hat{\boldsymbol{x}})^T \boldsymbol{d} / \nabla f_j(\hat{\boldsymbol{x}})^T \boldsymbol{d}$ but if the term $\nabla f_j(\hat{\boldsymbol{x}})^T \boldsymbol{d}$ is zero then $t_{ij}(\hat{\boldsymbol{x}}, \boldsymbol{d})$ is not defined. However, we can extend this definition in a similar way we did in the case of Definition 3.1.

It must be pointed out that trade-off definitions above do not make any assumptions related to the Pareto optimality. However, in this paper we are interested in considering trade-offs especially related to decision vectors belonging to the Pareto optimal set E. When we consider a Pareto optimal solution we know that none of the objective values can be improved without degrading a value of at least one other objective. In the terms of trade-offs this means that at a Pareto optimal solution at least one trade-off rate value involving objectives f_i and f_j must be negative, where $i, j \in \{1, \dots, k\}$ and $i \neq j$. In other words, if we want to improve the value of one objective then at least one other objective value must be degraded.

If we consider practical use of trade-off and trade-off rate we can see two main differences. The first one is that in a nonlinear case a trade-off (Definition 3.1) is always exact between two Pareto optimal solutions whereas the trade-off rate (Definition 3.3) is giving a linear approximation of what happens in a relatively small neighborhood of considered Pareto optimal solution when we move a small step away from the given solution to a certain direction. On the other hand, to compute trade-off by using Definition 3.1 we must generate two Pareto optimal solutions whereas trade-off rate in Definition 3.3 can be computed by using the information obtained at a single Pareto optimal solution. This is an important aspect, because we can use trade-off rate information to reflect what is happening in a neighborhood of some particular Pareto optimal solution without needing to produce any other Pareto optimal solution. This kind of information can be helpful to the DM when (s)he must decide how to state preferences in an interactive method in such a way that the method is able to produce improved solutions.

No matter in which form trade-off information is used in interactive multiobjective methods, the most important aspect is that it offers information that is useful for the DM and this way helps to make better decisions during an interactive solution procedure. If we consider Definition 3.3 the first problem is how to select the direction $\boldsymbol{d}$ in a meaningful way. Later on we will see that in the case of partial trade-off rates we do not have to consider directions explicitly. So far in the literature the partial trade-off rates have been the most used form of trade-off information in interactive methods. However, also other than partial trade-off directions are used. We will see that, for instance, such an intuitive directions as coordinate axes directions and directions of objective function gradients are considered in the existing methods. Even thought the direction can be selected freely it is important to notice that it must always produce trade-off information which have some meaning for the DM. The aim is to offer information that aids the DM in a way or another.

Figure 1 illustrates the concepts of trade-off (Definition 3.1) and trade-off rate (Definition 3.3) in the case of two objective functions. As can be seen the trade-off $T_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \Delta f_i(\boldsymbol{x}, \hat{\boldsymbol{x}}) / \Delta f_j(\boldsymbol{x}, \hat{\boldsymbol{x}})$ can be related to two Pareto optimal solutions $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ even if they are located at very different parts of the Pareto optimal set E. The aim with the trade-off rate, in the case of a Pareto optimal $\hat{\boldsymbol{x}}$, is to consider such a direction $\boldsymbol{d}$ which produces a tangent vector for the Pareto optimal surface at $\boldsymbol{f}(\hat{\boldsymbol{x}})$, in other words, in Definition 3.3 there must be such a $\bar{\alpha} > 0$ such that $\hat{\boldsymbol{x}} + \alpha \boldsymbol{d} \in E$ for all $\alpha \in [0, \bar{\alpha}]$.

Even if the trade-off rate information is an appealing way to reflect the local changes in some neighborhood of the considered Pareto optimal solution it must be emphasized that it offers only a linear approximation of the Pareto surface. In addition, this approximation may in a nonlinear case give a very misleading idea of how the underlying Pareto surface is actually behaving if we move further away from the considered Pareto optimal solution.

Like pointed out formerly, by definition the trade-off concept is not related only to Pareto optimal solutions. This means that, for example, trade-off can be computed between any two decision vectors. Furthermore, the trade-off rate can be computed at any decision vector to any

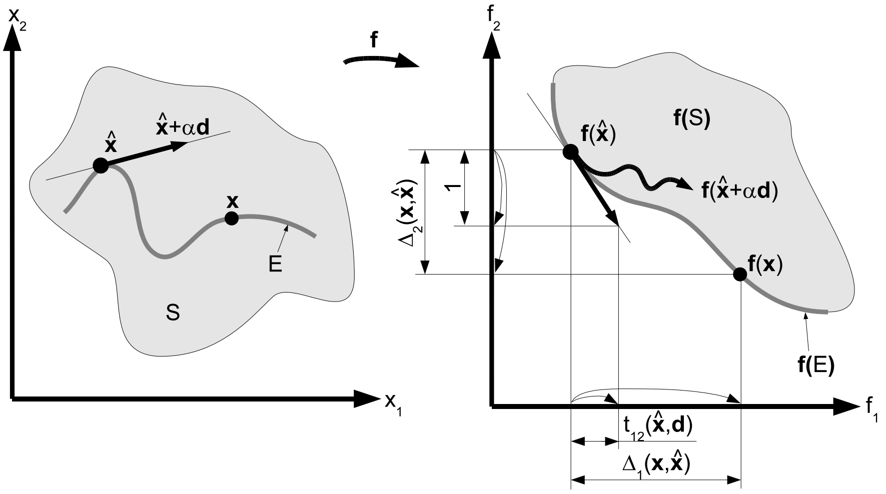

Figure 1: An illustration of trade-off and trade-off rate

direction, and even if the vector is Pareto optimal the direction might point out of set E. However, from the practical point of view it is preferable that these computed values have some concrete meaning in multiobjective solution methods. Therefore, it is favorable to consider only trade-offs between Pareto optimal solutions.

One other way to utilize trade-off kind of ideology in multiobjective optimization methods is to use the concept of subjective trade-off. This means that instead of computing objective trade-offs related to Pareto optimal solutions we can ask the DM to define preferred trade-offs related to some solution(s). This kind of subjective trade-off information can be, for instance, used in interactive methods to introduce new constraints that temporarily or permanently limit the search space (e.g. trade-off cut methods), or the DM can indicate his/her so called indifference direction at some Pareto optimal solution in a form of a marginal rate of substitution.

The use of subjective trade-off information is typically related to an assumption that the DM has in a local sense an *implicit value function* $V : \mathbb{R}^k \to \mathbb{R}$. Because of the Pareto optimality we assume that for the DM less is more. In such a case V is assumed to be monotonously decreasing. In this respect it is considered as if the DM is trying to "maximize" $V(\boldsymbol{f}(\boldsymbol{x}))$ subject to $\boldsymbol{x} \in S$. Basically the term 'implicit' means that the analytic form of V is not known but the DM is able to reflect the shape of this function by answering to questions related to preferences. Local means that preferences of the DM at some considered Pareto optimal solution are only related to that point and given preference information reflects only the local behavior of the implicit value function.

A value function represents the preferences of the DM related to objective vectors in such a way that if $\boldsymbol{f}(\boldsymbol{x}^a)$ and $\boldsymbol{f}(\boldsymbol{x}^b)$ are objective vectors and $V(\boldsymbol{f}(\boldsymbol{x}^a)) > V(\boldsymbol{f}(\boldsymbol{x}^b))$ then the DM prefers $\boldsymbol{x}^a$ to $\boldsymbol{x}^b$. If $V(\boldsymbol{f}(\boldsymbol{x}^a)) = V(\boldsymbol{f}(\boldsymbol{x}^b))$ then the DM is indifferent between these two objective vectors, in other words, $\boldsymbol{f}(\boldsymbol{x}^a)$ and $\boldsymbol{f}(\boldsymbol{x}^b)$ lie on the same *indifference curve*, that is, the contour curve of the value function V.

Definition 3.5 (Marginal rate of substitution (MRS))**.** A *marginal rate of substitution* (indifference trade-off rate) involving objectives f_i and f_j at $\boldsymbol{x}$, denoted by $m_{ij}(\boldsymbol{x})$, indicates the amount how much the value of the objective f_i must be improved to compensate a one unit degrade in the value of the objective f_j while other objectives remain unchanged. If for V partial derivatives with respect to objectives f_j and f_i exist then at $\boldsymbol{x}$ MRS is

$$m_{ij}(\boldsymbol{x}) = \frac{\frac{\partial V(\boldsymbol{f}(\boldsymbol{x}))}{\partial f_j}}{\frac{\partial V(\boldsymbol{f}(\boldsymbol{x}))}{\partial f_i}}$$

The usefulness of MRS information can be described as follows. Consider that by defining $m_{ij}(\boldsymbol{x})$ values between each pair of objectives we are able to construct in the objective space a tangent hyperplane for indifference curve of value function V at some $\boldsymbol{x}$. The hyperplane is separating the objective space into two halfspaces where the other halfspace contains directions from $\boldsymbol{x}$ where the value of V is decreasing and another halfspace contains directions where the value of V is increasing. This is due the fact that the gradient vector of an implicit value function indicates the direction of the greatest value improvement from the DM perspective. This kind of information can be utilized in methods, for instance, in such a way that at some $\boldsymbol{x}$ we can ask m_{ij} values from the DM and after that we can generate in a small neighborhood of $\boldsymbol{x}$ another Pareto optimal solution which is guaranteed to produce at least as good value of V as $V(\boldsymbol{x})$.

Let us assume that we have a problem which has a convex Pareto surface and that $\hat{\boldsymbol{x}} \in S$ is a Pareto optimal decision vector and $t_{ij}(\hat{\boldsymbol{x}})$ is corresponding to a partial trade-off involving objectives f_i and f_j. If at some point we have $-m_{ij}(\hat{\boldsymbol{x}}) = t_{ij}(\hat{\boldsymbol{x}})$ for all $i, j = 1, \ldots, k$, $i \neq j$, then $\hat{\boldsymbol{x}} \in S$ can be considered to be the best compromise solution for the DM (see, e.g., [22]). This is due to the fact that the set of Pareto optimal objective vectors $\boldsymbol{f}(E)$ is always located at the boundary of the set of feasible objective vectors $\boldsymbol{f}(S)$ and if the tangent of this boundary is perpendicular with the direction of indifference curve this means in the case of convex Pareto surface that it is not possible to move to any feasible direction in objective space in such a way that the value of V would increase. This relationship between MRS and trade-off rate is depicted in Figure 2 where $\boldsymbol{f}(\boldsymbol{x}^a)$ is optimal with respect to V. In the figure dashed line indicates the indifference curves (contour curves) of the value function V.

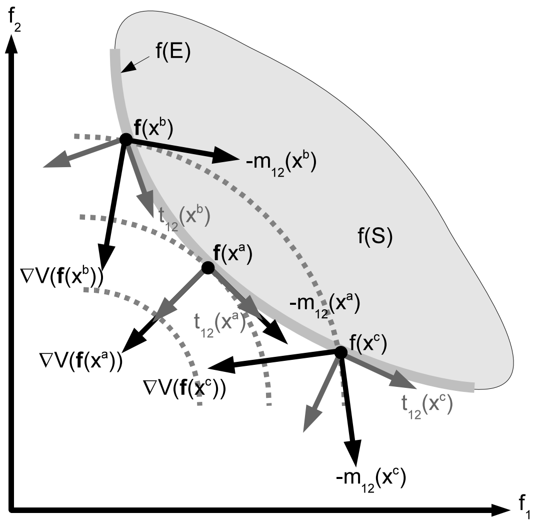

Figure 2: The connection between MRS and trade-off rate

4 Generating Pareto optimal solutions

As formerly pointed out, from the mathematical point of view, the solution to problem (1) is the set of Pareto optimal decision vectors E and the DM is needed to determine which solution is the most preferred one to him/her. The problem is that set E is not explicitly defined. Therefore, to evaluate Pareto optimal solutions we need some methods that are able to generate them from the set of feasible decision vectors S. One way to determine individual Pareto optimal solutions is to convert problem (1) to a single objective form, this task is called *scalarization*. In scalarization a relatively small set of parameters is connected to problem (1) in such a way that by altering these parameters we can form different single objective problems when solved produce as a solution a Pareto optimal solution. This kind of single objective formulation is called *scalarization problem* and it can be designed in several different ways.

Ideally it is assumed that by solving an scalarization problem we are able to produce any Pareto optimal solution related to problem (1), and furthermore, it is assumed that only Pareto optimal solutions are produced. However, these assumptions are not usually fulfilled [24, pp. 277]. Next we present two scalarization problem formulations that are referred to later on when interactive multiobjective optimization methods utilizing trade-offs are presented.

The *ϵ-constraint method* (e.g. in [4]) proposes a scalarization where one objective function f_i is selected as a *primary objective* while rest of the objectives f_j are treated as constraints and called *constraint objectives*. It is suggested that the most important objective for the DM is selected as a primary objective. This leads to the following scalarization problem formulation:

$$\begin{array}{lll} \underset{\boldsymbol{x}}{\text{minimize}} & f_i(\boldsymbol{x}), & \text{for some } i \in \{1, \dots, k\} \\ \text{subject to} & f_j(\boldsymbol{x}) \leq \epsilon_j, & \text{for all } j = 1, \dots, k, \text{ and }, j \neq i \\ & \boldsymbol{x} \in S, & \end{array} \tag{4}$$

where S is the feasible set of the original problem (1). A vector $\boldsymbol{\epsilon} \in \{(\epsilon_1, \dots, \epsilon_{i-1}, \epsilon_{i+1}, \dots, \epsilon_k)^T \in \mathbb{R}^{k-1} : X(\epsilon) \neq \emptyset\}$, where $X(\epsilon) := \{\boldsymbol{x} \in \mathbb{R}^n : f_j(\boldsymbol{x}) \leq \epsilon_j, j \neq i\}$, is set in such a way that there exists at least one vector in the set $S \cap X(\epsilon)$.

If scalarization (4) is used directly as a solution method then the DM is assumed to set upper bounds ϵ_j for constraint objectives f_j ($j = 1, \dots, k$ and $j \neq i$), and by altering these bounds different (weakly) Pareto optimal solutions can be generated. One drawback related to this problem is that especially in the case of several objectives it might be difficult to explicitly check whether bounds ϵ_j set by the DM are assigned in such a way that the set $S \cap X(\epsilon)$ contains at least one feasible decision vector.

In *weighted Chebyshev method* [2] another kind of scalarizing problem formulation is used. Later on, we use term *Chebyshev scalarization* to refer the following problem:

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{minimize}} & \max\limits_{i=1,\dots,k} w_i(f_i(\boldsymbol{x}) - z_i^{\star\star}) \\ \text{subject to} & \boldsymbol{x} \in S \end{array} \tag{5}$$

In this scalarization different (weakly) Pareto optimal solutions are obtained by projecting the utopian vector $\boldsymbol{z}^{\star\star}$ to the set of Pareto optimal solutions. This is done by assigning different scaling factors $w_i > 0$ for objectives f_i ($i = 1, \dots, k$).

The presented formulation (5) is able to produce all (weakly) Pareto optimal solutions and it is also guaranteed that every solution obtained is (weakly) Pareto optimal. The drawback with problem (5) is that the objective function is not differentiable and therefore usage of efficient gradient based solvers is not directly possible. However, if the functions f_i are all differentiable then problem (5) can be converted to an equivalent differentiable form by adding an auxiliary variable. This leads to the following formulation:

$$\begin{array}{ll} \underset{\alpha, \boldsymbol{x}}{\text{minimize}} & \alpha \\ \text{subject to} & w_i(f_i(\boldsymbol{x}) - z_i^{\star\star}) \leq \alpha \\ & \boldsymbol{x} \in S \end{array} \tag{6}$$

In the upcoming sections we present several multiobjective optimization methods that utilize either of the presented scalarization problems (except the ISTM method). To simplify notation, we will use the formulations above without so called augmentation terms. An augmentation term can be included to ensure that obtained solutions are properly Pareto optimal. However, it must be emphasized that all presented methods to be presented can be formulated in such a way that they produce properly Pareto optimal solutions instead of weakly Pareto optimal. Especially from the trade-off information utilization perspective it does not make a much difference whether we consider weakly or properly Pareto optimal solutions, as long as we consider trade-offs in the extended form in the sense of Remark 3.1. In what follows, we use the term Pareto optimal even if the considered method might produce only weakly Pareto optimal solutions.

Both the scalarization methods presented above can be basically used as an interactive multiobjective optimization methods directly. The DM just alters mutually constraint ϵ_i ($i = 1, \dots, k-1$) or scaling factor w_i ($i = 1, \dots, k$) parameters and obtains a set of Pareto optimal solutions and at some point selects the most preferred one as the final solution. However, in a general case it is intuitively quite easy to understand that setting parameters consistently in this way might be very difficult. This gives a motivation for more cultivated interactive methods where the aim is to offer to the DM a framework where scalarization parameters can be set in a systematic and meaningful way. The ideal situation is such where the DM can feel that (s)he is in control and the solution selected as a final one corresponds to his/her preferences and is therefore in subjective sense the best possible according to the current knowledge. Later on, we describe interactive methods that use the trade-off concept, in a way or another, to aid the DM to indicate preferences. The preferences are then used implicitly to set parameters in a scalarization function at each iteration. However, before presenting the methods, we discuss one popular way to compute (partial) trade-off rates as a side-product when a Pareto optimal solution is produced using some scalarization function and an appropriate single objective optimization method.

5 Trade-off rates and Karush-Kuhn-Tucker multipliers

As already pointed out, the trade-off concept does not necessitate by definition that the considered decision variables are Pareto optimal. However, the aim of this paper is to discuss how trade-off information is used in interactive multiobjective methods and therefore we are mostly interested in generating and analyzing trade-off information related to Pareto optimal solutions.

The computation of the trade-off $T_{ij}(\boldsymbol{x}, \hat{\boldsymbol{x}})$ (Definition 3.1) between two separate Pareto optimal solutions $\boldsymbol{x} \in E$ and $\hat{\boldsymbol{x}} \in E$ is a straightforward task. After we have generated these vectors by using some scalarization formulation we can use the value T_{ij} to indicate the relative objective value changes if we consider a vector $\boldsymbol{x}$ instead of $\hat{\boldsymbol{x}}$

If the objective functions f_i and f_j considered are continuously differentiable, then also trade-off rate $t_{ij}(\hat{\boldsymbol{x}}; \boldsymbol{d})$ can be easily computed at $\hat{\boldsymbol{x}}$ to some direction $\boldsymbol{d}$ by using Definition 3.3. However, for our purposes the direction $\boldsymbol{d}$ is meaningful only when it points to the Pareto optimal set E or at least is a tangent direction to E at the Pareto optimal solution $\hat{\boldsymbol{x}} \in E$ which is located at the boundary of the Pareto optimal set $E \subset S$. Furthermore, the direction $\boldsymbol{d}$ must be set in such a way that the corresponding trade-off has some special meaning for the DM because we are interested in offering supporting information for the DM. One possible meaningful direction $\boldsymbol{d}$ is such where $t_{ij}(\hat{\boldsymbol{x}}; \boldsymbol{d})$ turns to be a partial trade-off $t_{ij}(\hat{\boldsymbol{x}})$. Remember that $t_{ij}(\hat{\boldsymbol{x}})$ indicates how the value of the objective f_i is changing when the value of the objective f_j changes by one unit while the other objectives are remaining at their current levels. In existing methods, the partial trade-off rate is the most used form of trade-off information, and in what follows we study how partial trade-off rates can be computed without needing to explicitly consider the direction $\boldsymbol{d}$ appearing in Definition 3.3.

From the application perspective, the most popular way to compute partial trade-off rate information at the given Pareto optimal solution is to use the connection to the Karush-Kuhn-Tucker (KKT) multipliers [13] related to some Pareto optimal solution obtained using some scalarization function. In practice this means that the numerical single objective optimization method used to solve scalarization problems must be able to produce KKT-multipliers related to each Pareto optimal solution computed.

Because this approach plays such a central role in the practical trade-off rate computation, next we present, as an example, a theorem which connects KKT-multipliers to ϵ-constraint scalarization (4) (see e.g. [7]). The following theorem was presented originally in [7] and the necessary background to understand terminology used is given in Appendix. Without loss of generality, we assume that the objectives are indexed in such a way that the primary objective in the ϵ-constraint scalarization (4) is the objective f_k.

Theorem 5.1. *Let a Pareto optimal solution $\hat{\boldsymbol{x}} \in E$ be a solution for scalarization problem (4) for some $\hat{\boldsymbol{\epsilon}} = (\hat{\epsilon}_1, \ldots, \hat{\epsilon}_{k-1})^T$ such that the following conditions are satisfied*

i) $\hat{\boldsymbol{x}}$ is a regular point with respect to active constraints

ii) the second order sufficiency conditions hold at $\hat{\boldsymbol{x}}$

iii) all active constraints at $\hat{\boldsymbol{x}}$ are nondegenerate

Let $\hat{\lambda}_j$ denote the optimal KKT-multiplier corresponding to the constraint objective $f_j(\hat{\boldsymbol{x}}) \leq \epsilon_j$, $j = 1, \ldots, k-1$. Without loss of generality assume that these constraints are ordered in such a way that $\hat{\lambda}_j > 0$ for all $j = 1, \ldots, p$ and $\hat{\lambda}_j = 0$ for all $j = p+1, \ldots, k-1$. Based on the implicit function theorem we have *a) If $p = k-1$ (i.e. $\hat{\epsilon}_j = f_j(\hat{\boldsymbol{x}})$ for all $j = 1,\dots,k-1$) then there exists a neighborhood $N(\hat{\boldsymbol{x}})$ of $\hat{\boldsymbol{x}}$ and continuously differentiable vector-valued function $\bar{\boldsymbol{x}} : \mathbb{R}^{k-1} \to \mathbb{R}^n$ defined in some neighborhood $N(\hat{\boldsymbol{\epsilon}}) \subset \mathbb{R}^{k-1}$ of $\hat{\boldsymbol{\epsilon}}$ such that $E \cap N(\hat{\boldsymbol{x}}) \subseteq \bar{\boldsymbol{x}}(N(\hat{\boldsymbol{\epsilon}})) \subseteq E$.*

b) If $p = k-1$ and $N(\hat{\boldsymbol{x}})$ is as in a), let $Z^k = \{(f_1(\boldsymbol{x}),\dots,f_k(\boldsymbol{x}))^T : \boldsymbol{x} \in E \cap N(\hat{\boldsymbol{x}})\}$ and $Z^{k-1} = \{(f_1(\boldsymbol{x}),\dots,f_{k-1}(\boldsymbol{x}))^T : \boldsymbol{x} \in E \cap N(\hat{\boldsymbol{x}})\}$. There exists a continuously differentiable function $\bar{f}_k$ defined on Z^{k-1} such that for each $(f_1(\boldsymbol{x}),\dots,f_k(\boldsymbol{x}))^T \in Z^k$ we can get $f_k(\boldsymbol{x}) = \bar{f}_k(f_1(\boldsymbol{x}),\dots,f_{k-1}(\boldsymbol{x}))^T$. Moreover

$$\frac{\partial \bar{f}_k}{\partial f_j}(f_1(\hat{\boldsymbol{x}}),\dots,f_{k-1}(\hat{\boldsymbol{x}})) = -\hat{\lambda}_j$$

for each $j = 1,\dots,k-1$.

c) If $1 \le p < k-1$ then let $Z^k_\epsilon = \{(f_1(\boldsymbol{x}),\dots,f_k(\boldsymbol{x}))^T : \boldsymbol{x} \in \bar{\boldsymbol{x}}(N(\hat{\boldsymbol{\epsilon}}))\}$. There exist continuously differentiable functions $\bar{f}_{p+1},\dots,\bar{f}_{k-1}$ and $\bar{f}_k$ defined on $N(\hat{\boldsymbol{\epsilon}})$ such that for each $(f_1(\boldsymbol{x}),\dots,f_k(\boldsymbol{x}))^T \in Z^k_\epsilon$ we have

$$\bar{f}_j(\boldsymbol{x}) = \bar{f}_j(f_1(\boldsymbol{x}),\dots,f_p(\boldsymbol{x}),\hat{\epsilon}_{p+1},\dots,\hat{\epsilon}_{k-1}),\ \text{for all } j = p+1,\dots,k.$$

Moreover for each $i = 1,\dots,p$ we have

$$\left.\frac{\partial \bar{f}_k}{\partial f_i}\right|_{\boldsymbol{\epsilon}=\hat{\boldsymbol{\epsilon}}} = -\hat{\lambda}_j = \frac{\nabla f_k(\hat{\boldsymbol{x}})^T \boldsymbol{d}^i}{\nabla f_i(\hat{\boldsymbol{x}})^T \boldsymbol{d}^i}$$

where $\boldsymbol{d}^i$ is the direction of $\partial \bar{\boldsymbol{x}}(\hat{\boldsymbol{\epsilon}})/\partial \epsilon_i$, and for each $j = p+1,\dots,k-1$ we have

$$\left.\frac{\partial \bar{f}_j}{\partial f_i}\right|_{\boldsymbol{\epsilon}=\hat{\boldsymbol{\epsilon}}} = \nabla f_j(\hat{\boldsymbol{x}})^T \frac{\partial \bar{\boldsymbol{x}}(\hat{\boldsymbol{\epsilon}})}{\partial \epsilon_i}$$

Proof. See [7, pp. 163] □

Theorem 5.1 means that when conditions i), ii), and iii) are fulfilled at some Pareto optimal solution $\hat{\boldsymbol{x}}$ and the KKT-multiplier $\hat{\lambda}_j$ strictly positive for each constraint objective f_j, $j = 1,\dots,k-1$, in problem (4), then $-\hat{\lambda}_j$ (for some $j = 1,\dots,k-1$) reflects how primary objective f_k is going to change if constraint objective f_j is changing by one unit, while the other objectives remain at their current levels. In other words, we have a connection between optimal KKT-multipliers of problem (4) and partial trade-off rates. If all KKT-multipliers are not strictly positive at the considered Pareto optimal $\hat{\boldsymbol{x}}$ (i.e. one of the objective constraints is not active, $f_j(\hat{\boldsymbol{x}}) < \hat{\epsilon_j}$) we are not able to consider partial trade-off rates. In Theorem 5.1 case c) reflects this kind of situation, indicating that we can consider total trade-off. The practical implications of this are considered a bit later when the ISWT method is presented.

A technical problem related to case c) in Theorem 5.1 is that the value $\partial \bar{\boldsymbol{x}}(\hat{\boldsymbol{\epsilon}})/\partial \epsilon_j$ is not know. However, it can be approximated, for instance, by perturbing by a small amount the component $\hat{\epsilon}_j$ of vector $\hat{\boldsymbol{\epsilon}}$ that is related to constraint objective f_j having a strictly positive KKT-multiplier and solving this perturbed problem. In [4, pp. 376] some alternative ways are presented to deal with this situation but in every case some additional computation is always needed.

In practice, when some interactive multiobjective optimization method utilizing the ϵ-contraint scalarization (4) is used, it may be difficult to confirm that assumptions i), ii), and iii) of Theorem 5.1 are fulfilled at some Pareto optimal solution under consideration. In [14], it is claimed, without proof, that the following Lemma can be used to verify whether the assumptions in Theorem 5.1 hold at some Pareto optimal solution in the case of the ϵ-contraint scalarization (4).

Lemma 5.2. *Let us denote the Lagrangian function related to problem (4) by*

$$L(\boldsymbol{x},\boldsymbol{\lambda},\boldsymbol{\mu}) = f_k(\boldsymbol{x}) + \sum_{i=1}^{k-1} \lambda_i(\nabla f_i(\boldsymbol{x}) - \epsilon_i) + \sum_{i=1}^{m} \mu_i g_i(\boldsymbol{x})$$

and $\nabla^2_{xx} L(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\mu}})$ is the corresponding Hessian matrix with respect to $\boldsymbol{x}$. Furthermore, $\hat{\boldsymbol{\lambda}} \in \mathbb{R}^{k-1}$, $\hat{\boldsymbol{\mu}} \in \mathbb{R}^m$ are optimal KKT-multipliers related to the objective constraints and the original constraints, respectively. The vector $\boldsymbol{\epsilon} \in \mathbb{R}^{k-1}$ is the ϵ-constraint vector. The diagonal matrices $D_1(g_j(\hat{\boldsymbol{x}}))$, $D_2(f_i(\hat{\boldsymbol{x}}) - \epsilon_i)$ and $D_3(-\hat{\lambda}_i)$ are $m \times m$, $(k-1) \times (k-1)$ and $(k-1) \times (k-1)$ matrices which have $g_j(\hat{\boldsymbol{x}})$, $f_i(\hat{\boldsymbol{x}}) - \epsilon_i$ and λ_i as diagonal entries, respectively.

In Theorem 5.1, the conditions $i) - iii)$ are satisfied for problem (4) if and only if the matrix

$$\begin{pmatrix} \nabla^2_{xx} L(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\mu}}) & \nabla g_1(\hat{\boldsymbol{x}})^T \ldots \nabla g_m(\hat{\boldsymbol{x}})^T & \nabla f_1(\hat{\boldsymbol{x}})^T \ldots \nabla f_{k-1}(\hat{\boldsymbol{x}})^T & \mathbf{0} & \mathbf{0} \\ \begin{matrix} \hat{\mu}_1 \nabla g_1(\hat{\boldsymbol{x}})^T \\ \vdots \\ \hat{\mu}_m \nabla g_m(\hat{\boldsymbol{x}})^T \end{matrix} & D_1(g_j(\hat{\boldsymbol{x}})) & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \begin{matrix} \hat{\lambda}_1 \nabla f_1(\hat{\boldsymbol{x}})^T \\ \vdots \\ \hat{\lambda}_{k-1} \nabla f_{k-1}(\hat{\boldsymbol{x}})^T \end{matrix} & \mathbf{0} & D_2(f_i(\hat{\boldsymbol{x}}) - \epsilon_i) & D_3(-\hat{\lambda}_i) & \mathbf{0} \\ \begin{matrix} \nabla f_1(\hat{\boldsymbol{x}})^T \\ \vdots \\ \nabla f_{k-1}(\hat{\boldsymbol{x}})^T \end{matrix} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \nabla f_k(\hat{\boldsymbol{x}})^T & 0 & 0 & 0 & -1 \end{pmatrix}$$

is nonsingular at a Pareto optimal $\hat{\boldsymbol{x}} \in E$.

Even thought Theorem 5.1 is valid only for the ϵ-constraint scalarization (4), similar results have been presented for several other scalarization problem formulations. For example, Sakawa and Yano [33] showed that there is a connection between partial trade-off rates and KKT-multipliers related to Chebyshev scalarization problem (6).

As stated in Theorem 5.1 for the ϵ-constraint scalarization (4), a partial trade-off rate involving primary objective f_k and some constraint objective f_j, for some $j = 1, \ldots, k-1$, is

$$\frac{\partial f_k}{\partial f_j}(\hat{\boldsymbol{x}}) = -\hat{\lambda}_j. \tag{7}$$

For the Chebyshev scalarization problem (6), partial trade-off rates can be obtained from the formula

$$\frac{\partial f_i}{\partial f_j}(\hat{\boldsymbol{x}}) = -\frac{\hat{\lambda}_j w_j}{\hat{\lambda}_i w_i}, \tag{8}$$

where w_i, w_j and $\hat{\lambda}_i$, $\hat{\lambda}_j$ are scaling factors and optimal KKT-multipliers related to the objectives f_i and f_j, respectively.

In addition to the scalarizations (4) and (6) used in this paper in [34] corresponding connections are derived using so called hyperplane problem for several well-known scalarization problem formulations such as: the weighting method, and the augmented forms of the ϵ-constraint and the Chebyshev scalarization. In [14], results similar to Lemma 5.2 have also been presented for the weighting method and Chebyshev scalarization method. In [23] it is shown by using so called generalized hyperplane method that there is a connection between partial trade-off rates and KKT-multipliers also in the case of other than Pareto dominance structure (the case where optimality is defined by using some other convex cone than orthant cone). All these results are derived using the implicit function theorem and sensitivity results from the theory of single objective optimization (see, e.g., [5, Theorem , pp. 3.2.2 ,pp. 72], [16, Sensitivity theorem, pp. 236]).

In the case of multiobjective linear optimization problems the simplex method produces multipliers that have a bit similar kind of interpretation like KKT-multipliers have in the case of nonlinear problems. We discuss this connection when trade-off utilizing methods are introduced in the next section.

6 Objective Trade-off rates and interactive methods

In this section, we survey interactive multiobjective optimization methods that have presented some unique way to utilize objective trade-off information. We present only the methods that use objective trade-off information in some form either with or without connections to the concept of subjective trade-off information such as MRS. The presented methods can be seen as a prototypes and each of them characterize one unique way to use trade-off information in the context of multiobjective optimization methods. Later on, we analyze the methods presented and propose an classification based on the level at which the trade-off information is utilized in these methods.

In what follows, each of these methods is first shortly described and related trade-off ideas are explained. It must be emphasized that the methods are in most cases presented in a slightly simplified form to highlight the underlying trade-off ideas. For instance, most of the methods are actually using augmented scalarization problem formulations which enables them to deal with properly Pareto optimal instead of weakly Pareto optimal solutions. However, inclusion of such technical details makes the notation only more complicated while the actual trade-off utilization idea remains the same. Therefore, we omit some technical details. In most of the cases there may also exist extensions to a basic method but those are mentioned only if they make some relevant contribution to the underlying trade-off concept.

In what follows, it is important to notice that the methods are not presented in the order they have been appeared in the literature. This means that the methods are not necessarily always proposing an improvement to the methods that have already been existing before them. Thus, the presentation order is based on the way how the trade-off information is utilized in the methods. The following ways can be identified: trade-off to the coordinate direction in the decision space, trade-off to the direction of an objective function gradient, trade-off and approximation of implicit value function, automatic trade-off to aid the DM, determining allowed change using trade-off table, and trade-off rates and MRS as a stopping condition. We will notice that trade-off information is utilized at different levels ranging from learning to decision making.

6.1 Trade-off to the coordinate direction (ZW)

In a general case, in Definition 3.3 when defining a trade-off rate the direction $\boldsymbol{d}$ can be set in infinitely many ways. One intuitively meaningful direction is the coordinate direction in the decision space. This makes sense especially if we are solving a linear multiobjective optimization problem with the simplex method and want to consider relative changes in the objective function values when some nonbasic variable enters the basis. The method of Zionts and Wallenius (ZW) [36] is the first interactive multiobjective optimization method that directly utilizes trade-off information to aid the DM to reflect her/his preferences during the interactive solution process.

We consider a linear multiobjective optimization problem in the standard form

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & C\boldsymbol{x} \\ \text{subject to} \quad & A\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq 0 \end{aligned} \tag{9}$$

where we have vector $\boldsymbol{x} \in \mathbb{R}^p$, C is $k \times p$ matrix, A is $m \times p$ matrix, and vector $\boldsymbol{b} \in \mathbb{R}^m$. In the standard formulation, inequality constraints are converted to equality constraints by adding slack variables, so when compared to the problem (1) we have $p > n$, that is, in addition of decision variables $\boldsymbol{x}$ contains also slack variables. Let us assume that $\hat{\boldsymbol{x}}$ is a Pareto optimal solution for (9) and the corresponding simplex tableau is the following

	$\boldsymbol{x}^B$			$\boldsymbol{x}^N$			
$\hat{x}_1$	1			a	$\dots$	a	b
$\vdots$		$\ddots$		$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\hat{x}_n$			1	a	$\dots$	a	b
f_1				s	$\dots$	s	$-f_1$
$\vdots$				$\vdots$	$\ddots$	$\vdots$	$\vdots$
f_k				s	$\dots$	s	$-f_k$

where columns $\boldsymbol{x}^B$ are $\boldsymbol{x}^N$ containing basic and nonbasic variables, respectively.

The underlying idea of the method is to show at each iteration to the DM one Pareto optimal solution and ask whether the DM is interested in changing this solution to another one if this leads to a certain trade-off in the objective function values. According to the answers of the DM, an implicit value function is approximated and based on this information a new Pareto optimal solution is produced.

In more detail, it is assumed that the DM has at each iteration h an implicit linear value function of the form $V(\boldsymbol{w}^h, \boldsymbol{x}) = -(\boldsymbol{w}^h)^T C\boldsymbol{x}$, where the weighting vector $\boldsymbol{w}^h \in \mathbb{R}^k$ is not known explicitly. Components w_i^h of the weighting vector satisfy the conditions $\sum_{i=1}^k w_i^h = 1$ and $w_i^h > 0$ for all $i = 1, \dots, k$. At the each iteration, the preference information of the DM is obtained using trade-off information related to the current Pareto optimal solution $\hat{\boldsymbol{x}}^h$ and this preference information is used to determine a vector $\boldsymbol{w}^h$ which is used to form the value function $V(\boldsymbol{w}^h, \boldsymbol{x})$. At each iteration h, a new Pareto optimal solution $\boldsymbol{x}^h$ is obtained by solving the following subproblem

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{maximize}} & V(\boldsymbol{w}^h, \boldsymbol{x}) \\ \text{subject to} & A\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq 0 \end{array} \tag{10}$$

If we consider the optimal simplex tableau related to the Pareto optimal extreme solution $\boldsymbol{x}^h$, then the components s_i^j of the jth nonbasic column $\boldsymbol{s}^j$ indicate how much the value of the objective f_i is going to change with respect to one unit change in nonbasic variable x_j if this variable enters the basis. Hence, in this method objective trade-off information is shown to the DM in the form of vector $\boldsymbol{s}^j$. The method proceeds as follows:

1. Set $h = 1$ and initial weights $\boldsymbol{w}^h$ (e.g. equal weights). Set $W = \emptyset$, this set will contain additional constraints to be generated during the method.
2. Use the simplex method to solve problem (10) and to obtain a Pareto optimal extreme solution $\boldsymbol{x}^h$.
3. Determine the index set $J^h(\boldsymbol{x}^h)$ that contains all nonbasic columns which lead to adjacent Pareto optimal extreme solution from $\boldsymbol{x}^h$ and have not been considered before (see [36, Appendix 1] or [4, pp. 236] for more details).
4. Present $\boldsymbol{x}^h$ and $C\boldsymbol{x}^h$ to the DM. For each index $j \in J^h(\boldsymbol{x}^h)$, present values $\boldsymbol{s}^j \in \mathbb{R}^k$ to the DM. Each component s_i^j of column vector $\boldsymbol{s}^j$ indicates how much the value of objective f_i is going to change if the variable x_j changes by one unit. By comparing the values s_i^j $(i = 1, \dots, k)$ related to each objective function, the DM is getting an idea of trade-offs. If the DM is willing to make a proposed trade-off, add constraint $-(\boldsymbol{w}^h)^T \boldsymbol{s_j} > 0$ permanently to set W. If the DM is not willing to make a trade-off, add constraint $-(\boldsymbol{w}^h)\boldsymbol{s_j} < 0$ permanently to set W. If the DM is indifferent, add constraint $-(\boldsymbol{w}^h)\boldsymbol{s_j} = 0$ permanently to set W.
5. If for each $j \in J^h(\boldsymbol{x}^h)$ the DM is not willing to make a trade-off or is indifferent with all j, then the current Pareto optimal extreme solution $\boldsymbol{x}^h$ is the final solution and the procedure can be stopped. Otherwise continue.
6. Find weight vector $\boldsymbol{w}^{h+1}$ satisfying constraints $\sum_{i=1}^k w_i = 1$, $w_i > 0$, for all $i = 1, \dots, k$, and constraints stacked to the set W for each $j \in J^h(\boldsymbol{x}^h)$ during this and all the previous iterations. Set $h = h + 1$ and go to step 2.

Remark 6.1. In the above procedure, the trade-off information shown to the DM is related to the question whether to move to some Pareto optimal extreme solution or not if a certain trade-off will occur. As pointed out, in the original form of this method, trade-off information is not shown to the DM in the form given in Definition 3.3. However, it is possible to show all or some of the trade-off rate values $t_{pq}(\boldsymbol{x}^h, \boldsymbol{e}^j) = s_p^j / s_q^j$ to the DM, where we have a unit vector $\boldsymbol{e}^j \in \mathbb{R}^n$, having components $e_i^j = 0$ for $i \neq j$ and $e_i^j = 1$ for $i = j$. The value $t_{pq}(\boldsymbol{x}^h, \boldsymbol{e}^j)$ reflects the trade-off rate involving the objectives f_p and f_q if we move to a Pareto optimal extreme solution which corresponds to the simplex tableau operation where nonbasic variable x_j enters the basis.

One drawback of this method is that because the simplex method is used, the method produces only Pareto optimal solutions that are extreme points of the underlying linear problem. However, in some cases the DM might be interested in intermediate Pareto optimal solutions as well. Another drawback is the stopping condition which is build into the method, in other words, the method will terminate in a finite number of steps and the DM is not totally free to decide whether he/she is satisfied with the final solution or not. In addition, the assumption related to existence of a linear value function might be too restricting.

6.2 Trade-off rates to the direction of an objective function gradient (IMOOP)

In the previous section, we did consider coordinate directions as the direction $\boldsymbol{d}$ in Definition 3.3. In paper [27] Tappeta and Renaud have proposed the interactive multi-objective optimization procedure (IMOOP) in which the concept of so called trade-off matrix is used. The problem with this method is that in the original paper [27] the presentation is not precise and most of the results are stated without any rigorous proofs. Therefore, we do not describe here the complete algorithm of the IMOOP method but only outline the most relevant ideas which are directly related to the utilization of objective trade-off information. Later on, we will see that also ISTM method utilizes similar kind of idea to aid the DM. However, from the trade-off concept point of view the unique idea related to the IMOOP method is the way how trade-off rate information is computed to the direction of an objective function gradient instead of the direction of which is producing partial trade-off rates.

In the IMOOP method the Pareto optimal solutions are generated using the Chebychev scalarization problem (6). At each iteration h of this interactive solution procedure the DM determines an *aspiration level* $\bar{z}_i^h$ for each objective f_i, $i = 1, \dots, k$ to reflect the levels (s)he is interested in achieving. These aspiration levels are used to set scaling factors as $w_i^h = 1/(\bar{z}_i^h - z^{\star\star})$ in the Chebychev scalarization problem (6). Let us denote the corresponding solution by $\boldsymbol{x}^h$.

At the current Pareto optimal solution $\boldsymbol{x}^h$ the DM is able to do trade-off analysis by using so called *trade-off matrix* containing total trade-offs computed for each objective f_j to the direction which gives the largest feasible improvement for that particular objective. If there is no active constraints at the current Pareto optimal solution $\boldsymbol{x}^h$, that is, $g_i(\boldsymbol{x}^h) < 0$, for each $i = 1, \dots, m$, then largest improvement for objective f_j at $\boldsymbol{x}^h$ is of course to direction $-\nabla f_j(\boldsymbol{x}^h)$. In the light of the Definition 3.3 we can compute for each objective f_j total trade-offs $t_{ij}(\boldsymbol{x}^h, -\nabla f_j(\boldsymbol{x}^h))$, for each $i = 1, \dots, k$.

In the case where we have active constraints at $\boldsymbol{x}^h$, that is, $g_i(\boldsymbol{x}^h) = 0$, for some $i = 1, \dots, m$, the gradient vector $-\nabla f_j(\boldsymbol{x}^h)$ is not necessarily pointing to the feasible set S. To overcome this problem we can project corresponding $-\nabla f_j(\boldsymbol{x}^h)$ to the feasible set S. This can be done by introducing a matrix J where the gradient vectors $\nabla g_i(\boldsymbol{x}^h)$ of the active constraints at $\boldsymbol{x}^h$ are as a row vectors. The projection matrix $M = I - J^T(JJ^T)^{-1}J$, where I is the identity matrix, can be used to project the gradient vector $-\nabla f_i(\boldsymbol{x}^h)$ to the direction $\boldsymbol{d}^{i;h} = -M\nabla f_i(\boldsymbol{x}^h)$ which is a tangent direction for the feasible set S at $\boldsymbol{x}^h$.

Remark 6.2. For instance, the situation where $M = 0$ is not considered in [27]. However, this problem can be approached with a similar kind of technique which is used in the context of the gradient projection method of Rosen, proposed for the single objective optimization (see, e.g., [1]). Furthermore, the projection procedure might produce an improving feasible direction but it is in a general case unclear whether this direction is pointing to the set of Pareto optimal solutions.

We denote by $\boldsymbol{d}^{i;h}$ the projection direction related to gradient vector $-\nabla f_i(\hat{\boldsymbol{x}})$ at iteration h. At the current solution $\boldsymbol{x}^h$ the following *trade-off matrix* is shown to the DM.

$$\begin{pmatrix} t_{11}(\boldsymbol{x}^h; \boldsymbol{d}^{1;h}) & \dots & t_{k1}(\boldsymbol{x}^h; \boldsymbol{d}^{1;h}) \\ \vdots & \ddots & \vdots \\ t_{1k}(\boldsymbol{x}^h; \boldsymbol{d}^{k;h}) & \dots & t_{kk}(\boldsymbol{x}^h; \boldsymbol{d}^{k;h}) \end{pmatrix} \tag{11}$$

The row j in the matrix above reflects what is the total trade-off involving objectives f_i and f_j when we move to the feasible direction that (in a certain sense) produces the largest descent for the objective f_j. It must be pointed out that the diagonal entry $t_{ii}(\boldsymbol{x}^h; \boldsymbol{d}^{i;h}) = 1$, for each $i = 1, \dots, k$.

In [27], it is proposed that also partial trade-off rates $t_{ij}(\boldsymbol{x}^h)$ can be produced using the projection idea presented above. In such a case the matrix J containing gradients of active constraints will also contain the gradient vectors $\nabla f_p(\boldsymbol{x}^h)$ for $p \neq i, j$. In other words, we are projecting the direction $-\nabla f_j(\boldsymbol{x}^h)$ to the direction which is feasible (or at least a tangent direction at the boundary of S) and where the change in the objectives f_p is zero, for all $p = i \neq j$.

Trade-off rate information can be considered as a first-order approximation of the Pareto surface. In addition to this first order approximation, the IMOOP method produces also a second order approximation for the Pareto surface. However, we omit further details here because those are not directly related to the objective trade-off concept in the sense it is treated in this paper. Even thought, in [27] Tappeta and Renaud present this unique way to produce trade-off matrix, they do not give any concrete ideas how this information can be explicitly used by the DM to set aspiration levels for the next iteration of an interactive procedure. Therefore, in the context of the IMOOP method the trade-off matrix is offered for the analysis purposes and based on this information the DM is free to make any conclusion (s)he likes. The purpose of objective trade-off information is just to reflect overall behavior of the Pareto surface.

6.3 Trade-off and approximation of implicit value function (ISWT)

The previous methods did utilize trade-off rate information computed using simplex multipliers or Definition 3.3. In the rest of the methods to be presented we consider trade-off rates which are produced by using the connection to KKT-multipliers (this was discussed in Section 5).

If partial trade-off rate involving two objectives f_i and f_j is presented to the DM at some Pareto optimal solution, the DM can indicate how preferable the trade-off is. This kind of information can be interpreted in a way that it reflects the shape of an implicit value function. If at some Pareto optimal solution the DM is indifferent to the underlying trade-offs this indicates that the DM is satisfied with this solution. In other words, the DM does not prefer any kind of change.

The interactive surrogate worth trade-off (ISWT) method [3] which is an interactive extension of the surrogate worth trade-off (SWT) method [8], utilizes a technique where the DM indicates her/his preferences toward to objective trade-offs at a Pareto optimal solution. In the ISWT method, the ϵ-constraint scalarization (4) is used to produce Pareto optimal solutions and upper bounds ϵ_j in constraints $f_j(\boldsymbol{x}) \leq \epsilon_j$ are produced from a local approximation of an implicit value function. At each iteration, preference information is obtained from the DM by asking questions related to trade-off rates at the current Pareto optimal solution. In what follows, it is assumed that conditions of Theorem 5.1 are satisfied.

The basic steps of the ISWT method are as follows:

1. Set $h = 1$. Select one objective f_i as a primary objective, and set initial values of $\boldsymbol{\epsilon}^h$.
2. Obtain a Pareto optimal solution $\boldsymbol{x}^h$ by solving ϵ-constraint scalarization problem (4). Compute trade-off rates by using equation (7).
3. Obtain surrogate worth values W_{ij}^h from the DM. Values W_{ij}^h are used to indicate how preferable the trade-offs are at the current solution $\boldsymbol{x}^h$. If the DM does not want to move from current solution, then stop. Otherwise continue.
4. Use W_{ij}^h values and $\boldsymbol{\epsilon}^h$ to form $\boldsymbol{\epsilon}^{h+1}$. Go to step 2.

In step 2, trade-off rates are produced using the results of Theorem 5.1, where the KKT-multiplier λ_j was related to objective constraint f_j of the ϵ-constraint scalarization problem (4) in such a way that it reflects the amount of change in the value of the primary objective f_i when the value of the constraint objective f_j is changing by one unit. If at iteration h, the KKT-multipliers are positive, that is, $\lambda_j^h > 0$, for all $j = 1, \ldots, k$ and $i \neq j$, then in step 3 the following question is presented for each $j \neq i$ to the DM:

> Question 1: "Consider the current objective values $\boldsymbol{f}(\boldsymbol{x}^h)$. How keen are you to improve f_i by λ_j^h units for each one unit degrade in the value of the objective f_j if all the other objectives f_p, $p = 1, \ldots, k$ and $p \neq i, j$, keep their current values?"

The DM answers to the previous question using integers from -10 to 10 (surrogate worth values). Number 10 means that the DM wants to make the proposed trade-off and number -10 indicates that the DM wants to make an opposite trade-off. Number 0 indicates that the DM is indifferent with the proposed change. These integers are assigned to W_{ij}^h for all $j = 1, \ldots, k$ and $j \neq i$. In step 3, the procedure is stopped if $W_{ij}^h = 0$ for all $j = 1, \ldots, k$, $j \neq i$, which indicates that the DM does not want to make any changes to the current solution $\boldsymbol{x}^h$.

We recall from Theorem 5.1 that the KKT-multipliers related to objective constraints at the solution of the ϵ-constraint problem may not be strictly positive in a general case. In such a case for the objectives constraints f_j having KKT-multiplier $\lambda_j^h = 0$ the following question is presented to the DM:

Question 2: "Consider the current objective levels $\boldsymbol{f}(\boldsymbol{x}^h)$. How keen are you to improve f_i by λ_j^h units and change the values of objectives f_p (for $p = 1, \ldots, k$ and $p \neq i, j$), by $\nabla f_p(\boldsymbol{x}^h)^T \partial \bar{\boldsymbol{x}}(\boldsymbol{\epsilon}^h)/\partial \epsilon_j$ units, while the value of objective f_j degrades by one unit."

This question is related to the case c) in Theorem 5.1, and as pointed out earlier some additional computation is needed to produce the term $\partial \bar{\boldsymbol{x}}(\boldsymbol{\epsilon}^h)/\partial \epsilon_j$. In other words, this kind of a situation can be handled but it is unfavorable from a computational perspective.

When W_{ij}^h values are obtained, a new bound ϵ_j can be set for each constraint objective f_j, for all $j = 1, \ldots, k$, $j \neq i$, of ϵ-constraint scalarization problem (4). If constraint objective f_j ($j = 1, \ldots, k$, $j \neq i$) has a strictly positive KKT-multiplier, the constraint can be updated by setting

$$f_j \leq \epsilon^{h+1} = \epsilon^h + \alpha^h \left(W_{ij}^h |f_j(\boldsymbol{x}^h)| \right)$$

If the KKT-multiplier of some constraint objective f_j ($j = 1 \ldots, k$, $j \neq i$) is zero then the constraint can be updated by setting

$$f_j \leq \epsilon^{h+1} = \epsilon^h + \alpha \left(\nabla f_j(x^h)^T \left(\frac{\partial \bar{\boldsymbol{x}}(\boldsymbol{\epsilon}^h)}{\partial \epsilon_j} \right) W_{ij}^h |f_j(\boldsymbol{x}^h)| \right)$$

In the above formulas, the parameter α is varied in a relatively small interval $(0, \bar{\alpha}) \in \mathbb{R}$ and for each of these α values a new ϵ-constraint problem is solved. Solutions obtained can be then visualized to the DM who is able to point out the most preferred ϵ_j^{h+1} which determines a new upper bounds for the ϵ-constraint problem (4).

In the original method formulation surrogate worth values W_{ij}^h are assigned as integers in the range $[-10, 10]$. However, Tarvainen has suggested in [28] that limiting the range to $[-2, 2]$ might help the DM to make more consistent decisions because there is less where to choose from.

At most $(k - 1)$ trade-off related questions are asked at each iteration. This happens when all the KKT-multipliers related to constraint objectives f_j are strictly positive, when the DM is assumed to compare two objective at a time and indicate whether the change is favorable or not with the assumption that all other objectives remain at their current levels. When some of the KKT-multipliers are zero, then Question 2 is presented to the DM for every objective which has a positive KKT-multiplier. However, even though we have fewer questions, the DM has to consider changes involving several objectives and this may cause cognitive burden especially when the problem considered contains several objectives.

6.4 Automatic trade-off to aid the DM (STOM)

The satisficing trade-off method (STOM) [21] is based on a classification idea. During the interactive procedure, the DM is assumed to classify objective functions at each iteration h to the three classes: Objectives f_i ($i \in I_I^h$) that should be improved, objectives f_i ($i \in I_R^h$) that can be relaxed, and objectives f_i ($i \in I_A^h$) which are accepted as they are. At each iteration, the DM indicates at the current Pareto optimal solution his/her preferences as aspiration levels $\bar{z}_i$ for objectives to be improved and the method uses trade-off rate information to automatically propose values for aspiration levels related to objectives to be relaxed, this procedure is called an *automatic trade-off*. The DM is always allowed to change the automatically set aspiration levels, but especially in the case of problems that have very many objective functions this kind of support can be useful because it can be used to decrease cognitive burden set on the DM. Technically, in this method Chebyshev scalarization problem (6) is used, where the scaling parameter $w_i^h = 1/(\bar{z}_i^h - z_i^{\star\star})$ is set at each iteration h by using the aspiration level $\bar{z}_i^h$ related to objective f_i, for all $i = 1, \ldots, k$.

1. Set $h = 1$ and ask the DM to specify aspiration levels $\bar{z}_i^h$ for each f_i ($i = 1, \ldots, k$).
2. Solve Chebyshev problem (6) to obtain a Pareto optimal solution $\boldsymbol{x}^h$.
3. Show $\boldsymbol{f}(\boldsymbol{x}^h)$ to the DM. Ask the DM to classify objective functions f_i ($i = 1, \ldots, k$) to classes I_I^h, I_A^h, and I_R^h. If $I_I^h = \emptyset$, stop. Otherwise, ask the DM to specify new aspiration levels $\bar{z}_i^{h+1}$ for objective functions f_i, where $i \in I_I^h$. Set $\bar{z}_i^{h+1} = f_i(\boldsymbol{x}^h)$ for f_i, where $i \in I_A^h$.

4. Use automatic trade-off to set new aspiration levels $\bar{z}_i^{h+1}$ for objective functions f_i classified to set I_R^h. In other words, set $\bar{z}_i^{h+1} = f_i(\boldsymbol{x}^h) + \frac{1}{|I_R^h|\lambda_i w_i} \sum_{j \in I_I^h} \lambda_j w_j (f_j(\boldsymbol{x}^h) - \bar{z}_j)$, for $i \in I_R^h$, where λ_i is optimal KKT-multiplier related to f_i in problem (6) and $|I_R^h|$ is the number of objectives classified to class I_R^h.
5. Set $h = h + 1$ and go to step 3.

Like trade-off rate information generally, also automatic trade-off information is giving only a linear approximation about how much relaxed objectives are going to change with respect to a change in objectives the DM has selected to be improved. For linear and quadratic problems automatic trade-off information can be interpret as an exact information, this is means that it is possible to obtain Pareto optimal solutions without needing to solve a scalarization problem (see, e.g., [19] and [20]).

6.5 Determining allowed change using trade-off table (ISTM)

Even though, in the IMOOP method we already introduced the concept of a trade-off matrix we did not considered how this kind of information can be explicitly used by the DM in an interactive solution procedure. Next, we present the interactive step trade-off method (ISTM) [31] to demonstrate the usage of trade-off matrix when determining aspiration levels. The basic idea in the ISTM method is quite similar to the STOM method. At each iteration, the DM is shown some Pareto optimal solution and some supporting information. If the DM is not satisfied with the solution shown then (s)he classifies the objectives to the three classes which are the same as in the STOM method. In other words, at each iteration h all k objective functions of problem (1) are classified to classes I_I^h, I_A^h, and I_R^h. For each objective f_i which can be relaxed (class $i \in I_R^h$) the DM is assumed to determine amount of relaxation according to the trade-off rate information related to current Pareto optimal solution. The difference to the STOM method is that the DM determines the aspiration levels for the objectives that are relaxed by using the trade-off matrix, in the STOM method these aspiration levels are proposed automatically.

The scalarization problem solved in ISTM method differs slightly from scalarization problems (i.e. problems (4) and (6)) used in the other interactive methods presented in this paper. In the ISTM method the following (simplified) auxiliary problem is solved

$$\begin{array}{lll}
\underset{\boldsymbol{\alpha},\boldsymbol{x}}{\text{maximize}} & \sum_{i \in I_I^h} \alpha_i^h & \\
\text{subject to} & f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^{h-1}) - w_i \alpha_i^h, & \text{for } i \in I_I^h \\
& f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^{h-1}), & \text{for } i \in I_A^h \\
& f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^{h-1}) + \Delta f_i, & \text{for } i \in I_R^h
\end{array} \tag{12}$$

where, for instance, $w_i = (z_i^\dagger - z_i^{\star\star})$ reflecting the possible value range of the objective f_i. Let us denote by $|I_I^h|$ the number of objectives classified to class I_I^h at iteration h. The vector $\boldsymbol{\alpha^h} \in \mathbb{R}^{|I_I^h|}$ contains auxiliary variables $\alpha_i^h \geq 0$. By looking at the first constraint $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^{h-1}) - w_i \alpha_i$ we can see that the aim in the auxiliary problem is to maximize an improvement for the objectives classified to class I_I^h. At each iteration h, the DM is required to determine an amount of relaxation Δf_i related to objective f_i ($i \in I_R^h$) by using the trade-off rate information at the current solution.

The partial trade-off rates are determined by the optimal KKT-multiplier λ_i^h related to constraints based on class I_R^h. Therefore, assumptions similar to the presented in Theorem 5.1 must be satisfied. For problem (12) at obtained Pareto solution $\boldsymbol{x}^h$ partial trade-off involving objectives f_i, for $i \in I_I^h$, and f_j, for $j \in I_R^h$, is $t_{ij}(\boldsymbol{x}^h) = -w_i \lambda_j^h$ [31]. In other words, at the current solution the partial trade-off rate is used to produce a linear approximation for the change involving two objectives, with the assumption that the other objectives can be kept at their current levels. The ISTM method proceeds as follows:

1. Set $h = 1$ and produce an initial Pareto optimal solution $\boldsymbol{x}^h$ by solving the Chebyshev scalarization problem (6) (e.g. by using scaling parameters $w_i = 1/(z_i^\dagger - z_i^{\star\star})$).
2. The DM is required to classify the objectives to I_I^h, I_A^h, and I_R^h by using information based on vectors $\boldsymbol{f}(\boldsymbol{x^h})$, $\boldsymbol{z}^\star$ and $\boldsymbol{z}^\dagger$.

3. By examining the current solution and the corresponding partial trade-off rates the DM is required to assign values Δf_j^h, for all $i \in I_R^h$ and then answering the questions Q_j^h : "Suppose that objective f_p remains its current level $f_p(\boldsymbol{x}^h)$, for $p = 1, \ldots, k, p \neq i, j$. If the objective values $f_j(\boldsymbol{x}^h)$, for $j \in I_R^h$, are relaxed to the values $f_j(\boldsymbol{x}^h) + \Delta f_j$ while objective values $f_i(\boldsymbol{x}^h)$, for $i \in I_I^h$, are improved to the values $f_i(\boldsymbol{x}^h) - w_i \lambda_j \Delta f_j$, do you consider this trade-off worthwhile?". The DM must assign values Δf_j^h again if the corresponding trade-off is not satisfying her/him. If the DM sets all $\Delta f_j^h = 0$, the procedure is stopped.
4. Form and solve the auxiliary problem (12) using information obtained from the DM. The obtained Pareto optimal solution is $\boldsymbol{x}^{h+1}$ and λ_j^{h+1} is the optimal KKT-multiplier related to constraints of class I_R^h
5. If the DM is satisfied with the current solution $\boldsymbol{x}^{h+1}$, stop. Otherwise, set $h = h + 1$ continue to step 2

In step 3, the DM is assumed to use partial trade-off rates that are presented in a matrix form where component $t_{ij}(\boldsymbol{x}^h)$ is indicating partial trade-off between a relaxed ($i \in I_R^h$) and an improved ($j \in I_I^h$) objective. It is quite clear that when the number of objectives in the problem is very high the analysis of numerical trade-off matrix can be quite demanding for the DM. An important aspect to notice is the fact that the DM is not forced to use the trade-off matrix in any way or is allowed to interpret it very freely. However, unlike in the IMOOP method, in the ISTM method some outlines are given how the DM should use trade-off information to determine aspiration levels for the next iteration. In this sense the trade-off matrix can be considered as additional supporting information that is not directly related to the method itself. Therefore, it might be useful to consider the trade-off rate matrix as supporting information in the context of any interactive method.

6.6 Trade-off rate and MRS as a stopping condition (SPOT and GRIST)

From the trade-off perspective, the main idea in this method is the stopping condition that uses MRS's and partial trade-off rates. The use of MRS necessitates the assumption that the DM has an implicit value function. The aim is to determine such a Pareto optimal solution $\hat{\boldsymbol{x}}$ where the trade-off rate and MRS values are concurrent, that is, $m_{ij}(\hat{\boldsymbol{x}}) = -t_{ij}(\hat{\boldsymbol{x}})$, for each $j = 1, \ldots, k$ and $j \neq i$. This idea was already illustrated in Figure 2. This kind of stopping condition necessitates that the problem is convex, that is, it is assumed that functions f_i (i=1,...,k) and g_i ($i = 1, \ldots, k$) of problem (1) are convex. The sequential proxy optimization technique (SPOT) [22] utilizes this idea. The method uses the ϵ-constraint scalarization (4) and the MRS values are given by the DM. The MRS values are used to reflect the shape of an implicit value function.

The method starts from some Pareto optimal solution where the DM provides MRS information. If the stopping condition mentioned above is not fulfilled then a search direction is generated to the set of feasible objective vectors and using this direction a special proxy value function, that is used to simulate an implicit value function, is maximized. The solution maximizing proxy function is Pareto optimal. If the DM feels that the solution obtained is better than the previous one, then the DM is assumed to provide new MRS information and a new feasible direction is generated. Otherwise, more interesting solutions can be searched for between the previous and the current solution.

It is assumed that at each Pareto optimal solution $\boldsymbol{x}^h$ assumptions similiar to the presented in Theorem 5.1 are satisfied. This means that at a Pareto optimal solution considered partial trade-off rate involving the primary objective f_i and some objective constraint f_j in the ϵ-constraint scalarization (4) can be obtained using the optimal KKT-multiplier λ_j^h related to f_j. Furthermore, it is assumed that the DM has an implicit value function which is concave, strictly monotonically decreasing and continuously differentiable with respect to objectives f_i ($i = 1, \ldots, k$). Note that because we are dealing with the ϵ-constraint scalarization problem (4), we have the relation $t_{ij}(\hat{\boldsymbol{x}}) = -\lambda_j$ between partial trade-off rates and KKT-multipliers. The method proceeds as follows:

1. Set $h = 1$. Select one objective function f_i as a primary objective and set initial values for the upper bound vector $\boldsymbol{\epsilon}^h$ in ϵ-constraint scalarization problem (4).

2. Obtain a solution $\boldsymbol{x}^h$ by solving the ϵ-constraint scalarization problem (4) (all objective constraints must be active at the obtained solution, that is, $f_j(\boldsymbol{x}^h) = \epsilon_j^h$, for all $j = 1, \ldots, k-1$).
3. Ask the DM to determine MRS values m_{ij}^h related to the current objective vector $\boldsymbol{f}(\boldsymbol{x}^h)$.
4. If $\left|m_{ij}^h - \lambda_{ij}^h\right| < \delta$, for all $i = 1, \ldots, k$ and $i \neq j$, where δ is a small strictly positive real number, then stop. The vector $\boldsymbol{x}^h$ is a final solution. Otherwise continue.
5. Set $\Delta\epsilon_j^h = -m_{ij}^h - \lambda_{ij}^h$, for each $j = 1, \ldots, k-1$ and this determines the direction vector $\Delta\boldsymbol{\epsilon}^h$ where the value of an implicit value function is improving.
6. Use a proxy function P to locally approximate the implicit value function and determine a step size α^h in such a way that P is maximized at a point satisfying ϵ-constraints $f_j(\boldsymbol{x}^h) \leq \epsilon_j^h + \alpha^h \Delta\epsilon_j^h$ (see [22], for more details).

Above we have given only a very rough outline of the SPOT method. From trade-off information point of view, the most important aspect with this method is how MRS's and trade-off rates are utilized in the stopping condition. Therefore, we do not go into details, for instance, with consistency checks related to MRS information or possible proxy function forms. If the DM is consistent in giving MRS information then $k-1$ MRS values at each iteration is needed. However, it is not necessarily easy for the DM to be consistent with MRS information.

The GRIST method [30, 32] is using the similar stopping condition as SPOT. In the GRIST method partial trade-off rate information obtained at the current Pareto optimal solution $\boldsymbol{x}^h$ is used to produce the tangent hyperplane for the Pareto surface. The DM is assumed to reflect the shape of an implicit utility function by offering MRS information. The MRS information is used to produce a gradient vector of an implicit value function at $\boldsymbol{x}^h$ and this gradient is projected to the tangent hyperplane of the Pareto surface. By using the preference information given by the DM a step size is determined to the direction of the projected gradient of an implicit value function, this determines a reference point (aspiration levels) belonging to the tangent hyperplane. The reference point is then projected to the set of Pareto optimal solutions. When the procedure is repeated we end up eventually in a situation where the normal vector of the tangent hyperplane and the gradient of an implicit value function coincide. This is similar to the stopping condition described above and explained in the context of Figure 2.

7 Further analysis and discussion

In the methods presented trade-off information is mainly produced by using the trade-off interpretation related to the KKT-multipliers, like described in the case of ϵ-constraint scalarization problem (4) in Theorem 5.1. Theorem 5.1 (and similar results) can be used to produce partial trade-off rates instead of total trade-off rates. In a general case partial trade-off rates have probably more intuitive interpretation than total trade-off rates because by considering partial trade-offs the DM can concentrate on pairwise comparison of objectives instead of considering change in all objectives simultaneously. This might be the reason why in methods it is more popular to use results like Theorem 5.1 than to compute objective trade-off rates directly from the Definition 3.3.

One main drawback related to Theorem 5.1 (and similar results) is that an underlying multiobjective optimization problem must fulfill quite demanding assumptions. However, in this sense results like Lemma 5.2 are useful because these can be used to test whether the necessitated assumptions are satisfied at some Pareto optimal solution. Even thought, in practice it might be too expensive to produce necessitated second order information accurately enough (e.g. most of the numerical single objective optimization methods avoid direct computation of any kind of second order information, which means, that such information is not necessarily available without an additional computation.)

In addition, in Theorem 5.1, the case where the number of objectives is greater than the number of decision variables is likely to cause problems because of the degenerate objective constraints related to a scalarization function utilized. However, in practice this is not necessarily a serious problem because usually we have a relatively large amount of variables and only few objectives.

Even thought, the assumptions in Theorem 5.1 can be considered as a drawback it is of course favorable that an interactive method is able to handle such special cases when, for example, some of those assumptions is not fulfilled. However, if we consider the methods presented, in their basic form, only the ISWT method directly addresses the problem which arises when the assumptions of Theorem 5.1 (case c)) are not completely fulfilled.

As an alternative approach Definition 3.3 is convenient especially when we have some direction which produces meaningful trade-off rates for the DM (direction such as the gradient of an objective function). Like seen in the case of the IMOOP method a direction can be projected to the feasible set. However, the IMOOP method introduces only one possible way to do a projection, in a general case this can be done infinitely many ways (some possible examples are presented e.g. in [1, pp. 411]. Which is the best way to do the projection depends on what we want to achieve. Which ever is the utilized projection style it is important that the DM has a clear picture what the trade-off rates to the projected direction are trying to reflect. As pointed out, in the case of the IMOOP method also partial trade-off rates can be obtained through a projection. However, such approaches have not yet been studied in mathematically rigorous way.

Event thought, objective trade-off rate information is utilized successfully in many methods there is also many problems related to this kind of information. First of all, it must be kept in mind that trade-off rate information gives us only a linear approximation about what is happening at the Pareto surface in some local neighborhood of the select Pareto optimal solution. If the underlying multiobjective optimization problem is not highly nonlinear then linear approximation might be very useful when supporting the DM. On the other hand, if we are dealing with highly nonlinear problems linear approximation may give a very misleading idea about the interconnections between objective functions. Furthermore, if a linear approximation is accurate enough only a very small neighborhood of some considered Pareto optimal solution this kind of information might be irrelevant for the DM. The aim is to build confidence not to distract the DM. In the case of a general nonlinear multiobjective optimization problem it might be difficult to determine beforehand how reliable a linear approximation is in the case of some particular problem. Therefore, more research is needed to produce results which can be used to produce reliable error approximation for trade-off rate computations.

Figure 3 illustrates some problems that might occur when we are using trade-off rate information. In Figures a) and b) we can see that even if the Pareto surface is very smooth and connected trade-off rate information can give a completely misleading picture. In b) the Pareto surface is smooth and almost linear, excluding a small bump in the middle which causes trade-off rate information point totally irrelevant direction. It must be emphasized that especially in the case of several objectives this may be a serious problem because the DM do not usually have any kind of visualization of the objective space at hand. In c) the disconnected Pareto surface causes a problem when trade-off rate is evaluated at very close of the boundary of the relative interior of the Pareto surface. In d) the Pareto optimal set is disconnected, Pareto surface is nonsmooth and connected, and overlapping dark gray lines present two sets of objective vectors that can be considered as Pareto optimal if consideration is restricted one at a time to some local area in the decision space. This behavior might be relevant because the direction in Definition 3.3 is defined locally so it is not possible to directly identify the problem occurring in d).

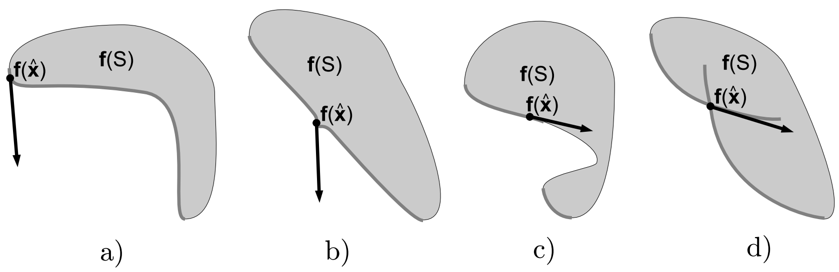

Figure 3: Some problems related to trade-off rate information

Despite the potential problems mentioned above trade-off rates, when available, can be very useful to capture the shape of the Pareto surface at a local neighborhood of some Pareto optimal solution. In the context of interactive multiobjective methods the most appealing feature related to the trade-off rate concept is that they can reflect available solution possibilities without additional computation. If we consider the methods surveyed in this paper we can notice that they utilize trade-off information at a couple of different levels. By examining the methods, we can identify at least the following three main classes.

A Trade-off rate information is used to describe the overall shape of the Pareto surface. Trade-off rates are shown to the DM, for instance, in a matrix form and the DM has not been given any explicit rule how to utilize this information. In other words, trade-off rate information is presented only for analysis purposes as additional supporting information. In this paper IMOOP and ISTM methods can be seen partly to fit into this class.

B Trade-off rate information is used in explicit way to predict what is happening for the other objectives when the DM has already stated some preference information related to some other objectives. The methods STOM and ISTM utilize this idea.

C Trade-off rate information is used with a connection to an implicit value function. The methods SPOT and GRIST use objective trade-off rates and MRS to determine a stopping condition to an interactive procedure. In the ISWT and ZW methods trade-off rates are used to produce questions which aim to capture the shape of an implicit value function.

The classification above can be also interpret in such a way that in class A methods offer trade-off information in a very general form to reflect what kind of trade-offs are available around of some Pareto optimal solution. This information can be very useful to quickly analyze whether that solution could be improved in a local neighborhood to produce solution which corresponds more closely the preferences of the DM. However, even though the idea seems to be useful the problem might be that the numerical trade-off rate matrix causes quite a much cognitive burden for the DM, especially when the number of objectives increases.

In the methods belonging to class B the DM indicates some preference information which is then used with underlying objective trade-off rate information to approximate into what kind of solutions the given preferences might lead. The methods in this class use trade-off information just to reflect or approximate what kind of Pareto optimal solutions are available.

The methods in class C are interpreting objective trade-off information in more explicit way. In other words, 'what should be' type of information (subjective trade-off information) is obtained from the DM and it is, in a way or another, compared to 'what is' type of information (objective trade-off information). The aim is to find such a Pareto optimal solution where objective and subjective information corresponds to each other.

It is also important to notice that the methods in class A are more learning-oriented than the methods in class B or C. On the other hand, methods in class C are more decision making oriented by assuming that the DM should know more precisely what (s)he wants. This kind interpretation suggests that trade-off ideas used in class A are more suitable for situations where the DM is just trying to capture an overall idea related to the available Pareto optimal solutions. Class C contains some ideas that are usable when the DM is aware of his/her preferences. The trade-off ideas in class B are somewhere in the middle. This observation is quite natural and fits well to the context of multiobjective optimization problem solving where it can been seen that many times so called learning phase precedes the decision making phase.

8 Conclusions

In this paper we have discussed about the concept of objective trade-off information and how it has been so far utilized in the interactive multiobjective optimization methods presented in the literature. We have outlined, from a practical perspective, some important theoretical ideas behind objective trade-off information and explained how this kind of information is typically produced in interactive methods. The methods have been selected to this paper based on the originality of the objective trade-off idea they are utilizing. Therefore, possible extensions and applications of the methods are not discussed unless they have contributed to the underlying trade-off utilization idea. Based on the analysis made, we can conclude that trade-off information can be used at different levels and different phases of an interactive solution process.

At which level the trade-off information should be used is of course a matter of taste. The most appropriate approach can be selected according to what kind of information the DM is able or wants to use in the solution process. It must be emphasized that trade-off information offers only one way to present and ask information in interactive methods. In an ideal case the DM is able to indicate which kind of method is the most suitable for some particular problem or solution phase. In other words, trade-off type of information can be useful but it necessitates that it is natural for the DM to consider relative changes between objectives. It is clear that some multiobjective problems might contain incommensurable objectives where it is very difficult (or even impossible) for the DM to consider a relative unit trade-offs between objective function values. Of course, in such a case trade-off information based methods should not be used.

It is appealing to think that objective trade-off information based ideas can be possibly easily integrated (in the sense of automatic trade-off) to existing methods as an additional decision support tools that can be made available for the DM when needed. In such a case without affecting the underlying method the DM could freely request trade-off information to grasp an idea what kind of solutions are available around a neighborhood of some particular solution.

As potential objective trade-off information related research topics we can point out at least the following ones. Definitely more research is needed related to quality of linear approximation obtained using trade-off rate information. For the DM it is important to know how reliable shown information is. It might be, for instance, possible to use sensitivity of KKT-multipliers to approximate quality of linear approximation (see e.g. [5]). It is also important to start to pay more attention to the situations where the assumptions in results like Theorem 5.1 are not fulfilled. Furthermore, a possibility of capturing a general behavior of Pareto surface around some Pareto optimal solution might be very useful for the DM, and therefore it must be studied in more detail how this kind of information can be efficiently presented to the DM without causing to much cognitive burden. In a broader context the presentation of all available trade-off directions (e.g. in the sense of [9]) at some Pareto optimal solution should be also studied in the context of practical multiobjective optimization methods.

A Appendix: Results from the single objective optimization

To present Theorem 5.1 which connects Karush-Kuhn-Tucker (KKT) multipliers to the concept of trade-off rate in the case of the ϵ-constraint scalarization (4) we need a couple of well-known definitions and results from the theory of single objective optimization. In what follows, we will present results only for inequality constrained problem. However, it must be emphasized, that all the results can be extended to include equality constraints as well. In the following we are dealing with a *nonlinear single objective optimization problem* of the form

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{minimize}} & f(\boldsymbol{x}) \\ \text{subject to} & g_i(\boldsymbol{x}) \leq 0, \quad \text{for all } i = 1, \ldots, m \end{array} \tag{13}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_i : \mathbb{R}^n \to \mathbb{R}$, are objective and constraints functions, respectively. In what follows, we denote by $I(\hat{\boldsymbol{x}}) := \{i : g_i(\hat{\boldsymbol{x}}) = 0\}$ an index set indicating constraints g_i of problem (13) which are active at some $\hat{\boldsymbol{x}}$.

Definition A.1 (Regular point)**.** It is said that $\hat{\boldsymbol{x}}$ is a *regular point* if vectors $\nabla g_j(\hat{\boldsymbol{x}})$, for $i \in I$, are linearly independent.

Now we present couple of optimality conditions related to problem (13).

Theorem A.1 (KKT Necessary Conditions)**.** *Let $\hat{\boldsymbol{x}}$ be a regular point of the constraints of problem (13). For $\hat{\boldsymbol{x}}$ to be a local minimum of problem (13) there must exist a set of KKT-multipliers $\boldsymbol{\lambda} \in \mathbb{R}^m$ such that*

$$\begin{array}{rcl} \nabla f(\hat{\boldsymbol{x}}) + \sum\limits_{i \in I(\hat{\boldsymbol{x}})} \lambda_i \nabla g_i(\hat{\boldsymbol{x}}) & = & \mathbf{0} \\ \lambda_i g_i(\hat{\boldsymbol{x}}) & = & 0, \quad \textit{for all } i = 1, \ldots, m \\ \lambda_i & \geq & 0, \quad \textit{for all } i = 1, \ldots, m \end{array} \tag{14}$$

Proof. See e.g. [16] □

Definition A.2 (Nondegenerate constraint)**.** An active constraint g_i is said to be *nondegenerate* if $\lambda_i > 0$ for each $i \in I(\hat{\boldsymbol{x}})$.

Let us divide index set $I(\hat{\boldsymbol{x}})$ into two separate sets $I_+(\hat{\boldsymbol{x}}) = \{i : \lambda_i > 0\}$ and $I_0(\hat{\boldsymbol{x}}) = \{i : \lambda_i = 0\}$ according to the quality of optimal KKT-multiplier λ_i assigned to active constraints g_i, for $i \in I(\hat{\boldsymbol{x}})$. The second order sufficiency conditions for a feasible vector $\hat{\boldsymbol{x}}$ to be local solution for problem (13) are the following.

Theorem A.2 (Second order KKT Sufficiency Conditions)**.** *Let us assume that functions f and g_i are twice continuously differentiable and ∇^2 is the Hessian matrix operator. If in addition, condition A.1 holds at $\hat{\boldsymbol{x}}$ and the matrix*

$$\nabla^2 f(\hat{\boldsymbol{x}}) + \sum_{i \in I(\hat{\boldsymbol{x}})} \lambda_i \nabla^2 g_i(\hat{\boldsymbol{x}})$$

is positive definite for all vectors $\mathbf{d} \neq \mathbf{0}$ which fulfill conditions

$$\begin{array}{rcll} \nabla g_i(\hat{\boldsymbol{x}})^T \boldsymbol{d} & = & 0, & \textit{for all } i \in I_+(\hat{\boldsymbol{x}}) \\ \nabla g_i(\hat{\boldsymbol{x}})^T \boldsymbol{d} & \leq & 0, & \textit{for all } i \in I_0(\hat{\boldsymbol{x}}) \end{array}$$

then $\hat{\boldsymbol{x}}$ is local minimum of problem (13).

Proof. See e.g. [16] □

Using the above results we can now present a theorem which can be used when KKT-multipliers are connected to the trade-off rate concept in Theorem 5.1.

Theorem A.3 (Sensitivity Theorem)**.** *Let functions f and g_i be twice continuously differentiable. In a single objective problem*

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\textit{minimize}} & f(\boldsymbol{x}) \\ \textit{subject to} & g_i(\boldsymbol{x}) \leq c_i, \quad \textit{for all } i = 1, \dots, m \end{array} \tag{15}$$

and let $\hat{\boldsymbol{x}}$ be a local feasible solution for this problem when $c_i = 0$, for all $i = 1, \dots, m$. If $\hat{\boldsymbol{x}}$ satisfies conditions

i) The vector $\hat{\boldsymbol{x}}$ is a regular point

ii) Second-order sufficiency conditions are satisfied at $\hat{\boldsymbol{x}}$

iii) The active constraints are nondegenerate at $\hat{\boldsymbol{x}}$

then there is a continuously differentiable function $\bar{\boldsymbol{x}} : \mathbb{R}^m \to \mathbb{R}^n$ defined on a neighborhood $\mathbf{0} \in \mathbb{R}^m$ such that $\bar{\boldsymbol{x}}(\mathbf{0}) = \hat{\boldsymbol{x}}$ and such that for every $\mathbf{c}$ in a vicinity of $\mathbf{0}$, a vector $\bar{\boldsymbol{x}}(\boldsymbol{c})$ is a strict local solution for problem (15). Moreover optimal KKT-multipliers related to constraints at solution $\hat{\boldsymbol{x}}$ have following interpretation

$$\frac{\partial f(\bar{\boldsymbol{x}}(\mathbf{0}))}{\partial c_i} = -\lambda_i, \quad \textit{for all } i = 1, \dots, m \tag{16}$$

Proof. See e.g. [16] □

Above Theorem A.3 can be interpreted in such a way that optimal KKT-multipliers $\boldsymbol{\lambda}$ related to the constraints of problem (15) at some solution $\hat{\boldsymbol{x}}$ indicate a relative change in the value of the objective function when the constraint requirements are changed by one unit. In other words, if active constraint g_i, for some $i = 1, \dots, m$, is relaxed by one unit then value of objective function f is improved an amount of λ_i. Of course this approximation is only linear and reflects only infinitesimal changes, but anyhow it gives an idea what is happening if constraints are altered at some solution.

References

[1] Bazaraa, M. S., Sherali, H. D., and Shetty, C. M. *Nonlinear Programming: Theory and Algorithms*, second ed. John Wiley & Sons, New York, 1993.

[2] Bowman, V. J. On the relationship of the Tchebycheff norm and the efficient frontier of multi-criteria objectives. In *Multiple Criteria Decision Making*, H. Thiriez and S. Zionts, Eds. Springer-Verlag, New York, 1976, pp. 76–85.

[3] Chankong, V., and Haimes, Y. Y. The interactive surrogate worth trade-off (ISWT) method for multiobjective decision making. In *Multiple Criteria Problem Solving*, S. Zionts, Ed. Springer-Verlag, 1977, pp. 42–67.

[4] Chankong, V., and Haimes, Y. Y. *Multiobjective Decision Making Theory and Methodology.* Elsevier Science Publishing Co., Inc., New York, 1983.

[5] Fiacco, A. V. *Inroduction to Sensitivity and Stability Analysis in Nonlinear Programming.* Academic Press, 1983.

[6] Geoffrion, A. M., Dyer, J. S., and Feinberg, A. An Interactive approach for multi-criterion optimization, with an application to the operation of an academic department. *Management Science 19*, 4 (1972), 357–368.

[7] Haimes, Y. Y., and Chankong, V. Kuhn-Tucker multipliers as trade-offs in multiobjective decision-making analysis. *Automatica 17* (1979), 59–72.

[8] Haimes, Y. Y., and Hall, W. A. Multiobjectives in water resource systems analysis: The surrogate worth trade-off method. *Water Resources Research 10*, 4 (1974), 615–624.

[9] Henig, M. I., and Buchanan, J. T. Tradeoff directions in multiobjective optimization problems. *Mathematical Programming 78*, 3 (1997), 357–374.

[10] Hunt, B. J., Blouin, V. Y., and Wiecek, M. M. Modeling relative importance of design criteria with a modified Pareto preference. *Journal of Mechanical Design 129*, 9 (2006), 907–914.

[11] Hwang, C.-L., and Masud, A. S. M. *Multiple Objective Decision Making, Methods and Applications*. Springer-Verlag, Berlin, 1979.

[12] Kaliszewski, I., and Michalowski, W. Efficient solutions and bounds on tradeoffs. *Journal of Optimization Theory and Applications 94*, 2 (1997), 381–394.

[13] Kuhn, H. W., and Tucker, A. W. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, J. Neyman, Ed. University of California Press, 1951, pp. 481–492.

[14] Kuk, H., Tanino, T., and Tanaka, M. Trade-off analysis for vector optimization problems via scalarization. *Journal of Information & Optimization Sciences 18* (1997), 75–87.

[15] Lee, G. M., and Nakayama, H. Generalized trade-off directions in multiobjective optimization problems. *Applied Mathematics 10* (1997), 119–122.

[16] Luenberger, D. G. *Linear and Nonlinear Programming.* Addison-Wesley, Reading, Massachusetts, 1973.

[17] Miettinen, K. *Nonlinear Multiobjective Optimization.* Kluwer Academic Publishers, Boston, 1999.

[18] Miettinen, K., and Mäkelä, M. M. On generalized trade-off directions in nonconvex multiobjective optimization. *Mathematical Programming 92* (2002), 141–151.

[19] Nakayama, H. Trade-off analysis based upon parametric optimization. In *Multiple Criteria Decision Support, Lecture Notes in Economics and Mathematical Systems 356*, P. Korhonen, A. Lewandowski, and J. Wallenius, Eds. Springer-Verlag, 1991, pp. 42–52.

[20] Nakayama, H. Trade-off analysis using parametric optimization techniques. *European Journal of Operational Research 60*, 1 (1992), 87–98.

[21] Nakayama, H., and Sawaragi, Y. Satisficing trade-off method for multiobjective programming. In *Interactive Decision Analysis, Proceedings of an International Workshop on Interactive Decision Analysis and Interpretative Computer Intelligence*, M. Grauer and A. P. Wierzbicki, Eds. Springer-Verlag, New York, 1984, pp. 113–122.

[22] Sakawa, M. Interactive multiobjective decision making by the sequential proxy optimization technique: SPOT. *European Journal of Operational Research 9* (1982), 386–396.

[23] Sakawa, M., and Yano, H. Generalized hyperplane methods for characterizing Λ-extreme point and trade-off rates for multiobjective optimization problems. *European Journal of Operation Research 57* (1992), 368–380.

[24] Sawaragi, Y., Nakayama, H., and Tanino, T. *Theory of Multiobjective Optimization*. Academic Press, Inc., Orlando, Florida, 1985.

[25] Shin, W. S., and Ravindran, A. A comprative study of interactive trade-off cutting plane methods for MOMP. *European Journal of Operational Research 56* (1992), 380–393.

[26] Steuer, R. E. *Multiple Criteria Optimization: Theory, Computation, and Applications*. John Wiley & Sons, 1986.

[27] Tappeta, R. V., and Renaud, J. E. Interactive multiobjective optimization procedure. *AIAA Journal 37* (1999), 881–889.

[28] Tarvainen, K. On the Implementation of the Interactive Surrogate Worth Trade-Off (ISWT) Method. In *Interactive Decision Analysis*, M. Grauer and A. P. Wierzbicki, Eds. Springer-Verlag, Heidelberg, 1984, pp. 154–161.

[29] Weidner, P. Tradeoff directions and dominance sets. In *Multi-objective Programming and Goal Programming, Advances in Soft Computing*, T. Tanino, T. Tanaka, and M. Inuiguchi, Eds. Springer-Verlag, Berlin, 2003, pp. 275–280.

[30] Yang, J.-B. Gradient projection and local region search for multiobjective optimisation. *European Journal of Operational Research 112* (1999), 432–459.

[31] Yang, J.-B., Chen, C., and Zhang, Z.-J. The interactive step trade-off method (ISTM) for multiobjective optimization. *IEEE Transactions on Systems, Man, and Cybernetics 20*, 3 (1990), 688–695.

[32] Yang, J.-B., and Li, D. Normal vector identification and interactive tradeoff analysis using minimax formulation in multiobjective optimization. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans 32*, 3 (2002), 305–319.

[33] Yano, H., and Sakawa, M. Trade-off rates in the weighted Tchebycheff norm method. *Large Scale Systems 13* (1987), 167–177.

[34] Yano, H., and Sakawa, M. Trade-off rates in the hyperplane method for multiobjective optimization problems. *European Journal of Operational Research 44* (1990), 105–118.

[35] Yu, P.-L. *Multiple-Criteria Decision Making: Concepts Techniques, and Extensions*. Plenum Press, New York, 1985.

[36] Zionts, S., and Wallenius, J. An interactive programming method for solving the multiple criteria problem. *Management Science 22* (1976), 652–663.