Discrete representations of the nondominated set for multi-objective optimization problems

Florian Jamain

UNIVERSITÉ PARIS-DAUPHINE

N° attribué par la bibliothèque

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ PARIS-DAUPHINE

Spécialité : **Informatique**

préparée au **LAMSADE**

dans le cadre de l'**École Doctorale de Dauphine**

présentée et soutenue publiquement par

Florian JAMAIN

le 27 Juin 2014

Représentations discrètes de l'ensemble des points non dominés pour des problèmes d'optimisation multi-objectifs

Discrete representations of the nondominated set for multi-objective optimization problems

Jury

Directeur de thèse :

Cristina BAZGAN, Professeur — LAMSADE, Université Paris-Dauphine

Co-encadrant :

Daniel VANDERPOOTEN, Professeur — LAMSADE, Université Paris-Dauphine

Rapporteurs :

Patrice PERNY, Professeur — LIP6, Université Paris-VI

Denis TRYSTRAM, Professeur — LIG, Grenoble INP

Examinateurs :

Hassene AISSI, Maître de conférences — LAMSADE, Université Paris-Dauphine

Kathrin KLAMROTH, Professor — University of Wuppertal

Margaret WIECEK, Professor — Clemson University

A la mémoire de mon grand-père.

Remerciements

Je tiens à remercier toutes les personnes m'ayant soutenu durant mes années de thèse.

Tout d'abord, je veux remercier Cristina et Daniel pour m'avoir encadré et accepté tel que j'ai pu être durant ces quatre années. Il va de soi que sans vous je n'aurais pas pu avancer, je vous remercie pour votre investissement. Ces années de thèse ont été, grace à vous, très agréables à mes yeux.

Je remercie également les membres du jury pour avoir accepté de relire mon manuscrit et d'être présents le jour de ma soutenance, et particulièrement Patrice Perny et Denis Trystram pour avoir accepté d'être rapporteurs.

Je veux aussi remercier les différents collègues de travail que j'ai eu le plaisir de cotoyer durant ma thèse.

Tout d'abord mes co-auteurs et amis Edouard et Abdallah. Tous les thésards qui ont fait leur thèse durant une période à peu près similaire à la mienne, à commencer par Renaud le chef de thésards, Amine le thésard modèle, Sonia dont les conseils ont toujours été précieux, Émeric pour ses nombreux coups de main, Lydia et Dalal pour leur joie communicative. Mais aussi tous ceux dont j'ai eu le plaisir de plus ou moins faire la connaissance, Lyes, Morgan, Yann, Satya, Marek, Romain, Nathanaël, Tom, Nico, Rafaël, Liang-liang, Youcef, Mohamed Amine, Samy, Basile, Afef, Raouia, Amel, Soumaya ...

Je remercie aussi certains permanents avec lequelles j'ai apprécié discuter et rigoler, tout d'abord les fameux Florian et Mohamed-Ali, mais aussi Olivier, Laurent, Jérôme, Vangélis et Tristan.

Je tiens à remercier aussi tous ceux qui m'ont soutenu en dehors du cadre de travail.

Tout d'abord, je vous remercie papa et maman de m'avoir laissé faire de si longues études, je vous aime. Je pense aussi à toute ma famille [1], notamment à ma grand-mère et à ma géniale petite soeur qui a toujours cru en son grand frère ! Je remercie aussi mes amis, qui m'ont soutenu et dont je suis fier autant

1. Sans t'oublier Cassou, chien de compagnie.

qu'il est possible de l'être, Beber, Kriril, Morgann, Jaï-dy, Romain, Lolo, Damien, Emy, Clotilde ...

Enfin, je tiens à te remercier particulièrement grand-pa. Tu ne pourras pas être là le jour de ma soutenance, j'aurais tellement aimé que ça puisse être le cas. Sans toi je n'aurais très probablement jamais été amené à faire cette thèse, je te la dédie, je pense encore beaucoup à toi, je t'aime.

Contents

1 Introduction

In many decision contexts, a decision maker bases his choice on several criteria, often conflicting. For example, when traveling by car we are interested in minimizing the fuel used and minimizing the time of the travel, when buying a product we are interested in minimizing the price and maximizing the quality, when constructing a building we are interested in minimizing the cost and maximizing the safety of the structure. This kind of practical *multi-objective* or *multicriteria* problems led to the field of research called multi-objective optimization (see [24, 25, 34] for general overviews).

In multi-objective optimization, in opposition to single objective optimization, there is typically no optimal solution i.e. one that is best for all the criteria. Therefore, the standard situation is that any solution can always be improved on at least one criterion. The solutions of interest, called *efficient* solutions, are those such that any other solution which is better on one criterion is necessarily worse on at least one other criterion. In other words, a solution is efficient if its corresponding vector of criterion values is not dominated by any other vector of criterion values corresponding to a feasible solution. These vectors, are called *feasible points* when associated to feasible solutions and *nondominated points* when associated to efficient solutions. The set of all nondominated points is usually called the *Pareto set*. Since each efficient solution could be interesting for a decision maker, the question is: How can we "solve" a multi-objective problem?

A first approach consists in aggregating the different objective functions in one global objective function and then consider the problem as a single objective optimization problem. Many ways to aggregate the different objective functions have been studied and developed [21, 35, 39, 42, 60, 83]. Nevertheless, this approach is relevant only if it is possible to define such a function that represents the preferences of the decision maker and if it is then possible to optimize the aggregated function. In general, it is not easy to formalize the preferences of a decision maker and moreover sometimes we may have different decision makers who have different preferences.

The second approach consists in producing all the efficient set (or a set of solutions that represents as well as possible the efficient set). This provides a representation of the trade-offs between objectives and may support the decision maker in identifying solutions of interest.

Note that, how a decision maker will make his choice among the proposed solutions, is an important field of research called *Decision Aiding* which has grown significantly since Bernard Roy introduced in 1968 the well-known method *ELECTRE* [70]. This kind of questions will not be discussed in this thesis. We are only focused in producing an interesting set of solutions for a decision maker, not in helping him to make his choice among this set of solutions.

For many multi-objective optimization problems, one of the main difficulties is the large cardinality of the set of nondominated points, and the even larger cardinality of the set of efficient solutions (considering that several solutions can have the same image in the criterion space). Similarly to single objective optimization where we usually look for one among all optimal solutions, we usually look for all nondominated points and a corresponding efficient solution for each such point. Thus, the study can be restricted to the set of nondominated points. Even with this restriction, it is well-known, that most multi-objective combinatorial optimization problems are *intractable*, in the sense that they admit families of instances for which the number of nondominated points is exponential in the size of the instance [24].

It appears that the intractability situation arises when the number of values taken on each criterion is itself exponential in the size of the instance. Then, either it is possible to have some *a priori* information on the different criteria to avoid this case, or it is not possible to produce the full set of nondominated points and we have to contend with an approximation of this set. In this last case, the goal is to compute and give to a decision maker a set of solutions that represents as well as possible the different choices, i.e. providing a good approximation of the Pareto set. There exists two different approaches to compute an approximation set.

The first one is based on heuristics or metaheuristics which are very useful in practice but give no guarantee on the quality of the returned solutions. The running times of these algorithms is also often competitive but no theoretical results can be established.

The second approach is to produce approximation algorithms with guaranteed performance. The main and most ambitious approximation algorithms are (fully) polynomial time approximation schemes, called *PTAS* (resp. *FPTAS*). A PTAS (resp. FPTAS) produces, for any $\varepsilon > 0$, a particular approximation of the Pareto set called ε-Pareto set in a time polynomial in the size of the instance (and $1/\varepsilon$). An ε-Pareto set is a set P_ε of feasible points that approximately dominates every other feasible points, i.e. such that for every feasible point z it contains a feasible point z' that is better within a factor $1 + \varepsilon$ than z in all the objectives.

This approach, which is central in this thesis, has grown significantly when Papadimitriou and Yannakakis [66] showed that for every multi-objective optimization problem and every $\varepsilon > 0$ there exists an ε-Pareto set whose size is polynomial in the size of the instance and $1/\varepsilon$. Moreover, they gave a necessary and sufficient condition for its efficient computability.

Note that, there may exist ε-Pareto sets of very different sizes. An interesting problem introduced by Vassilvitskii and Yannakakis [80] and continued by Diakonikolas and Yannakakis [20] is the efficient construction of ε-Pareto sets of size as small as possible. Indeed, the construction of an ε-Pareto set of polynomial size could not be sufficient for multiple practical reasons. For example, when buying a new apartment, we could have time to visit only a few apartments. Anyway, it is why even when the full Pareto set has a polynomial size, we may still prefer produce an ε-Pareto set of small size instead of the full Pareto set.

The notion of *representation* of the Pareto set, which is even more general than the notion of ε-Pareto set, can also be considered. A *representation* is evaluated according to three main dimensions: the quality of the *coverage*, i.e. providing a good approximation, the *cardinality*, i.e. it does not contain too many points, the *spacing*, i.e. it does not include any redundancies [33, 72]. Indeed, still when buying our apartment, we do not want to visit several apartments that are very similar. The notion of representation of the Pareto set is meaningful since it is closer to a set of solutions that a decision maker would receive to make his choice.

Goal of the thesis The goal of this thesis is to propose new general methods to get around the intractability of multi-objective optimization problems and to present some algorithms that produce good representations of the nondominated set.

First, we try to give some insight on this intractability by determining an, easily computable, upper bound on the number of nondominated points, knowing the number of values taken on each criterion (chapter 3). Then, we are interested in producing some discrete and tractable representations of the set of nondominated points for each instance of multi-objective optimization problems. These representations must satisfy some conditions of *coverage*, *cardinality* and if possible *spacing*. Starting from works aiming to produce ε-Pareto sets of small size, we first propose a direct extension of these works (chapter 4), then we focus our research on ε-Pareto sets satisfying an additional condition of *stability* (chapter 5).

Our results are mainly theoretical. When we propose a bound, it is easily computable, when we propose an algorithm, it is computable in polynomial time and with guaranteed performance. Moreover, no multi-objective optimization problem is studied in particular in this thesis, the methods are generic and relevant for every multi-objective optimization problem.

The work presented in this thesis has been performed in the context of the ANR (French Research Agency) project GUEPARD "GUaranteed Efficiency for PAReto optimal solutions Determination in multiobjective combinatorial optimization problems" (2009-2013). It included three partners: LIP6 (Université Paris-VI), LAMSADE (Université Paris-Dauphine), and LINA (Université de Nantes). The project was divided into five complementary tasks that cover the main scientific questions to be dealt with in any multi-objective combinatorial optimization study:

1. Analysis of instances of multi-objective combinatorial optimization problems.
2. Complexity and approximability of multi-objective combinatorial optimization problems.
3. Exact methods for the determination of the Pareto set.
4. Efficient approximation algorithms for the Pareto set with provable guarantees.
5. Preference-based optimization for compromise search.

Our contributions are related to tasks 2 and 4.

Organization of the thesis This thesis is organized as follows:

In chapter 2, entitled *Preliminaries*, we define the basic concepts, give details on the different ways to study multi-objective optimization problems, formalize the different problems studied in this thesis and recall some previous related works.

In chapter 3, entitled *Computation of upper bounds*, we investigate the number of nondominated points when we know (or have an upper bound on) the number of values taken on each criterion. We propose some bounds, easily computable, on the number of nondominated points. We also study the tightness of these bounds and the possible reduction of these bounds using some known feasible solutions.

In chapter 4, entitled *Approximation of small size*, we investigate the problem of determining a small ε-Pareto set and we compare the size of the ε-Pareto set produced with the size of a smallest ε-Pareto set. For the bicriteria case, we propose a new polynomial time algorithm that produces an ε-Pareto set of size at most three times the size of a smallest ε-Pareto set. For the tricriteria case, we study the performance of a greedy algorithm when the points are given explicitly in the input, thus answering an open problem of Koltun and Papadimitriou [54].

In chapter 5, entitled *Representation using kernels*, we focus on special ε-Pareto sets, called $(\varepsilon, \varepsilon')$-kernels, which satisfy a property of stability. The points in an $(\varepsilon, \varepsilon')$-kernel have to be spaced by at least a $(1 + \varepsilon')$ factor in some dimensions. We give some general results on $(\varepsilon, \varepsilon')$-kernels. We propose some polynomial time algorithms that produce small $(\varepsilon, \varepsilon')$-kernels for the bicriteria case and we give some negative results for the tricriteria case and beyond.

In the last chapter, we provide some general conclusions and perspectives.

2 Preliminaries

Contents

2.1 Framework and definitions

In a general multi-objective optimization problem, there are $p \geq 2$ objective functions, where we may have both minimization and maximization objectives. Since it is convenient not to consider all the combinations of types of objectives in the different proofs and algorithms, we assume in this thesis that all are objectives to be minimized. If some or all objective functions are to be maximized, our results are directly extendable.

Thus, in this thesis, we consider multi-objective optimization problems formulated as:

$$\min_{x \in S} \{f_1(x), \ldots, f_p(x)\}, \tag{2.1}$$

where $f_1, \ldots, f_p$ are $p \geq 2$ objective functions to be minimized and S is the set of feasible solutions.

We distinguish the decision space X which contains the set S of feasible solutions from the criterion space $Y \subseteq R^p$ which contains the criterion vectors associated to these solutions. We denote by $f(x) = (f_1(x), \dots, f_p(x))$ the feasible point associated to a feasible solution $x \in S$, and by $Z = f(S)$ the set of images of the feasible solutions. In the criterion space Y, we denote by y_i the coordinate on criterion f_i of a point $y \in Y$ for $i = 1, \dots, p$. We define, in the criterion space Y, the following partial strict order. Relation $\leq$ corresponds to the standard dominance relation used in multi-objective optimization.

Definition 1. *Let $\leq$ be the partial strict order defined such that for any $y, y' \in Y$, $y \leq y'$ if $y_i \leq y'_i$ for all $i \in \{1, \dots, p\}$ and $y \neq y'$. We say that point y dominates point y'.*

Then we define *efficient* solutions and *nondominated* points, respectively, in the decision space X and in the criterion space Y, as follows:

Definition 2. *A feasible solution $x \in S$ is called* efficient *if there is no other feasible solution $x' \in S$ such that $f(x') \leq f(x)$. If x is efficient, $f(x)$ is a* nondominated point *in the criterion space.*

We denote by S_E the set of efficient solutions and by Z_{ND} the set of nondominated points. In some context we also denote by P, the set of nondominated points, called *Pareto set*.

Definition 3. *A feasible solution $x \in S$ is called supported if there exists a linear combination of the criterion functions for which x is optimal.*

Definition 4. *A feasible point $z \in Z$ is* weakly nondominated *if there is no feasible point $z' \in Z$ such that $z'_i < z_i$ for all $i = 1, \dots, p$.*

We also define, in the criterion space Y, the following partial order:

Definition 5. *Given a constant $c > 0$, let $\preceq_c$ be the partial order defined such that for any $y, y' \in Y$, $y \preceq_c y'$ if $y_i \leq (1 + c)y'_i$ for all $i \in \{1, \dots, p\}$. We say that point y* $(1 + c)$-dominates *point y'.*

Now, we can define the following central notion of ε-Pareto sets.

Definition 6. *For any rational $\varepsilon > 0$, an* ε-Pareto set *P_ε is a subset of feasible points such that for all $z \in P$, there exists $z' \in P_\varepsilon$ such that z' $(1 + \varepsilon)$-dominates z.*

In the context of ε-Pareto sets, the central relation is the $(1 + \varepsilon)$-dominance relation, denoted by $\preceq_\varepsilon$. The asymmetric part of the $(1 + \varepsilon)$-dominance relation is denoted by $\prec_\varepsilon$.

Note that sometimes we can also be interested in a set of solutions whose images in the criterion space are the points of an ε-Pareto set P_ε. We make no distinction between these two sets in the following and an ε-Pareto set refers to a set of points or to a set of solutions.

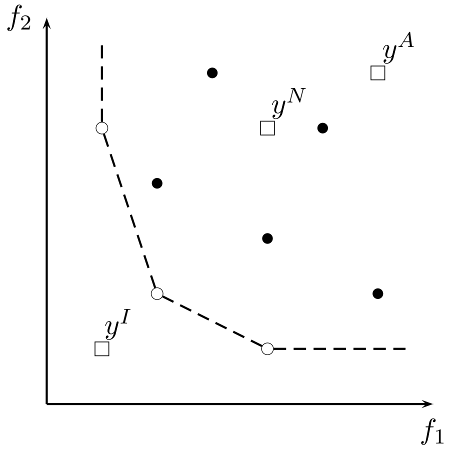

Figure 2.1: The ideal, anti-ideal and nadir points

We recall the definitions of three particular points in the criterion space. Anti-ideal, ideal and nadir points are defined as follows:

Definition 7. *The ideal and anti-ideal point are defined respectively by:*

$$y_i^I = \min_{x \in S} f_i(x), i = 1, \ldots, p$$

$$y_i^A = \max_{x \in S} f_i(x), i = 1, \ldots, p$$

The nadir point is a refinement of the anti-ideal point, defined by:

$$y_i^N = \max_{x \in S_E} f_i(x), i = 1, \ldots, p$$

These three notions are illustrated in the bicriteria case in Figure 2.1.

Observe that, when the underlying single objective minimization (resp. maximization) problem is solvable in polynomial time, the coordinates of the ideal (resp. anti-ideal) point are computable in polynomial time for any number of objectives. As pointed out in [27], for more than two objectives, the coordinates of the nadir point are not easily computable.

2.2 Difficulties in computing the nondominated set

For a multi-objective optimization problem, when we wish to compute the nondominated set, we are generally faced with two main difficulties. On the one hand, it is hard to verify if a feasible point is nondominated and, on the other hand the number of nondominated points can be very large.

2.2.1 Complexity of the decision problem

For a given instance of a multi-objective optimization problem, the associated *multi-objective decision problem* consists, for a given point y, in deciding if there is a feasible point z such that $z_i \leq y_i$ for $i = 1, \ldots, p$.

Usually, even for an "*easy*" multi-objective optimization problem, i.e. when the underlying single objective problem can be solved in polynomial time, the associated multi-objective decision problem is NP-hard and even strongly NP-hard in some cases. We recall that a problem is strongly NP-hard if it remains so even when all its numerical parameters are bounded by a polynomial in the size of the input. Of course, if the underlying single objective problem is NP-hard, the associated multi-objective decision problem is also NP-hard.

We recall some main multi-objective optimization problems, with an underlying single objective problem that can be solved in polynomial time, whose associated multi-objective decision problem is NP-hard even for the bi-ojective case: BI-OBJECTIVE SHORTEST PATH [74], BI-OBJECTIVE SPANNING TREE [13], BI-OBJECTIVE $s - t$ CUT (that is even strongly NP-hard) [66]. Of course, if a multi-objective decision problem is NP-hard in the bi-objective case, it is also NP-hard for more than two objectives.

Remark that solving this decision problem for a given point y means establishing if point y is dominated. Thus, determining if a point is dominated is generally an NP-hard problem.

2.2.2 Intractability

Usually, for a multi-objective optimization problem, it is possible to find a family of instances for which the cardinality of the nondominated set is very large, i.e. exponential in the input size.

Definition 8. *A multi-objective optimization problem is intractable if there are families of instances for which the number of nondominated points is not polynomial in the input size.*

Of course, if a multi-objective optimization problem is intractable in the bi-objective case, it is also intractable for more than two objectives. Most of the classical multi-objective combinatorial optimization problems are proved intractable even for the bi-ojective case: BI-OBJECTIVE KNAPSACK [51], BI-OBJECTIVE SHORTEST PATH [45], BI-OBJECTIVE TSP [29], BI-OBJECTIVE SPANNING TREE [44], BI-OBJECTIVE $s - t$ CUT [51] are intractable. We present in Figure 2.2 a classical instance of BI-OBJECTIVE SHORTEST PATH showing that the problem is intractable. In Figure 2.3, the points in the criterion space of this instance are represented, showing that all the 2^{n-1} paths from v_1 to v_n are efficient.

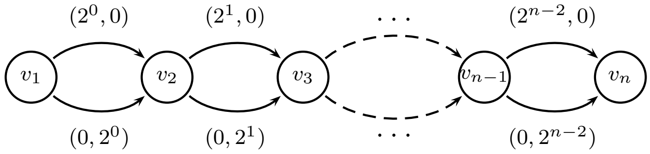

Figure 2.2: Intractability of SHORTEST PATH

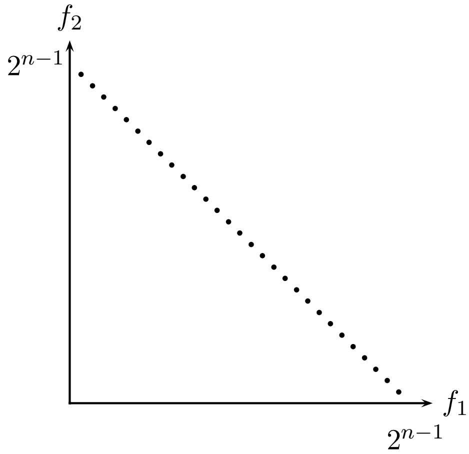

Figure 2.3: Intractability in the criterion space

Up to our knowledge, only one multi-objective combinatorial optimization problem is proved to be tractable, BI-OBJECTIVE GLOBAL MIN CUT [1].

However, many algorithms for generating all the Pareto set have been developed in the literature. There exists several exact methods for generating all the Pareto set. We give some examples of these methods in the following. The well-known *Two-phase method* introduced by Ulungu and Teghem [79] is the main one and was much developed these last years [67, 69, 81]. It consists first in computing all the supported efficient solutions (see [65, 68] for different approaches) and then in the definition and the exploration of the search area inside which non-supported nondominated points may exist. The second phase is mainly based on *Branch and Bound* algorithms [63, 81] or *Ranking* algorithms [67]. The *Two-phase method* is mainly relevant for the bi-objective case since the search area of the second phase is not defined by trivial geometric constructions in the multi-objective case [27, 69]. There exists also some algorithms for generating all the Pareto set based on the *ε-constraint method* [11, 56, 62] or on *Dynamic Programming* for instance for MULTI-OBJECTIVE KNAPSACK [53, 64].

Of course, since the number of nondominated points can be exponential in the input size, such algorithms cannot run in polynomial time.

Finding some ways to get around the intractability of multi-objective optimization problems is one of the main issue in multi-objective optimization. We present in the following section a review of existing results on this question.

2.3 Problem formulation and state of the art

The situation of intractability arises when the number of values taken on each criterion is itself exponential in the size of the instance. For example, it is the case in the instance for BI-OBJECTIVE SHORTEST PATH presented in Figure 2.2. So, there are two options to "hope" to get around intractability. Either it is possible to have some *a priori* information on the different criteria to avoid this case and so produce the set of nondominated points could be done in polynomial time, or it is not possible to produce the full set of nondominated points in polynomial time and then we have to contend with an approximation of this set. We present in the following some existing results based on these two approaches. These results are the main ones that are directly related to the cardinality of the set of points produced.

2.3.1 Bounds on the nondominated set

In this part, the goal is to avoid the intractability assuming that we can get some information on the different criteria. More precisely, the goal is to investigate the number of nondominated points when it is assumed that we know (or have an upper bound on) the number of values taken on each criterion. Remark that, how to find (in general or more probably depending on the underlying problem) these numbers, or even some small bounds on it, is in itself an interesting problem.

Up to our knowledge, this problem has not been dealt with, except very recently by Stanojević et al. in [75]. The best bound on the number of nondominated points that they give is obtained by a recursion formula which is well-known in ordered set theory [57] and that we recall in chapter 3. Unfortunately, this formula becomes quickly impractical when the number of values on each criterion increases. One of our purposes is to provide an alternative formula which does not depend on the number of values taken on each criterion.

2.3.2 Approximation of the Pareto set

First, we discuss about the notion of ε-Pareto set and present some works related to the size of ε-Pareto sets. Then, we deal with the more general notion of representation of the Pareto set, including a dimension of "stability".

2.3.2.1 ε-Pareto sets

For a given instance of a multi-objective optimization problem, there may exist several ε-Pareto sets, and these may have different sizes. Papadimitriou and Yannakakis showed in [66] that for every classical multi-objective optimization problem and every $\varepsilon > 0$ an ε-Pareto set of size polynomial in the input size and

$1/\varepsilon$ always exists. Thus, producing an approximation of the Pareto set is a way to get around the intractability. They obtained this result subdivising the objective space into hyperrectangles, such that, in each dimension the ratio of the largest to the smallest coordinate of an hyperrectangle is $1+\varepsilon$. Then, observing that the number of hyperrectangles thus defined is polynomial in the input size and $1/\varepsilon$, can be obtained by taking, when it exists, one point in each hyperrectangle. Note that, to obtain an ε-Pareto set that does not contain unnecessary points, it is possible to remove the dominated points of the obtained ε-Pareto set.

Moreover, it is also shown in [66] that the computation of an ε-Pareto set is related to the computation of the following routine *GAP*$_\delta$.

Definition 9. (GAP) *Given an instance I of a given problem, a point y and a rational $\delta \geq 0$, the routine* GAP$_\delta(y)$ *either returns a feasible point that dominates y or reports that there does not exist any feasible point z such that $z_i \leq \frac{y_i}{1+\delta}$ for all $i = 1, \ldots, p$.*

We say that a routine *GAP*$_\delta(y)$ runs in polynomial time (resp. fully polynomial time when $\delta > 0$) if its running time is polynomial in $|I|$ and $|y|$ (resp. $|I|$, $|y|$, $|\delta|$ and $1/\delta$).

Note that, for $\delta = 0$, the existence of a routine *GAP*$_\delta$ running in polynomial time is quite unlikely, since the routine has to solve a decision problem that is often NP-hard (see section 2.2.1).

We recall the central result relating the computation of an ε-Pareto set and the computation of the routine *GAP*$_\delta$. We sketch the proof of this result since we use some similar techniques in the thesis.

Theorem 1. *(Papadimitriou and Yannakakis [66]) An ε-Pareto set is computable in polynomial time (resp. fully polynomial time) if and only if the routine* GAP$_\delta$ *runs in polynomial time (resp. fully polynomial time).*

Proof : The one direction of this equivalence is quite easy. If it is possible to compute an ε-Pareto set in polynomial time (resp. fully polynomial time), then we can solve the routine *GAP*$_\delta(b)$ in polynomial time (resp. fully polynomial time) by checking if P_ε contains a point which dominates b. In such a case return the corresponding solution, else return NO.

In the other direction, if we can solve the routine *GAP*$_\delta$ in polynomial time (resp. fully polynomial time), then it is possible to compute an ε-Pareto set of polynomial size (resp. fully polynomial time) using the following procedure: Subdivise the objective space into hyperrectangles, such that, in each dimension the ratio of the largest to the smallest coordinate of an hyperrectangle is $1+\varepsilon'$ where $1+\varepsilon' \leq \sqrt{1+\varepsilon}$. Then, use the routine *GAP*$_{\varepsilon'}$ on all the corner points of the grid and then remove the obtained dominated points. □

Even if algorithms for computing ε-Pareto sets were given before for specific problems, most notably MULTI-OBJECTIVE SHORTEST PATH, for which Hansen [45] and Warburton [82] showed how to contruct an ε-Pareto set in polynomial time, many PTAS (FPTAS) have been developed and improved for many multi-objective optimization problems since Theorem 1 was established.

We give a brief list of these results for main multi-objective optimization problems: MULTI-OBJECTIVE SHORTEST PATH [78], MULTI-OBJECTIVE KNAPSACK [7, 31], MULTI-OBJECTIVE TSP [59], BI-OBJECTIVE WEIGHTED MAX-CUT [2], scheduling problems [4, 52]. Some particular conditions have also been studied for the existence of an FPTAS for the computation of ε-Pareto sets for different combinatorial problems including network flow and scheduling problems [71]. There are few works that also deal with the non-approximability of some multi-objective problems, for instance BI-OBJECTIVE $s-t$ MIN CUT [66], MULTI-OBJECTIVE TSP(1,2) [3] or some versions of MAX CUT [48]. Naturally, a result of non-approximability for a single objective problem is still valid for the multi-objective version of the problem.

In this thesis we are interested in the efficient construction of ε-Pareto sets. Our algorithms are guaranteed to run in polynomial time and to produce an ε-Pareto set for a fixed $\varepsilon > 0$. However, in the literature there are also heuristics that have been developed for the approximation of the Pareto set [18, 19].

Finally, there are also some works that deal with the approximation of the Pareto set using a fixed number of solutions. In [6], for a given constant k, some general properties are studied allowing the construction of approximation sets using at most k solutions for some bi-objective optimization problems. However, this kind of problems is more likely studied in the literature with only one solution [32, 36, 40, 41, 76]. This approach (using only one solution) is similar to the notion of max-min fair-ness [5, 10, 55]. In these settings, the different objectives represent the part of an agent A_i, where $i = 1, \ldots, p$, and the goal of the max-min fairness criterion is to maximize the satisfaction of the least satisfied agent.

2.3.2.2 ε-Pareto sets of minimal size

The existence of ε-Pareto sets of polynomial size is interesting for a computational point of view. However, such sets can still be quite large. Small ε-Pareto sets, if they exist, would be quite interesting for decision makers. For this reason, Vassilvitskii and Yannakakis introduced in [80] the study of the determination of an ε-Pareto set of minimal size. Formally, we can summarize these ideas as the following *primal* and *dual* problems:

Definition 10. (primal and dual problems)

– Primal problem: *Given an instance of a p-objective problem and an $\varepsilon > 0$, the goal is to obtain an ε-Pareto set of minimal size.*

– Dual problem: *Given an instance of a p-objective problem and an integer $k > 0$, the goal is to obtain an ε-Pareto set of size k with a minimal ε.*

One can remark that in order to obtain results on the primal and dual problems, just using a routine *GAP* on the "grid" considered by Papadimitriou and Yannakakis [66] is not sufficient. For instance, for the primal problem, the objective space must be explored with more precision to obtain some guarantees on the size of the ε-Pareto set returned. For this purpose, an interesting *generic* approach was introduced by Vassilvitskii and Yannakakis in [80]. They explored the objective space not using a routine *GAP* in all the corner points of the "grid" but performing, in a specific order, binary searches which call a routine *GAP* at each step. Thus, they obtained an ε-Pareto set whose size is close to the size of a smallest ε-Pareto set.

Definition 11. *An algorithm that uses a routine is called* generic *if only the routine is specific to a particular problem.*

In such algorithms, it is only required to have bounds on the minimum and maximum values of the criteria functions. We assume that the values taken by the objective functions are positive rational numbers whose numerators and denominators have at most m bits, and so that any feasible point has a value between 2^{-m} and 2^m. Then, the algorithm calls a routine as a black box for some values between 2^{-m} and 2^m. This hypothesis is quite general and not too restrictive. Moreover, note that with such hypothesis, the minimum difference between two points of the criterion space is at least 2^{-2m}. Indeed, let a/b and a'/b' be rational numbers belonging to the range $[2^{-m}, 2^m]$, it is clear that $a/b - a'/b' = (ab' - ba')/bb' \geq 1/2^{2m}$.

Remark that if the objective function values are even polynomially bounded, then the minimum difference between two points of the criterion space is also polynomially bounded. Thus, an algorithm based on the routine *GAP*$_\delta$ for $\delta > 0$ is polynomially equivalent to the same algorithm but using the exact routine *GAP*$_0$ instead of *GAP*$_\delta$ since it is possible to choose δ less than the minimum difference between two points of the criterion space.

In order to use generic algorithms, Diakonikolas and Yannakakis introduced in [20] two other main routines called *Restrict*$_\delta$ and *DualRestrict*$_\delta$ for the bi-objective case.

Definition 12. (Restrict and DualRestrict)

– *Given an instance I, a bound b and a rational $\delta \geq 0$, the routine* Restrict$_\delta(f_1, f_2 \leq b)$ *either returns a feasible point z satisfying $z_2 \leq b$ and $z_1 \leq (1+\delta). \min\{f_1(x) : x \in S$ and $f_2(x) \leq b\}$ or correctly reports that there does not exist any feasible point z such that $z_2 \leq b$.*
– *Given an instance I, a bound b and a rational $\delta \geq 0$, the routine* DualRestrict$_\delta(f_1, f_2 \leq b)$ *either returns a feasible point z satisfying $z_2 \leq b(1+\delta)$ and*

$z_1 \leq \min\{f_1(x) : x \in S$ *and* $f_2(x) \leq b\}$ *or correctly reports that there does not exist any feasible point* z *such that* $z_2 \leq b$.

We say that a routine *Restrict*$_\delta(f_1, f_2 \leq b)$ or *DualRestrict*$_\delta(f_1, f_2 \leq b)$ runs in polynomial time (resp. fully polynomial time when $\delta > 0$) if its running time is polynomial in $|I|$ and $|b|$ (resp. $|I|$, $|b|$, $|\delta|$ and $1/\delta$). Routines *Restrict*$_\delta(f_1, f_2 \leq b)$ and *DualRestrict*$_\delta(f_2, f_1 \leq b')$ are polynomially equivalent as proved in [20]. It means that one of these routines can be used a polynomial number of times to simulate the other one.

Remark that routines *Restrict*$_\delta(f_1, f_2 < b)$ and *DualRestrict*$_\delta(f_1, f_2 < b)$ with a strict constraint, can easily be simulated respectively by routines *Restrict*$_\delta(f_1, f_2 \leq b')$ and *DualRestrict*$_\delta(f_1, f_2 \leq b')$ using $b' = b - 2^{-2m}$.

Such routines have been studied for many multi-objective optimization problems. We give a list of main multi-objective optimization problems for which there exists a routine *Restrict*$_\delta$ that runs in polynomial (or fully polynomial) time: BI-OBJECTIVE SHORTEST PATH [30, 46, 58, 82], BI-OBJECTIVE SPANNING TREE [37, 47], BI-OBJECTIVE MATCHING [9], BI-OBJECTIVE MATROID INTERSECTION [9].

Remark that in the routines considered it is assumed that the error δ is a rational number, otherwise it is approximated from below by a rational one.

We denote by P^*_ε a smallest ε-Pareto set and by opt_ε its cardinality. There exist only a few results on primal and dual problems, we summarize these results in the following.

Bi-objective case, primal problem In the bi-objective case, for the primal problem, a generic algorithm that computes an ε-Pareto set of size at most $3opt_\varepsilon$ using routines *GAP*$_\delta$ was established by Vassilvitskii and Yannakakis in [80]. Moreover, if the routine *GAP*$_\delta$ runs in polynomial time (resp. fully polynomial time) then the algorithm also runs in polynomial time (resp. fully polynomial time). Then, Diakonikolas and Yannakakis showed in [20] that an ε-Pareto set of size at most $2opt_\varepsilon$ is computable in polynomial time if there exists a routine *Restrict*$_\delta$ computable in polynomial time for at least one objective. These approximation results are tight for the class of problems admitting such routines. Remark that these two classes of problems are distinct. For instance, BI-OBJECTIVE KNAPSACK, the knapsack problem with two objective functions to minimize and a capacity constraint to satisfy, belongs to the class of problems admitting a routine *GAP*$_\delta$ that runs in polynomial time but not to the class of problems admitting a routine *Restrict*$_\delta$ that runs in polynomial time. Indeed, solving *Restrict*$_\delta$ for BI-OBJECTIVE KNAPSACK requires to solve SINGLE OBJECTIVE KNAPSACK to produce a solution that respects the constraints of *Restrict*$_\delta$.

Definition 13. *An algorithm that computes an ε-Pareto set of size at most $k.opt_\varepsilon$ is called a k-approximation algorithm.*

Observe that the generic algorithms proposed in [20, 80] run in *fully* polynomial time if and only if the routines called run in *fully* polynomial time. For instance, for BI-OBJECTIVE SPANNING TREE, since there exists a routine *GAP*$_\delta$ that runs in fully polynomial time [50], the generic algorithm of Vassilvitskii and Yannakakis [80] is a 3-approximation that runs in fully polynomial time. Moreover, since the generic algorithm of Diakonikolas and Yannakakis [20] produces a 2-approximation, the computation of a 2-approximation that runs in fully polynomial time is related to the existence of a routine *Restrict*$_\delta$ for BI-OBJECTIVE SPANNING TREE that runs in fully polynomial time (which is an interesting open question). Nevertheless, the best current algorithm that solves *Restrict*$_\delta$ has a running time $O((1/\varepsilon)^{1/\varepsilon}n^3)$ with $1/\delta = O(1/\varepsilon)$ [47]. Thus, using this algorithm to solve *Restrict*$_\delta$, the generic algorithm of Diakonikolas and Yannakakis [20] gives an *efficient polynomial time approximation scheme (EPTAS)* which is a 2-approximation.

Bi-objective case, dual problem In the bi-objective case, for the dual problem, it is shown in [80], that the problem is NP-hard even in simple cases but has a PTAS if the bi-objective problem admit a routine *GAP*$_\delta$ that runs in polynomial time.

Multi-objective case For more than two objectives, it is shown for the primal problem that any generic algorithm based on routine *GAP* is not a k-approximation for any constant k [80] and for the dual problem that it is NP-hard to approximate the minimum ratio even within any polynomial multiplicative factor [20].

There is no algorithm in the literature that is not generic that produces an ε-Pareto set with a guarantee on its size. It seems hard to take benefit of the underlying problem to obtain some guarantees on the size of the produced ε-Pareto set. Moreover, the works mentioned ([80], [20]) of Diakonikolas, Vassilvitskii and Yannakakis are, to our knowledge, the only ones in the literature that give some approximability and non-approximability results for primal and dual problems.

2.3.2.3 Good representations of the Pareto set

Measures of the quality of a discrete representation of the Pareto set have been discussed in [33, 72]. As outlined in these papers three dimensions are relevant:

- *coverage* which ensures that any nondominated point is represented or *covered* by at least one point in the representation,

- *spacing (also called stability or uniformity)* which ensures that any two points in the representation are sufficiently *spaced*, avoiding redundancies,
- *cardinality* which should be minimal so as to make the representation as tractable as possible.

Coverage is the most important dimension for the representation to be meaningful. However, it must be counterbalanced by the two other dimensions which favor a uniform and small cardinality representation. While coverage on the one hand and spacing and cardinality on the other hand are clearly conflicting, the relationship between spacing and cardinality is not obvious. At first sight it could seem that improving spacing will lead to a decrease of the number of points in the representation. It must be observed, however, that imposing spacing is an additional constraint that may impact negatively on the cardinality. An interesting result in our work is that no negative impact is to be expected in the bi-objective case, but it is no longer true when dealing with at least three objectives. This shows the interest of considering all three dimensions.

Many works in the literature refer to the computation of discrete representations. These works mainly consider that a representation is a subset of the nondominated set, this makes a difference with an approximation set which can contain some dominated points. As outlined in [33], we can classify these works in different categories.

Some of these works use a predefined measure into algorithms to produce representations satisfying some prespecified goals on coverage, spacing or cardinality. Most of these algorithms are only applicable to particular classes of problems. For instance, Helbig [49] proposed an approach applicable to bi-objective problems and Sylva and Crema [77] an approach applicable to mixed-integer linear problems, both give a "good" coverage, the method suggested by Sayin [73] which is applicable for multiple objective linear problems gives a representation with a target coverage or target cardinality. Finally Eichfelder [28] presented a scalarization method which controls the spacing.

Another approach is to generate in a first phase a discrete representation which is quite large and then in a second phase remove some points so that the resulting set satisfies some conditions of coverage, spacing or cardinality [61].

Finally, there exist some heuristics that produce some discrete representations. In this case, many authors just improve the coverage or spacing of existing methods. Observe that the framework introduced by Sayin in [72] allows *a posteriori* to measure the quality of a discrete representation using linear programming. The author also identify the cases in which coverage or spacing conditions can be computed by solving a simple mathematical program. It is particularly useful for evaluating the set returned by some heuristics.

In the following we give some examples of heuristics. The *global shooting method*, a heuristic presented by Benson and Sayin in [8], gives a "good" coverage but the spacing can be "bad". Conversely, the *normal boundary intersection method*, a heuristic presented by Das and Dennis in [19] gives a "good" spacing but the coverage can be "bad". The *revised boundary intersection method*, a heuristic recently presented by Ehrgott and Shao [26] which combine the ideas of the two previous heuristics gives satisfying coverage and spacing.

Of course, in these approaches, the quality of the coverage, the spacing or the cardinality is not compared to the optimum, i.e. to the best coverage, spacing and cardinality that is possible to obtain. Our goal is to compare our representations to these optimum values, it means to transpose the primal and dual problems in terms of representations. For this purpose, we only need to take into account the dimension of spacing.

One way to ensure *spacing* is to impose a condition of stability with respect to an $(1+\varepsilon')$-dominance relation. An ε-Pareto set satisfying this additional condition will be called $(\varepsilon, \varepsilon')$-kernel and is defined precisely as follows.

Definition 14. *Given a set Z of feasible points and $\varepsilon, \varepsilon' > 0$, an $(\varepsilon, \varepsilon')$-kernel is a set of points $K_{\varepsilon,\varepsilon'}$ satisfying the two following conditions:*

(i) *for any point $z' \in Z \setminus K_{\varepsilon,\varepsilon'}$, there exists $z \in K_{\varepsilon,\varepsilon'}$ such that $z \preceq_\varepsilon z'$ (ε-coverage).*

(ii) *for any two distinct points $z, z' \in K_{\varepsilon,\varepsilon'}$, we do not have $z \preceq_{\varepsilon'} z'$ (ε'-stability).*

Notice that we chose to use the same metric (the classical one in approximation of the Pareto set) for the coverage and the stability, which seems more natural than using different metrics.

Observe that an $(\varepsilon, \varepsilon')$-kernel is an $(\varepsilon, \varepsilon'')$-kernel for all $\varepsilon'' \leq \varepsilon'$. Thus, when we refer to an $(\varepsilon, \varepsilon')$-kernel, we always consider the largest known ε' for which the ε'-stability condition is satisfied.

If $\varepsilon' > \varepsilon$ it is easy to see that an $(\varepsilon, \varepsilon')$-kernel does not always exist. Consider for instance $Z = \{z^1, z^2\}$ such that neither $z^1 \preceq_\varepsilon z^2$ nor $z^2 \preceq_\varepsilon z^1$ but $z^1 \preceq_{\varepsilon'} z^2$ or $z^2 \preceq_{\varepsilon'} z^1$. Therefore, for a given ε, the goal is to find an $(\varepsilon, \varepsilon')$-kernel with the largest ε'. When $\varepsilon' = \varepsilon$ an $(\varepsilon, \varepsilon')$-kernel is called an ε-kernel.

These special ε-Pareto sets seem interesting for two reasons. First, a decision maker could be interested in an ε-kernel instead of a classical ε-Pareto set because of its better representation of the space of solutions. Secondly, since we are also interested in small ε-Pareto sets, the ε-stability condition seems meaningful. Indeed, since the points of an ε-kernel have to be spaced by at least a $1+\varepsilon$ factor in some dimension it could induce a character of minimality. In fact, it is the case only for the bi-objective case.

In Figure 2.4 we present a small instance to illustrate the interest of this concept. In this figure, the set constituted by points z^3 and z^4 is an ε-Pareto set of minimum cardinality but not an ε-kernel. This set gives no real choice

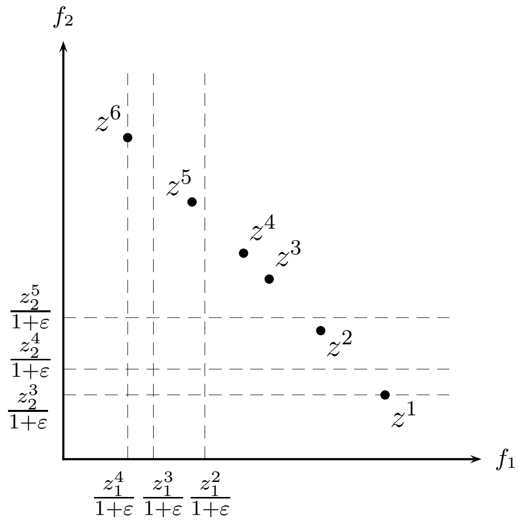

Figure 2.4: ε-kernels compared to ε-Pareto sets.

to a decision maker since the two points are very close. The set constituted by points z^2 and z^5, which is an ε-kernel, provides a much better representation of the nondominated set.

Our goal is to establish some general properties on ε-kernels and propose some solutions to the primal and dual problems for the case of ε-kernels. In chapter 5, when we consider the primal and dual problems, we refer to the version where we look for ε-kernels instead of ε-Pareto sets.

3 Computation of upper bounds

Abstract

In this chapter, we propose an upper bound on the maximal number of nondominated points of a multicriteria optimization problem. Assuming that the number of values taken on each criterion is known, the criterion space corresponds to a comparability graph or a product of chains. Thus, the upper bound can be interpreted as the stability number of a comparability graph or, equivalently, as the width of a product of chains. Standard approaches or formulas for computing these numbers are impractical. We develop a practical formula which only depends on the number of criteria. We also investigate the tightness of this upper bound and the reduction of this bound when feasible, possibly efficient, solutions are known.

Contents

The content of this chapter is based on the following paper.

▶ C. Bazgan, F. Jamain, and D. Vanderpooten. On the number of nondominated points of a multicriteria optimization problem, *Discrete Applied Mathematics*, 161(18):2841-2850, 2013.

In the following section, we define the basic concepts and formalize the problem both in the context of graphs and ordered sets. Then, in section 3.2, we deal with simple cases and provide, in the general case, a formula using a combinatorial version of the inclusion-exclusion principle [15]. The time for computing this formula is only exponential in the number of criteria. We also make comparisons with other bounds which are easier to compute. In section 3.3, we show that the proposed bound is tight for many classical multicriteria optimization problems. In section 3.4, we try to reduce the maximal number of nondominated points using known feasible solutions, possibly efficient. We conclude with some possible extensions to this work.

3.1 Problem statements

We assume that each criterion function f_i can take up to $c_i + 1$ values, where c_i is a nonnegative integer. Since we are interested in the *number* of nondominated points, we can assume for simplicity and without loss of generality, that each f_i takes integer values between 0 and c_i for $i = 1, \ldots, p$.

In some cases, the c_i values are known precisely, e.g. for qualitative criteria which take values on a scale whose grades correspond to predefined judgements. In other cases, these values can only be approximated. We can find an upper bound on c_i by computing the coordinates of the *ideal* and *anti-ideal* points, corresponding, respectively, to the best and the worst possible values on each criterion. Better bounds can be given if we can compute the coordinates of the *nadir* point, which corresponds to the worst possible values over the set of nondominated points. Moreover, how to find some small bounds on these numbers, is in itself an interesting problem. For some problems and some particular families of instances, it is possible to compute (or find some very small bounds on) the c_i values. For instance, consider the following family of instances of MULTI-OJECTIVE SPANNING TREE: The complete graphs with n vertices, where each edge cost is randomly chosen between 0 and k on each criterion, with $k \ll n$. Since each spanning tree consists of $n - 1$ edges, with high probability we have $c_i = (n-1)k$ for all $i = 1, \ldots, p$.

The problem of determining the maximum cardinality of the nondominated set can be stated as follows.

MAX SIZEND
Input: an integer p and p integers c_i, $i = 1, \ldots, p$.
Output: maximum cardinality of the nondominated set Z_{ND} associated to a set Z of p-dimensional points such that at most $c_i + 1$ values are taken on the i^{th} dimension, $i = 1, \ldots, p$.

Let $(\overline{c_i + 1}) = \{0, \ldots, c_i\}$, $i = 1, \ldots, p$ and $P = (\overline{c_1 + 1}) \times \ldots \times (\overline{c_p + 1})$. Any relevant set Z, and in particular any of those leading to a nondominated set of maximum cardinality, is included in P.

More precisely, this problem can be stated within different theoretical frameworks. Using graph theory, the maximal cardinality of a set of nondominated points corresponds to the *stability number* of a given graph. Using ordered set theory, this maximal cardinality corresponds to the *width* of a *product of chains*. These two frameworks provide different insights on our problem that will be discussed in the following.

3.1.1 Statement as a graph theory problem

Consider the graph $G = (P, E)$ whose set of vertices is $P = (\overline{c_1 + 1}) \times \ldots \times (\overline{c_p + 1})$ and set of edges is $E = \{(u, v) \in P \times P : u \leq v\}$. By construction, G is a *comparability* graph (i.e. a graph that admits a transitive orientation), since relation $\leq$ is transitive.

In this context, determining the maximum number of nondominated points amounts to determining the maximum cardinality of a stable set in G, i.e. computing $\alpha(G)$, the stability number of G. It is well-known that $\alpha(G)$ can be determined in polynomial time when G is a comparability graph [38]. In our case, this is achieved by computing a minimum flow in the digraph $G' = (P, \leq)$ from $(0, \ldots, 0)$ to $(c_1, \ldots, c_p)$ where each vertex has a lower bound of 1. Then $\alpha(G)$ corresponds to the value of this minimum flow in G'.

Computing a minimum flow in G' can be performed in a time polynomial in the number of vertices $P = \prod_{i=1}^{p}(c_i + 1)$. Since the input of MAX SIZEND is not G' but only values $c_1, \ldots, c_p$, which are encoded in binary, this approach only gives us a pseudo-polynomial time algorithm to solve MAX SIZEND.

3.1.2 Statement as an ordered set theory problem

Given a partially ordered set (S, R), we recall that a *chain* is a totally ordered subset and an *antichain* is a subset whose elements are pairwise incomparable. Moreover, the *height* of (S, R), denoted by $h(S)$, is the maximal cardinality of a chain in S, and the *width* of (S, R), denoted by $\alpha(S)$, is the maximal cardinality of an antichain in S. (S, R) is said to be *ranked* if we can define a function r such that for any $x, y \in S$, whenever xRy and there is no element $z \in S$ such that $xRzRy$, we have $r(y) = r(x) + 1$. Calling L_k the level of rank k in S, i.e.

the subset of elements of S with rank k, we define $n_k = |L_k|$ and $\sigma(S) = \max n_k$. Since the levels are antichains, we have $\alpha(S) \geq \sigma(S)$. Finally, a partially ordered ranked set S is said to satisfy the Sperner property if $\alpha(S) = \sigma(S)$ [14].

In our case, $P = (\overline{c_1 + 1}) \times \ldots \times (\overline{c_p + 1})$, which is a product of chains, is partially ordered by the dominance relation $\leq$. The resulting partially ordered set $(P, \leq)$ has height $h(P) = \sum_{k=1}^{p} c_k$ (denoted for short by h in the following). Moreover, $(P, \leq)$ can be ranked using rank function r which associates to each element $(y_1, \ldots, y_p) \in P$ its rank $r(y_1, \ldots, y_p) = \sum_{k=1}^{p} y_k$.

In this context, solving MAX SIZEND is equivalent to determining the width $\alpha(P)$. We recall the following result.

Theorem 2. *(De Bruijn et al. [12]) A product of chains satisfies the Sperner property.*

Therefore, since P is a product of chains, we have $\alpha(P) = \sigma(P)$. Thus, we are interested in determining the cardinality of a level of P which has the largest number of elements. It is well-known that the levels of maximum cardinality are all centered around the level $L_{h/2}$ if h is even, and the levels $L_{(h-1)/2}$ and $L_{(h+1)/2}$ if h is odd [14, 57]. Thus, determining $\alpha(P)$ amounts to computing $n_{\lfloor \frac{h}{2} \rfloor}$.

Leclerc [57] and Caspar et al. [14] proposed induction formulas to compute $n_{\lfloor \frac{h}{2} \rfloor}$. Nevertheless, these induction formulas depend on p but also on the values c_i, $i = 1, \ldots, p$. Since the values c_i may often be large, these formulas are not really usable in practice, as acknowledged by the previous authors. This is, however, another pseudo-polynomial time method to solve MAX SIZEND. The motivation is to obtain a new more practical formula, the complexity of which does not depend on the values c_i, that is a strongly polynomial time algorithm.

3.2 Computation of the width of a product of chains

We first provide an upper bound on the width of a product of chains P, showing that this bound is tight in a special case, which includes the bicriteria case. Then, we propose and compare two formulas for computing exactly $\alpha(P)$.

We assume w.l.o.g. that the criteria are numbered by non-increasing order of values c_i, that is $c_1 \geq ... \geq c_p$.

3.2.1 A simple upper bound on $\alpha(P)$

A first simple upper bound on $\alpha(P)$ is given by the following result.

Lemma 1. $\alpha(P) \leq \prod_{i=2}^{p}(c_i + 1)$.

Proof : By contradiction, if $\alpha(P) > \prod_{i=2}^{p}(c_i + 1)$ there exist at least two nondominated points with the same values on criteria f_i, $i = 2, \ldots, p$. Then, among these two points, the point with a worse value on f_1 is dominated by the other one. □

This upper bound is tight in a particular case, as shown in the following lemma:

Lemma 2. $\alpha(P) = \prod_{i=2}^{p}(c_i + 1)$ *if and only if* $c_1 \geq \sum_{i=2}^{p} c_i$.

Proof :
⇐ If $c_1 \geq \sum_{i=2}^{p} c_i$ then all possible $\prod_{i=2}^{p}(c_i+1)$ configurations on the last $p-1$ criteria can be completed on criterion f_1 so as to define nondominated points. Indeed, any point with value v_j on criterion f_j, $j = 2, \ldots, p$ is nondominated if it is assigned the (nonnegative) value $\sum_{i=2}^{p} c_i - \sum_{i=2}^{p} v_j$ on criterion f_1.
⇒ If $\alpha(P) = \prod_{i=2}^{p}(c_i+1)$, all possible configurations on the last $p-1$ criteria must correspond to nondominated points. In particular, the $\sum_{i=2}^{p} c_i + 1$ following configurations, which constitute a chain on the last $p - 1$ criteria, must correspond to nondominated points:
$(*, 0, \ldots, 0), (*, 1, 0, \ldots, 0), \ldots, (*, c_2, 0, \ldots, 0),$
$(*, c_2, 1, 0, \ldots, 0), \ldots, (*, c_2, c_3, 0, \ldots, 0),$
$\ldots$
$(*, c_2, c_3, \ldots, c_{p-1}, 1), \ldots, (*, c_2, c_3, \ldots, c_{p-1}, c_p)$
For this chain on the last $p - 1$ criteria to become an antichain on the p criteria, we need $\sum_{i=2}^{p} c_i + 1$ different values on criterion f_1, and thus $c_1 \geq \sum_{i=2}^{p} c_i$. □

In the particular case where $p = 2$, we obtain the following corollary, since $c_1 \geq c_2$.

Corollary 1. *If* $p = 2$*, we have* $\alpha(P) = c_2 + 1$.

3.2.2 Exact computation of $\alpha(P)$

Since P satisfies the Sperner property, we noticed at the end of section 3.1.2 that $\alpha(P) = n_{\lfloor \frac{h}{2} \rfloor}$. We first review a well-known recursion formula for computing $n_{\lfloor \frac{h}{2} \rfloor}$, which is not practicable as values c_i grow. Then, we propose an alternative analytical formula, which is shown to be much easier to implement.

3.2.2.1 A recursion formula

As indicated in [57], the following result is known from "folklore".

Proposition 1. *Let* $P' = P \times (\overline{c+1})$ *where* P *is a product of chains. The values* n'_k*, the size of level of rank* k *in* P'*, can be obtained from values* n_k *by the following* *recursion:*

$$n'_k = \sum_{i=0}^{c} n_{k-i}$$

which can be rewritten as

$$n'_k = n'_{k-1} + n_k - n_{k-c-1} \qquad (3.1)$$

where $n_k = 1$*, for all* $k \geq 0$ *when* P *is a chain and* $n_k = 0$ *for* $k < 0$.

As outlined in [57], this recursion is relevant in practice only for a small number of criteria and small values c_i. More precisely, the complexity of this induction formula is given by the following result.

Lemma 3. *The computation of the width of the product of chains* $P = (\overline{c_1 + 1}) \times \ldots \times (\overline{c_p + 1})$ *using formula (3.1) of Proposition 1 is done in* $\Theta(p^2 c_{max})$ *operations, where* $c_{max} = \max\{c_1, \ldots, c_p\}$.

Proof: At each step i of the recursion for $i = 1, \ldots, p$, the computation of the cardinality of $((\sum_{j=p-(i-1)}^{p} c_j) + 1)/2$ levels is needed. Since each of these cardinalities is computed in constant time, the computation of $\alpha(P)$ is performed in $((\sum_{i=1}^{p} ic_i) + p)/2 = \Theta(p^2 c_{max})$ operations. □

Observe additionally that these recursions require to keep in memory all the sizes of the levels of the previous step, which requires a space $\Theta(c_{max})$. In most multicriteria optimization problems, the number of criteria is rather small and can thus assumed to be constant. On the other hand, values c_i may be rather large. This makes this recursion quickly useless. This is the motivation to obtain a formula computing the width of P whose complexity does not depend on the values c_i.

3.2.2.2 An analytical formula

We need to compute the number of points on a level of maximum cardinality, which amounts to computing the number of integer solutions of the equation

$$x_1 + ... + x_p = k \qquad (3.2)$$

with $k = \lfloor h/2 \rfloor$, under the constraints $0 \leq x_i \leq c_i$.

We recall the following result, presented in standard textbooks on combinatorics such as [15], which is a combinatorial version of the inclusion-exclusion principle.

Lemma 4. *The number of integer solutions of equation (3.2) with the restrictions*

$$s_i \leq x_i \leq m_i, \quad i = 1, \ldots, p$$

where s_i and m_i are given for $i = 1, \ldots, p$ with $s \leq k \leq m$, $s = s_1 + \ldots + s_p$ and $m = m_1 + \ldots + m_p$, with $u_i = m_i - s_i \geq 0$, $i = 1, \ldots, p$ is given by

$$\binom{p+k-s-1}{p-1} + \sum_{r=1}^{p} (-1)^r \sum_{I \subseteq \{1,\ldots,p\}:|I|=r} \binom{p+k-s-\sum_{i\in I} u_i - r - 1}{p-1}$$

Applied in our context, the previous lemma gives the following result.

Theorem 3. *The width $\alpha(P)$ of a product of chains $P = (\overline{c_1+1}) \times \cdots \times (\overline{c_p+1})$ is given by the following formula:*

$$\alpha(P) = \sum_{I \subseteq \{1,\ldots,p\}:|I| \leq \lfloor \frac{h}{2} \rfloor - c_I} (-1)^{|I|} \prod_{k=1}^{p-1} \left(1 + \frac{\lfloor h/2 \rfloor - c_I - |I|}{k}\right) \tag{3.3}$$

where $c_I = \sum_{i\in I} c_i$ and $c_\varnothing = 0$.

Proof : Using the formula of Lemma 4 with $s_i = 0$ and $m_i = c_i$ for $i = 1, \ldots, p$ we obtain the following formula:

$$\alpha(P) = \binom{p+\lfloor h/2 \rfloor - 1}{p-1} + \sum_{r=1}^{p} (-1)^r \sum_{I \subseteq \{1,\ldots,p\}:|I|=r} \binom{p+\lfloor h/2 \rfloor - \sum_{i\in I} c_i - r - 1}{p-1}$$

Combining the two members of this formula we have:

$$\alpha(P) = \sum_{I \subseteq \{1,\ldots,p\}:|I| \leq \lfloor \frac{h}{2} \rfloor - c_I} (-1)^{|I|} \binom{p+\lfloor h/2 \rfloor - c_I - |I| - 1}{p-1}$$

where $c_I = \sum_{i\in I} c_i$ and $c_\varnothing = 0$, which can be rewritten as (3.3) using $\binom{n}{t} = \frac{1}{t!}(n-t+1)\ldots n$. □

In the particular case where $p = 3$, the formula can be simplified as follows.

Corollary 2. *If $p = 3$, we have*

$$\alpha(P) = \begin{cases} (c_2+1)(c_3+1) & \text{if } c_1 \geq c_2 + c_3 \\ 1 + \left(\frac{h}{2}\right)^2 + \frac{h}{2} - \frac{c_1^2+c_2^2+c_3^2}{2} & \text{if } c_1 < c_2 + c_3 \text{ and } h \text{ is even} \\ \frac{1}{2} + \left(\frac{h+1}{2}\right)^2 - \frac{c_1^2+c_2^2+c_3^2}{2} & \text{if } c_1 < c_2 + c_3 \text{ and } h \text{ is odd} \end{cases}$$

Moreover, if $c_i = q, i = 1, 2, 3$, we have

$$\alpha(P) = \begin{cases} \frac{3}{4}(q+1)^2 + \frac{1}{4} & \text{if } q \text{ is even} \\ \frac{3}{4}(q+1)^2 & \text{if } q \text{ is odd} \end{cases}$$

Proof : The first case is a consequence of Lemma 2. The second and third cases are obtained from formula (3.3), observing that the only subsets $I \subseteq \{1, 2, 3\}$ such that $|I| \leq \lfloor h/2 \rfloor - c_I$ are $\varnothing, \{1\}, \{2\}$, and $\{3\}$ when $c_1 < c_2 + c_3$. □

The next lemma gives the complexity for computing $\alpha(P)$, using (3.3).

Lemma 5. *The computation of the width of a product of chains* $P = (\overline{c_1 + 1}) \times \cdots \times (\overline{c_p + 1})$ *using formula (3.3) is performed in* $O(p2^p)$ *operations.*

Proof : The product $\prod_{k=1}^{p-1}(1 + \frac{\lfloor h/2 \rfloor - c_I - |I|}{k})$ requires $O(p)$ operations and the sum is over $O(2^p)$ subsets, so the computation of $\alpha(P)$ needs $O(p2^p)$ operations. □

Thus, this complexity is exponential in the number of criteria p, but does not depend on the values c_i. Actually, since p is usually small in practice and thus considered constant in theory, the previous discussion can be summarized through the following result.

Theorem 4. MAX SIZEND *is solvable in constant time when* p *is constant.*

3.2.3 Comparison of the different bounds

We propose to compare $\alpha(P)$ to simpler bounds on the number of nondominated points. Let us first illustrate this comparison on a large class of instances of TRI-OBJECTIVE SPANNING TREE. Let $G = (V, E)$ be a complete graph with $n = 101$ vertices, where each edge cost is randomly chosen between 0 and 10 on each criterion. We wish to compute, in the worst case, the number of nondominated points.

Considering that for some instances all feasible solutions can give rise to different nondominated points [44], a first bound is the total number of spanning trees in a complete graph, i.e. $n^{n-2} = 101^{99}$. This huge bound, which can be achieved only when edge costs are exponential, does not take account of values c_i.

A second bound corresponds to the product $\prod_{i=2}^{p}(c_i + 1)$, where $c_1 = c_2 = c_3 = 1000$ and $p = 3$ which gives $1001^2 = 1.002.001$.

Finally, our proposed bound, computed from Corollary 2 with $c_1 = c_2 = c_3 = 1000$, gives $\frac{3}{4}(1001)^2 + \frac{1}{4}$= 750.751.

It is interesting to quantify the ratio between $\alpha(P)$ and $\prod_{i=2}^{p}(c_i + 1)$. The smallest ratio is reached, as in the previous example, when all c_i are equal. Let $\alpha_{p,q}(P)$ be the result of the formula which computes the maximal number of nondominated points in the worst case when there are p criteria and for all $i,\ c_i = q - 1$. In the following proposition, we determine $\lim_{q \to \infty} \alpha_{p,q}(P)/q^{p-1}$, where q^{p-1} corresponds to the product $\prod_{i=2}^{p}(c_i + 1)$.

Proposition 2. *For p criteria, we have*

$$\lim_{q\to\infty} \frac{\alpha_{p,q}(P)}{q^{p-1}} = \sum_{l=1}^{\lceil \frac{p}{2} \rceil -1} -(1)^l \frac{p}{l!(p-l)!}(\frac{p}{2}-l)^{p-1}.$$

Proof: Using formula (3.3) and keeping only the coefficients of the terms of degree $p-1$.

□

This way, we can compute all these limits when p is fixed. For instance $\lim_{q\to\infty} \frac{\alpha_{3,q}(P)}{q^2} = \frac{3}{4}$ and $\lim_{q\to\infty} \frac{\alpha_{4,q}(P)}{q^3} = \frac{2}{3}$. When the number of criteria increases, we note that the proposed bound is more and more interesting, as compared with the bound q^{p-1}.

3.3 Tightness of the bound

The determination of the maximum number of nondominated points is particularly relevant for multicriteria combinatorial optimization problems, for which it is well-known that this number can be exponential in the size of the instance [24]. Considering such a problem Π, the problem of determining the maximum cardinality of the nondominated set associated to Π, knowing values c_i, $i = 1, \dots, p$, is denoted by MAX SIZEND Π in the following.

We show in this part that our bound $\alpha(P)$ is tight for the multicriteria version of some classical optimization problems such as SELECTION, KNAPSACK, SHORTEST PATH, SPANNING TREE, TSP, s-t CUT. We propose some relatively simple families of instances of these problems where the number of nondominated points is exactly $\alpha(P)$.

We first introduce some notations used in the definitions of these problems. SELECTION and KNAPSACK require to define a set O of objects, a capacity b and a nonnegative integer t. Each object $o \in O$ has a criterion vector $v(o) = (v_1(o), \dots, v_p(o))$ and a weight $w(o)$. We define the criterion functions on a set $O' \subseteq O$ as $v_i(O') = \sum_{o\in O'} v_i(o)$ for all $i \in \{1, \dots, p\}$.

SELECTION consists in selecting a subset $O' \subseteq O$ of t objects maximizing $v_i(O')$, $i = 1, \dots, p$. KNAPSACK consists in selecting a subset $O' \subseteq O$ satisfying the constraint $\sum_{o\in O'} w(o) \leq b$ maximizing $v_i(O')$, $i = 1, \dots, p$.

The other problems are defined on a graph. Consider $G = (V, E)$ a graph where $V = \{1, \dots, n\}$ is the set of vertices and $E \subseteq V \times V$ is the set of edges. Each edge $e \in E$ has a criterion vector $v(e) = (v_1(e), \dots, v_p(e))$. We define the value function v on a subset E' of edges as follows: $v(E') = (v_1(E'), \dots, v_p(E'))$ where $v_i(E') = \sum_{e\in E'} v_i(e)$ for all $i \in \{1, \dots, p\}$.

Proposition 3. *The bound $\alpha(P)$ is tight for* MAX SIZEND SELECTION *and* MAX SIZEND KNAPSACK.

Proof: Consider p integers $c_1, \ldots, c_p$ and p subsets O_j, $j = 1, \ldots, p$, where each subset O_j contains c_j identical objects $o^i_j, i = 1, \ldots, c_j$ with $v_j(o^i_j) = 1$ and $v_k(o^i_j) = 0$ for $k \neq j$. Let $O = \cup_{j=1}^{p} O_j$ with $|O| = \sum_{j=1}^{p} c_j = n$ and $t = \lfloor \frac{n}{2} \rfloor$.

Selecting $t = \lfloor \frac{n}{2} \rfloor = \lfloor \frac{h}{2} \rfloor$ objects can be seen as selecting x_j objects in subset O_j, $j = 1, \ldots, p$ such that $\sum_{j=1}^{p} x_j = \lfloor \frac{h}{2} \rfloor$ and $0 \leq x_j \leq c_j$, with a resulting nondominated criterion vector $(x_1, x_2, \ldots, x_p)$. The number of such vectors is the number of integer solutions of equation (3.2) and thus corresponds to $\alpha(P)$.

Since SELECTION is a particular case of KNAPSACK, the result also holds for MAX SIZEND KNAPSACK. □

Proposition 4. *The bound $\alpha(P)$ is tight for* MAX SIZEND SHORTEST PATH, MAX SIZEND SPANNING TREE, *and* MAX SIZEND TSP.

Proof: Assume that p is even and let q be a nonnegative integer. We consider the following gadget consisting of a graph with two vertices, which are connected by edges corresponding to all the p-tuples containing $p/2$ values 0 and $p/2$ values 1, with the corresponding values on these edges (see Figure 3.1).

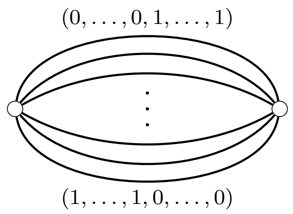

Figure 3.1: Gadget

Let G be the concatenation of q times this gadget (see Figure 3.2).

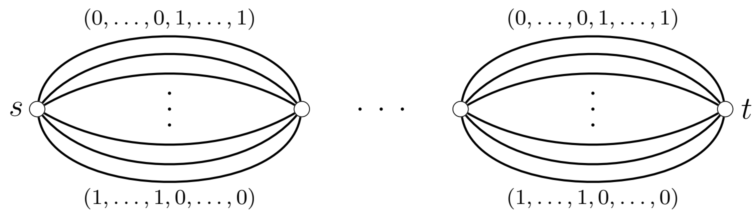

Figure 3.2: Graph G

Any path between s and t in G uses exactly one edge of each gadget and corresponds to a nondominated point $(v_1, \ldots, v_p)$ with $0 \leq v_i \leq q$ and $\sum_{i=1}^{p} v_i = \frac{pq}{2}$. The number of such points is the number of integer solutions of equation (3.2), with $c_i = q$, for $i = 1, \ldots, p$, and thus corresponds to $\alpha(P)$.

Since in the previous construction paths and spanning trees are equivalent, the proof holds for MAX SIZEND SPANNING TREE. Adding edge (s, t) to the above construction with criterion value $(0, \ldots, 0)$, the proof holds also for MAX SIZEND TSP. □

Proposition 5. *The bound $\alpha(P)$ is tight for* MAX SIZEND s-t CUT.

Proof : The proof is essentially the same as in Proposition 4 but using the following gadget consisting of a path whose edges correspond to all the p-tuples containing $p/2$ values 0 and $p/2$ values 1 (see Figure 3.3) and the following graph G (see Figure 3.4), where this gadget is duplicated q times, each of these being connected at each end.

Figure 3.3: Gadget

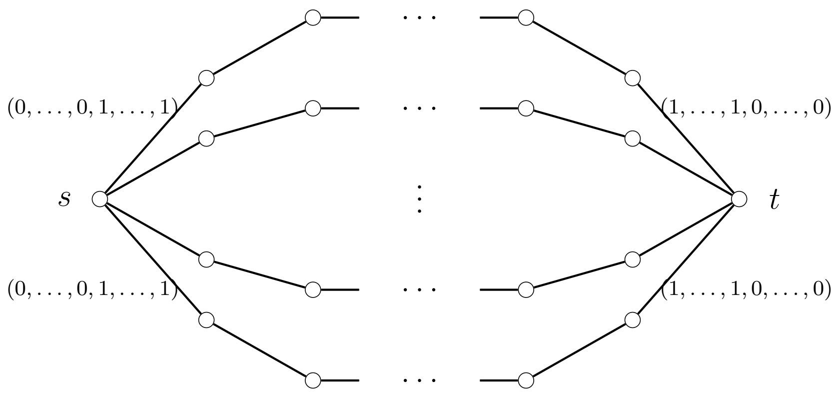

Figure 3.4: Graph G

In the same way, we have $c_i = q$ for $i = 1, \ldots, p$, and the number of nondominated points is exactly $\alpha(P)$. □

3.4 Reduction of the bound using known feasible solutions

We investigate now if it is possible to improve the upper bound on the number of nondominated points when a subset of feasible solutions or a subset of efficient solutions is known. Indeed, feasible solutions can often be easily computed. Moreover, *supported* efficient solutions, which are obtained by optimizing a weighted sum of the criteria, are easily computable, when the corresponding single criterion problem is polynomially solvable.

The knowledge of feasible criterion vectors, possibly known to be nondominated, involves the elimination of some points in P. More precisely, if a feasible point z is known, all the points dominated by z cannot be part of the nondominated set and can thus be removed from P. Moreover, if z is known to be nondominated, we can also remove from P all the points which dominate z. In the graph theory setting, this leads to subgraphs which are still comparability graphs. Therefore, the computation of the maximal number of nondominated points in this context is still achievable in pseudo-polynomial time. We investigate the problem under the ordered set theory setting.

3.4.1 When feasible solutions are known

Given $P = (\overline{c_1 + 1}) \times \ldots \times (\overline{c_p + 1})$ and k points $z^1, \ldots, z^k$ in the criterion space, representing feasible solutions, let D be the subset of P dominated by at least one point from $\{z^1, \ldots, z^k\}$, that is the set of points y of P such that there is $j \in \{1, \ldots, k\}$ with $z^j \leq y$. We want to study if the set $Q = P - D$ still satisfies the Sperner property and we want to compute $\alpha(Q)$.

3.4.1.1 Two objectives

In the bicriteria case we have the following result.

Proposition 6. *When $p = 2$, Q satisfies the Sperner property and we have $\alpha(Q) = min(c_2, min_{j=1}^{k} r(z^j)) + 1$ where $r(z^j)$ is the rank of point z^j.*

Proof: When there is no point in $\{z^1, \ldots, z^k\}$ located below the first level of maximum cardinality of P we have $\alpha(Q) = \alpha(P) = c_2 + 1$. Otherwise, let L be the lowest level of P containing an element of the set $\{z^1, \ldots, z^k\}$ and $z^m = (z_1^m, z_2^m)$ such a point. Since points z^j, $j \neq m$, located above level L do not eliminate any point on L, we have $\alpha(Q) \geq |L| = r(z^m) + 1$.

Consider now the set $W \subset P$ of points belonging either to the chains containing all the points with a first constant coordinate v_1, for each $v_1 \in \{0, \ldots, z_1^m\}$ or to the chains containing all the points with a second constant coordinate v_2, for each $v_2 \in \{0, \ldots, z_2^m - 1\}$. We have $Q \subset W$ and we use $|L|$ chains to cover W. Therefore, any antichain of Q contains at most $|L|$ points, i.e. we have $\alpha(Q) \leq |L| = r(z^m) + 1$.

In any case, $\alpha(Q)$ corresponds to the cardinality of a level of Q, meaning that Q satisfies the Sperner property. □

3.4.1.2 More than two objectives

When $p \geq 3$, the observed structure does not satisfy the Sperner property as will be shown in the next result. We observed in the bicriteria case that $\alpha(Q)$ is determined either from the first level of maximum cardinality or from the level of one of the points z^j. We could expect that, for $p \geq 3$, only these levels are relevant when computing $\alpha(Q)$. Unfortunately, we also show that other levels may contribute to $\alpha(Q)$. This suggests that the determination of $\alpha(Q)$ is difficult.

Proposition 7. *For any $p \geq 3$, Q does not satisfy the Sperner property. Moreover, other levels than the first level of maximum cardinality of P and levels of the points z^j may contribute to $\alpha(Q)$.*

Proof : We first construct a simple example with three criteria. Let $P = \overline{3}\times\overline{3}\times\overline{3}$ be the product of chains and $z = (0, 1, 0)$ a known feasible solution (see Figure 3.5).

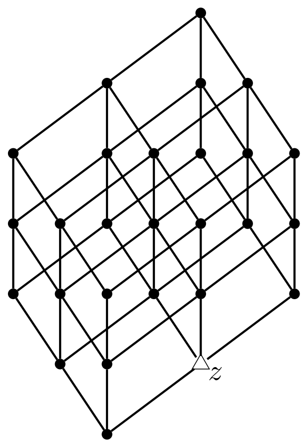

Figure 3.5: $P = \overline{3} \times \overline{3} \times \overline{3}$

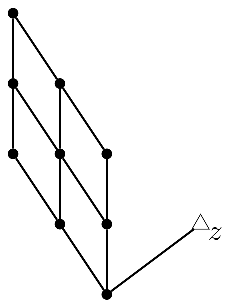

Figure 3.6: The set $Q = P - D$

The set $Q = P - D$, represented in Figure 3.6, does not satisfy the Sperner property. Indeed, we have $\sigma(Q) = |L_1| = |L_2| = 3$, while $\alpha(Q) = |L_2| + 1 = 4$ since point z, which belongs to level L_1, is incomparable to the 3 points belonging to level L_2. Observe that L_2 is neither the first level of maximum cardinality of P (L_3) nor the level of z (L_1).

This example can be extended easily to $p \geq 4$ criteria. We just need to extend z with values 0 on the $p - 3$ other criteria and add $p - 3$ new points z^i, $i = 1, \ldots, p - 3$, where z^i has coordinate 1 on criterion $i + 3$ and 0 on the other criteria. Doing so, we obtain the same set Q as for $p = 3$ (except that points in Q have now all their $p - 3$ last coordinates equal to 0). □

3.4.2 When efficient solutions are known

We consider now the same problem when the feasible solutions are known to be efficient.

Given $P = (\overline{c_1 + 1}) \times \ldots \times (\overline{c_p + 1})$ and k nondominated points $z^1, \ldots, z^k$ in the criterion space, representing efficient solutions, let D be the subset of P corresponding to the set of points y of P such that there is $j \in \{1, \ldots, k\}$ with $z^j \leq y$ or $y \leq z^j$. We are interested in computing $\alpha(Q)$, where $Q = P - D$.

3.4.2.1 Two objectives

In this case, the set Q does not satisfy the Sperner property. We illustrate this on an instance where $P = \overline{8} \times \overline{6}$, and two known nondominated points $z^1 = (5, 1)$ and $z^2 = (1, 4)$ (see Figure 3.7). Here Q consists of the points represented by squares and the two points z^1 and z^2. A largest antichain in Q is $\{z^1, z^2, y^1, \ldots, y^4\}$ and thus we have $\alpha(Q) = 6$, whereas $\sigma(Q) = 4$.

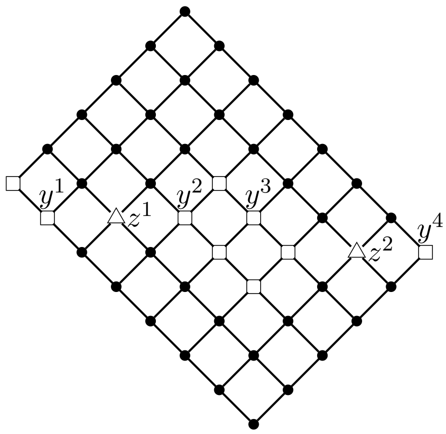

Figure 3.7: $P = \overline{8} \times \overline{6}$

We show, however, that $Q \setminus \{z^1, \ldots, z^k\}$ is a disjoint union of products of two chains, which allows the computation of $\alpha(Q)$. We assume in this part that the k nondominated points z^j, $j = 1, \ldots, k$ are ranked by non increasing value on the first criterion, i.e. $z_1^1 \geq \ldots \geq z_1^k$.

Proposition 8. *When $p = 2$, we have* $\alpha(Q) = k + \min(c_1 - z_1^1, z_2^1) + \min(z_1^k, c_2 - z_2^k) + \sum_{j=1}^{k-1} \min(z_1^j - z_1^{j+1} - 1, z_2^{j+1} - z_2^j - 1)$.

Proof : The first term k in the proposed formula corresponds to the k given nondominated points. These k points delimit exactly $k + 1$ disjoint products of two chains, some of them being possibly empty. The first product of chains is of size $c_1 - z_1^1$ on the first criterion and z_2^1 on the second one, the $(k+1)^{\text{th}}$ product of chains is of size z_1^k on the first criterion and $c_2 - z_2^k$ on the second one, whereas the products of chains located between two points z^j and z^{j+1} are of size $z_1^j - z_1^{j+1} - 1$ on the first criterion and $z_2^{j+1} - z_2^j - 1$ on the second one. Each point of any of these $k+1$ products of chains is incomparable with any point of any other product and incomparable with each z^j. Since the width of a product of two chains $\overline{c_1} \times \overline{c_2}$ is $\min(c_1, c_2)$, the formula is proved. □

We remark that, to determine $\alpha(Q)$, we can consider only nondominated points located on the levels which contain the known nondominated points z^j. Referring again to the instance presented in Figure 3.7, we illustrate this remark with the largest antichain $\{z^1, z^2, y^1, \ldots, y^4\}$.

3.4.2.2 More than two objectives

We observed in the bicriteria case that $\alpha(Q)$ is determined by considering points on the levels of points z^j. Unfortunately, for $p \geq 3$, other levels may contribute to $\alpha(Q)$, as shown in the next result. This suggests that the determination of $\alpha(Q)$ is difficult.

Proposition 9. *For any $p \geq 3$, other levels than the first level of maximum cardinality of P and levels of the points z^j may contribute to $\alpha(Q)$.*

Proof : We consider the same counter-example as in the proof of Proposition 7 and Figure 3.5. The set $Q = P - D$ is represented in Figure 3.8.

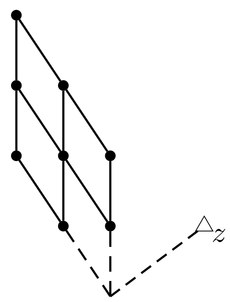

Figure 3.8: The set $Q = P - D$

We have $\alpha(Q) = |L_2| + 1$. Observe that L_2 is neither the first level of maximum cardinality of P (L_3) nor the level of z (L_1). □

3.5 Conclusions

The purpose of this chapter was to develop tight and easily computable bounds on the cardinality of the set of nondominated points. Graph theory and ordered set theory provided complementary insights on this topic. Two main questions require further investigation.

A basic assumption in our work is the a priori knowledge on the number of values taken on each criterion. Obviously, obtaining a good upper bound on these values is itself a difficult question which depends on the problem at hand as well as on the specific instances.

Knowing feasible, possibly efficient, solutions may improve our bound on the number of nondominated points. The impact is clear in the bicriteria case. For $p \geq 3$, nice properties (the Sperner property, the fact that only the levels of known points are relevant) are no longer valid. Even if we know, from graph theory, that this upper bound can be computed in pseudo-polynomial time, further structural insights are still required.

4 Approximation of small size

Abstract

In this chapter, we are interested in a problem introduced by Vassilvitskii and Yannakakis [80], the computation of a minimum set of solutions that approximates within an accuracy ε the Pareto set of a multi-objective optimization problem. We first establish a new 3-approximation algorithm for the bicriteria case. We also propose a study of the greedy algorithm performance for the tricriteria case when the points are given explicitly, answering an open question raised by Koltun and Papadimitriou in [54].

Contents

The content of this chapter is based on the following paper.

► C. Bazgan, F. Jamain, and D. Vanderpooten. Approximate Pareto set of minimal size for multi-objective optimization problems: Approximation and Greedy algorithms, submitted.

In the following section we formalize the problem and recall some previous related works. In section 4.2, we mainly propose a new polynomial time 3-approximation algorithm of the size of a smallest ε-Pareto set for the bicriteria case. In section 4.3, we give a performance analysis of the greedy algorithm when the points of the objectives space are given explicitly in the input and the number of criteria is three. We conclude with some possible extensions to this work.

4.1 Problem statements

In this chapter, we are interested in determining efficiently small ε-Pareto sets which is a problem introduced by Vassilvitskii and Yannakakis in [80]. Formally, the problem is the following:

Primal problem: Given an instance of a p-objective problem and an $\varepsilon > 0$, the goal is to obtain an ε-Pareto set of minimal size.

We denote by P_ε^* a smallest ε-Pareto set and by opt_ε its cardinality. We recall that an algorithm that uses a routine is called *generic* if only the routine is specific to a particular problem.

4.1.1 Two objectives

In the bi-objective case, for the primal problem, a generic algorithm that computes an ε-Pareto set of size at most $3opt_\varepsilon$ using routines *GAP*$_\delta$ with $\delta > 0$ was established in [80]. Moreover, if the routine *GAP*$_\delta$ runs in polynomial time (resp. fully polynomial time) then the algorithm also runs in polynomial time (resp. fully polynomial time). This approximation result is tight for the class of problems admitting such a routine. They also showed the existence of a bi-objective problem admitting such a routine that cannot be approximated within a factor better than 3 in polynomial time unless P = NP. It means that there are problems whose the size of a smallest ε-Pareto set is 3-approximable using a generic algorithm and for which it is not possible to do better than 3 even using an algorithm that directly depends on the problem. In the next section, we first show that a main classical bi-objective problem belongs to this class.

We study the problem of determining small ε-Pareto sets with generic algorithms but using a routine called *SoftRestrict* instead of *GAP*.

Definition 15. (SoftRestrict) *Given a positive rational bound b and a parameter $\delta > 0$, the routine* SoftRestrict$_\delta(f_1, f_2 \leq b)$ *either returns a feasible point z satisfying $z_2 \leq (1+\delta)b$ and $z_1 \leq (1+\delta).\min\{f_1(x) : x \in S$ and $f_2(x) \leq b\}$ or correctly reports that there does not exist any feasible point z such that $z_2 \leq b$.*

We say that a routine *SoftRestrict*$_\delta(f_1, f_2 \leq b)$ runs in polynomial time (resp. fully polynomial time when $\delta > 0$) if its running time is polynomial in $|I|$ and $|b|$ (resp. $|I|$, $|b|$, $|\delta|$ and $1/\delta$).

Remark that a routine *SoftRestrict*$_\delta(f_1, f_2 < b)$, with a strict constraint, can easily be simulated by a routine *SoftRestrict*$_\delta(f_1, f_2 \leq b')$ using $b' = b - 2^{-2m}$.

Such a routine was proposed for several problems. For instance, for the BI-OBJECTIVE SPANNING TREE problem, the best known running time of the routine is $O(mn^5\tau(\lfloor(n-1)/\delta\rfloor, \lfloor(n-1)/\delta\rfloor))$ where n is the number of vertices in the graph, m the number of edges in the graph and $\tau(a, b)$ is the time to multiply polynomials of maximum degrees less than or equal to a and b [50]. For a BI-OBJECTIVE SINGLE MACHINE SCHEDULING problem, an algorithm is given in [16] and the running time of the routine is $O(n^5R/(\delta^3L))$ where n is the number of jobs, L and R are respectively lower and upper bounds on the first coordinate of an optimal solution.

4.1.2 More than two objectives

For $p \geq 3$ objectives, Vassilvitskii and Yannakakis [80] showed that any generic algorithm based on routine *GAP*$_\delta$ cannot establish a c-approximation of the size of a smallest ϵ-Pareto set for any constant c.

Note that for several problems the routine *Restrict*$_\delta$ extended to p criteria (one criterion has to be minimized and there is a bound to satisfy on each other criterion) is not computable in polynomial time [43] but it could be the case for the extension of *DualRestrict*$_\delta$ to p criteria. Essentially since the error δ is represented in the $p-1$ constraints in *DualRestrict*$_\delta$ but only once in the minimization in *Restrict*$_\delta$. Recall that for $p = 2$ when routines *DualRestrict*$_\delta$ are computable in polynomial time for both objectives, an ε-Pareto set of size at most $2opt_\varepsilon$ is computable in polynomial time [20]. An interesting problem is to obtain an ε-Pareto set of smallest size for a p-criteria problem using the extension of *DualRestrict*$_\delta$, but it seems quite hard.

To get around this difficulty, the standard way is to assume that the feasible points are given explicitly in the input. Assuming that feasible points are given explivitly in the input, we propose a study of the greedy algorithm performance, which is an open question of Koltun and Papadimitriou [54].

4.2 Two objectives

We first present a hardness result for the BI-OBJECTIVE KNAPSACK problem then we propose a new generic algorithm that approximates the size of a smallest ε-Pareto set to a factor 3, which is much simpler and, in some cases, more efficient than the one presented in [80].

4.2.1 Approximation hardness for Bi-objective Knapsack

Diakonikolas and Yannakakis [20] showed that the size of a smallest ε-Pareto set of Bi-objective Shortest Path and Bi-objective Spanning Tree cannot be approximated within a factor better than 2 in polynomial time, unless P = NP. These results are tight since these two problems admit a routine *Restrict*$_\delta$ that runs in polynomial time, and thus an ε-Pareto set of size at most $2opt_\varepsilon$ is computable in polynomial time as shown in [20]. Vassilvitski and Yannakakis [80] showed that the size of a smallest ε-Pareto set of an artificial variant of Knapsack, called Bi-objective 2-Type-Knapsack, cannot be approximated within a factor better than 3 in polynomial time, unless P = NP. This result is also tight since this problem has a routine *GAP*$_\delta$ that runs in polynomial time, and thus an ε-Pareto set of size at most $3opt_\varepsilon$ is computable in polynomial time as shown in [80].

In this part, we investigate the status of the classical version, called Bi-objective Knapsack, with as input a set Q of items, a capacity c and for each item i two values $v_1(i)$, $v_2(i)$ and a weight $w(i)$. Values and weights are positive rationals. A solution is a nonempty subset Q' of items with total values $v_1(Q') = \sum_{i\in Q'} v_1(i)$, $v_2(Q') = \sum_{i\in Q'} v_2(i)$ and a total weight $w(Q') = \sum_{i\in Q'} w(i) \leq c$. The goal is to maximize the values. First, note that the size of a smallest ε-Pareto set of Bi-objective Knapsack is approximable in polynomial time to a factor 3 since this problem admits an FPTAS [31]. We prove that the size of a smallest ε-Pareto set of Bi-objective Knapsack is not approximable in polynomial time within a factor better than 3, if P $\neq$ NP.

Theorem 5. *For* Bi-objective Knapsack *the size of a smallest ε-Pareto set cannot be approximated within a factor better than 3 in polynomial time, unless* P = NP.

Proof : We construct a reduction from the Partition problem. Thus, from any instance I of Partition, we construct an instance I' of Bi-objective Knapsack such that if the answer of I is 'yes' then the size of the smallest ε-Pareto set of I' is 1 and if the answer of I is 'no' then the size of the smallest ε-Pareto set of I' is 3. Recall that in Partition, the input is a set N of n positive integers $a_1, \ldots, a_n$, and we have to determine if it is possible to partition N into two subsets with equal sum. Starting with such an instance we construct an instance of Bi-objective Knapsack as follows. Let $b = \sum_{i=1}^{n} a_i/2$. For each $i = 1, \ldots, n$, we have one item i with values $v_1(i) = v_2(i) = a_i$ and weight $w(i) = a_i$. In addition, we have two special items α and β with $v_1(\alpha) = (1+\varepsilon)b$, $v_2(\alpha) = 0$, $w(\alpha) = b$ and $v_1(\beta) = 0$, $v_2(\beta) = (1+\varepsilon)b$, $w(\beta) = b$. The capacity of the knapsack is b. Note that if a solution contains a special item, it cannot contain any other item. Let z^α and z^β be the points corresponding to the solution with special item α and β respectively. Consider now solutions without special items. The corresponding points having the same value on each criterion, let z^* be the point with the largest value v^* on each criterion. z^* dominates all other such points.

If I is a 'yes' instance, we have $v^* = b$. Thus, z^* $(1+\varepsilon)$-dominates both z^α and z^β, and $\{z^*\}$ is an ε-Pareto set. If I is a 'no' instance, we have $v^* < b$. Thus z^α and z^β must make part of any ε-Pareto set and $\{z^*, z^\alpha, z^\beta\}$ is a smallest ε-Pareto set. □

Remark that we can generalize the previous result proving that for p-OBJECTIVE KNAPSACK the size of a smallest ε-Pareto set cannot be approximated within a factor better than $p + 1$ in polynomial time, unless P = NP.

4.2.2 A new 3-approximation algorithm

We propose in this section a new 3-approximation algorithm, based on the routine *SoftRestrict*$_\delta$.

Our approximation algorithm has the same approximation ratio as the algorithm presented in [80] but is much simpler, both in its description and in its proof, owing to the use of the routine *SoftRestrict*$_\delta$ instead of *GAP*$_\delta$. Its running time is comparable to the one of [80] and better under some conditions.

Before presenting and analyzing this new 3-approximation algorithm, we first compare the two routines *GAP* and *SoftRestrict*.

4.2.2.1 Comparison of the routines

Proposition 10. *The routines SoftRestrict and GAP are polynomially equivalent.*

Proof : We first show that we can answer to *GAP*$_\delta(y)$ using *SoftRestrict*$_\delta(f_1, f_2 \leq y_2/(1+\delta))$. Indeed, if *SoftRestrict*$_\delta(f_1, f_2 \leq y_2/(1+\delta))$ returns NO or returns a feasible point z with $z_1 > y_1$, we return NO and if *SoftRestrict*$_\delta(f_1, f_2 \leq y_2/(1+\delta))$ returns a feasible point z with $z_1 \leq y_1$ we return z.

We give in the following an algorithm that computes the function *SoftRestrict*$_\delta(f_1, f_2 \leq b)$ using a polynomial number of calls to *GAP*$_{\delta'}$ where $\delta' = \sqrt{1+\delta} - 1$. We first call *GAP*$_{\delta'}((1+\delta')2^m, (1+\delta')b)$. If it returns NO, then we also return NO for *SoftRestrict*$_\delta(f_1, f_2 \leq b)$. Otherwise, we partition the objective space by defining intervals, on the first objective, from $2^{-m}/(1+\delta')$ to 2^m such that the ratio between the upper and lower bounds of each interval is $1+\delta'$. We perform a binary search on the upper bounds of the previous intervals calling *GAP*$_{\delta'}(a, (1+\delta')b)$ for some a until one finds a value a^* such that (i) *GAP*$_{\delta'}(a^*(1+\delta'), (1+\delta')b)$ returns a feasible point z^* and (ii) *GAP*$_{\delta'}(a^*, (1+\delta')b)$ returns NO. Then we return z^*.

The number of subdivisions on the first coordinate is $2m/\log(1+\delta') \approx \Theta(4m/\delta')$. Hence, the number of calls to *GAP*$_{\delta'}$ is $\Theta(\log(m/\delta'))=\Theta(\log(m/\delta))$. □

Corollary 3. *Consider the class of bi-objective problems that possess a fully polynomial time routine SoftRestrict*$_\delta$ *with* $\delta > 0$ *for both objectives. Then, for any* $\varepsilon > 0$*, there is no polynomial time generic algorithm using SoftRestrict*$_\delta$ *that computes an* ε*-Pareto set of size less than or equal to* $3opt_\varepsilon$*.*

Proof : Follows from Proposition 10 and the fact that the same result holds for the routine *GAP*$_\delta$ [80]. □

4.2.2.2 Algorithm description

We first describe briefly the idea of the algorithm. We compute by f_1^{min} and f_2^{min} lower bounds on the minimum values on the first and second objectives using *SoftRestrict*$_\delta$. The algorithm iteratively generates a sequence of points $r^1, q^1, \ldots, r^s, q^s$. Points $q^1, \ldots, q^s$ are selected in decreasing order according to f_1 and increasing order according to f_2. Point q^1 is selected so as to $(1+\varepsilon)$-dominate the feasible points that have an optimal second coordinate while getting the best possible value on f_1. The algorithm stops when generates a point q^s that $(1+\varepsilon)$-dominates the feasible points that have a first coordinate equal to f_1^{min}. Routines *SoftRestrict*$_\delta(f_2, f_1 \leq b)$ and *SoftRestrict*$_\delta(f_1, f_2 \leq b)$ are alternatively used to construct points r^i and points q^i respectively. Point r^i is a point with a smallest second coordinate that we can determine with the routine *SoftRestrict*$_\delta$ that is not $(1+\varepsilon)/(1+\delta)$-dominated by the points q^j with $j < i$. Point q^i is a point with a smallest first coordinate that we can determine with routine *SoftRestrict*$_\delta$ that $(1+\varepsilon)$-dominates point r^i.

A formal description of this algorithm is given in Algorithm 4.1. In order to obtain a 3-approximation algorithm, we consider in the following that $\delta \leq \sqrt[3]{1+\varepsilon} - 1$.

We propose an illustration of Algorithm 4.1 in Figure 4.1 where 3 points q^1, q^2, q^3 are already selected and such that, instead of these three points, only one point p^{*1} was sufficient.

4.2.2.3 Algorithm analysis

We show now that Algorithm 4.1 produces a 3-approximation of the size of a smallest ε-Pareto set. Let $Q = \{q^1, \ldots, q^s\}$ and $R = \{r^1, \ldots, r^s\}$ be the sets of feasible points produced by the algorithm. We show in the following that set Q is an ε-Pareto set, then that its size is at most three times the size of P_ε^*, an ε-Pareto set of minimal size. The proof is essentially the same as the one in [20] for the 2-approximation algorithm. We first show some preliminarily results regarding points in Q and R.

Lemma 6. *For all* $i = 2, \ldots, s$ *we have* (i) $r_1^i < q_1^{i-1}(1+\delta)/(1+\varepsilon)$ *and* (ii) *for each feasible point* z *with* $z_1 < q_1^{i-1}/(1+\varepsilon)$*, we have* $z_2 \geq max\{\overline{f_2}^{i-1}, r_2^i/(1+\delta)\}$*.*

Algorithm 4.1: Algorithm SoftGreedy

---

input : An instance of a bi-objective problem for which routines *SoftRestrict*$_\delta(f_1, f_2 \leq b)$ and *SoftRestrict*$_\delta(f_2, f_1 \leq b)$ are available

output : An ε-Pareto set of size at most $3opt_\varepsilon$

1 $f_1^{min} \leftarrow f_1(\mathit{SoftRestrict}_\delta(f_1, f_2 \leq 2^m))/(1+\delta)$;
   $f_2^{min} \leftarrow f_2(\mathit{SoftRestrict}_\delta(f_2, f_1 \leq 2^m))/(1+\delta)$;
2 $r^1 \leftarrow \mathit{SoftRestrict}_\delta(f_2, f_1 \leq 2^m)$;
3 $\overline{f_2}^1 \leftarrow \frac{1+\varepsilon}{(1+\delta)^2} r_2^1$;
4 $q^1 \leftarrow \mathit{SoftRestrict}_\delta(f_1, f_2 \leq \overline{f_2}^1)$;
5 $\overline{f_1}^1 \leftarrow q_1^1/(1+\varepsilon)$;
6 $Q \leftarrow \{q^1\}$;
7 $i \leftarrow 1$;
8 **while** $\overline{f_1}^i > f_1^{min}$ **do**
9     $i \leftarrow i+1$;
10     $r^i \leftarrow \mathit{SoftRestrict}_\delta(f_2, f_1 < \overline{f_1}^{i-1})$;
11     $\overline{f_2}^i \leftarrow \frac{1+\varepsilon}{1+\delta} max\{\overline{f_2}^{i-1}, r_2^i/(1+\delta)\}$;
12     $q^i \leftarrow \mathit{SoftRestrict}_\delta(f_1, f_2 \leq \overline{f_2}^i)$;
13   **if** $q_1^i > r_1^i$ **then**
14         $q^i \leftarrow r^i$;
15     $\overline{f_1}^i \leftarrow q_1^i/(1+\varepsilon)$;
16     $Q \leftarrow Q \cup \{q^i\}$;
17 **return** Q;

Proof : This results from the definition of the routine *SoftRestrict*$_\delta$ and steps 10-12 and 15 of the algorithm. □

Lemma 7. *For all* $i = 1, \ldots, s$ *we have* (i) $q_2^i \leq (1+\delta)\overline{f_2}^i$ *and* (ii) *for each feasible point* z *with* $z_2 \leq \overline{f_2}^i$*, we have* $z_1 \geq q_1^i/(1+\delta)$.

Proof : This results from the definition of the routine *SoftRestrict*$_\delta$ and steps 10-12 of the algorithm. □

We can now prove the following results.

Proposition 11. *Set* Q *is an* ε*-Pareto set.*

Proof : We show that the points in Q cover all the feasible points by partitioning the range of feasible values on f_1. More precisely, we show that:

(i) Point q^1 $(1+\varepsilon)$-dominates all the feasible points with an f_1 value greater than or equal to $q_1^1/(1+\varepsilon)$.

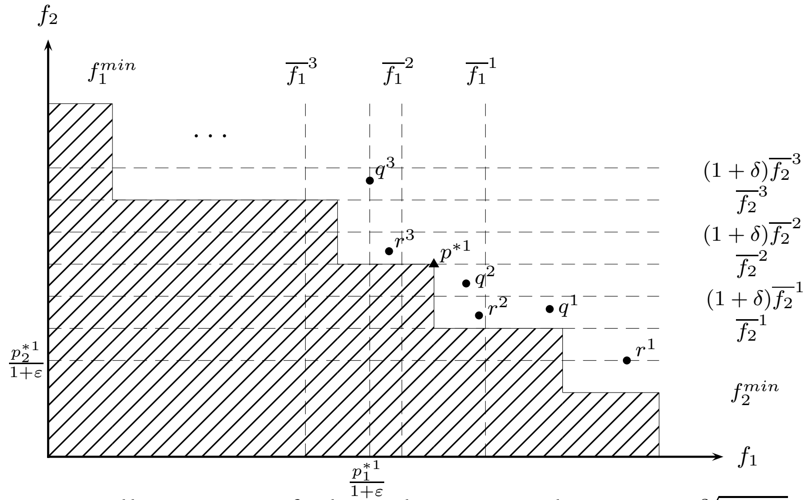

Figure 4.1: Illustration of Algorithm 4.1 with $1+\delta = \sqrt[3]{1+\varepsilon}$

(ii) For each $i = 2, \dots, s$ point q^i $(1+\varepsilon)$-dominates all the feasible points that have their f_1 value in the interval $\left[q_1^i/(1+\varepsilon), q_1^{i-1}/(1+\varepsilon)\right)$.

(iii) There is no feasible point with a f_1 value smaller than $q_1^s/(1+\varepsilon)$.

(i) Let z be a feasible point with $z_1 \geq q_1^1/(1+\varepsilon)$. We need to show that z is $(1+\varepsilon)$-dominated by q^1, i.e. that $z_2 \geq q_2^1/(1+\varepsilon)$. From steps 2-4 we get where $q_2^1 \leq r_2^1(1+\varepsilon)/(1+\delta) \leq f_2^{min}(1+\varepsilon)$ and thus $q_2^1/(1+\varepsilon) \leq f_2^{min} \leq z_2$.

(ii) Let z be a feasible point satisfying $q_1^i/(1+\varepsilon) \leq z_1 < q_1^{i-1}/(1+\varepsilon)$. We need to show that z is $(1+\varepsilon)$-dominated by q^i, i.e. that $z_2 \geq q_2^i/(1+\varepsilon)$. From Lemma 6-$(ii)$ we have $z_2 \geq max\{\overline{f_2}^{i-1}, r_2^i/(1+\delta)\}$. Furthermore from Lemma 9-(i) we have $q_2^i \leq (1+\delta)\overline{f_2}^i$. Hence, from the definition of $\overline{f_2}^i$ (step 11), we get $q_2^i \leq max\{\overline{f_2}^{i-1}, r_2^i/(1+\delta)\} \leq (1+\varepsilon)z_2$.

(iii) The stopping condition of the algorithm (step 8) is $\overline{f_1}^s = q_1^s/(1+\varepsilon) \leq f_1^{min}$. □

Therefore, we only need to show that the size of Q is at most three times the size of an optimal ε-Pareto set.

Proposition 12. *Set Q is such that $|Q| \leq 3opt_\varepsilon$.*

Proof: Let $P_\varepsilon^* = \{p^{*1}, \dots, p^{*k}\}$ be an optimal ε-Pareto set, where its points p^{*i} for $i = 1, \dots, k$ are in increasing order of their coordinates on f_2 and decreasing order of their coordinates on f_1. We have to show that $|Q| \leq 3k$. For this purpose, we show by induction on i that if the algorithm selects a feasible point q^{3i-2} then there must exist a point p^{*i} in P_ε^*, if the algorithm selects a point q^{3i-1} then $p_1^{*i}(1+\delta) \geq q_1^{3i-1}$ and if the algorithm selects a point q^{3i} then $p_1^{*i} > (1+\delta)q_1^{3i}$.

Initialization ($i = 1$). The first statement trivially holds. To prove the second statement we first show that $p_2^{*1} \leq \overline{f_2}^2$. Since the feasible point r^1 must be

$(1+\varepsilon)$-dominated by a point of P^*_ε and in particular by point p^{*1}, we have $p^{*1}_2 \le r^1_2(1+\varepsilon)$. From the definitions of $\overline{f_2}^1$ and $\overline{f_2}^2$ (step 11) and using that $\delta \le \sqrt[3]{1+\varepsilon}-1$, we have $r^1_2(1+\varepsilon) = \overline{f_2}^1(1+\delta)^2 \le \overline{f_2}^2$. Using that $p^{*1}_2 \le \overline{f_2}^2$ and Lemma 9-(ii) we obtain $p^{*1}_1(1+\delta) \ge q^2_1$. Since we have $q^3_1 \le r^3_1$ (steps 13 and 14) Lemma 6-(i) implies that $q^3_1 < q^2_1(1+\delta)/(1+\varepsilon)$, and so that the third statement is a consequence of the second one, using that $\delta \le \sqrt[3]{1+\varepsilon}-1$.

Induction step. Assume the result is true until index $i-1$, we prove it for index i. First, if the algorithm selects a point q^{3i-2}, we show that P^*_ε contains a point p^{*i}. By the termination condition of the algorithm (step 8), we have $q^{3i-3}_1 > (1+\varepsilon)f^{min}_1$ and by the induction hypothesis that $p^{*i-1}_1 > (1+\delta)q^{3i-3}_1$, it follows that $p^{*i-1}_1 > (1+\delta)(1+\varepsilon)f^{min}_1$. Thus, point p^{*i-1} does not $(1+\varepsilon)$-dominate the feasible points that have a minimum first coordinate, and so P^*_ε must contain another point p^{*i}. To prove the second statement we first show that $p^{*i}_2 \le \overline{f_2}^{3i-1}$. Since the feasible point r^{3i-2} must be $(1+\varepsilon)$-dominated by a point of P^*_ε and in particular by point p^{*i}, we have $p^{*i}_2 \le r^{3i-2}_2(1+\varepsilon)$. From the definitions of $\overline{f_2}^{3i-2}$ and $\overline{f_2}^{3i-1}$ (step 11) and using that $\delta \le \sqrt[3]{1+\varepsilon}-1$, we have $r^{3i-2}_2(1+\varepsilon) = \overline{f_2}^{3i-2}(1+\delta)^2 \le \overline{f_2}^{3i-1}$. Using that $p^{*i}_2 \le \overline{f_2}^{3i-1}$ and Lemma 9-(ii) we obtain $p^{*i}_1(1+\delta) \ge q^{3i-1}_1$. Since we have $q^{3i}_1 \le r^{3i}_1$ (steps 13 and 14) Lemma 6-(i) implies that $q^{3i}_1 < q^{2i-1}_1(1+\delta)/(1+\varepsilon)$, and so that the third statement is a consequence of the second one, using that $\delta \le \sqrt[3]{1+\varepsilon}-1$. □

Thus, we obtain the following result:

Theorem 6. *Algorithm 4.1 computes an ε-Pareto set of size less than or equal to $3opt_\varepsilon$ using $O(opt_\varepsilon)$ routine calls to SoftRestrict$_\delta$, when $\delta \le \sqrt[3]{1+\varepsilon}-1$.*

Proof: Since the algorithm uses $2\,|Q|$ times the routine *SoftRestrict*$_\delta$, the number of routine calls is bounded by $6opt_\varepsilon$, the result is a direct consequence of Propositions 11 and 12. □

4.2.2.4 Comparison to existing algorithms

For the class of problems admitting a routine *GAP* that runs in polynomial time, the algorithm of Vassilvitski and Yannakakis presented in [80] was the only one ensuring some guarantee on the size of the returned ε-Pareto set. This algorithm, called *ZIGZAG* is a generic algorithm based on the routine *GAP*, that establishes a 3-approximation of the size of a smallest ε-Pareto set and it needs $O(opt_\varepsilon \cdot \log(m/\varepsilon))$ routine calls. Since algorithm *ZIGZAG* and our algorithm run in polynomial time for the same class of problems and give the same approximation ratio of a smallest ε-Pareto set we can compare them with regard to their running times. The running time of a generic algorithm is defined as the product between the number of routine calls and the running time of the routine called. This way, the running time of algorithm *ZIGZAG* is $O(opt_\varepsilon \cdot \log(m/\varepsilon)) \cdot T_{GAP_\delta}$

with $\delta = \sqrt[4]{1+\varepsilon} - 1$ and the running time of the algorithm *SoftGreedy* is $O(opt_\varepsilon) \cdot T_{SoftRestrict_\delta}$ with $\delta = \sqrt[3]{1+\varepsilon} - 1$ where T_{GAP_δ} and $T_{SoftRestrict_\delta}$ are the running times of the routines GAP_δ and $SoftRestrict_\delta$ respectively.

The running times of algorithms *SoftGreedy* and *ZIGZAG* are comparable since we can solve the routine $SoftRestrict_\delta$ using $\Theta(\log(m/\varepsilon))$ calls to $GAP_{\delta'}$ with $\delta' = \sqrt{1+\delta} - 1$ (see the proof of Proposition 10). Moreover, if we can solve faster the routine $SoftRestrict_\delta$, the running time of algorithm *SoftGreedy* will be smaller than the one of algorithm *ZIGZAG*. Especially, if the best known algorithm to solve GAP_δ solves $SoftRestrict_\delta$ in the same time, we gain the time of the binary searches. It is the case for BI-OBJECTIVE SPANNING TREE since the algorithm in [50] that solves $SoftRestrict_\delta$ is the best known to solve GAP_δ.

4.3 More than two objectives

For $p \geq 3$ objectives, Vassilvitski and Yannakakis [80] showed that any generic algorithm based on routine GAP_δ cannot establish a c-approximation of the size of a smallest ϵ-Pareto set for any constant c. Note that for several problems the routine $Restrict_\delta$ extended to p criteria (one criterion has to be minimized and there is a bound to satisfy on each other criterion) is not computable in polynomial time [43] but it could be the case for the extension of $DualRestrict_\delta$ to p criteria. Essentially since the error δ is represented in the $p-1$ constraints in $DualRestrict_\delta$ but only once in the minimization in $Restrict_\delta$. Recall that for $p=2$ when routines $DualRestrict_\delta$ are computable in polynomial time for both objectives, an ε-Pareto set of size at most $2opt_\varepsilon$ is computable in polynomial time [20]. An interesting problem is to obtain an ε-Pareto set of smallest size for a p-criteria problem using the extension of $DualRestrict_\delta$, but it seems quite hard. To get around this difficulty, we assume in this section that the feasible points are given explicitly in the input. In this case, we can easily filter out the dominated points and thus we consider in the following that the input contains only nondominated points.

When $p = 3$, the problem of finding an ε-Pareto set of smallest size is c-approximable, for some constant c [54]. Moreover, for any number of criteria there is a $O(\log n)$-approximation since the instances of the problem are particular ones of the SET COVER problem. We prove that the *greedy* algorithm for SET COVER does not perform better than its worst case, $O(\log n)$, on these particular instances. Thus we answer an open question of Koltun and Papadimitriou [54]. The greedy algorithm iteratively selects the point that covers the largest number of non-covered points.

Note that for 2 criteria, the greedy algorithm gives a 2-approximation, but it is not really satisfying because the algorithm of Diakonikolas and Yannakakis [20] finds an optimal solution when the points are given explicitly in the input.

Theorem 7. *For $p \geq 3$ objectives, when the feasible points are given explicitly in the input, the solution produced by the greedy algorithm for* SET COVER *has a size $\Theta(\log n) \cdot opt_\varepsilon$ in the worst case.*

Proof: We prove the result for $p = 3$. The result clearly extends to $p \geq 4$ since we can consider the same points extended with the last $p - 3$ coordinates to 0. In order to prove it, we give a family of instances where the algorithm produces a $\Theta(\log n)$-approximation. Let a, b, c be three nonnegative integers, $n = 2^\ell$ and consider $N = 2^\ell + \ell - 3 = n + \log n - 3$ nondominated points in the criterion space, defined as follows.

For all $i = 1, \ldots, \log n - 1$ let:
$z^i = (\frac{a}{(1+\varepsilon)^{i/(\log n-1)}}, \frac{b}{(1+\varepsilon)^{i/(\log n-1)}}, c(1+\varepsilon)^{(i-1)/(\log n-1)+1})$,

Γ_i be a set of $\frac{n}{2^{i+1}}$ points lying uniformly on the line from $(a, \frac{b}{(1+\varepsilon)^{(i-1)/(\log n-1)+1}} - 1, c(1+\varepsilon)^{\frac{i-1}{\log n-1}})$ to $(a(1+\varepsilon), \frac{b}{(1+\varepsilon)^{i/(\log n-1)+1}}, c(1+\varepsilon)^{(i-1)/(\log n-1)})$ if $i \neq \log n - 1$ and be the singleton $\{(a, \frac{b}{(1+\varepsilon)^{(i-1)/(\log n-1)+1}} - 1, c(1+\varepsilon)^{(i-1)/(\log n-1)})\}$ if $i = \log n - 1$,

Γ'_i be a set of $\frac{n}{2^{i+1}}$ points lying uniformly on the line from $(\frac{a}{(1+\varepsilon)^{(i-1)/(\log n-1)+1}} - 1, b, c(1+\varepsilon)^{(i-1)/(\log n-1)})$ to $(\frac{a}{(1+\varepsilon)^{i/(\log n-1)+1}}, b(1+\varepsilon), c(1+\varepsilon)^{(i-1)/(\log n-1)})$ if $i \neq \log n - 1$ and be the singleton $\{(\frac{a}{(1+\varepsilon)^{(i-1)/(\log n-1)+1}} - 1, b, c(1+\varepsilon)^{(i-1)/(\log n-1)})\}$ if $i = \log n - 1$.

Note that if the coordinates of points z^i are not rational, we approximate its coordinates from below by rational ones. Moreover, if the coordinates of points in $\Gamma \cup \Gamma'$ are not rational, we approximate its coordinates from above by rational ones.

Let $\xi = \cup_i \{z^i\}$ and $\Gamma = \cup_i \Gamma_i$, $\Gamma' = \cup_i \Gamma'_i$. Remark that a, b, c can be set sufficiently large to have the following $(1+\varepsilon)$-dominance relations:

1. for any $z^i, z^j \in \xi$, $z^i \preceq_\varepsilon z^j$
2. for any $i, j = 1, \ldots, \log n - 1$, for any $z \in \Gamma_j \cup \Gamma'_j$, $z^i \preceq_\varepsilon z$ iff $i = j$
3. for any $z, z' \in \Gamma$, $z \preceq_\varepsilon z'$
4. for any $z, z' \in \Gamma'$, $z \preceq_\varepsilon z'$
5. for any $z \in \Gamma$, $z' \in \Gamma'$, $z \not\preceq_\varepsilon z'$ and $z' \not\preceq_\varepsilon z$

We show in the following that the greedy algorithm selects the points z^i, $i = 1, \ldots, \log n - 1$ in this order. The proof is by induction on i.

Initialization (i=1). Note that $|\Gamma| = |\Gamma'| = \frac{n}{2} - 1$. From 5 it follows that any point in $\Gamma \cup \Gamma'$ $(1+\varepsilon)$-dominates at most $\frac{n}{2} + \log n - 2$ points. From 1 and 2 point z^i $(1+\varepsilon)$-dominates exactly the points in $\xi \cup \Gamma_i \cup \Gamma'_i$, where $|\xi \cup \Gamma_i \cup \Gamma'_i| = \frac{n}{2^i} + \log n - 1$. In particular point z^1 $(1+\varepsilon)$-dominates $\frac{n}{2} + \log n - 1$ points. Therefore, point z^1 is the first point selected by the greedy algorithm.

Induction step. Assuming that the first $i - 1$ points selected by the greedy algorithm are $z^1, \ldots, z^{i-1}$, we prove that the next one is point z^i. The points

$(1+\varepsilon)$-dominated by $\{z^1, \ldots, z^{i-1}\}$ are exactly the points in $\xi \cup (\cup_{j=1}^{i-1}(\Gamma_j \cup \Gamma'_j))$. Therefore, any point in $\Gamma \cup \Gamma'$ $(1+\varepsilon)$-dominates exactly $|\cup_{j=i}^{\log n-1}(\Gamma_j \cup \Gamma'_j)| = \frac{n}{2^i} - 1$ points that are not already covered. Point z_i exactly $(1+\varepsilon)$-dominates, among the non-covered points, the points in $\Gamma_i \cup \Gamma'_i$ where $|\Gamma_i \cup \Gamma'_i| = \frac{n}{2^i}$. Thus point z^i is selected by the greedy algorithm at step i.

Observe now, that the first point of Γ_1 $(1+\varepsilon)$-dominates all the points in ξ. Thus, from 3 and 4, it follows that a set constituted by the first point of Γ_1 and any point in Γ' $(1+\varepsilon)$-dominate the N points. Therefore, the greedy algorithm returns a set of points of size $\log n - 1$ while an optimal set of points contains only two points. □

4.4 Conclusions

In this chapter, we investigated a problem introduced by Vassilvitskii and Yannakakis [80], computing a minimum set of solutions for a multi-objective optimization problem that represents approximately the Pareto set within an accuracy ε. First, for the bi-objective case, we presented a new 3-approximation algorithm of the size of a smallest ε-Pareto set. We showed that for a classical bi-objective problem this approximation is tight unless P $=$ NP. Since this problem is particularly difficult when the number of criteria is greater than two, when $p \geq 3$, we standardly considered that the points of the objective space are given explicitly in the input. In these settings, we studied the greedy algorithm performance, answering an open question of Koltun and Papadimitriou [54].

There is an interesting problem that remains open for a number of criteria greater than or equal to 3. When $p \geq 3$ and the points are given explicitly in the input, it is NP-hard to determine an ε-Pareto set of minimal size but there is a 100-approximation of the size of a smallest ε-Pareto set [54]. The open problem is to make this lower and upper bounds closer.

5 Representation using kernels

Abstract

In this chapter, we are interested in producing discrete and tractable representations of the set of non-dominated points for multi-objective optimization problems. These representations must satisfy some conditions of *coverage*, i.e. providing a good approximation of the non-dominated set, *spacing*, i.e. whitout redundancies, and *cardinality*, i.e. with the smallest possible number of points. This leads us to focus on ε-Pareto sets of small size satisfying an additional condition of ε'*stability*, called $(\varepsilon, \varepsilon')$-*kernels* or ε-*kernels* when $\varepsilon' = \varepsilon$ is possible.

We first establish some general properties on ε-kernels. Then, for the bi-objective case, we propose some generic algorithms computing in polynomial time either an ε-kernel of small size or, for a fixed size k, an ε-kernel with a nearly optimal approximation ratio $1 + \varepsilon$. For more than two objectives, we show that ε-kernels do not necessarily exist but that $(\varepsilon, \varepsilon')$-kernels with $\varepsilon' \leq \sqrt{1+\varepsilon} - 1$ always exist. Nevertheless, we show that the size of a smallest $(\varepsilon, \varepsilon')$-kernel can be very far from the size of a smallest ε-Pareto set.

Contents

The content of this chapter is based on the following paper.

► C. Bazgan, F. Jamain, and D. Vanderpooten. Representation of the non-dominated set for multi-objective optimization problems using kernels, submitted.

In the following section, we define the concept of representation, recall the notion of $(\varepsilon, \varepsilon')$-kernels and we formalize the problem. In section 5.2, we study the bi-objective case. We show some general results and present polynomial time algorithms to construct small $(\varepsilon, \varepsilon')$-kernels under some conditions. In section 5.3, we study the case of three or more objectives, pointing out specific difficulties. We conclude with some possible extensions to this work.

5.1 Problem statements

A good representation of the Pareto set is evaluated according to three main dimensions. We already deal with two of them in the previous chapter, the quality of the *coverage*, i.e. providing a good approximation, and the *cardinality*, i.e. that does not contain too many points. The last dimension, which is the goal of this chapter, is the *spacing*, i.e. that does not include any redundancies [72].

One way to assure the spacing is to consider an $(\varepsilon, \varepsilon')$-*kernel* that is a special ε-Pareto set for which we impose a condition of stability. We recall the definition:

Definition 16. *Given a set Z of feasible points and $\varepsilon, \varepsilon' > 0$, an $(\varepsilon, \varepsilon')$-kernel is a set of points $K_{\varepsilon,\varepsilon'}$ satisfying the two following conditions:*

(i) for any point $z' \in Z \setminus K_{\varepsilon,\varepsilon'}$, there exists $z \in K_{\varepsilon,\varepsilon'}$ such that $z \preceq_\varepsilon z'$ (ε-coverage).

(ii) for any two distinct points $z, z' \in K_{\varepsilon,\varepsilon'}$, we do not have $z \preceq_{\varepsilon'} z'$ (ε'-stability).

Since when $\varepsilon' > \varepsilon$ an $(\varepsilon, \varepsilon')$-kernel does not always exist, for a given ε the goal is to find an $(\varepsilon, \varepsilon')$-kernel with the largest ε'. When $\varepsilon' = \varepsilon$ an $(\varepsilon, \varepsilon')$-kernel is called an ε-kernel.

In this chapter, our goal is to establish some general properties on ε-kernels and propose some solutions to the primal and dual problems for the case of ε-kernels. In the following sections, when we talk about primal and dual problems, we refer to the version where we want ε-kernels instead of ε-Pareto sets.

5.2 Two objectives

We first give some general results on ε-kernels in the bi-objective case (section 5.2.1). Then we consider the computation of ε-kernels when an exact Restrict routine is available (section 5.2.2) and when we only have an approximate Restrict routine (section 5.2.3).

5.2.1 General results

We first show that, in the bi-objective case, an ε-kernel always exists.

Lemma 8. *In the bi-objective case, relation $\prec_\varepsilon$ does not contain cycles.*

Proof: Suppose that we have the cycle $z^1 \prec_\varepsilon z^2 \ldots \prec_\varepsilon z^n \prec_\varepsilon z^1$. Thus, for all $i \in \{1, \ldots, n-1\}$ we have (i) $z^i_j/(1+\varepsilon) \leq z^{i+1}_j$ for each $j \in \{1,2\}$ and (ii) there exists $j \in \{1,2\}$ such that $z^i_j(1+\varepsilon) < z^{i+1}_j$. Moreover, we have (i) $z^n_j/(1+\varepsilon) \leq z^1_j$ for each $j \in \{1,2\}$ and (ii) there exists $j \in \{1,2\}$ such that $z^n_j(1+\varepsilon) < z^1_j$.

Considering this cycle, assume that we are t_j times in case (ii) for each $j \in \{1,2\}$. We must have $t_1 + t_2 \geq n$. First, remark that it is not possible that $t_j = 0$ for each $j \in \{1,2\}$. Indeed, assuming wlog that $t_1 = 0$, we get $t_2 = n$ leading to $(1+\varepsilon)^n < 1$. Now, observe that when we are t_j times in case (ii) for criterion j, we are also $n - t_j$ times in case (i). Since $t_j > 0$ for each $j \in \{1,2\}$, we have $z^1_j/(1+\varepsilon)^{n-2t_j} < z^1_j$, which implies $t_j < n/2$ for each $j \in \{1,2\}$, contradicting $t_1 + t_2 \geq n$. □

Proposition 13. *In the bi-objective case, an ε-kernel always exists.*

Proof: It is a direct consequence of Lemma 8 since any relation that does not admit cycles in its asymmetric part admits kernels as proved in Duchet [22]. □

In general ε-kernels may contain dominated points. We prove the existence of ε-kernels containing only nondominated points.

Proposition 14. *In the bi-objective case, an ε-kernel that contains only nondominated points always exists.*

Proof: Let K_ε be an ε-kernel of the Pareto set P. Proposition 13 implies that such an ε-kernel does exist. K_ε is clearly an ε-Pareto set and contains only nondominated points by definition. □

In the following we give some bounds on the size of any ε-kernel.

Theorem 8. *In the bi-objective case, any ε-kernel has a cardinality less than or equal to $3opt_\varepsilon$.*

Proof : The proof is by contradiction. Let P^*_ε be an ε-Pareto set of minimal size opt_ε. Now assume that there exists an ε-kernel K_ε of size at least $3opt_\varepsilon + 1$. It means that at least one point z^* of P^*_ε $(1+\varepsilon)$-dominates at least 4 points of K_ε.

Let z^i for $i = 1,2,3,4$ be 4 points of K_ε such that $z^* \preceq_\varepsilon z^i$ for each $i = 1,2,3,4$. Assume wlog that $z^{i+1}_1 < z^i_1$ and $z^{i+1}_2 > z^i_2$ for $i = 1,2,3$. Since K_ε is an ε-kernel, the coordinates of the points z^i must satisfy the following inequalities: $z^{i+1}_1 < z^i_1/(1+\varepsilon)$ and $z^{i+1}_2 > z^i_2(1+\varepsilon)$. Using these inequalities and since $z^* \preceq_\varepsilon z^i$ for each $i = 1,2,3,4$, its coordinates satisfy $z^*_1 \leq z^4_1(1+\varepsilon) < z^3_1 < z^2_1/(1+\varepsilon) < z^1_1/(1+\varepsilon)$ and $z^*_2 \leq z^1_2(1+\varepsilon) < z^2_2 < z^3_2/(1+\varepsilon) < z^4_2/(1+\varepsilon)$. Thus no point z^i for $i = 1,\ldots,4$ $(1+\varepsilon)$-dominates z^*. If another point z of K_ε $(1+\varepsilon)$-dominates z^* the previous inequalities give $z_1 \leq z^*_1(1+\varepsilon) < z^3_1(1+\varepsilon) < z^2_1$ and $z_2 \leq z^*_2(1+\varepsilon) < z^2_2(1+\varepsilon) < z^3_2$ which involves that point z $(1+\varepsilon)$-dominates points z^2 and z^3. This contradicts the fact that K_ε is an ε-kernel. Thus, no point of K_ε $(1+\varepsilon)$-dominates z^*, contradiction. □

If we consider ε-kernels containing nondominated points only, we obtain a smaller upper bound on their size. The following result is even slightly stronger since it deals with ε-kernels containing *weakly* nondominated points only.

Theorem 9. *In the bi-objective case, any ε-kernel that contains only weakly nondominated points has a cardinality less than or equal to $2opt_\varepsilon$.*

Proof : The proof is by contradiction. Let P^*_ε be an ε-Pareto set of minimal size opt_ε. Now assume that there exists an ε-kernel K_ε of size at least $2opt_\varepsilon + 1$ containing only weakly nondominated points. It means that at least one point z^* of P^*_ε $(1+\varepsilon)$-dominates at least 3 points of K_ε.

Let z^i for $i = 1,2,3$ be 3 points of K_ε such that $z^* \preceq_\varepsilon z^i$ for each $i = 1,2,3$. Assume wlog that $z^{i+1}_1 < z^i_1$ and $z^{i+1}_2 > z^i_2$. Since K_ε is an ε-kernel, the coordinates of the points z^i must satisfy the following inequalities: $z^{i+1}_1 < z^i_1/(1+\varepsilon)$ and $z^{i+1}_2 > z^i_2(1+\varepsilon)$ for $i = 1,2$. Since $z^* \preceq_\varepsilon z^i$ for each $i = 1,2,3$, the coordinates of point z^* must satisfy $z^*_1 \leq z^3_1(1+\varepsilon) < z^2_1$ and $z^*_2 \leq z^1_2(1+\varepsilon) < z^2_2$. This contradicts the fact that z^2 is a weakly nondominated point. □

Corollary 4. *In the bi-objective case, there exists an ε-kernel with a cardinality less than or equal to $2opt_\varepsilon$.*

Proof : It is a direct consequence of Theorem 9 and Proposition 14. □

We are interested now on ε-kernels of minimal size. An important fact is that an ε-kernel of minimal size is not larger than an ε-Pareto set of minimal size opt_ε.

Theorem 10. *In the bi-objective case, there exists an ε-kernel of size opt_ε.*

A constructive proof of Theorem 10 is given in the next section, where an algorithm that computes an ε-kernel of size opt_ε is provided.

5.2.2 Algorithms for ε-kernels using exact routines

In this section, we provide algorithms for the primal problem (section 5.2.2.1) and the dual problem (section 5.2.2.2) considering that a routine *Restrict*$_0$ is available for both objectives.

5.2.2.1 Primal problem

We propose an algorithm that produces an ε-kernel of minimal size that contains only nondominated points.

Algorithm description The algorithm proceeds in two phases. The first phase (greedy phase) corresponds to a slightly modified version of the algorithm presented in [20] which returns a set $\{q_1, \ldots, q_s\}$ of nondominated points as an ε-Pareto set of minimal size. The second phase (verification phase) ensures ε-stability by checking, and possibly modifying, the returned set. We denote by f_1^{min} and f_2^{min} the minimum values on the first and second objectives respectively. In the first phase, the algorithm iteratively generates points $r^1, q^1, \ldots, r^s, q^s$ in decreasing order according to f_1 and increasing order according to f_2. Point r^1 corresponds to an optimal solution on objective f_2. Point q^1 is the nondominated point with the best possible value on f_1 which $(1+\varepsilon)$-dominates r^1. Routines *Restrict*$_0(f_2, f_1 \leq b)$ and *Restrict*$_0(f_1, f_2 \leq b)$ are used alternatively to construct points r^i and points q^i respectively. Point r^i is a point with the smallest value on f_2 that is not $(1+\varepsilon)$-dominated by the point q^{i-1}. Point q^i is the nondominated point with the smallest value on f_1 that $(1+\varepsilon)$-dominates point r^i. The first phase of the algorithm stops when it determines a point q_s that $(1+\varepsilon)$-dominates the feasible points that have a first coordinate equal to f_1^{min}. At the end of the first phase, ε-stability is ensured on the first objective, but not on the second one. In the second phase, points q^i are scanned from q^s in decreasing order according to f_2. If point q^i $(1+\varepsilon)$-dominates point q^{i-1}, we replace point q^{i-1} by the nondominated point with the smallest f_1 value which is not $(1+\varepsilon)$-dominated by q^i while having a strictly larger value on f_1 than q^{i-1}. This ensures ε-stability on the second objective, while preserving ε-stability on the first one.

A formal description of this algorithm is given in Algorithm 5.1.

We propose an illustration of this algorithm in Figure 5.1 where 4 points q^1, q^2, q^3, q^4 are selected during the first phase. During the second phase, the algorithm detects that point q^3 $(1+\varepsilon)$-dominates point q^2 and, therefore, replaces q^2 by q'^2 which is not $(1+\varepsilon)$-dominated by q^3 but $(1+\varepsilon)$-dominates all the points that were $(1+\varepsilon)$-dominated by q^2 only.

Algorithm 5.1: Algorithm Greedy and Verification

input : An instance of a bi-objective problem for which routines *Restrict*$_0(f_1, f_2 \leq b)$ and *Restrict*$_0(f_2, f_1 \leq b)$ are available
output : An ε-kernel of size opt_ε

$f_1^{min} \leftarrow f_1(\textit{Restrict}_0(f_1, f_2 \leq 2^m))$; $f_2^{min} \leftarrow f_2(\textit{Restrict}_0(f_2, f_1 \leq 2^m))$;
$r^1 \leftarrow \textit{Restrict}_0(f_2, f_1 \leq 2^m)$;
$\overline{f_2}^1 \leftarrow (1+\varepsilon) r_2^1$;
$q^1 \leftarrow \textit{Restrict}_0(f_1, f_2 \leq \overline{f_2}^1)$;
$q^1 \leftarrow \textit{Restrict}_0(f_2, f_1 \leq q_1^1)$;
$\overline{f_1}^1 \leftarrow q_1^1/(1+\varepsilon)$;
$Q \leftarrow \{q^1\}$;
$i \leftarrow 1$;
`/* greedy phase */`
while $\overline{f_1}^i > f_1^{min}$ **do**
    $i \leftarrow i+1$;
    $r^i \leftarrow \textit{Restrict}_0(f_2, f_1 < \overline{f_1}^{i-1})$;
    $\overline{f_2}^i \leftarrow (1+\varepsilon) r_2^i$;
    $q^i \leftarrow \textit{Restrict}_0(f_1, f_2 \leq \overline{f_2}^i)$;
    $q^i \leftarrow \textit{Restrict}_0(f_2, f_1 \leq q_1^i)$;
    $\overline{f_1}^i \leftarrow q_1^i/(1+\varepsilon)$;
    $Q \leftarrow Q \cup \{q^i\}$;
`/* verification phase */`
$i \leftarrow i-1$;
while $q_2^{i+1}/(1+\varepsilon) > f_2^{min}$ **do**
    if $q_2^{i+1}/(1+\varepsilon) \leq q_2^i$ **then**
        $Q \leftarrow Q - \{q^i\}$;
        $q^i \leftarrow \textit{Restrict}_0(f_1, f_2 < q_2^{i+1}/(1+\varepsilon))$;
        $q^i \leftarrow \textit{Restrict}_0(f_2, f_1 \leq q_1^i)$;
        $Q \leftarrow Q \cup \{q^i\}$;
    $i \leftarrow i-1$;
return Q;

Algorithm analysis We show now that Algorithm 5.1 produces an ε-kernel of minimal size. Let $R = \{r^1, \ldots, r^s\}$ and $Q = \{q^1, \ldots, q^s\}$ be the set of feasible points produced by the algorithm. We first show some preliminary results regarding points in Q and R.

Proposition 15. *Set Q contains only nondominated points.*

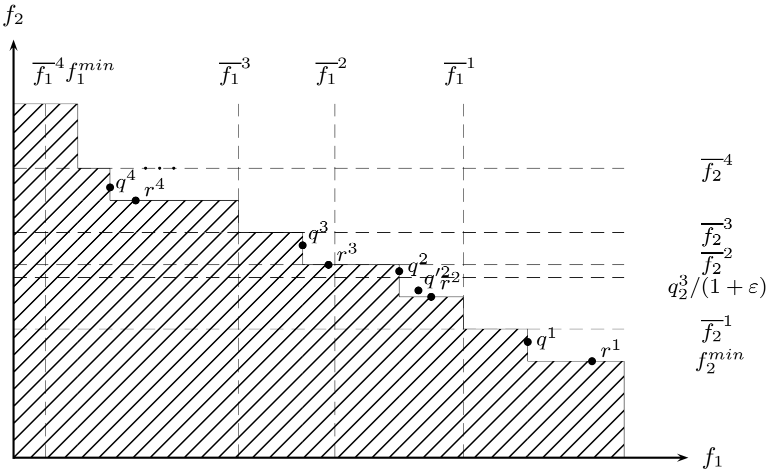

Figure 5.1: Illustration of Algorithm 5.1

Proof: Points $q^i \in Q$ are computed in two steps, both in the greedy phase (steps 13-14) and in the verification phase (steps 21-22). The first step returns a point q^i such that there exists no point $z \in Z$ such that $z_1 < q_1^i$ and $z_2 \leq q_i^2$. Thus, at this step, q^i is only guaranteed to be weakly nondominated since there may exist a point z such that $z_1 = q_1^i$ and $z_2 < q_2^i$. The second step rules out this possibility, ensuring that q^i is nondominated. □

Observe that the algorithm proposed in [20], which corresponds to the greedy phase, does not include this second step optimization. Therefore, the returned ε-Pareto set in [20] consists of weakly nondominated points.

Lemma 9. *During the verification phase, if a point q'^i replaces a point q^i in Q, we have (i) $q_2'^i < q_2^i$ and (ii) $q_1'^i > q_1^i$.*

Proof: (i) Point q'^i computed at steps 21-22 satisfies $q_2'^i < q_2^{i+1}/(1+\varepsilon) \leq q_2^i$.

(ii) Since points in Q are nondominated, including q^i and q'^i, (i) implies that $q_1'^i > q_1^i$. □

Lemma 10. *Any feasible point $z \in Z$ $(1+\varepsilon)$-dominates at most one point from R.*

Proof: Suppose, by contradiction, that z $(1+\varepsilon)$-dominates two points from R. Clearly, the most favorable situation is when these points are consecutive. Thus, let r^i and r^{i-1} be two consecutive points in R such that z $(1+\varepsilon)$-dominates them. Assuming that $z \preceq_\varepsilon r^{i-1}$, we have $z_2 \leq (1+\varepsilon)r_2^{i-1}$. By steps 13-14, this inequality implies that $q_1^{i-1} \leq z_1$, which implies $q_1^{i-1}/(1+\varepsilon) \leq z_1/(1+\varepsilon)$. From step 11, we have $r_1^i < q_1^{i-1}/(1+\varepsilon)$ and thus $r_1^i < z_1/(1+\varepsilon)$, contradicting $z \preceq_\varepsilon r^i$. □

Lemma 11. *The only point in R which is $(1+\varepsilon)$-dominated by q^i is r^i, for $i = 1, \ldots, s$.*

Proof : By Lemma 10, we just need to show that $q^i \preceq_\varepsilon r^i$, for $i = 1, \ldots, s$. We proceed by induction. By steps 13-14, the assertion is clear if q^i has not be modified. In particular, for q^s which is not modified, the assertion is true. Assuming now that $q^{i+1} \preceq_\varepsilon r^{i+1}$, we prove that $q^i \preceq_\varepsilon r^i$. The only case that could me problematic is when q^i has been modified during the second phase. By Lemma 10, we have not $(q^{i+1} \preceq_\varepsilon r^i)$, which means that $q_2^{i+1} > (1+\varepsilon)r_2^i$. Hence, by steps 21-22, we get $q_1^i \leq r_1^i$. Moreover, regarding the second criterion, since q^i computed during the first phase $(1+\varepsilon)$-dominates r^i, we have $q_2^i \leq (1+\varepsilon)r_2^i$. Considering that q^i has been modified, using Lemma 9-(i) we get $q_2^i < (1+\varepsilon)r_2^i$. Therefore, we get finally $q^i \preceq_\varepsilon r^i$. □

Proposition 16. *Set Q satisfies the ε-coverage condition.*

Proof : We show that the points in Q cover all the feasible points by partitioning the range of feasible values on f_1. More precisely, we show that:

(i) Point q^1 $(1+\varepsilon)$-dominates all the feasible points with an f_1 value greater than or equal to $q_1^1/(1+\varepsilon)$.

(ii) For each $i = 2, \ldots, s$, point q^i $(1+\varepsilon)$-dominates all the feasible points that have their f_1 value in the interval $\left[q_1^i/(1+\varepsilon), q_1^{i-1}/(1+\varepsilon)\right)$.

(iii) There is no feasible point with a f_1 value smaller than $q_1^s/(1+\varepsilon)$.

(i) Let z be a feasible point with $z_1 \geq q_1^1/(1+\varepsilon)$ and, by definition, $z_2 \geq f_2^{min}$. Point q^1 computed in steps 4-5 satisfies $q_2^1 \leq (1+\varepsilon)f_2^{min} \leq (1+\varepsilon)z_2$, which shows that q^1 $(1+\varepsilon)$-dominates z. If point q^1 is modified during the verification phase, using Lemma 9-(i) we also have $q_2^1 \leq (1+\varepsilon)z_2$.

(ii) Let z be a feasible point satisfying $q_1^i/(1+\varepsilon) \leq z_1 < q_1^{i-1}/(1+\varepsilon)$. In order to prove that z is $(1+\varepsilon)$-dominated by q^i, we have to show that $q_2^i \leq z_2(1+\varepsilon)$. We consider three cases.

- If points q^i and q^{i-1} have not been modified during the verification phase, then q^i, which is defined in step 13-14, verifies $q_2^i \leq (1+\varepsilon)r_2^i$. From step 11, we have $r_2^i \leq z_2$, which leads to $q_2^i \leq (1+\varepsilon)z_2$.
- If point q^i has been modified but not point q^{i-1}, then by Lemma 9-(i), the inequality is preserved.
- Finally if point q^{i-1} has been modified during the verification phase, step 21 ensures that there is no feasible point z' such that $z'_2 < q_2^i/(1+\varepsilon)$ and $z'_1 < q_1^{i-1}$. Since $z_1 < q_1^{i-1}/(1+\varepsilon)$, it follows that $z_1 < q_1^{i-1}$ and thus $z_2 \geq q_2^i/(1+\varepsilon)$.

(iii) Point q^s, which is not modified in the verification phase, is the last point obtained in the while loop 9-16. By step 15 and condition in step 9, we have $q_1^s/(1+\varepsilon) \leq f_1^{min}$. □

Proposition 17. *Set Q satisfies the ε-stability condition.*

Proof : We just need to show that ε-stability holds for consecutive points in Q, that is for all $i = 2, \ldots, s$ we have (i) not $(q^{i-1} \preceq_\varepsilon q^i)$ and (ii) not $(q^i \preceq_\varepsilon q^{i-1})$.

(i) From Lemma 11, we have not$(q^{i-1} \preceq_\varepsilon r^i)$. This occurs because we have on the first criterion $q_1^{i-1} > (1+\varepsilon) r_1^i$. Since we have $r_1^i \geq q_1^i$, we get $q_1^{i-1} > (1+\varepsilon) q_1^i$, that is not$(q^{i-1} \preceq_\varepsilon q^i)$.

(ii) Test 19-23 ensures that $q_2^{i-1} < q_2^i/(1+\varepsilon)$. □

Theorem 11. *For any $\varepsilon > 0$, Algorithm 5.1 computes an ε-kernel of minimal size opt_ε that contains only nondominated points using $O(opt_\varepsilon)$ routine calls to Restrict$_0$.*

Proof : Q is an ε-kernel containing only nondominated points from Propositions 15, 16, and 17. Moreover, set Q has minimal size opt_ε since, from Lemma 10, at least $|R|$ points are required for any ε-Pareto set, whereas Algorithm 5.1 returns a set Q with $|Q|=|R|$.

Since the algorithm uses at most $3\,|Q| + 2\,|Q| = 5\,|Q|$ times the routine *Restrict*$_0$, the number of routine calls is bounded by $5opt_\varepsilon$. □

Since opt_ε is polynomially bounded in the input size and $1/\varepsilon$ [66], we have the following corollary.

Corollary 5. *For any $\varepsilon > 0$, if routines Restrict$_0$ are computable in polynomial time for both objectives, then we can determine an ε-kernel of minimal size that contains only nondominated points in polynomial time in the size of the input and $1/\varepsilon$.*

5.2.2.2 Dual problem

We show that the minimal ratio $1+\varepsilon^*$ is approximable within any factor $1+\theta$ in polynomial time in the input size and $1/\theta$.

Theorem 12. *Let k be a nonnegative integer and let $1+\varepsilon^*$ be the minimal ratio for which an ε^*-kernel of size at most k exists. For any rational $\theta > 0$, we can determine an ε-kernel of size at most k with $1+\varepsilon \leq (1+\varepsilon^*)(1+\theta)$. This can be done using $O(k \log(m/\theta))$ subroutine calls to Restrict$_0$.*

Proof : We first apply Algorithm 5.1 with $\varepsilon = \theta$. If the returned ε-kernel has size at most k, then the required condition is satisfied. Otherwise, the minimal ratio $1+\varepsilon^*$ belongs to the range $[1+\theta, 2^{2m}]$, where the upper bound corresponds to the extreme situation with $k = 1$ and $Z = \{z^1 = (2^m, 1/2^m), z^2 = (1/2^m, 2^m)\}$. Let $1+\varepsilon_i = (1+\theta)^i$ be the candidate ratios for $i = 1, \ldots, \lceil 2m/\log(1+\theta) \rceil$. We perform a binary search on i values. At each step we call Algorithm 5.1 in order to obtain an ε_i-kernel of minimal size. If this size is greater than k then we continue the search in the right part, otherwise in the left part. Observe that, at each step, the search is between the indices i_ℓ and i_r such that the size of the smallest i_ℓ-kernel is more than k and the size of the smallest i_r-kernel is at most k. Thus, $1+\varepsilon_\ell < 1+\varepsilon^* \leq 1+\varepsilon_r$. The search is stopped when $i_r = i_\ell + 1$, i.e. when $1+\varepsilon_r = (1+\varepsilon_\ell)(1+\theta)$. Then the current ε_{i_r}-kernel is of size at most k and such that $1+\varepsilon_{i_r} = (1+\varepsilon_{i_\ell})(1+\theta) \leq (1+\varepsilon^*)(1+\theta)$.

The number of calls to Algorithm 5.1 is $O(\log(2m/\log(1+\theta))) \approx O(\log(m/\theta))$. Since we can stop each call to Algorithm 5.1 when it tries to compute a $(k+1)^{th}$ point, each such call uses $O(k)$ calls to *Restrict*$_0$. Thus, the total running time is $O(k\log(m/\theta))$ *Restrict*$_0$ calls. □

Corollary 6. *Let k be a nonnegative integer and let $1+\varepsilon^*$ be the minimal ratio for which an ε^*-kernel of size at most k exists. If routines Restrict$_0$ are computable in polynomial time for both objectives, then we can determine an ε-kernel of size at most k with $1+\varepsilon \leq (1+\varepsilon^*)(1+\theta)$ in polynomial time in the size of the input and $1/\theta$.*

5.2.3 Algorithms for $(\varepsilon, \varepsilon')$-kernels using approximate routines

In this section, we provide algorithms for the primal problem (section 5.2.3.1) and the dual problem (section 5.2.3.2) considering that a routine *Restrict*$_\delta$ is available for both objectives.

If a routine *Restrict*$_\delta$ with $\delta > 0$ is available for at least one objective, Diakonikolas and Yannakakis [20] showed that no generic algorithm is able to compute an ε-Pareto set of minimal size but it is possible to compute an ε-Pareto set of size $2opt_\varepsilon$ in polynomial time. Then, from Theorem 8, it follows that using routines *Restrict*$_\delta$ with $\delta > 0$ we can only compute an ε-kernel of size between $2opt_\varepsilon$ and $3opt_\varepsilon$ in polynomial time. In fact, adapting an argument from [20] which shows that there is no polynomial time generic algorithm based on routines *Restrict*$_\delta$ with $\delta > 0$ that approximates the size of a smallest ε-Pareto set to a factor better than 2, we can show the following proposition.

Proposition 18. *Consider the class of bi-objective problems that possess a fully polynomial time routine Restrict$_\delta$ with $\delta > 0$ for both objectives. Then, for any $\varepsilon > 0$, there is no polynomial time generic algorithm using Restrict$_\delta$ that computes an ε-kernel.*

Proof: Consider the following set of feasible points $Z = \{z, z^1, z^2, z^3, z^4\}$ (see Figure 5.2) where: $z = (z_1, z_2)$, with $z_1, z_2 \geq 1/\varepsilon$, $z^1 = ((z_1{+}1)(1{+}\varepsilon), z_2/(1{+}\varepsilon)^2)$, $z^2 = (z_1+1, z_2)$, $z^3 = (z_1, z_2+1)$ and $z^4 = (z_1/(1+\varepsilon)^2, (z_2+1)(1+\varepsilon))$. Then, note that each point of $\{z, z^2, z^3\}$ $(1+\varepsilon)$-dominates only these three points, and that z^1 $(1+\varepsilon)$-dominates z^2 and z^4 $(1+\varepsilon)$-dominates z^3. Then, there are exactly

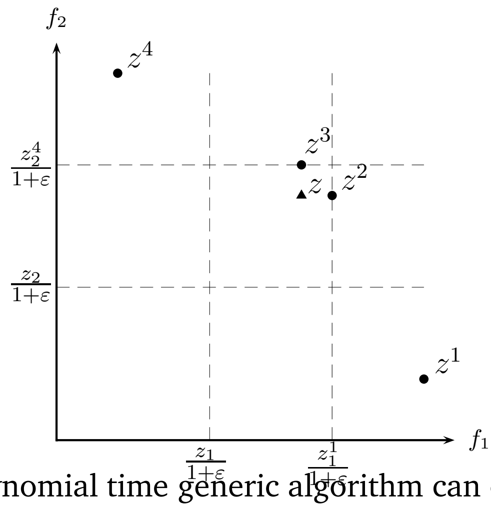

Figure 5.2: No polynomial time generic algorithm can compute an ε-kernel.

three minimal ε-Pareto sets: $P_\varepsilon = \{z, z^1, z^4\}$, $P'_\varepsilon = \{z^2, z^1, z^4\}$, $P''_\varepsilon = \{z^3, z^1, z^4\}$ and only P_ε is an ε-kernel.

We show that a generic algorithm using Restrict$_\delta$ is guaranteed to return the ε-kernel only if $1/\delta$ is exponential in the size of the input. Let $z_1 = z_2 = M$, where M is an integer value exponential in the size of the input and $1/\varepsilon$. Assume that we use the routine *Restrict*$_\delta(f_1, f_2 \leq C)$ with $C \in [M, M+1)$. Then, the routine can return the point z^2 instead of z as long as $\delta \geq 1/M$. Symmetrically, if we use the routine *Restrict*$_\delta(f_2, f_1 \leq C)$ with $C \in [M, M+1)$ we can obtain z^3 instead of z. But, since we want a polynomial time algorithm, $1/\delta$ has to be polynomial in $\log M$. Therefore, a polynomial time generic algorithm cannot guarantee to compute the unique ε-kernel which contains the point z. □

This previous proposition shows that it is not possible to obtain a similar result to Corollary 5 when only approximate routines Restrict$_\delta$ with $\delta > 0$ are available. However, if we relax the stability condition to a value $\varepsilon' < \varepsilon$, we show that we can achieve the computation in polynomial time. Therefore, in the following, we assume that $\varepsilon' < \varepsilon$.

5.2.3.1 Primal problem

We propose an algorithm that produces an $(\varepsilon, \varepsilon')$-kernel of size at most twice the size of a minimal ε-Pareto set.

Algorithm description The algorithm proceeds in two phases. The first phase (greedy phase) corresponds to the algorithm presented in [20] which returns a 2-approximation algorithm for finding an ε-Pareto set of minimal size. The second phase (verification phase) is basically the same as Algorithm 5.1 but using ε' instead of ε.

Algorithm 5.2: Algorithm Greedy and Verification Extended

input : An instance of a bi-objective problem for which routines *Restrict*$_\delta(f_1, f_2 \leq b)$ and *Restrict*$_\delta(f_2, f_1 \leq b)$ are available

output : An $(\varepsilon, \varepsilon')$-kernel of size at most $2opt_\varepsilon$

$f_1^{min} \leftarrow f_1(\textit{DualRestrict}_\delta(f_1, f_2 \leq 2^m))$;
$f_2^{min} \leftarrow f_2(\textit{DualRestrict}_\delta(f_2, f_1 \leq 2^m))$;
$r^1 \leftarrow \textit{Restrict}_\delta(f_2, f_1 \leq 2^m)$;
$\overline{f_2}^1 \leftarrow \frac{1+\varepsilon}{(1+\delta)^2} r_2^1$;
$q^1 \leftarrow \textit{DualRestrict}_\delta(f_1, f_2 \leq \overline{f_2}^1)$;
$\overline{f_1}^1 \leftarrow q_1^1/(1+\varepsilon)$;
$Q \leftarrow \{q^1\}$;
$i \leftarrow 1$;
`/* greedy phase */`
while $\overline{f_1}^i > f_1^{min}$ **do**
  $i \leftarrow i + 1$;
  $r^i \leftarrow \textit{Restrict}_\delta(f_2, f_1 < \overline{f_1}^{i-1})$;
  $\overline{f_2}^i \leftarrow \frac{1+\varepsilon}{1+\delta} max\{\overline{f_2}^{i-1}, r_2^i/(1+\delta)\}$;
  $q^i \leftarrow \textit{DualRestrict}_\delta(f_1, f_2 \leq \overline{f_2}^i)$;
  $\overline{f_1}^i \leftarrow q_1^i/(1+\varepsilon)$;
  $Q \leftarrow Q \cup \{q^i\}$;
`/* verification phase */`
$s \leftarrow i$, $i \leftarrow i - 1$;
while $q_2^{i+1}/(1+\varepsilon) > f_2^{min}$ **do**
  if $q_2^{i+1}/(1+\varepsilon') \leq q_2^i$ **then**
    $Q \leftarrow Q - \{q^i\}$;
    $q^i \leftarrow \textit{Restrict}_\delta(f_1, f_2 < q_2^{i+1}/(1+\varepsilon))$;
    if $q_1^i < q_1^{i-1}/(1+\varepsilon')$ **then**
      $Q \leftarrow Q \cup \{q^i\}$;
  else
    reindex $\{q^{i+1}, \ldots, q^s\}$ by $\{q^i, \ldots, q^{s-1}\}$;
    $s \leftarrow s - 1$;
  $i \leftarrow i - 1$;
return Q;

The algorithm is shown to produce an $(\varepsilon, \varepsilon')$-kernel when $\delta < (1+\varepsilon)/(1+\varepsilon') - 1$ (Propositions 19 and 20) and the size of this $(\varepsilon, \varepsilon')$-kernel is proved to be at most $2opt_\varepsilon$ if $\delta \leq \sqrt[3]{1+\varepsilon} - 1$ (Theorem 13). Therefore, we assume that $\delta < \min\{(1+\varepsilon)/(1+\varepsilon') - 1, \sqrt[3]{1+\varepsilon} - 1\}$.

A formal description of this algorithm is given in Algorithm 5.2.

Note that, when a routine *Restrict*$_\delta$ is available only for one objective, we have another version of this algorithm that requires $\delta < \min\{\sqrt{(1+\varepsilon)/(1+\varepsilon')}-1, \sqrt[3]{1+\varepsilon}-1\}$ by replacing step 19 by $q^i \leftarrow$ *DualRestrict*$_\delta(f_1, f_2 < q_2^{i+1}/(1+\varepsilon')(1+\delta)^2)$;.

Algorithm analysis We show now that Algorithm 5.2 produces an $(\varepsilon, \varepsilon')$-kernel whose size is at most $2opt_\varepsilon$. Let $Q = \{q^1, \ldots, q^s\}$ be the set of feasible points produced by the algorithm.

Observe that in steps 23-24 Algorithm 5.2 does discards points that are proved unnecessary in the next result. The returned set may thus be of smaller cardinality than the ε-Pareto set obtained at the end of the greedy phase.

Lemma 12. *During the verification step, if a point q'^i, replacing a point q^i, is such that $q_1'^i \geq q_1^{i-1}/(1+\varepsilon')$, then point q'^i is unnecessary.*

Proof: Point q'^i, with $q_1'^i \geq q_1^{i-1}/(1+\varepsilon')$, is computed in step 19 using *Restrict*$_\delta(f_1, f_2 < q_2^{i+1}/(1+\varepsilon))$ where $\delta < (1+\varepsilon)/(1+\varepsilon') - 1$. This implies that any feasible point z satisfying $z_2 < q_2^{i+1}/(1+\varepsilon)$ is such that $z_1 \geq q_1'^i/(1+\delta) > q_1'^i(1+\varepsilon')/(1+\varepsilon) \geq q_1^{i-1}/(1+\varepsilon)$. Therefore, there is no feasible point z such that $z_1 < q_1^{i-1}/(1+\varepsilon)$ and $z_2 < q_2^{i+1}/(1+\varepsilon)$. Thus, a point that is $(1+\varepsilon)$-dominated by point q'^i is $(1+\varepsilon)$-dominated by point q^{i-1} or q^{i+1}. □

In the following, for proving the correctness of our algorithm, the case of points which are not included (steps 23-24) can be ignored. Indeed, when this happens, the consequence of reindexing at step 23 is that points q^{i+1} and q^{i-1} become respectively points q^{i+1} and q^i at the next iteration, without any impact on the ε-coverage condition as shown by Lemma 12.

Lemma 13. *During the verification step, if a point q'^i replaces a point q^i in Q, we have (i) $q_2'^i < q_2^i$ and (ii) $q_1'^i \geq q_1^i$.*

Proof: (i) Point q'^i computed at step 19 satisfies $q_2'^i < q_2^{i+1}/(1+\varepsilon) < q_2^{i+1}/(1+\varepsilon') \leq q_2^i$.

(ii) Remark that point q^i was computed in step 12 using routine *DualRestrict*$_\delta$ during the greedy phase. It follows that there is no feasible point z such that $z_1 < q_1^i$ and $z_2 < q_2^i/(1+\delta)$. Since $\varepsilon' < (1+\varepsilon)/(1+\delta) - 1$, point q'^i is computed in step 19 such that $q_2'^i < q_2^{i+1}/(1+\varepsilon) < q_2^{i+1}/(1+\varepsilon')(1+\delta) \leq q_2^i/(1+\delta)$. It follows that $q_1'^i \geq q_1^i$. □

Proposition 19. *Set Q satisfies the ε-coverage condition.*

Proof: We show that the points in Q cover all the feasible points by partitioning the range of feasible values on f_1. More precisely, we show that:

(i) Point q^1 $(1+\varepsilon)$-dominates all the feasible points with an f_1 value greater than or equal to $q_1^1/(1+\varepsilon)$.

(ii) For each $i = 2, \ldots, s$, the point q^i $(1+\varepsilon)$-dominates all the feasible points that have their f_1 value in the interval $\left[q_1^i/(1+\varepsilon), q_1^{i-1}/(1+\varepsilon)\right)$.

(iii) There is no feasible point with a f_1 value smaller than $q_1^s/(1+\varepsilon)$.

(i) Let z be a feasible point with $z_1 \geq q_1^1/(1+\varepsilon)$ and, by definition, $z_2 \geq f_2^{min}$. Point q^1 computed in step 4 satisfies $q_2^1 \leq (1+\varepsilon)f_2^{min} \leq (1+\varepsilon)z_2$, which shows that q^1 $(1+\varepsilon)$-dominates z. If point q^1 is modified during the verification phase, using Lemma 13-(i) we also have $z_2 \geq q_2^1/(1+\varepsilon)$.

(ii) Let z be a feasible point satisfying $q_1^i/(1+\varepsilon) \leq z_1 < q_1^{i-1}/(1+\varepsilon)$. In order to prove that z is $(1+\varepsilon)$-dominated by q^i, we have to show that $q_2^i \leq (1+\varepsilon)z_2$. We consider three cases.

- If points q^i and q^{i-1} have not been modified during the verification phase, then q^i, which is defined in step 12, verifies $q_2^i \leq (1+\varepsilon)max\{\overline{f_2}^{i-1}, r_2^i/(1+\delta)\}$. From step 10 we have $z_2 \geq r_2^i/(1+\delta)$ and from step 12 for $i-1$ we have $z_2 \geq \overline{f_2}^{i-1}$. Thus $max\{\overline{f_2}^{i-1}, r_2^i/(1+\delta)\} \leq z_2$ which leads to $q_2^i \leq (1+\varepsilon)z_2$.
- If point q^i has been modified but not point q^{i-1}, then by Lemma 13-(i), the inequality is preserved.
- Finally if point q^{i-1} has been modified during the verification phase, step 19 ensures that there is no feasible point z' such that $z'_2 < q_2^i/(1+\varepsilon)$ and $z'_1 < q_1^{i-1}/(1+\delta)$. Since $z_1 < q_1^{i-1}/(1+\varepsilon)$ it follows that $z_1 < q_1^{i-1}/(1+\delta)$ and thus $z_2 \geq q_2^i/(1+\varepsilon)$.

(iii) Point q^s, which is not modified during the verification phase, is the last point obtained in the while loop 8-14. By step 13 and condition in step 8, we have $q_1^s/(1+\varepsilon) \leq f_1^{min}$. □

Proposition 20. *Set Q satisfies the ε'-stability condition.*

Proof : We just need to show that ε'-stability holds for consecutive points in Q, that is for all $i = 2, \ldots, s$ we have (i) not $(q^{i-1} \preceq_{\varepsilon'} q^i)$ and (ii) not $(q^i \preceq_{\varepsilon'} q^{i-1})$.

(i) We consider three cases.

- If points q^i and q^{i-1} have not been modified during the verification phase, then point r^i, computed in step 10, is such that $r_1^i < q_1^{i-1}/(1+\varepsilon)$. Moreover since point q^i, computed in step 12, is such that $q_1^i \leq r_1^i$, we get $q_1^i < q_1^{i-1}/(1+\varepsilon) < q_1^{i-1}/(1+\varepsilon')$, that is not $(q^{i-1} \preceq_{\varepsilon'} q^i)$.
- If point q^i is modified and point q^{i-1} is not modified, then since q^i is added to Q in step 21, it satisfies $q_1^i < q_1^{i-1}/(1+\varepsilon')$, that is not $(q^{i-1} \preceq_{\varepsilon'} q^i)$.
- The final case is when point q^{i-1} changes during the verification phase and is replaced by a point q'^{i-1}. Then, according to Lemma 13-(ii) the inequality is preserved.

(ii) Test 17-24 and the definition of point q^{i-1} at step 19 ensures that $q_2^{i-1} < q_2^i/(1+\varepsilon) < q_2^i/(1+\varepsilon')$. □

Lemma 14. *Any point $z \in Z$ $(1+\varepsilon)$-dominates at most two points from R.*

Proof : Suppose, by contradiction, that z $(1+\varepsilon)$-dominates three points from R. Clearly, the most favorable situation is when these points are consecutive. Thus, let r^i, r^{i-1}, and r^{i-2} be three consecutive points in R such that z $(1+\varepsilon)$-dominates them. Assuming that $z \preceq_\varepsilon r^{i-2}$, we have $z_2 \leq (1+\varepsilon)r_2^{i-2}$. By step 11, for $i-2$ and $i-1$, we get $\overline{f_2}^{i-2} \geq \frac{1+\varepsilon}{(1+\delta)^2} r_2^{i-2}$ and $\overline{f_2}^{i-1} \geq \frac{1+\varepsilon}{1+\delta} \overline{f_2}^{i-2}$ and thus $\overline{f_2}^{i-2} \geq \frac{(1+\varepsilon)^2}{(1+\delta)^3} r_2^{i-2}$. Since $(1+\delta)^3 < 1+\varepsilon$, we have $z_2 \leq \overline{f_2}^{i-1}$. From this last inequality, by step 12, for $i-1$, we have $q_1^{i-1} \leq z_1$, which implies $q_1^{i-1}/(1+\varepsilon) \leq z_1/(1+\varepsilon)$. From step 10, we have $r_1^i < q_1^{i-1}/(1+\varepsilon)$ and thus $r_1^i < z_1/(1+\varepsilon)$, contradicting $z \preceq_\varepsilon r^i$. □

Theorem 13. *For any $\varepsilon, \varepsilon'$ such that $\varepsilon > \varepsilon' > 0$, Algorithm 5.2 computes an $(\varepsilon, \varepsilon')$-kernel of size less than or equal to $2opt_\varepsilon$ using $O(opt_\varepsilon)$ routine calls to Restrict$_\delta$ or DualRestrict$_\delta$, where $\delta < \min\{(1+\varepsilon)/(1+\varepsilon') - 1, \sqrt[3]{1+\varepsilon} - 1\}$.*

Proof : Q is an $(\varepsilon, \varepsilon')$-kernel from Propositions 19 and 20. Moreover, set Q has a size less than or equal to $2opt_\varepsilon$ since, from Lemma 14, at least $\lceil |R|/2 \rceil$ points are required for any ε-Pareto set, whereas Algorithm 5.2 returns a set Q with $|Q| \leq |R|$.

Since the algorithm uses at most $2\,|Q| + |Q| = 3\,|Q|$ times the routines *Restrict$_\delta$* or *DualRestrict$_\delta$*, the number of routine calls is bounded by $3opt_\varepsilon$. □

Since opt_ε is polynomially bounded in the input size and $1/\varepsilon$ [66], we have the following corollary.

Corollary 7. *For any $\varepsilon, \varepsilon'$ such that $\varepsilon > \varepsilon' > 0$, if routines Restrict$_\delta$ and DualRestrict$_\delta$ with $\delta > 0$ are computable in (fully) polynomial time for both objectives, then we can determine an $(\varepsilon, \varepsilon')$-kernel of size less than or equal to $2opt_\varepsilon$ in (fully) polynomial time.*

We recall that it is not possible to produce an ε-Pareto set of size opt_ε in polynomial time using routines *Restrict$_\delta$* [20]. Nevertheless, Vassilvitskii and Yannakakis showed in [80] that it is possible to produce in polynomial time an ε-Pareto set of size bounded $opt_{\hat{\varepsilon}}$ for any $\hat{\varepsilon} < \varepsilon$. In the following we present a similar result for $(\varepsilon, \varepsilon')$-kernels. More precisely, we show that Algorithm 5.2 used with $\delta < \min\{\sqrt{(1+\varepsilon)/(1+\hat{\varepsilon})} - 1, (1+\varepsilon)/(1+\varepsilon') - 1\}$ computes an $(\varepsilon, \varepsilon')$-kernel of size bounded by $opt_{\hat{\varepsilon}}$, for any $\hat{\varepsilon} < \varepsilon$ and $\varepsilon' < \varepsilon$. Let Q be the set of feasible points produced by the algorithm.

Since $\delta < (1+\varepsilon)/(1+\varepsilon') - 1$, Q is an $(\varepsilon, \varepsilon')$-kernel from Propositions 19 and 20. Therefore, we only need to show that set Q has a size less than or equal to $opt_{\hat{\varepsilon}}$.

Proposition 21. *When $\delta \le \sqrt{(1+\varepsilon)/(1+\hat{\varepsilon})} - 1$, Algorithm 5.2 returns a set Q with $|Q| \le opt_{\hat{\varepsilon}}$.*

Proof: Let $P^*_{\hat{\varepsilon}} = \{p^{*1}, \dots, p^{*k}\}$ be an $\hat{\varepsilon}$-Pareto set of minimal size, where points p^{*i} for $i = 1, \dots, k$ are in increasing order of their coordinates on f_2 and decreasing order of their coordinates on f_1. Let $\tilde{Q} = \{\tilde{q}^1, \dots, \tilde{q}^r\}$ be the set of points returned by the greedy phase of Algorithm 5.2. We have $|\tilde{Q}| \ge |Q|$ due to the possible omission of points in steps 23-24 of the verification step. We show now that $|\tilde{Q}| \le |P^*_{\hat{\varepsilon}}|$. For this purpose, we show by induction on i that for each point $\tilde{q}^i$ in $\tilde{Q}$ there exists a point p^{*i} in $P^*_{\hat{\varepsilon}}$ such that $\tilde{q}^i_1 \le p^{*i}_1$.

Initialization ($i = 1$). The fact that P^*_{ε} contains at least one point is trivially true. We need to show that $\tilde{q}^1_1 \le p^{*1}_1$. Since point $\tilde{q}^1$ is computed in step 4 using *DualRestrict*$_\delta(f_1, f_2 \le \overline{f_2}^1)$, to show the statement it suffices to prove that $\overline{f_2}^1 \ge p^{*1}_2$. Since $P^*_{\hat{\varepsilon}}$ is an $\hat{\varepsilon}$-Pareto set where its points p^{*j} for $j = 1, \dots, k$ are in increasing order of their coordinates on f_2, it follows that point p^{*1} must $(1+\hat{\varepsilon})$-dominates f_2^{min} and so $p^{*1}_2 \le (1+\hat{\varepsilon}) f_2^{min}$. Since $\delta \le \sqrt{(1+\varepsilon)/(1+\hat{\varepsilon})} - 1$, it follows that $p^{*1}_2 \le \frac{1+\varepsilon}{(1+\delta)^2} f_2^{min}$. From step 2 we have $r^1_2 \ge f_2^{min}$ and from step 3 we have $\overline{f_2}^1 = \frac{1+\varepsilon}{(1+\delta)^2} r^1_2$, thus it follows that $\overline{f_2}^1 \ge p^{*1}_2$.

Induction step. Assume the result is true until index $i-1$, we prove it for index i. By the termination condition of the greedy phase of Algorithm 5.2 (step 8), we have $\tilde{q}^{i-1}_1 > (1+\varepsilon) f_1^{min}$ and by the induction hypothesis that $p^{*i-1}_1 \ge \tilde{q}^{i-1}_1$, it follows that $p^{*i-1}_1 > (1+\varepsilon) f_1^{min}$. Thus, point p^{*i-1} does not $(1+\varepsilon)$-dominate the feasible points that have a first coordinate equal to f_1^{min}, and so $P^*_{\hat{\varepsilon}}$ must contain another point p^{*i}. Since point $\tilde{q}^i$ is computed in step 12 using *DualRestrict*$_\delta(f_1, f_2 \le \overline{f_2}^i)$, to show the statement it suffices to prove that $\overline{f_2}^i \ge p^{*i}_2$. Since P^*_{ε} is an $\hat{\varepsilon}$-Pareto set where its points p^{*j} for $j = 1, \dots, k$ are in increasing order of their coordinates on f_2, it follows that point p^{*i} must $(1+\hat{\varepsilon})$-dominates point r^i and so $p^{*i}_2 \le (1+\hat{\varepsilon}) r^i_2$. Since $\delta \le \sqrt{(1+\varepsilon)/(1+\hat{\varepsilon})} - 1$, it follows that $p^{*i}_2 \le \frac{1+\varepsilon}{(1+\delta)^2} r^i_2$. From step 11 we have $\overline{f_2}^i \ge \frac{1+\varepsilon}{(1+\delta)^2} r^i_2$, thus it follows that $\overline{f_2}^i \ge p^{*i}_2$. □

Theorem 14. *For any $\hat{\varepsilon}, \varepsilon, \varepsilon'$ such that $\varepsilon > \hat{\varepsilon} > 0$ and $\varepsilon > \varepsilon' > 0$, Algorithm 5.2 computes an $(\varepsilon, \varepsilon')$-kernel of size less than or equal to $opt_{\hat{\varepsilon}}$ using $O(opt_{\hat{\varepsilon}})$ routine calls to Restrict$_\delta$ or DualRestrict$_\delta$, with $\delta < \min\{\sqrt{(1+\varepsilon)/(1+\hat{\varepsilon})} - 1, (1+\varepsilon)/(1+\varepsilon') - 1\}$.*

Proof: Set Q returned by Algorithm 5.2 is an $(\varepsilon, \varepsilon')$-kernel since Propositions 19 and 20 hold. Moreover, the size of Q is less than or equal to $opt_{\hat{\varepsilon}}$ by Proposition 21. Since the algorithm uses $3|Q|$ times the routine *Restrict*$_\delta$ or *DualRestrict*$_\delta$, the number of routine calls is bounded by $3opt_{\hat{\varepsilon}}$. □

Corollary 8. *For any $\hat{\varepsilon}, \varepsilon, \varepsilon'$ such that $\varepsilon > \hat{\varepsilon} > 0$ and $\varepsilon > \varepsilon' > 0$, if routines Restrict$_\delta$ and DualRestrict$_\delta$ with $\delta > 0$ are computable in (fully) polynomial time for both objectives, then we can determine an $(\varepsilon, \varepsilon')$-kernel of size less than or equal to $opt_{\hat{\varepsilon}}$ in (fully) polynomial time.*

5.2.3.2 Dual problem

We show that the minimal ratio $1+\varepsilon^*$ is approximable within any factor $1+\theta$ in polynomial time in the input size and $1/\theta$.

Theorem 15. *Let k be a nonnegative integer and let $1+\varepsilon^*$ be the minimal ratio for which an ε^*-kernel of size at most k exists. For any rational $\theta > 0$, we can determine an $(\varepsilon, \varepsilon')$-kernel with $1 + \varepsilon \leq (1 + \varepsilon^*)(1 + \theta)$, for all $\varepsilon' < \varepsilon$, of size at most k using $O(k \log(m/\theta))$ routine calls to Restrict$_\delta$ or DualRestrict$_\delta$.*

Proof: We first apply Algorithm 5.2 with $\varepsilon = \theta$, $\varepsilon' < \varepsilon$, and $\delta < \min\{\sqrt[4]{1+\theta} - 1, (1+\varepsilon)/(1+\varepsilon') - 1\}$, where $\delta < \sqrt[4]{1+\theta} - 1$ results from considering $1 + \hat{\varepsilon} = \sqrt{1+\theta}$ in Theorem 14. If the returned $(\varepsilon, \varepsilon')$-kernel has size at most k, then the required condition is satisfied. Otherwise, from Theorem 14, the minimal ratio $1 + \varepsilon^*$ belongs to the range $[\sqrt{1+\theta}, 2^{2m}]$. Let $1 + \varepsilon_i = (\sqrt{1+\theta})^i$ be the candidate ratios for $i = 2, \ldots, \lceil 4m/\log(1+\theta) \rceil$ and let $1 + \hat{\varepsilon}_i = (1+\varepsilon_i)/\sqrt{1+\theta}$. We perform a binary search on i values. At each step we call Algorithm 5.2 with $\delta < \min\{\sqrt[4]{1+\theta} - 1, (1+\varepsilon_i)/(1+\varepsilon'_i) - 1\}$, where ε'_i is an arbitrary number such that $\varepsilon'_i < \varepsilon_i$, in order to obtain an $(\varepsilon_i, \varepsilon'_i)$-kernel of size at most $opt_{\hat{\varepsilon}_i}$ (see Theorem 14). If this size is greater than k then we continue the search in the right part, otherwise in the left part. Observe that, at each step, the search is between the indices i_ℓ and i_r such that the size of the $(\varepsilon_{i_\ell}, \varepsilon'_{i_\ell})$-kernel is more than k and the size of the $(\varepsilon_{i_r}, \varepsilon'_{i_r})$-kernel is at most k. Thus, $(1+\varepsilon_{i_\ell})/\sqrt{1+\theta} < 1 + \varepsilon^* \leq 1 + \varepsilon_{i_r}$. The search is stopped when $i_r = i_\ell + 1$, i.e. when $1 + \varepsilon_{i_r} = (1+\varepsilon_{i_\ell})\sqrt{1+\theta}$. Then, the current $(\varepsilon_{i_r}, \varepsilon'_{i_r})$-kernel is of size at most $opt_{\hat{\varepsilon}_{i_r}} \leq k$ and such that $1 + \varepsilon_{i_r} = (1+\varepsilon_{i_\ell})\sqrt{1+\theta} \leq (1+\varepsilon^*)(1+\theta)$.

The number of calls to Algorithm 5.2 is $O(\log(4m/\log(1+\theta))) \approx O(\log(m/\theta))$. Since we can stop each call to Algorithm 5.2 when it tries to compute a $(k+1)^{th}$ point, each such call uses $O(k)$ calls to *Restrict*$_\delta$ or *DualRestrict*$_\delta$. Thus, the total running time is $O(k \log(m/\theta))$ *Restrict*$_\delta$ or *DualRestrict*$_\delta$ calls. □

Corollary 9. *Let k be a nonnegative integer and let $1+\varepsilon^*$ be the minimal ratio for which an ε^*-kernel of size at most k exists. If routines Restrict$_\delta$ and DualRestrict$_\delta$ with $\delta > 0$ are computable in (fully) polynomial time for both objectives, for any rational $\theta > 0$, we can determine an $(\varepsilon, \varepsilon')$-kernel with $1 + \varepsilon \leq (1+\varepsilon^*)(1+\theta)$, for all $\varepsilon' < \varepsilon$, of size at most k in (fully) polynomial time.*

5.3 More than two objectives

For more than two objectives, the notion of ε-kernel is not really operational since an ε-kernel does not always exist.

Proposition 22. *For $p \geq 3$ objectives, an ε-kernel may not exist.*

Proof: Let $p = 3$ and z^1, z^2, and z^3 be three points with the following coordinates: $z^1 = (a(1+\varepsilon), b/(1+\varepsilon), c)$, $z^2 = (a, b(1+\varepsilon), c/(1+\varepsilon))$, $z^3 = (a/(1+\varepsilon), b, c(1+\varepsilon))$ where a, b, and c are three nonnegative rational numbers.

Clearly z^1 $(1+\varepsilon)$-dominates z^2, z^2 $(1+\varepsilon)$-dominates z^3 and z^3 $(1+\varepsilon)$-dominates z^1. Since any ε-kernel must satisfy the ε-stability condition, it follows that an ε-kernel must contain at most one point. Moreover, no point $(1+\varepsilon)$-dominates the two others. Since any ε-kernel must satisfy the ε-coverage condition, it follows that an ε-kernel must contain at least two points. This is clearly impossible. □

Moreover, even if an ε-kernel exists, we have no guarantee on its size like Theorems 8, 9, and 10 for the bi-objective case. On the opposite, we can show that a smallest ε-kernel may have a very large size compared with opt_ε.

Proposition 23. *For $p \geq 3$ objectives, the size of a smallest ε-kernel, when it exists, can be greater than $k.opt_\varepsilon$ for any integer k.*

Proof: Let $p = 3$ and z^1, z^2, and z^3 be defined as in the proof of Proposition 22. Let $z = (z^2_1, z^3_2, z^1_3) = (a, b, c)$. Fix any rational $\hat{\varepsilon} > \varepsilon$ and consider the $3k$ points $z^{1j} = (z^1_1(1+\hat{\varepsilon})^j, z^1_2/(1+\varepsilon), z^1_3(1+\hat{\varepsilon})^{k-j})$, $z^{2j} = (z^2_1(1+\hat{\varepsilon})^{k-j}, z^2_2(1+\hat{\varepsilon})^j, z^2_3/(1+\varepsilon))$ and $z^{3j} = (z^3_1/(1+\varepsilon), z^3_2(1+\hat{\varepsilon})^{k-j}, z^3_3(1+\hat{\varepsilon})^j)$ for $j = 1, \ldots, k$.

For this instance, the only cases of $(1+\varepsilon)$-dominance are: $z^1 \preceq_\varepsilon z^2$, $z^2 \preceq_\varepsilon z^3$, $z^3 \preceq_\varepsilon z^1$, $z \preceq_\varepsilon z^i$ and $z^i \preceq_\varepsilon z$ for $i = 1, 2, 3$, and $z^i \preceq_\varepsilon z^{ij}$ for $i = 1, 2, 3$ and $j = 1, \ldots, k$.

The set constituted by points z^1, z^2, and z^3 is clearly an ε-Pareto set of minimal size. Moreover, any ε-kernel must contain point z and thus points z^{ij} for $i = 1, 2, 3$ and $j = 1, \ldots, k$. This is the only ε-kernel and it contains $3k+1$ points. □

However, if we consider $\varepsilon' \leq \sqrt{1+\varepsilon} - 1$, we can show that an $(\varepsilon, \varepsilon')$-kernel always exists. For this purpose, we recall the notion of quasi-kernel (also called semi-kernel).

Definition 17. *Given a directed graph $G = (V, A)$, a quasi-kernel is a set $S \subseteq V$ such that (i) for any $v \in V - S$, there exists $v' \in S$ such that $(v', v) \in A$ or there exist $v' \in S$ and $v'' \in V - S$ such that $(v', v'') \in A$ and $(v'', v) \in A$ (ii) for any $u, v \in S$, $u \neq v$, $(u, v) \notin A$.*

The following result is well-known.

Theorem 16. *(Chvátal and Lovász [17]) Any finite directed graph G admits a quasi-kernel.*

Applied in our context, this gives rise to the following result.

Proposition 24. *For any number of objectives $p \geq 3$ and any finite set Z of points an $(\varepsilon, \varepsilon')$-kernel exists if and only if $\varepsilon' \leq \sqrt{1+\varepsilon} - 1$.*

Proof :

$\Leftarrow$ Consider the graph $G = (Z, \preceq_{\varepsilon'})$ and apply Theorem 16.

$\Rightarrow$ Assuming that $\varepsilon' > \sqrt{1+\varepsilon} - 1$, we show the existence of an instance which does not admit an $(\varepsilon, \varepsilon')$-kernel.

Let $Z = \{z^1, z^2, z^3\}$ where z^1, z^2, and z^3 are three points in the criterion space and assume that their coordinates are the following: $z^1 = (a(1+\varepsilon'), b/(1+\varepsilon'), c)$, $z^2 = (a, b(1+\varepsilon'), c/(1+\varepsilon'))$, $z^3 = (a/(1+\varepsilon'), b, c(1+\varepsilon'))$ with a, b, and c three nonnegative rational numbers.

Remark that z^1 $(1+\varepsilon')$-dominates z^2, z^2 $(1+\varepsilon')$-dominates z^3 and z^3 $(1+\varepsilon')$-dominates z^1. In order to satisfy the ε'-stability condition an $(\varepsilon, \varepsilon')$-kernel contains at most one point among z^1, z^2, and z^3. Moreover, since $\varepsilon' > \sqrt{1+\varepsilon} - 1$, no point $(1+\varepsilon)$-dominates the two others and thus in order to satisfy the ε-coverage condition, an $(\varepsilon, \varepsilon')$-kernel must contain at least two points. This is clearly impossible. □

Moreover, when the points Z are given explicitly and $\varepsilon' \leq \sqrt{1+\varepsilon} - 1$ it is possible to compute an $(\varepsilon, \varepsilon')$-kernel in polynomial time. Indeed, the problem can be reduced to finding a kernel in a directed acyclic graph [23]. We briefly describe the method of Duchet et al. from [23]. Consider the directed graph $G = (Z, \preceq_{\varepsilon'})$ and any arbitrary order $<$ on the vertices. We first partition the set of arcs into two disjoint subsets $A_1 = \{(i,j) \in \preceq_{\varepsilon'} : i < j\}$, $A_2 = \{(i,j) \in \preceq_{\varepsilon'} : i > j\}$. The two directed graphs (Z, A_1) and (Z, A_2) contain no cycle. Since a (unique) kernel can be easily computed in polynomial time in directed acyclic graphs, first construct the kernel K of (Z, A_1) and then the kernel K' of (K, A_2). The resulting subset K' is a quasi-kernel of G, i.e. an $(\varepsilon, \varepsilon')$-kernel.

In the general case, when the points of the criterion space are not given explicitly, we have the following result.

Proposition 25. *For $p \geq 3$ objectives and any $0 < \varepsilon' \leq \sqrt[3]{1+\varepsilon} - 1$, an $(\varepsilon, \varepsilon')$-kernel is computable in polynomial time when the associated routine* GAP$_\delta$ *runs in polynomial time.*

Proof : First we construct a grid in the criterion space as in the proof of the efficient constructability of an ε-Pareto set presented in [66]. Consider a subdivision of the criterion space into hyperrectangles such that, in each dimension, the ratio of the largest to the smallest coordinate of each hyperrectangle is $\sqrt[6]{1+\varepsilon}$. In each corner point, call the *GAP*$_\delta$ routine with $\delta = \sqrt[6]{1+\varepsilon} - 1$ and denote by S the resulting set of points. Set S (after removing the dominated points) is clearly an $(\sqrt[3]{1+\varepsilon}-1)$-Pareto set.

On set S, we use the method of Duchet et al. [23] to construct a quasi-kernel in a directed graph. Thus, we obtain a subset $K \subseteq S$ which is an $((\sqrt[3]{1+\varepsilon})^2 - 1, \sqrt[3]{1+\varepsilon}-1)$-kernel for the points in S. Since S is an $(\sqrt[3]{1+\varepsilon}-1)$-Pareto set, it implies that K is an $((\sqrt[3]{1+\varepsilon})^2 \times \sqrt[3]{1+\varepsilon} - 1, \sqrt[3]{1+\varepsilon}-1)$-kernel i.e. an $(\varepsilon, \varepsilon')$-kernel. □

Nevertheless, we can show a similar result to Proposition 23 for $(\varepsilon, \varepsilon')$-kernels.

Proposition 26. *For $p \geq 3$ objectives and any $0 < \varepsilon' \leq \sqrt{1+\varepsilon} - 1$, the size of a smallest $(\varepsilon, \varepsilon')$-kernel can be greater than $k.opt_\varepsilon$ for any integer k.*

Proof: Let $p = 3$ and z^1, z^2, and z^3 be three points with the following coordinates: $z = (a, b, c)$, $z^1 = (a(1+\varepsilon'), b/(1+\varepsilon'), c)$, $z^2 = (a, b(1+\varepsilon'), c/(1+\varepsilon'))$, $z^3 = (a/(1+\varepsilon'), b, c(1+\varepsilon'))$ where a, b, and c are three nonnegative rational numbers. Fix any rational $\hat{\varepsilon}$ and consider $6k$ points $z^{1j} = (z^1_1(1+\hat{\varepsilon})^j, z^1_2/(1+\varepsilon), z^1_3(1+\hat{\varepsilon})^{2k-j})$, $z^{2j} = (z^2_1(1+\hat{\varepsilon})^{2k-j}, z^2_2(1+\hat{\varepsilon})^j, z^2_3/(1+\varepsilon))$ and $z^{3j} = (z^3_1/(1+\varepsilon), z^3_2(1+\hat{\varepsilon})^{2k-j}, z^3_3(1+\hat{\varepsilon})^j)$ for $j = 1, \ldots, 2k$.

Remark that points z, z^1, z^2, and z^3 $(1+\varepsilon)$-dominates each other and $z^i \preceq_\varepsilon z^{ij}$ for $i = 1, 2, 3$ and $j = 1, \ldots, k$. For this instance, the only cases of $(1+\varepsilon')$-dominance are: $z^1 \preceq_{\varepsilon'} z^2$, $z^2 \preceq_{\varepsilon'} z^3$, $z^3 \preceq_{\varepsilon'} z^1$, $z \preceq_{\varepsilon'} z^i$ and $z^i \preceq_{\varepsilon'} z$ for $i = 1, 2, 3$.

The set constituted by points z^1, z^2, and z^3 is clearly an ε-Pareto set of minimal size. Moreover, a smallest $(\varepsilon, \varepsilon')$-kernel contains a point z^i with $i = 1, 2, 3$ and all the points $z^{i'j}$ for $i' = 1, 2, 3$ with $i \neq i'$ and $j = 1, \ldots, 2k$, and it contains $4k + 1$ points. □

5.4 Conclusions

The purpose of this chapter was to produce discrete and tractable representations of the set of nondominated points for multi-objective optimization problems. We considered that representations should satisfy some conditions of *coverage*, *spacing*, and *cardinality*. For this purpose, we introduced the notion of $(\varepsilon, \varepsilon')$-kernel which is a particular ε-Pareto set that satisfies an additional condition of stability implementing *spacing*. We proposed some generic methods to produce $(\varepsilon, \varepsilon')$-kernels. Our algorithms run in polynomial time if and only if the subroutines called in the algorithms run in polynomial time.

The situation for the bi-objective case is quite clear and the concept of $(\varepsilon, \varepsilon')$-kernel, or even ε-kernel, seems relevant to provide a good discrete representation of the nondominated set. For more than two objectives, we showed that imposing a condition of spacing may impact negatively on the cardinality. Since a coverage condition must necessarily be imposed, the choice is between emphasizing spacing or cardinality. If the condition of spacing prevails, we showed that it is possible to construct an $(\varepsilon, \varepsilon')$-kernel, with $\varepsilon' \leq \sqrt{1+\varepsilon} - 1$, provided that the *GAP*$_\delta$ routine is available, but without any guarantee on its cardinality. If the condition of cardinality prevails, known guarantees are very weak, even without any condition on spacing. The only known result, by Koltun and Papadimitriou [54], in the tri-objective case and when the points of the objective space are explicitly given, is the existence of a polynomial time algorithm which returns an ε-Pareto set of size at most $100opt_\varepsilon$.

6 Conclusions and perspectives

This thesis mainly deals with the difficulty of obtaining, for a multi-objective optimization problem, a good represention of the set of nondominated points, due to the intractability of the problem. We proposed new general methods to get around this intractability.

First, assuming that it is possible to obtain some information on the different criteria, more precisely, the number of values taken on each criterion, we developed some bounds on the cardinality of the set of nondominated points. Our main bound is computable in constant time if the number of criteria is assumed to be constant. We also showed that this bound is tight for many multi-objective optimization problems. We showed how these bounds can be reduced when some feasible or efficient solutions are known.

Then, we were interested in producing some discrete and tractable representations of the nondominated set for each instance of multi-objective optimization problems. These representations must satisfy some conditions of *coverage*, *cardinality* and if possible *spacing*. For this purpose, we investigated the problem, introduced by Vassilvitskii and Yannakakis, of computing a minimum set of solutions for a multi-objective optimization problem that represents approximately the Pareto set within an accuracy ε. For the bi-objective case, we presented a new 3-approximation algorithm of the size of a smallest ε-Pareto set. We showed that for a classical bi-objective problem this approximation is tight unless P = NP. This problem become really hard when the number of criteria is greater than or equal to three. In the case where the points of the objective space are given explicitly in the input, we studied the performance of a greedy algorithm, answering an open question of Koltun and Papadimitriou.

We introduced the concept of $(\varepsilon, \varepsilon')$-kernel to take into consideration the dimension of *spacing*. We showed some general properties on $(\varepsilon, \varepsilon')$-kernels and proposed algorithms to compute small $(\varepsilon, \varepsilon')$-kernels in polynomial time for the bi-objective case assuming that there exists a routine *Restrict*$_\delta$ that runs in polynomial time. The number of calls to *Restrict*$_\delta$ is linear in the number of points in a smallest ε-Pareto set.

There are many perspectives to this work. We already presented most of them at the end of each chapter of the thesis. We recall and develop the main ones here and add some general comments.

In chapter 3, for a multi-objective optimization problem, a basic assumption in our work is the a priori knowledge on the number of values taken on each criterion. Obviously, obtaining a good upper bound on these values is itself a difficult question which depends on the problem as well as on particular characteristics of specific instances. Thus, for a given multi-objective optimization problem, finding some large classes of instances for which tight upper bounds on the number of values taken on each criterion, appears as an interesting issue.

In chapter 4, we mainly showed that for more than two objectives, when the feasible points of a multi-objective optimization problem are given explicitly in the input, a greedy algorithm cannot be used to construct *small* ε-Pareto sets. In this case, it is known on the one hand that it is NP-hard to determine an ε-Pareto set of minimal size but on the other hand that there is a 100-approximation of the size of a smallest ε-Pareto set. Thus, there is a large gap between the NP-hardness and the 100-approximation. Making these lower and upper bounds closer is an interesting open question.

In chapter 5, for a multi-objective optimization problem, in the case of more than two objectives, we showed that an $(\varepsilon, \varepsilon')$-kernel with $\varepsilon' \leq \sqrt{1+\varepsilon} - 1$ always exists and moreover that its size could be arbitrarily far from the size of a smallest ε-Pareto set. This means that obtaining a stability condition implies to lose a good property on the size of the set obtained. It means that the dimensions of *spacing* and *cardinality* are conflicting, at least in the worst case. An interesting problem is to define a different stability condition ensuring the existence of a small ε-Pareto set satisfying this stability condition.

An interesting approach would be to consider a particular multi-objective optimization problem and study the issues discussed in this thesis applied to this particular problem. Especially, it seems not easy to take benefit of the underlying problem to obtain some results for primal and dual problems. It could also be interesting to implement the different algorithms presented in this thesis for a particular multi-objective problem, which of course requires to use the best known algorithm to solve the subroutines called.

We mentioned that finding an ε-Pareto set when the points of the objective space are given explicitly is a set covering problem, with particular instances. A very interesting open problem is to obtain a characterization of these instances. This requires to have a characterization of the relation $\preceq_\varepsilon$.

To conclude, we hope that this work, brings some insights and helps to develop different ways to get around the intractability of multi-objective optimization problems.

Bibliography

[1] H. Aissi, A. R. Mahjoub, S. T. McCormick, and M. Queyranne. A strongly polynomial time algorithm for multicriteria global minimum cuts. In *Proceedings of the 17th International Conference on Integer Programming and Combinatorial Optimization (IPCO'14)*, 2014, to appear.

[2] E. Angel, E. Bampis, and L. Gourvès. Approximation algorithms for the bi-criteria weighted max-cut problem. *Discrete Applied Mathematics*, 154(12):1685–1692, 2006.

[3] E. Angel, E. Bampis, L. Gourvès, and J. Monnot. (non)-approximability for the multi-criteria TSP(1,2). In *Proceedings of the 15th International Symposium on Fundamentals of Computation Theory (FCT'05)*, LNCS 3623, pages 329–340, 2005.

[4] E. Angel, E. Bampis, and A. Kononov. On the approximate tradeoff for bicriteria batching and parallel machine scheduling problems. *Theoretical Computer Science*, 306(1-3):319–338, 2003.

[5] A. Asadpour and A. Saberi. An approximation algorithm for max-min fair allocation of indivisible goods. *SIAM Journal on Computing*, 39(7):2970–2989, 2010.

[6] C. Bazgan, L. Gourvès, and J. Monnot. Approximation with a fixed number of solutions of some multiobjective maximization problems. *Journal of Discrete Algorithms*, 22:19–29, 2013.

[7] C. Bazgan, A. Hugot, and D. Vanderpooten. Solving efficiently the 0-1 multi-objective knapsack problem. *Computers and Operations Research*, 36(1):260–279, 2009.

[8] H. P. Benson and S. Sayin. Towards finding global representations of the efficient set in multiple objective mathematical programming. *Naval Research Logistics*, 44(1):47–67, 1997.

[9] A. Berger, V. Bonifaci, F. Grandoni, and G. Schäfer. Budgeted Matching and Budgeted Matroid Intersection via the Gasoline Puzzle. In *Proceedings of the 13th International Conference on Integer Programming and Combinatorial Optimization (IPCO'08)*, LNCS 5035, pages 273–287, 2008.

[10] D. Bertsimas, V.F. Farias, and N. Trichakis. The price of fairness. *Operations Research*, 59(1):17–31, 2011.

[11] J-F. Bérubé, M. Gendreau, and J-Y. Potvin. An exact [epsilon]-constraint method for bi-objective combinatorial optimization problems: Application to the traveling salesman problem with profits. *European Journal of Operational Research*, 194(1):39–50, 2009.

[12] N.G. De Bruijn, C. Tengbergen, and D. Kruyswijk. On the set of divisors of a number. *New Archive for Mathematics*, 23:191–193, 1951.

[13] P. M. Camerini. The complexity of finding multi-constrained spanning tree problems. *Discrete Applied Mathematics*, 5(1):39–50, 1983.

[14] N. Caspar, B. Leclerc, and B. Monjardet. *Finite ordered sets: concepts, results and uses*. Cambridge University Press, 2012.

[15] C. A. Charalambides. *Enumerative combinatorics*. Chapman, 2002.

[16] T.C.E. Cheng, A. Janiak, and M. Kovalyov. Bicriterion single machine scheduling with resource dependent processing times. *SIAM Journal on Optimization*, 8(2):617–630, 1998.

[17] V. Chvátal and L. Lovász. Every directed graph has a semi-kernel. *Lecture Notes in Mathematics*, 411:175, 1974.

[18] J. Cohon. *Multiobjective Programming and Planning*. Dover, 2004.

[19] I. Das and J. E. Dennis. Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization*, 8(3):631–657, 1998.

[20] I. Diakonikolas and M. Yannakakis. Small approximate Pareto sets for bi-objective shortest paths and other problems. *SIAM Journal on Computing*, 39(4):1340–1371, 2009.

[21] D. Dubois and H. Prade. A review of fuzzy set aggregation connectives. *Information Sciences*, 36(1-2):85–121, 1985.

[22] P. Duchet. Graphes noyau-parfaits. *Annals of Discrete Mathematics*, 9:93–101, 1980.

[23] P. Duchet, Y.O. Hamidoune, and H. Meyniel. Sur les quasi-noyaux d'un graphe. *Discrete Mathematics*, 65(3):231–235, 1987.

[24] M. Ehrgott. *Multicriteria Optimization*. Springer, 2000.

[25] M. Ehrgott and X. Gandibleux. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR Spektrum*, 42(4):425–460, 2000.

[26] M. Ehrgott and L. Shao. Multi-objective optimization for supporting radiation therapy treatment planning. In *22nd International Conference on Multiple Criteria Decision Making (MCDM'13*, 2013.

[27] M. Ehrgott and D. Tenfelde-Podehl. Computation of ideal and nadir values and implications for their use in MCDM methods. *European Journal of Operational Research*, 151(1):119–139, 2002.

[28] G. Eichfelder. An adaptive scalarization method in multiobjective optimization. *SIAM Journal on Optimization*, 19(4):1694–1718, 2009.

[29] V. A. Emelichev and V. A. Perepelista. On cardinality of the set of alternatives in discrete many-criterion problems. *Discrete Mathematics and Applications*, 2(5):461–471, 1992.

[30] F. Ergun, R. Sinha, and L. Zhang. An improved FPTAS for Restricted Shortest Path. *Information Processing Letters*, 83(5):237–239, 2002.

[31] T. Erlebach, H. Kellerer, and U. Pferschy. Approximating multiobjective knapsack problems. *Management Science*, 48(12):1603–1612, 2002.

[32] B. Escoffier, L. Gourvès, and J. Monnot. Fair solutions for some multiagent optimization problems. *Autonomous Agents and Multi-Agent Systems*, 26(2):184–201, 2013.

[33] S. L. Faulkenberg and M. M. Wiecek. On the quality of discrete representations in multiple objective programming. *Optimization and Engeneering*, 11(3):423–440, 2010.

[34] J. Figueira, S. Greco, and M. Ehrgott. *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer, 2005.

[35] J. C. Fodor and M. Roubens. *Fuzzy Preference Modelling and Multicriteria Decision Aid*. Kluwer Academic Publisher, 1994.

[36] A. Goel and A. Meyerson. Simultaneous optimization via approximate majorization for concave profits or convex costs. *Algorithmica*, 44(4):301–323, 2006.

[37] M. X. Goemans and R. Ravi. The Constrained Minimum Spanning Tree Problem. In *Proceedings of the 5th Scandinavian Workshop on Algorithm Theory (SWAT'96)*, LNCS 1097, pages 66–75, 1996.

[38] M.C. Golumbic. *Algorithmic graph theory and perfect graphs*. Annals of Discrete Mathematics Series, vol 57, North Holland, 2004.

[39] C. Gonzales, P. Perny, and S. Queiroz. Preference aggregation with graphical utility models. In *Proceedings of the 23rd Conference on Artificial Intelligence (AAAI'08)*, pages 1037–1042, 2008.

[40] L. Gourvès, J. Monnot, and L. Tlilane. Approximate tradeoffs on matroids. In *Proceedings of the 20th European Conference on Artificial Intelligence (ECAI'12)*, pages 360–365, 2012.

[41] L. Gourvès, J. Monnot, and L. Tlilane. A matroid approach to the worst case allocation of indivisible goods. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, pages 136–142, 2013.

[42] M. Grabisch and P. Perny. Agrégation multicritère. In B. Bouchon and C. Marsala, editors, *Utilisations de la logique floue*. Hermès, 1999.

[43] F. Grandoni and R. Zenklusen. Approximation schemes for multi-budgeted independence systems. In *Proceedings of the 18th Annual European Conference on Algorithms (ESA'10)*, LNCS 6346, pages 536–548, 2010.

[44] H.W. Hamacher and G. Ruhe. On spanning tree problems with multiple objectives. *Annals of Operations Research*, 52(4):209–230, 1994.

[45] P. Hansen. Bicriterion path problems. In *Proceedings of the 3rd conference on Multiple Criteria Decision Making: Theory and Applications*, volume LNEMS 177, pages 109–127, 1979.

[46] R. Hassin. Approximation schemes for the restricted shortest path problem. *Mathematics of Operations Research*, 17(1):36–42, 1992.

[47] R. Hassin and A. Levin. An efficient polynomial time approximation scheme for the constrained minimum spanning tree problem using matroid intersection. *SIAM Journal on Computing*, 33(2):261–268, 2004.

[48] J. Hastad. Some optimal inapproximability results. *Journal of ACM*, 48(4):798–859, 2001.

[49] S. Helbig. On a constructive approximation of the efficient outcomes in bicriterion vector optimization. *Journal of Global Optimization*, 5(1):35–48, 1994.

[50] S-P. Hong, S-J. Chung, and B-H. Park. A fully polynomial bicriteria approximation scheme for the constrained spanning tree problem. *Operations Research Letters*, 32(3):233–239, 2004.

[51] H. Hugot. *Approximation et énumération des solutions efficaces dans les problèmes d'optimisatio combinatoire multi-objectifs*. PhD thesis, Université Paris-Dauphine, 2007.

[52] E. Jeannot, E. Saule, and D. Trystram. Bi-objective approximation scheme for makespan and reliability optimization on uniform parallel machines. In *Proceedings of the 14th International Conference on Parallel Processing (Euro-Par'08)*, LNCS 5168, pages 877–886, 2008.

[53] K. Klamroth and M. Wiecek. Dynamic programming approaches to the multiple criteria knapsack problem. *Naval Research Logistics*, 47(1):57–76, 2000.

[54] V. Koltun and C. Papadimitriou. Approximately dominating representatives. *Theoretical Computer Science*, 371(3):148–154, 2007.

[55] A. Kumar and J.M. Kleinberg. Fairness measures for ressource allocation. *SIAM Journal on Computing*, 36(3):657–680, 2006.

[56] M. Laumanns, L. Thiele, and E. Zitzler. An adaptive scheme to generate the pareto front based on the epsilon-constraint method. *European Journal of Operational Research*, 169:932–942, 2005.

[57] B. Leclerc. Sur le nombre d'éléments des niveaux des produits de chaînes et des treillis permutoèdres. *Mathématiques et sciences humaines*, 112:37–48, 1990.

[58] D. H. Lorenz and D. Raz. A simple efficient approximation scheme for the restricted shortest path problem. *Operations Research Letters*, 28(5):213–219, 2001.

[59] B. Manthey. On approximating multicriteria TSP. *ACM Transactions on Algorithms*, 8(2), 17, 2012.

[60] J. L. Marichal. *Aggregation operators for multicriteria decision aid*. PhD thesis, University of Liège, 1998.

[61] C.A. Mattson, A.A. Mullur, and A. Messac. Smart pareto filter: Obtaining a minimal representation of multiobjective design space. *Optimization and Engineering*, 36(6):721–740, 2004.

[62] G. Mavrotas. Effective implementation of the e-constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation*, 213(2):455–465, 2009.

[63] G. Mavrotas and D. Diakoulaki. A branch and bound algorithm for mixed zero-one multiple objective linear programming. *European Journal of Operational Research*, 107(3):530–541, 1998.

[64] G. L. Nemhauser and Z. Ullmann. Discrete dynamic programming and capital allocation. *Management Science*, 15(9):494–505, 1969.

[65] Y. Okamoto and T. Uno. A polynomial-time-delay and polynomial-space algorithm for enumeration problems in multi-criteria optimization. In *Proceedings of the 18th International Symposium on Algorithms and Computation (ISAAC'07)*, LNCS 4835, pages 609–620, 2007.

[66] C. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science (FOCS'00)*, pages 86–92, 2000.

[67] A. Przybylski, X. Gandibleux, and M. Ehrgott. Two phase algorithms for the bi-objective assignment problem. *European Journal of Operational Research*, 185(2):509–533, 2008.

[68] A. Przybylski, X. Gandibleux, and M. Ehrgott. A recursive algorithm for finding all nondominated extreme points in the outcome set of a multiobjective integer programme. *INFORMS Journal on Computing*, 22(3):371–386, 2010.

[69] A. Przybylski, X. Gandibleux, and M. Ehrgott. A two phase method for multi-objective integer programming and its application to the assignment problem with three objectives. *Discrete Optimization*, 7(3):149–165, 2010.

[70] B. Roy. Classement et choix en présence de points de vue multiples: La méthode ELECTRE. *Revue Francaise d'Informatique et de Recherche Opérationnelle*, 8:57–75, 1968.

[71] H. M. Safer, J. B. Orlin, and M. Dror. Fully polynomial approximation in multi-criteria combinatorial optimization. 2004.

[72] S. Sayin. Measuring the quality of discrete representations of efficient sets in multiple objective mathematical programming. *Mathematical Programming*, 87(3):543–560, 2000.

[73] S. Sayin. A procedure to find discrete representations of the efficient setwith specified coverage errors. *Operational Research*, 51(3):427–436, 2003.

[74] P. Serafini. Some considerations about computational complexity for multiobjective combinatorial problems. *Recent advances and historical development of vector optimization*, 294:222–232, 1986.

[75] M. Stanojević, M. Vujošević, and B. Stanojević. On the cardinality of the nondominated set of multi-objective combinatorial optimization problems. *Operations Research Letters*, 41(2):197–200, 2013.

[76] C. Stein and J. Wein. On the existence of schedules that are near-optimal for both makespan and total weighted completion time. *Operations Research Letters*, 21(3):115–122, 1997.

[77] J. Sylva and A. Crema. A method for finding well-dispersed subsets of non-dominated vectors for multiple objective mixed integer liear programs. *European Journal of Operational Research*, 180(3):1011–1027, 2007.

[78] G. Tsaggouris and C. Zaroliagis. Multiobjective optimization: Improved fptas for shortest paths and non-linear objectives with applications. In *Proceedings of the 17th International Symposium on Algorithms and Computation (ISAAC'06)*, LNCS 4288, pages 389–398, 2006.

[79] E.L. Ulungu and J. Teghem. The two phases method: An efficient procedure to solve bi-objective combinatorial optimization problems. *Foundations of Computing and Decision Sciences*, 20(2):149–165, 1995.

[80] S. Vassilvitskii and M. Yannakakis. Efficiently computing succinct trade-off curves. *Theoretical Computer Science*, 348(2-3):334–356, 2005.

[81] M. Visée, J. Teghem, M. Pirlot, and E.L. Ulungu. Two-phases method and branch and bound procedures to solve the bi–objective knapsack problem. *Journal of Global Optimization*, 12(2):139–155, 1998.

[82] A. Warburton. Approximation of pareto optima in multiple-objective shortest-path problems. *Operations research*, 35(1):70–79, 1987.

[83] R. R. Yager. Connectives and quantifiers in fuzzy sets. *Fuzzy Sets and Systems*, 40(1):39–75, 1991.

Résumé

Le but de cette thèse est de proposer des méthodes générales afin de contourner l'intractabilité de problèmes d'optimisation multi-objectifs.

Dans un premier temps, nous essayons d'apprécier la portée de cette intractabilité en déterminant une borne supérieure, facilement calculable, sur le nombre de points non dominés, connaissant le nombre de valeurs prises par chaque critère. Nous nous attachons ensuite à produire des représentations discrètes et tractables de l'ensemble des points non dominés de toute instance de problèmes d'optimisation multi-objectifs. Ces représentations doivent satisfaire des conditions de *couverture*, i.e. fournir une bonne approximation, de *cardinalité*, i.e. ne pas contenir trop de points, et si possible de *stabilité*, i.e. ne pas contenir de redondances. En s'inspirant de travaux visant à produire des ensembles ε-Pareto de petite taille, nous proposons tout d'abord une extension directe de ces travaux, puis nous axons notre recherche sur des ensembles ε-Pareto satisfaisant une condition supplémentaire de *stabilité*. Formellement, nous considérons des ensembles ε-Pareto particuliers, appelés $(\varepsilon, \varepsilon')$-noyaux, qui satisfont une propriété de stabilité liée à ε'. Nous établissons des résultats généraux sur les $(\varepsilon, \varepsilon')$-noyaux puis nous proposons des algorithmes polynomiaux qui produisent des $(\varepsilon, \varepsilon')$-noyaux de petite taille pour le cas bi-objectif et nous donnons des résultats négatifs pour plus de deux objectifs.

Mots-clés : Représentations discrètes, ensemble de Pareto, approximation, points non dominés, noyaux, problèmes d'optimisation multi-objectifs.

Abstract

The goal of this thesis is to propose new general methods to get around the intractability of multi-objective optimization problems.

First, we try to give some insight on this intractability by determining an, easily computable, upper bound on the number of nondominated points, knowing the number of values taken on each criterion. Then, we are interested in producing some discrete and tractable representations of the set of nondominated points for each instance of multi-objective optimization problems. These representations must satisfy some conditions of *coverage*, i.e. providing a good approximation, *cardinality*, i.e. it does not contain too many points, and if possible *spacing*, i.e. it does not include any redundancies. Starting from works aiming to produce ε-Pareto sets of small size, we first propose a direct extension of these works then we focus our research on ε-Pareto sets satisfying an additional condition of *stability*. Formally, we consider special ε-Pareto sets, called $(\varepsilon, \varepsilon')$-kernels, which satisfy a property of stability related to ε'. We give some general results on $(\varepsilon, \varepsilon')$-kernels and propose some polynomial time algorithms that produce small $(\varepsilon, \varepsilon')$-kernels for the bicriteria case and we give some negative results for the tricriteria case and beyond.

Keywords: Discrete representations, Pareto set, approximation, Pareto set, nondominated points, kernels, multi-objective optimization problems.